GPS-Guided Mobile Robot Platform Featuring Modular Design Elements for Agricultural Applications

A thesis presented in partial fulfilment of the requirements for the degree of

Masters of Engineering
in
Mechatronics

at Massey University Turitea Campus
Palmerston North
New Zealand.

Samuel John Oldfield Corpe

2016

Abstract

The agricultural industry has not seen significant innovation in development of low-cost automated farming solutions, with current systems costing several thousands of dollars to implement. Currently these automated solutions are primarily implemented around crop planting and harvesting, and the large implementation cost of these systems makes them unfeasible for small-scale operations. Within many agricultural industries, workers expend a considerable amount of time undertaking simple tasks that are labour intensive. Many of these tasks could instead be completed using a self-driving robotic platform outfitted with the appropriate devices required for the tasks.

This thesis covers the research work aiming to produce a solution that could turn an existing farming vehicle into a multipurpose low-cost agricultural platform, to act as the platform for an autonomous vehicle capable of performing pre-programmed tasks within an agricultural environment. A quad bike was selected as the vehicle platform for this research in which the control modules would control the speed and direction of this farm bike.

Four modules were developed to control the vehicle components that would normally be operated by a human operator. These modules are comprised of mechanical actuators coupled with a microcontroller control system and includes some specific designs to maintain the user's ability to manually control the pre-existing systems. A gear-changing module controls the vehicles manual gearbox, providing a method to detect and control the vehicles current gear. A speed control module was developed to control the vehicles throttle and braking system and detects the vehicles speed. A steering module controls the vehicles steering system, allowing for accurate control of the vehicles direction. Finally, a vehicle controller module provides a central command interface that ties the previous three modules together and controls the vehicles electrical components and engine.

Development of a low-cost differential GPS (DGPS) system was also undertaken to reduce the implementation cost of the system. Due to inconclusive results in relation to the positional accuracy of this system is was decided that a standard GPS system would be used for the vehicle prototype with further development on the DGPS system would be undertaken in future development of the research.

The successful development of a farm automated vehicle platform was achieved through this research. With further improvement on software, intelligent control and the development of a low-cost differential global positioning satellite (GPS) system, a fully autonomous farm platform that can be outfitted with different tools or devices for the required farm tasks is feasible and practical.

Acknowledgements

I would like to express my appreciation to my supervisor Dr Liqiong Tang who provided me with valued guidance and support throughout this research project.

A huge thank you to Mr Clive Bardell, Mr Ian Thomas, Mr Kerry Griffiths, Mr Anthony Wade and Mr Morio Fukuoka for providing technical advice during the design and manufacture of the mechanical and electrical hardware within this project.

I would like to thank all other support staff within the School of Engineering and Advanced Technology at Massey University Palmerston North, particularly the lovely women in the finance office and at reception for all of your help.

I would like to thank all my friends in particular Mr Christopher Scott and Mr Miguel Contreras for their advice, support and friendship throughout our time completing our undergraduate and postgraduate studies at Massey University.

Finally, I want to thank my parents and extended family for all the support they have always provided.

List of Figures

List of Tables

List of Abbreviations

Global Positioning Satellite - GPS

Differential Global Positioning Satellite - DGPS

Universal Time Coordinated - UTC

Land Information New Zealand - LINZ

Earth centred Earth fixed - ECEF

Antilock Brake Systems - ABS

Defence Advanced Research Projects Agency - DARPA

Light Detection and Ranging - LIDAR

Ohio State University's - OSU

A-Star - A*

Infrared - IR

Pulse Width Modulation - PWM

Ground - GND

Transmit Line - TX

Receive Line - RX

Programmable Logic Device - PLD

Uninterruptible Power Supply - UPS

General-Purpose Input-Output pins - GPIO

Most Significant Bit - MSB

Proportional-Integral - PI

Chapter One Introduction

Rapid development within the areas of robotics, automation, navigation and sensing technologies has occurred over the past 20 years. These changes and new technologies developed affect the daily lives of humans, and provide improved methodologies and practices within the working place. While technological developments have produced innovations within the several industries such as self-driving cars within the automotive industry. The agricultural and farming sectors have only seen minor changes in the development of self-driving vehicles for agriculture, with many areas awaiting improvement. Traditional farming methods do not include labour intensive activities such as soil sampling on a regular basis. While this is a relatively simple activity, high equipment and high labour cost associated with accurate soil sampling limit most farmers to only undertaking these tasks annually.

This research aims to fill in a gap relating to autonomous vehicles within the agricultural sector, through the development of a low cost base vehicle for a multipurpose agricultural vehicle platform with a focus on modular design elements. The functionality of this vehicle is augmented through the addition of external modules. These modules alter the individual tasks that the system is capable of undertaking within its operating environment. This base vehicle features individual control modules that are capable of manipulating the mechanical and electrical controls that would normally require a human operator in order for the vehicle to function, and includes an onboard GPS sensor augmented with a magnetometer compass to provide directional navigation.

A portion of this research is devoted to the development of a low cost differential system to improve the vehicles ability to sense its position within an outdoor environment, in an attempt to provide sub metre accuracy to a the vehicle control system that will be developed at a later date.

1.1 The Research Topic

The aim of this research is to develop the a mobile platform that is suitable for use within the agricultural industry, which must be have systems in place to enable navigation between two GPS defined coordinates within an outdoor environment. A user control interface must be developed, including a software system to control the operation of the platform.

The objectives are:

- Design a mobile platform capable of traversing roads, fields over uneven terrain autonomously. The vehicle must be capable of serving multiple purposes, fit through gateways and down existing roadways.
- Develop a software control system that will allow a controller to set the speed and direction of the developed vehicle platform
- Investigate control and mapping systems that could generate an area map for system to use as well as a develop path finding that will allow navigation within this area using this map along with external sensor data to perform preset tasks.

Users could benefit from several safety features developed as a part of this research that are aimed to reduce the risk of vehicle accidents that can cause death. According to statistics available on the Work Safe New Zealand website[19], 850 people are injured on farms riding quad bikes every year, with five deaths being attributed to this activity. These deaths can be caused by users rolling their quad bikes and becoming pinned under the bikes, it often may take hours or sometimes days before help is able to arrive. In many of these cases, no one is aware that an accident has occurred or where the accident has happened.

Figure 1-1 Work Safe New Zealand Quad Bike Injury Statistics[19]

Due to the modular nature of this system, a simple long-range communication system with integrated GPS positioning could be produced to track the quad bike, and would automatically notify a base station on the farm that an accident has occurred and provide a GPS position to rescue workers, increasing the safety around the operation of quad bikes.

1.2 The Scope of Research

The scope of this research is to develop a small, cheap, reliable, and robust mobile platform featuring modular design elements through the conversion of an existing vehicle. This platform must provide a method to control an existing vehicle via a computer system, as well as withstanding environmental weather and terrain conditions found within agricultural environments.

- Allow the integration of additional modules to determine the functionality of the mobile platform
- Develop control modules that can be applied to multiple vehicles
- Sensors must detect the vehicle position and surrounding objects
- An onboard interface should be present for infield use
- Implementation of long range wireless communication will provide users with real-time updates and allow infield commands via remote access

1.3 Organisation of Dissertation

This dissertation contains nine chapters and appendices containing relevant GPS equations and datasheets. These chapters contain information on the development of the modular control modules, including their mechanical and electronic design as well as the controlling software developed for these systems. Details on the hardware components used within the implementation of research are outlined within the appendices, including formula and theories relating to GPS systems and serial commands developed to control the modules. The following paragraphs provide a description of each chapter within this dissertation.

Chapter 1 details the introduction and scope of the research conducted within this thesis.

Chapter 2 outlines the background research into the accuracy of current GPS technologies, the development of autonomous vehicles including key technologies, path-finding methodologies and existing autonomous vehicles within the agricultural sector.

Chapter 3 describes the proposed system developed as a part of this research, including design considerations for real world applications within the industry and outlines the expected final system.

Chapter 4 outlines the development of a low cost differential GPS system. This chapter covers selection of a GPS module, control system and long-range communication system necessary for this system. The final design for the differential system is outlined with the developed software system.

Chapter 5 contains the development of the mechanical systems necessary for the steering, braking, gear changing and cable control systems used by the various system modules. Requirements for the vehicle platform used for this system are outlined, including details on vehicles types used within the agricultural industry and the chosen vehicle in which this research is based upon.

Chapter 6 contains the electrical control circuits developed for the various mechanical modules outlined in chapter 5 and the vehicle controller that links the various modules together. The development of a user interface system is also outlined for this system.

Chapter 7 outlines the software developed for the hardware within the individual modules including detailing on the capabilities of each module based on the commands received. An overview of the user interface display and a preliminary path-finding system is also covered.

Chapter 8 details the testing undertaken on the various modules developed for this dissertation and the associated results.

Chapter 9 discusses the research presented at the conclusion of this thesis along with recommendations for the future development of this system based on the prototype developed.

Chapter Two Literature Review

Currently, there is a lack of autonomous vehicles within the farming and agricultural industries, with many of the jobs and activities preformed by workers being relatively simple in nature, but are time intensive making them undesirable to be performed on a regular basis. This research focuses around the development of a base vehicle for a small autonomous platform, designed for use within the agricultural sector. This review will focus on the technologies necessary in order for this vehicle to be controlled autonomously. These being; detection of its position within its environment, sensing systems required to detect objects within the environment, control systems used within autonomous vehicles, and investigating existing autonomous vehicles currently on the market.

For the mobile system to navigate to a particular location, it first must be capable of detecting its current location within its specified operational area. Range finders or image sensing can detect the position of landmarks, or with a GPS system to obtain a coordinate position. The resulting accuracy of the system determines the repeatable positioning accuracy of the vehicle system.

A number of existing autonomous vehicles have been developed using government research grants, primarily in relation to the Defence Advanced Research Project Agency (DARPA) grand challenges and urban challenges, as well as the Google self-driving car project. The various path-finding algorithms used within these projects should be identified and related to these projects. Several autonomous agricultural vehicles already exist, but these are limited to larger vehicles such as tractors and combine harvesters where they are used to steer the vehicle to ensure crop rows are planted and harvest in straight lines to provide optimum capacity within fields.

Unexpected and unmapped obstacles are expected to be present in the platforms operating environment. Therefore, the system requires a method of detecting these obstacles at a distance so that it can map and avoid them. Redundant sensors should also be included to detect the speed, direction, and angle of the vehicle in the event that the GPS positioning system requires supplementing.

2.1 GPS Systems

This section investigates the various GPS systems available and includes an overview of how the positioning system functions, as well as the known factors that affect the accuracy of the system. Various methods and systems will be investigated for both hardware and software to obtain accurate positional readings, constrained by any resulting system being low cost. An area of particular interest will be Differential GPS and the resulting accuracy of this these systems.

A brief background regarding the principles of the system as well as sources of error will be investigated, to aid in the design/selection of a GPS system for this research. The following GPS systems have been excluded from this review, as they were unavailable for this projects development and testing, as they are unavailable within New Zealand. These are the Russian GLONASS system, the European GALILEO satellite system and the Chinese Beidou/

Compass satellite system, all of which would provide improved accuracy over the GPS satellite system.

2.1.1 GPS Development and Working Principle

The Global Positioning System (GPS) comprises of a series of satellites in orbit of the planet, which act together to transmit a series of signals to the surface of the planet. Using this system, two data variables can be determined; firstly, the exact coordinate position of the receiver in terms of latitude, longitude and altitude, which will be accurate to within a range of 1mm to 20m depending on the system used[1, 6]. The second variable is the exact time as represented by the Universal Time Coordinated (UTC) format, and will be accurate within 60ns[6].

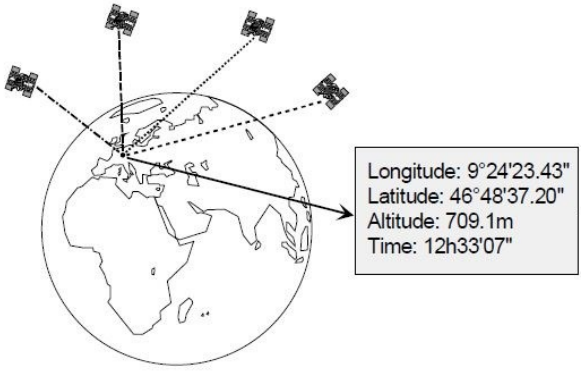


Figure 2-1 The basic function of satellite navigation[6]

The signal transmitted by each satellite comprises of data relating to the orbit of all the GPS satellites within the network as well as a time index for when the signal was transmitted[6]. This time index is used by the receiver to calculate the time taken for the signal to travel from the satellite, by comparing the receivers' clock to that of the satellite at time of transmission. The signals emitted from the satellite, travel at the speed of light, therefore, by calculating the time taken for the signal to travel to the receiver the distance between the satellite and receiver can be calculated.

Triangulation is the fundamental principle behind GPS technology[2], therefore using this calculated distance, as well as the calculated position in space of the satellite at the time of transmission, the receivers position can be represented as any point, which meets the specified distance from the satellite. In order to calculate the position of the receiver, the distance from three satellites is required, with a fourth satellite distance required to calculate the bias between the satellites clock and receiver clock. Therefore, a minimum of four satellites is required to achieve a full GPS fix for the receiver [2, 11].

GPS satellites feature high precision atomic clocks that use the resonance frequency of Caesium to track time[6]. This provides a time index of sufficient accuracy for positioning calculations, to ensure this precision is maintained with a synchronized time between all the satellites, a ground control system ensures the clocks are within a few nanoseconds of each other[2]. GPS devices typically feature a crystal controlled clock source, which drifts over time resulting in a time error relative to the satellite clock[2]. To prevent this from introducing an unknown distance error into the positioning system, a satellite signal must be devoted to

synchronizing the receiver clock, thus to obtain a position in three-dimensional space, at least four satellite signals must be available[6].

Satellite visibility and geometric configuration has a significant impact on the accuracy of GPS measurements, with objects such as trees, buildings or terrain blocking satellite signals or create multipath reflections which increases the calculated distance from satellite to receiver[2]. Geometric configuration refers to the spread of visible satellites within the receivers' field of view, with optimal satellite geometry obtained when all satellites are spread out in the sky [2, 3].

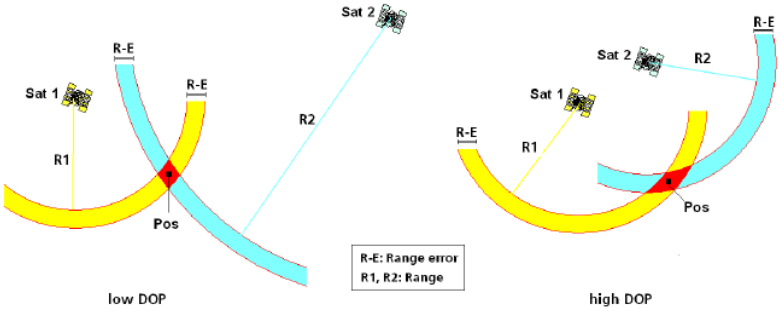


Figure 2-2 The flatter the angle with which the circles with ranges R1 and R2 intersect the higher the dolution of precision[6]

A GPS satellite broadcasts two signals, designated L1 and L2, to the ground from an altitude of 20,200Km, with each satellite completing two orbits per day. Both frequencies transmit the same broadcast message, with the L1 band oscillating at a frequency of 1,575.42 MHz and the L2 band operating at 1,227.60 MHz, with a wavelength of 19cm and 24cm respectively. Both frequencies broadcast the same message featuring the system clock, clock offset, highly accurate orbital data for the specific satellite, approximate orbital data for all satellites within the network[6].

2.1.2 Standalone GPS

Standalone GPS systems are commonly found in cell phones, handheld GPS systems and drones, and are referred to as standalone systems as they only rely on signals from the GPS satellite network. These receivers use the L1 band of signals to calculate its position on the planet, with the L2 band reserved for military use only. In theory, GPS is known as a precise positioning system, though there are several error sources. These reduce the accuracy of the system and are primarily categorized into three error groups: satellite, propagation and receiver[11].

For a standalone receiver, following error values can be expected in the calculated distance from the satellite:

Error Source	Error (metres)
Broadcast ephemeris (satellite position and orbit data)	2.5
Satellite clocks	1.5
Ionosphere delays	5
Troposphere delays	0.7
Receiver noise and numerical computations	0.7
Multipath and diffraction	1.5
Total Estimated Error	11.9

Table 2-1 Estimated effect on positional error from expected sources [2].

Due to these error sources, it is common for the calculated GPS position to wander around the true position, with many of the above signal errors varying slowly over time [1, 2].

2.1.3 Differential GPS

A differential GPS (DGPS) system uses the publicly available L1 satellite signals to provide high accurate positioning within the sub-metre range. DGPS operates on the principle that when two GPS receivers are within a specified distance with one another, both receivers will observe the same satellites and thus will be subject to the same atmospheric errors[2, 3, 11]. These errors are outlined in Table 2-1. While this is agreed, the distance at which the corrections are valid varies between sources, with some stating corrections are valid within 150-200km radius of the station[2, 6], with others stating corrections are only valid up to 50 km[12] of the station. To prevent inaccuracies within the system, a maximum operating radius of 50km should be employed on the system.

By fixing and measuring the precise position of a high-quality GPS receiver, a reference/base station is produced[2, 6]. With its exact position known, the pseudorange measurement can be instead used to calculate the current time difference/error on the transmitted signal for the individual satellite[2, 6]. This time error is calculated from the difference between the pseudorange measurement and the calculated distance based on the known position of the base station and the calculated nominal position of the satellite at the specified time index[3, 6].

The corrections for each satellite must then be transmitted via a suitable broadcaster[6, 11] to the receiver in field, providing real-time compensation or the corrections can be applied later or post-processing. The pseudorange corrections can in fact result in increased inaccuracies within the compensated positioning if the update rate for the corrections is too slow[11]. This error must be calculated for the individual pseudorange measurements, as opposed to adjusting the infield receiver by the specific deviation from the longitude, latitude and altitude values[6]. This is due to the specific satellites used in the base stations positional calculations are not guaranteed to match the satellites used by the roving receiver[6].

For other receivers to use these adjustments, the individual pseudorange errors must be formatted for use with the receiver with many of the GPS chips featuring automatic compensation of the satellites pseudorange based on corrections supplied using the RTCM format.

Therefore, a differential system can be simplified into three steps[6]:

1. Correction calculations for individual satellites at the fixed base station.
2. Packaging and transmission of the corrections from station to mobile GPS receiver.
3. Compensation of the measured pseudorange value from the individual satellites based on data from the base station, resulting in an adjusted GPS position.

Two kinds of DGPS signal compensation are possible resulting in differing levels of accuracy to the end position, with the corrections calculated based on pseudorange or carrier range measurements with an obtained accuracy ranging from a few metres to a few centimetres respectively[3]. Though to achieve this increase in accuracy, an extensive series of mathematical equations must be used to obtain an accuracy within a few millimetres[6].

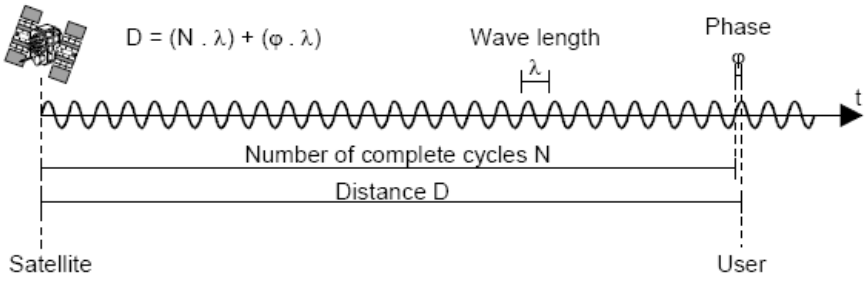


Figure 2-3 Principle of phase measurement[6]

For this prototype system, sub metre accuracy is not required, thus a Differential system based on pseudorange compensation is adequate for this research and the resulting prototype. Values obtained from the U-blox book[6] indicate the typical positioning accuracy is shown in Table 2-2, with the main sources of error for DGPS systems being attributed to multipath error and horizontal error.

Error cause and type	Error without DGPS/SBAS	Error with DGPS/SBAS
Ephemeris data	1.5m	0.1m
Satellite clocks	1.5m	0.1m
Effect of the ionosphere	3.0m	0.2m
Effect of the troposphere	0.7m	0.2m
Multipath reception	1.0m	1.4m
Effect of the receiver	0.5m	0.5m
Total RMS value	4.0m	1.2m
Horizontal error (1-Sigma (68%) HDOP=1.3)	6.0m	1.8m
Horizontal error (2-Sigma (95%) HDOP=1.3)	**12.0m**	**3.6m**

Table 2-2 Positioning accuracy with/without DGPS[6]

To calculate the distance between the satellite and base station requires the calculation of the satellites orbital position at a specified time, t. This position is precisely computed using 16 parameters[3] contained within the ephemeris data within the satellites transmission. Using a mathematical model of satellite orbital dynamics, the satellites calculated position in space can be determined at time, t [2]. The ephemeris parameters for

each satellites is updated hourly[3] via a series of GPS ground control stations that track the satellites[2, 6], with the almanac parameters updated every six days[3].

RINEX formatting is the standard ASCII format for GPS data and while it has a larger file size than binary data, it provides a more flexible solution for distribution[3]. This format is comprised of seven file types, each containing a header and data section as follows[3].

1. GNSS observation data file
2. GPS navigation message file
3. meteorological data file
4. GLONASS navigation message file
5. Geostationary satellites (GPS signal payloads) navigation message file
6. Satellite and receiver clock data file
7. SBAS broadcast data file

Worldwide, a number of ground-based DGPS services are available with many countries offering multiple services[6]. Within New Zealand, Land Information New Zealand (LINZ) operates a series of GPS base stations throughout the country. This LINZ website claims to offer users with an accuracy approaching a few centimetres relative to the NZGD2000 datum, in both the horizontal and vertical planes. Real-time GPS data streams are available for use, allowing users to determine an accurate position in real-time without the need for their own base stations. This data is formatted to RTCM and NTRIP standards, though the LINZ webpage[20] does recommend that users be within 15km of the connected PositioNZ station.

2.1.4 GPS Formulas

Several formulas are needed for calculations in determining the users GPS position, in particular evaluating the pseudorange.

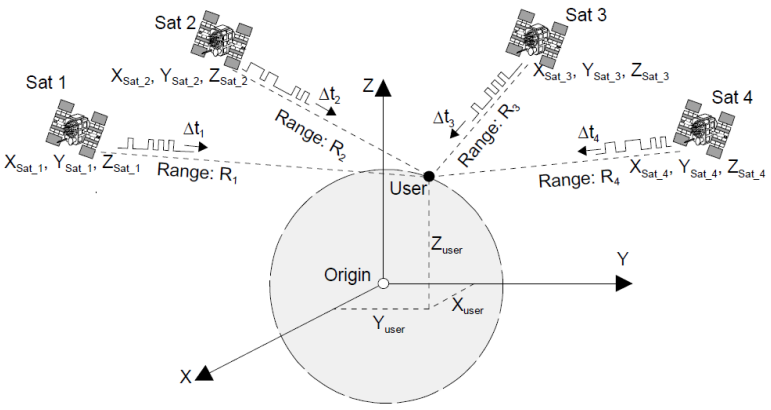


Figure 2-4 Three-dimensional Cartesian coordinate system for earth[6]

Pseudorange distance, R as calculated based on Cartesian coordinate system as well as signal travel time is shown in equation 2.12.1.

$$R = \sqrt{(X_{Sat} - X_{User})^2 + (Y_{Sat} - Y_{User})^2 + (Z_{Sat} - Z_{User})^2} = \Delta t * c \quad \textbf{2.1}$$

With the measured pseudorange distance, PSR, is represented by equation 2.2.

$$PSR = \sqrt{(X_{Sat} - X_{User})^2 + (Y_{Sat} - Y_{User})^2 + (Z_{Sat} - Z_{User})^2} + c * \Delta t_0 \quad \textbf{2.2}$$

c: Speed of light

Δt: Signal travel time to user

Δt_0: difference between the transmitting satellites clock and the receivers

Using the pseudorange formula in equation 2.2, as well as the known fixed coordinate of the base station as represented in Cartesian format and calculating the satellites Cartesian coordinates in space using the formulas found in 0 Figure 9-3, with variables from the satellites ephemeris data, a calculated Cartesian pseudorange can be produced. This Cartesian pseudorange can then be compared to the transmitted, to provide error adjustments. The received pseudorange can be accurately modelled by equation 2.3 defined in [12] with the variables defined in Table 2-3.

$$PSR = \sqrt{(X_{Sat} - X_{User})^2 + (Y_{Sat} - Y_{User})^2 + (Z_{Sat} - Z_{User})^2} + c\Delta t_r(t) + MP(t) + n(t) + c\Delta t_{sv}(t) + SA(t) + E(t) + c\Delta t_{ion}(t) + c\Delta t_{tr}(t) \quad \textbf{2.3}$$

c	Speed of light
$\Delta t_r(t)$	Receiver clock bias
Δt_{sv}	Satellite clock bias
SA(t)	Selective availability error
E(t)	Error in the calculated ephemeris
$\Delta t_{ion}(t)$	Dispersive ionsphere errors
$\Delta t_{tr}(t)$	Non dispersive atmospheric errors
MP(t)	Multipath error
n(t)	Random measurement noise

Table 2-3 Satellite Error Variables

The majority of GPS receivers only provide pre-processed data on the output, providing information such as positional, accuracy and satellite information. To retrieve the required satellite information necessary to analyse and process the signals and calculate the required data for signal error corrections, the GPS receiver must be capable of providing raw GPS data. This data is transmitted by the satellite is comprised of 25 frames each of which contains five sub-frames containing the satellite information, ephemeris data, almanac and other relevant data[6, 7, 10]. Parameters within these frames are stored as hexadecimal numbers, with the number of bits representing each parameter defined for each frame along with a scale factor to shift the decimal, information on these sub-frames can be found in [10]

Using the ephemeris parameters decoded from the GPS transmission, the earth centred, Earth fixed (ECEF) coordinates can be calculated as a X, Y, Z data point with the formulas outlining these calculations outlined in table 2 of [10], including all mathematical constants required on [10].

With the calculated satellite position and the known ECEF coordinated of the base station, the satellite range can be calculated using equation 2.1. This calculated range can be

subtracted from the measured pseudorange giving the error distance incurred from the sources listed in Table 2-3 and shown in equation 2.3. The accuracy of this pseudorange error is limited by the system's ability to remove the non-common mode errors associated with this value, which comprise of the receiver measurement noise and the multipath errors present on the received signal[12]. This provides an error correction for the atmospheric delays, which must then be applied to the infield GPS to provide the real-time corrected coordinates.

2.2 Autonomous Vehicles

Autonomy in vehicles has been available for a number of years, with cars capable of making driving decisions without human intervention, prime examples of these are cruise control and antilock brake systems (ABS) [2]. Rising population and rapid growth within the automotive industry has resulted in a rapid increase in the number of cars on the roads[1, 14]. In an attempt to alleviate this growing problem, governments have increased funding in areas that have allowed research institutes to research projects into intelligent vehicle systems[1].

Since 2004, several vehicles have been developed from the U.S. Defence Advanced Research Projects Agency (DARPA) grand challenge as well as the urban challenge, which have the capabilities to perceive environments, including off-road trails, roads and objects within urban areas[1]. Fundamentally, these vehicles are mobile robot platforms, with functionality determined by the onboard sensors that determine its capabilities in navigation as well as the tasks in which it can perform.

Development of autonomous vehicles has not been limited to self-driving cars, with tractors retrofitted for farming applications, allowing for improved crop sowing over long distances through GPS guided steering.

2.2.1 Key Sensor Technologies of intelligent vehicles

For a vehicle to operate autonomously it must first be able to sense its own state, thus it must be capable of measuring its speed, using wheel encoders or monitoring from GPS, engine status (RPM, temperature), and fuel supply[2]. To travel from point A to point B, the vehicle must know its relative position and direction of travel. In order to alter its direction of travel the angle of the steering wheel must be known[2].

A critical component of autonomous vehicles is object detection and avoidance. Therefore, the vehicle platform must be capable for the addition of a sensor system to detect the presence objects within its environment. Several sensor systems are used in order to detect these objects, including LIDAR (Light Detection and Ranging), radar, radio and vision system, which when processed are used to build a virtual representation of the environment surrounding the vehicle [1, 2]. LIDAR, radar and radio are known as active sensors, with vision classed as a passive. Different sensors provide varying degrees of accuracy over both short and long range, combining data from these sensors ensures robust sensing across various weather conditions including rain and fog[1].

Figure 2-5, shows the sensor layout for Ohio State University's (OSU) off-road autonomous vehicle that participated in the 2005 DARPA Grand Challenge, as described in [2]. Ultrasonic sensors are mounted to the left, right and rear of the vehicle to detect objects out of view of the main sensor array located at the front of the vehicle. The vehicle features three LIDAR scanners, mounted at varying heights of 0.6, 1.1, and 1.68 metres above the ground.

The first scanner scans parallel to the ground, the second is set to intercept the ground at a point 30 metres in front of the vehicle, with the final scanner set to intercept the ground 50 metres in front of the vehicle.

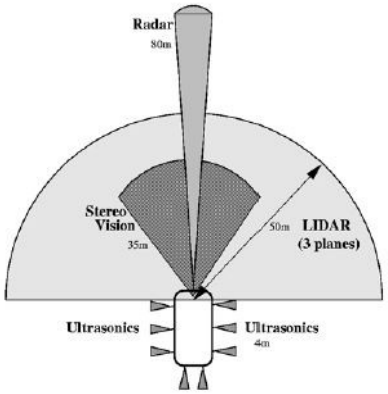


Figure 2-5 Sensor Overlay for the OSU ION off-road vehicle[2]

Though not shown on Figure 2-5, there is a fourth LIDAR scanner mounted 1.68 metres above the ground, orientated in a vertical position that provides an estimate of the ground profile directly in front of the vehicle. Radar provides long-range object detection, and an image processing system uses two cameras to analyze objects/terrain in view.

In order to avoid these objects, their detected position must be monitored over time, which can be equated to provide a vector for the object[1]. This vector allows the system to predict the future position of the object and undertake any necessary measures to avoid collision. This vector must be recalculated often as it may change over time.

Based upon the above research, the following sensor systems are required for a basic agricultural vehicle relating to three areas; vehicle monitoring, object detection and vehicle location. Vehicle monitoring covers everything to do with the operation of the vehicle. This includes speed (encoders/GPS), fuel levels (whether fossil fuels or electric), engine status (temperature, RPM) and physical control systems such as steering mechanism positioning, braking and throttle control. Object detection relates to the autonomous systems ability to detect objects around the vehicle that would hinder the vehicles ability to perform its task. This may be in relation to object avoidance so it can travel from A to B, which in an agricultural scenario would involve avoiding trees, people, animals and navigating through gateways.

Finally, vehicle location is the vehicles ability to know where it is in the world, either by measuring the distance and direction it has travelled from a know position, or with a GPS system. A combination of both of these provides a rugged system, with GPS accuracy affected by line of sight to satellites within the sky that can be blocked by trees and buildings. In these cases, measuring the distance and direction travelled can provide an approximation of the current position of the vehicle. This can be further improved with area maps containing the positions of fixed landmarks such as buildings and fencing posts that can be identified by range finder equipment to locate the vehicle on internal navigational maps.

This research focuses on the vehicle status monitoring and vehicle location estimation of a basic agricultural vehicle.

2.2.2 Path Finding and Obstacle Avoidance Methodologies

Path planning is key for autonomous vehicles, to ensure the vehicle is operated in accordance with rules specified for the environment, whether it be on roads within an urban environment or off road in an agricultural area[1]. In [2], the purpose of path planning software has been outlined as software that provides the coordinates for a traversable route which allows the vehicle to travel from point A to point B, avoiding any obstacles along the designated route.

For autonomous vehicles, path planning can be broken into two subtasks[2], global and local path planning. Global planning determines the best route to reach the destination using a known map, i.e. street maps for self-driving cars, breaking this path into small segments, which in relation to cars would be turning directions for streets. Whereas, local path planning is responsible for object avoidance as the vehicle travels through the individual global segments.

The base vehicle for the autonomous agricultural system should only be operating within areas that have been specified by the user. Because these areas are known to the system, maps can be produced that define the operational area, with fixed obstacles such as gateways, trees, buildings and terrain features being unchanged. Thus, the system can perform path planning based on these maps.

Two types of maps will be discussed as described in [2] covering Raster and Vector data maps. Raster maps[2], aka 2-D Occupancy maps, can be used with simple data processing techniques, though they have a high storage cost, with the map divided into a collection of cells, generally consisting of a uniform size. In comparison, Vector maps[2] require complex data processing methods to process the map data stored as curves, connected line segments, and discrete points. Vector data is generally used for on-road driving, with the complex data requiring less storage requirements than Raster data maps. The resolution of the maps will be limited by the accuracy of the GPS system, this as well as the small testing area that this unit will be operating in, makes Raster maps suitable for this system.

According to [2], Dijkstra's algorithm is generally considered base algorithm to calculate the lowest cost path between two points on the map. Path-planning algorithms used in video games use similar grid based maps environments, similar to Raster maps.

The algorithm as described in [2], follows this format:

1. Given some initial node, mark its value as 0 and the value of all other nodes as infinite;
2. Mark all nodes as unvisited.
3. Mark the given initial node as the current node.
4. For any unchecked nodes that surround the current node:
    a. Calculate its cost value as the sum of the value of the current node and the connecting edge and add it to a list of open nodes.
    b. If the current value of the node is greater than the newly calculated cost, set the value of the node equal to the newly calculated cost and store a pointer to the current node.
5. Mark the current node as visited; this finalizes the cost value of the current node.
6. If any unvisited nodes remain within the open list, find the lowest cost unvisited node, set it as the current node, and continue from step 4.

Other algorithms include, Voronoi diagrams[2], and A* (A-Star) search algorithm, with the Voronoi diagram deemed to be unfeasible due to its complexity when compared to Dijkstra's, where as the A* algorithm is derived from Dijkstra's and is classed as an off-road path planning algorithm due to its flexibility in applications to grid based terrain map[2].

The A* search algorithm outlined in [2] uses a grid based search method, employing a heuristic cost + distance function to determine the order in which grid cells are searched. The A* algorithm maintains two lists, an open and closed list, with the open list tracking all the cells have been in encountered but yet to be evaluated. The closed list tracks all cells that have been evaluated. Initially the closed list is empty, with the open list containing the initial starting cell, with each list item contains the cell position, the cell that preceded it (i.e. its parent cell) and the cost, $f(n)$, associated with that cell known as determined by the formula in equation 2.4.

$$f(n) = g(n) + h(n) \quad \textbf{2.4}$$

$g(n)$: The cost of travelling from the initial cell to cell 'n';
$h(n)$: The heuristic estimate cost of travelling from cell 'n' to the goal cell.

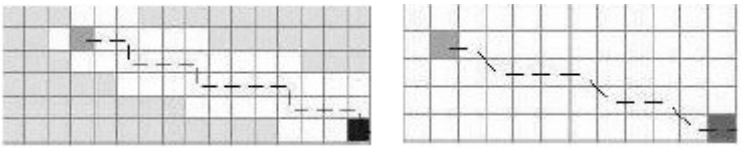


Figure 2-6 Illustration of Manhattan distance (Left) and Diagonal distance (Right)[2]

The heuristic cost, $h(n)$ defined in equation 2.4 can be represented as either the Manhattan distance or the Diagonal distance depending if the vehicle is capable of moving diagonally. The Euclidean distance is used for robots that can move in any direction, regardless of grid direction, corresponding to a straight-line path as illustrated in Figure 2-7 and is calculated using equation 2.5.

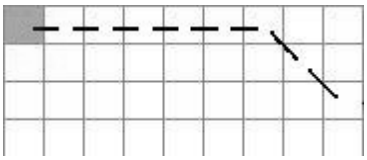


Figure 2-7 Illustration of Euclidean distance movement[2]

$$h(n) = \sqrt{\left(x - x_{goal}\right)^2 + \left(y - y_{goal}\right)^2} \quad \textbf{2.5}$$

The A* operates as a loop, evaluating the each item in the open list with the lowest f cost, its neighbouring cells are added to the open list (if not already contained within the open or closed list). If this cell is not the goal cell, it is added to the closed list, with the loop breaking under one of two conditions. The first being that the cell being evaluated is the goal cell and the second that open list is empty. Upon reaching the goal cell, the algorithm can produce a path linking the initial cell and the goal cell using the parent cells starting from the cell that encountered the goal cell. The algorithm returns a single path that connects the starting and goal positions, and should be rerun if new information is added to the map that would interfere with the calculated current path.

2.3 Sensing Technologies and Sensor Systems

For any autonomous vehicle to operate within a non-static environment i.e. an environment where not every object is fixed and in a known position, the system must have the ability to detect any object within the surrounding environment that could interfere with its operations. It must also be capable of positioning itself within its operating environment, as the use of GPS modules has already been discussed in Chapter 2.1, it will be excluded from this discussion.

2.3.1 Range Finding Sensors (Distance Measuring)

Several different range finding devices are available, with several of these used on the OSU ION autonomous vehicle mentioned in section 2.2.1. Four systems will be discussed within this section: Ultrasonic, LIDAR, Radar and Image processing, with each system differing in cost, accuracy, range and computational power requirements.

Ultrasonic sensors detect reflected sound waves emitted from the device, calculating the distance to the object reflecting the sound waves by measuring the time elapsed between when the signal was emitted and when it was received. Using this time and the known speed of sound within air of 330m/s, the distance from emitter to target can be calculated with the total time halved; this equals the time taken for the signal to interact with the object. These sensors are cheap, though they are short-range devices with a maximum object detection range of less than 10 metres measured in centimetre increments, and feature a wide detection angle. The accuracy of these sensors can be affected by environmental conditions including air pressure and humidity, which alters the speed of sound within air, though some higher quality sensors feature onboard sensors pressure/humidity sensors to automatically compensate for this change in the speed of sound.

LIDAR systems use a combination of a laser range finder and a rotating mirror to scan an area in front of the sensor, providing a range reading and angle for each reading. The resulting data points can be plotted to produce a 2-D cross section of the surrounding environment as demonstrated in Figure 2-8. Laser range finders feature medium distance object detection with narrow beam angles resulting in a higher precision measurement than ultrasonic sensors, particularly at long-range.

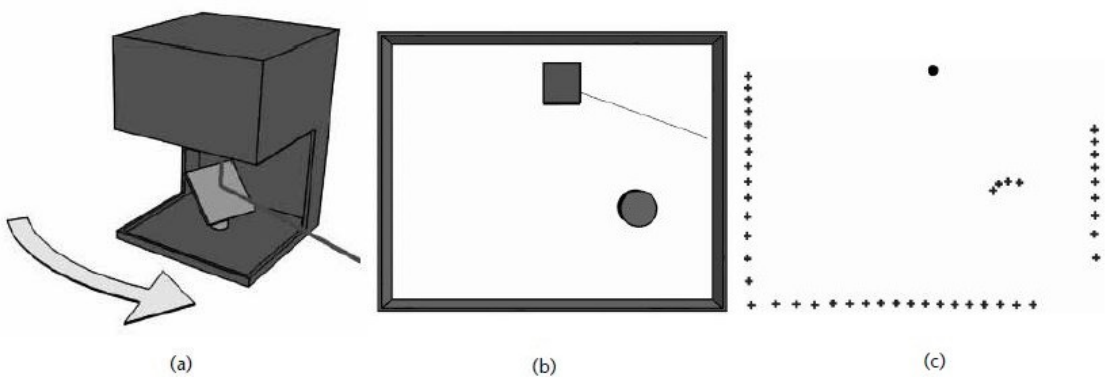


Figure 2-8 LIDAR Sensor System[2]

With the rise of hobbyist enthusiasts in robotics, equipment previously costing thousands of dollars has been redesigned and developed to tailor to this market. This has resulted in a number of products decreasing in price, including laser range finders, and while LIDAR systems are still priced out of reach of this projects budget, hobbyist laser range finders can be purchased for under $150 US. This can be combined with a mirror, motor and microcontroller can produce a low-cost LIDAR system that will be sufficient for the prototype system. As LIDAR sensors provide centimetre accuracy to objects within range, an area that had previously been mapped, with any objects with fixed positions such as buildings recorded on the map, would allow the system to detected its position on the map relative to these objects.

Radar provides a robust system for long-range object detection, and while providing significantly longer detection range when compared to LIDAR, though this system features a smaller detection angle as shown in Figure 2-5. Radar sensors are generally unaffected by environmental weather conditions, including rain, fog, snow and dust which would otherwise inhibit sensors system including LIDAR. Radar detects the reflection of radio waves from objects to compute the distance and azimuth measurements and are generally available at a lower cost than LIDAR systems.[2]

Image Processing is the use of cameras to detect the position on objects within the surrounding environment using two cameras operating in a similar fashion to binocular vision in humans and animals. By separating the cameras by a known distance, the two images captured by the cameras can be compared using the difference in pixel positions between the images. This with the known camera separation can be used to compute the range of the objects in view.

Image processing systems require significant processing to compensate for fish eye distortions in the image generated from the lens, as well as the calculating the range of all

pixels within the image at a rate in which the autonomous system can act on the data. The volume of processing required is dependent on the resolution of the cameras uses, with higher resolutions resulting in more pixels to compensate for distortions and range calculations, resulting in higher range accuracy for objects in view. Although, the effectiveness of this system is reduced depending on environmental conditions, extreme lights, rain, fog, etc.

2.3.2 Positioning and Orientation

As previously mentioned, the vehicles ability to locate its position and orientation within its environment is critical in enabling it to act autonomously for navigation. Methods of positioning though the use of GPS sensor systems is covered in section 2.1, as well as the use of range finder positioning sensors in conjunction with a map of the area indicating the position of all fixed objects. This section will explore the use of three or more point triangulation systems, RFID tags for area classification and magnetometer compass for heading detection.

Triangulation systems use three points to provide an extremely accurate position overlarge areas using simple calculations, with one paper[15] showing a resulting worse case error in the order of 10 cm through testing within a 2500m^2 area. This paper[15] measured the observed angle between the three known points via a optical scanner, and triangulated the position as shown in Figure 2-9.

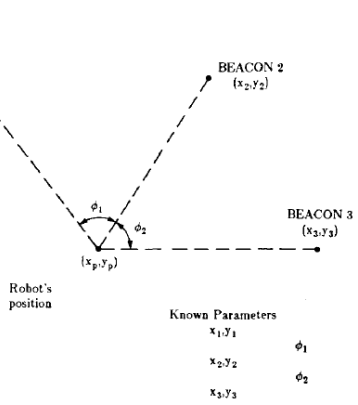


Figure 2-9 Positional Triangulation[15]

This could be feasible within an agricultural environment within open areas so long as line on sight was maintained to the beacons. Another ground based triangulation method is to use synchronized radio transmissions containing a station position and time of transmission, producing a system that operating using a similar principle to GPS. This system would have several advantages over GPS, mainly involving reduced interference from atmospheric sources, though it would not provide significant improvements over a differential GPS system that used phase correction.

RFID allows information to be stored electronically on tags, which can be read wirelessly using the appropriate reader and has been used in experiments to increase the accuracy of GPS positioning systems with tags embedded within roadways[16]. While it is impractical to embed these tags within the ground within fields to improve positioning, they could be installed to label gateways and rows within orchards.

For a vehicle to self-navigate from one point to another, orientation of the system must be known. A GPS system can provide the systems orientation from the direction in which the system is travelling, but requires a GPS fix in order to determine the direction of travel. Mapped landmarks detected via LIDAR sensors can provide a direction of travel for the vehicle, though the simplest method is using a magnetometer to determine the direction of magnetic north. This heading must be compensated to account for the variation between magnetic north and true north, according to the GNS Science webpage, the magnetic declination for the Palmerston North area between 21.5-22 degrees bases on Figure 2-10. To ensure that the measured compass heading is accurate, the system must compensate for errors induced on the sensor output caused by the sensor angle to the planet's surface.

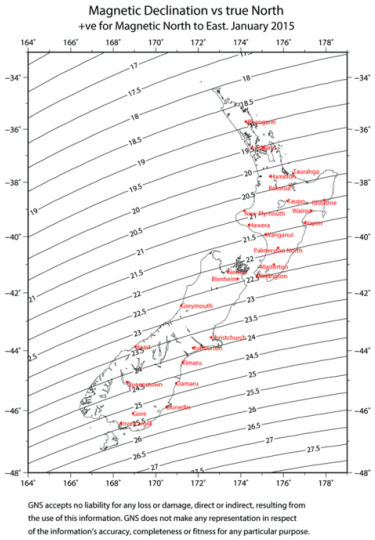


Figure 2-10 Magnetic Declination verses True North in New Zealand, January 2015[24]

Chapter Three Autonomous Farming System Development

Farmers have been facing labour shortages with a lack of skilled workers present within the farming industry, with a shortage of New Zealand workers that are willing or able to work on farms[33]. Organisations like the Federated Farmers of New Zealand work with government officials to grant work permits to skilled overseas workers, based on a government controlled skill shortage list[33]. Unfortunately, most agricultural work does not align with this list, leaving farmers with difficulties in finding skilled workers for farms, with dairy farm workers not qualifying for this list[33]. Because of this, the dairy industry has struggled to find an adequate supply of workers for current labour needs[4].

For the existing workers, time is spent performing tasks that do not require anything more than rudimentary skills, such as rounding up livestock, spraying fields, and inspecting crops. As these tasks only require a simple skill set, the skilled worker could be replaced with an autonomous system, which is designed to perform a series of pre-programmed tasks. These tasks, whether it is to travel to the location or used to carry a sprayer, are generally preformed using a quad bike at some point in the process and maybe preformed at any stage of the day or night.

Although these tasks maybe simple in nature, the requirement of a quad bike can complicate the task during certain weather and terrain conditions, with reduced visibility due to fog and heavy rain making navigation within the environment difficult. Steep and uneven terrain can result in injury and death to users riding quad bikes, due to the repositioning of the vehicles centre of gravity when ridden by a human, with 850 injuries every year caused by quad bikes in New Zealand[19]. Additional tasks including soil sampling are rarely undertaken as the high cost associated to both time and equipment make this task unfeasible to be conducted on a regular basis. This leads to a generalized application of fertilizer to crops and fields, which causes a nutrient run off and pollution of waterways.

These problems not only reduce the productivity of farms due to a lack of workers, they can cause injury to works and pollute waterways. To combat these affects, smarter farming techniques need to be developed, namely with intelligent autonomous vehicles that will reduce unskilled workload of farm workers by automating specific tasks on the farm. These vehicles can operate in a variety of weather conditions that would normally be uncomfortable and difficult for workers to operate within, and allow precision farming methods to be implemented.

This precision farming would allow for the accurate measurement and application of nutrients to patches of soil, as well as the precise application of sprays such as weed killers, to plants. By limiting these products to the required areas, the costs associated to the application of these products will be reduced, as well as the associated run of in to waterways cause by superfluous products. Night-time farming can also be performed autonomously, when daytime conditions such as rain, high winds, and intense sunlight may not be present. This would otherwise not be possible or desirable due to variation of working hours adversely effecting workers.

3.1 Proposed Autonomous System

This research aims to develop a base platform for an autonomous vehicle specifically for farming, featuring a modular system design that will allow for flexible integration into existing vehicles. Each of these modules is designed for a generic vehicle that offers the most basic functional system, unassisted steering, manual gear control, etc. Basing the design on a generic vehicle allows for simplified integration into alternative vehicles; with only minor changes required to adapt these modules to the vehicles system. The modular design can also be extended to augment the vehicles functionality, using interchangeable sensor and units to tailor the functionality of the autonomous system to the user's requirements.

A stable platform is a key feature for a farming vehicle while traversing over rough and uneven terrain. Farming vehicles primarily use wheeled designs featuring three, four or six wheels. To ensure the maximum possible platform stability on hills and over rough terrain, with four or six wheeled designs providing the maximum stability. Traversing rough terrain can prove difficult for some vehicle types, with loss of ground contract for the driven wheels due to terrain geometry on two-wheel drive systems. While the proposed system should not unknowingly encounter these terrain obstacles, a four-wheel driven platform increases its ability to navigate through rough and muddy terrain.

Using an existing vehicle such as a quad bike will reduce the design and implementation costs and increase the likelihood of farmers implementing the proposed modular systems. For autonomous vehicle navigation, the entire system must be capable of automatically controlling all the required manual interfaces on the vehicle including starter switches, light and horn controls. While this vehicle is an autonomous system, it must also allow for manual vehicle control, particularly where a farmers vehicle has been converted. By allowing the farmer to operate the vehicle manually, a simpler system setup is achieved by manually driving the vehicle to set its operational area, while also improving on the vehicles safety with an autonomous override if the vehicle is operated dangerously.

The purpose of this research is to develop an autonomous vehicle for the agricultural industry, with a specific focus on modular design. For this initial design, the system design focus is on three primary modules, which are speed, steering and gear changing modules. The modules will be designed for a generic vehicle, allowing them to be applied to various vehicle types and models with little to no modification. To operate autonomously, the system must realise the controls that a human operator normally controls in order to navigate the vehicle within its operating environment.

The autonomous system operating on the proposed platform will be designed to self-navigate between points in order to perform predefined tasks, and will use a self-generated map of the surrounding environment. In order for this system to operate autonomously within an area, the user will be required to perform an initial setup for the operational area. This will require manually operation of the vehicle around the area in which it will operate, with the system generating an internal map based upon the measured GPS coordinates outlining the boundary as well as any obstacles it encounters.

The intension of the system is to provide a mobile platform in which individual modules can be attached to augment the functionality of the system, allowing the system to

be tailored to the requirements of the farmer, and reducing the cost associated with implementing the system. An existing second-hand farming vehicle was chosen for modification act as the platform for this system. Using an existing farming vehicle, with a proven history of service within the industry as the systems platform, ensures that the system will be mechanically reliable within the intended operating environment. This will also provide farmers with a familiarity towards the system and allow the farmers existing vehicles to be retrofitted for this system.

This should result in a greater uptake from the farming community, with personalized functionality allowing the user to select only what they need from the system. Therefore, this will minimize the cost of implementing the system through functionality selection as well as by retrofitting an existing farm vehicle.

3.2 Design Considerations

To ensure platform stability, the vehicle should have a minimum of four wheels, along with a low centre of gravity to prevent rolling when traversing sloped surfaces. A second-hand quad bike that features a manual gearbox and four-wheel drive was selected as a base for the platform, with the reasoning behind this selection outlined in section 5.1.

In order to operate autonomously the bike requires modification, with mounting points added for the steering and gear changing mechanism. The steering mechanism is designed to bolt to the existing steering column mounts with no additional modifications besides a mounting bracket attached to the bikes frame. The gear changing mechanism requires a plate to be welded to the frame in which the gear changing mechanism will be attached, with the position of the plate set using a temporary guide plate. Modifications to the existing braking system to include braking sensors and a method of actuation that allows the brakes to remain fully function to the user.

3.3 Expected final system design and features

3.3.1 Features of the mechanical System

The mechanical modification and components designed for the implementation of the system and its testing is outlined in Chapter Five, featuring designs to control the vehicles steering, speed and gear changing controls. The mechanical steering system is capable of travelling between the mechanical limits of the bike, and features a manual electronically controlled manual mode providing the user with full control when required. The position of the bikes steering system is tracked regardless of whether the system is being manually or autonomously controlled.

The gear changing mechanism features a design that allows manual operation of the gear leaver by the user at any time in case of system malfunction, with the system designed to change gears automatically based on the load on the engine. A preload feature provides the system with quicker gear change by reducing the gear actuation time of the gear system by up to half. The speed control system features a servo controlled throttle cable and a linear actuator to operate the modified foot brake pedal. The position of the braking actuator does not impair the user ability to apply either the front or rear braking controls, with sensors positioned to detect the application of the braking controls via either the user or braking

actuator. This maintains the user ability to manually actuate the brakes when required, with a sensor integrated into the existing handle throttle control providing the user with a manual control mechanism for the vehicles throttle.

3.3.2 Control Systems and capabilities

The vehicles autonomous system consists of numerous sub-control systems as shown in Figure 3-1. These systems are present on-board the autonomous vehicle, with each mechanical system controlled by an individual control device that perform specific tasks based on the commands received from the vehicles central management controller. The vehicle controller developed links the mechanical modules, controlling them through pre-set commands based on the specified inputs from the navigation system. These commands are also determined by the outputs of each module, with the speed, steering angle and gear change status interact in a mutually inclusive manner, in which the output of one can affect the input command to another. An example being the restriction of the vehicles current speed based on the current steering angle of the system, with the steering angle also affected by the angle of the vehicle particularly on sloped surfaces.

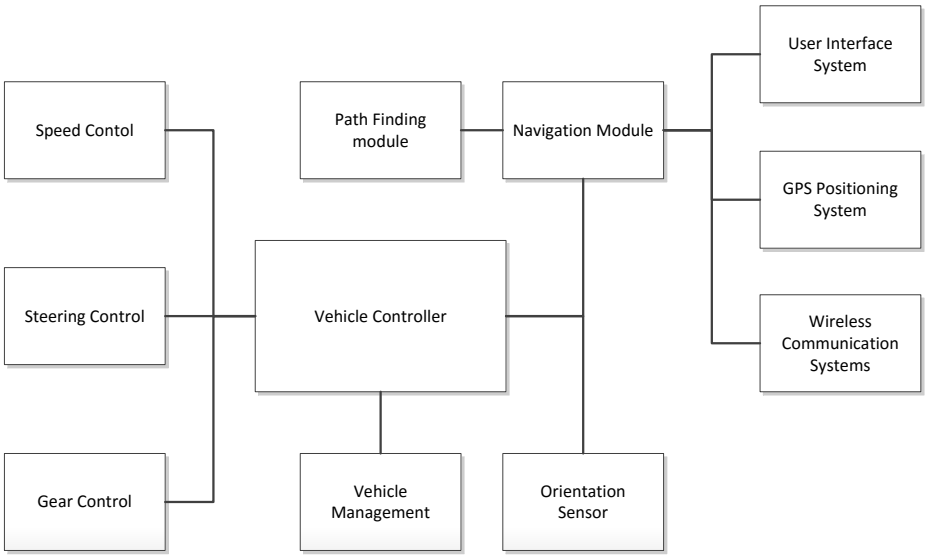


Figure 3-1 Autonomous System Diagram

3.3.3 Navigation Module and functionality

The Navigational module determines the vehicles physical position and orientation within its operating environment with the details of the surrounding area stored as a digital map. This digital map contains the terrain details and obstacle information for its operating environment, allowing this module to determine a traversable path connecting its position and destination. This generated path is simplified into a series of straight lines, producing the minimum number of nodes that the vehicle will have to traverse with a heading for the next node sent to the vehicle controller.

The navigation module is designed to receive information from sensor systems about any objects detected near the vehicle, this increases the accuracy of the vehicles detected position using landmarks as well as locating previously unknown obstacles or obstacles that

have moved from their previous position. If the navigational map is modified, the system will recalculate the vehicles route to ensure a traversable path is still available for navigation.

3.3.4 Monitoring system and features

Remote system monitoring allows the user to check the vehicles position and speed along with the status of the vehicles onboard systems and fuel status. Externally mounted sensor modules can be attached to provide monitoring for agricultural and farming land, providing soil nutrient information and moisture levels as well as information on grass levels with the possibility of additional visual systems to detect unwanted weeds and to count fruit yields on trees. Additional sensor systems such as those described above, will augment the functionality of the final vehicle platform developed as a part of this research.

Chapter Four Development of a Low Cost Differential GPS System

For an autonomous robot to operate within any environment, a system must be put in place that allows the robot to determine its position relative to known points, within its operating environment. This robot will primarily be operating in an outdoor environment, within fields, though the system must be capable of operating with other areas such as orchards as well as indoor environments such as sheds.

For successful operations within an outdoor environment, the system must be capable of accurately detect its location within an area that may contain obstacles. Therefore, a positioning system with a high degree of accuracy is required to allow the robot to, successfully and reliably navigate fixed obstacles such as fences, trees and buildings. The simplest and most reliable system that can provide accurate positioning over a wide area is using GPS.

Multiple types of GPS systems are available, with standard standalone GPS systems found in cars, phones and other devices provide a positioning accuracy[1] is about 10 metres[13]. While these may be sufficient for handheld navigation devices, autonomous robots require a much higher positional accuracy to ensure sufficient resolution to navigate tracks and avoid obstacles within the operating environment. Depending on the size of the vehicle, the minimum clearance between the vehicle and obstacles would vary, with larger vehicles such as trucks requiring a larger clearance than a quad bike or small tractor.

This system is designed to operate by using a map of its operational area, containing any obstacles detected or marked within the specified operating area. To avoid these objects, their exact position must be known so the vehicle can navigate to past these obstacles to reach its destination.

For a vehicle operating in this environment, a minimum repeatable accuracy of at least 20 centimetres for basic object avoidance, as well as providing sufficient resolution for repeated sensor measurements. However, an existing system that could provide this level of resolution would exceed the projects budget, costing upward of US ten thousand dollars. Therefore, the development of a differential GPS system that would fit within the projects low cost constraints was attempted, with an acceptable accuracy in the order of one metre. While this would not be a particularly accurate positioning system, the positioning of the vehicle in terms of object avoidance could be enhanced to provide sufficient resolution to navigate through gateways and other obstacles through the addition of additional sensors.

This object avoidance resolution would be enhanced with the integration of range finder sensors to detect objects at short to medium range, that the system was unaware of within its maps and allow increased precision in object avoidance and navigation. While this resolution will provide a sufficient level of accuracy for basic navigation, it would not provide sufficient resolution for work such as soil sampling, where the unit would have to return to the

[1] This accuracy quoted as the CEP (Circular Error Probability), which is defined as the radius of a circle centred on the true position. This value is based on a 50% probability, with the receiver with a 10-metre CEP accuracy will be within 10 metres of the true position 50% of the time[34].

same position to collect samples. Therefore, future development should include a system with a resolution of 2 centimetres to provide the level of accuracy required to perform a variety of tasks, so long as the vehicle was capable of physically positioning the sensor to this accuracy.

In recent years[2], there have been advances in the algorithms that the GPS modules use to calculate their position; this has allowed new modules to provide unaided accuracy to within 2.5m, which is suitable for many outdoor applications. However, a GPS system that provides sub-meter accuracy is needed to allow the platform to navigate narrow gaps between obstacles. To achieve this level of accuracy there are several different types of GPS that can do this: Military Grade (Dual Band), Geostationary and Differential GPS Systems.

Differential GPS (DGPS) is the only system that is widely available, with Dual Frequency reserved for military use only and Geostationary only available in areas where a satellite is in Geosynchronous orbit over the operating area (currently only in the United States and some parts of Europe). Differential GPS provides localized corrections to a specific area with a radius of the systems base station, providing accurate error corrections up to a distance of 50km[12] from a base station.

Several components are required for a differential GPS, two GPS receivers are required, a base station GPS in which to make the corrections and a rover receiver to implement the corrections, providing enhanced positioning. A processing unit is necessary at the base station to compute the satellite corrections, with a long-range data transmission system to deliver the RTCM corrections for real-time corrections.

4.1 GPS Chip Considerations

As outlined in section 2.1.3, most GPS devices available are not capable of transmitting the raw data transmitted from the satellites that is required for differential error calculations. Therefore, a GPS chip that can provide raw GPS data is a requirement of the system. Antenna selection is important for GPS systems to ensure the chip is capable of receiving the data streams transmitted from the satellites. GPS receivers operate using either active or passive antenna, with active antennas including a low noise preamplifier to boost the signal strength[6].

Three types of antenna are available as described in [6], patch antennas are the standard antenna type, helix antenna have an increased directionality than patch antennas which may reduce the effects of multipath, though this will also decrease the receivers field of view. Finally, on-chip antennas provide a small, cheap antenna solution, which is built on to the GPS board, though they are less effective than patch antennas. For this research, patch antennas were decided upon due to their signal gain and relatively large field of view when compared to the helix antennas.

Several components are required for a differential GPS; including two GPS receivers, where one receiver must be capable of RAW data output and will remain in a fixed position. The second GPS receiver must be capable of either accepting satellite correction data stored in

[2] From the year of this thesis, 2016.

the RINEX format, or must be capable of RAW data output to allow an external processor to calculate the adjusted GPS position.

The u-blox series of GPS receivers was selected for this project, featuring a wide range of standard and high precision GPS chips as well as a series of active antennas, which have been designed and tested with their products. Selection of the base station GPS was limited to those capable of RAW data output which operate using active patch antenna. Two modules were available which meet these criteria, the NEO-6P and the NEO-6T. Both modules are rated with a CEP value of 2.5 metres, therefore 50% of the calculated positions will fall within 2.5 metres of the actual position.

Apart from cost, the primary difference between these two GPS modules is the NEO-6Ts stationary mode feature. This stationary mode allows the module to obtain the satellite clock time from a single satellite, while this is not particularly useful; one supporting sub-feature for this is Survey-in. This survey-in procedure is used to determine the stationary receiver's position by averaging the all-valid 3D position solutions. This averaging continues until two stop criteria are met; these criteria being the a minimum observation time, with one day recommended for high accuracy, with the second criteria determined by the standard deviation of the 3D position.

The NEO-6P offers the same positional accuracy and RAW data output capabilities as the 6T, but without the survey in feature. While useful for quick and accurate base station setups, the lack of this survey-in feature will not affect the performance of system. The NEO-6P is capable of accepting RTCM files, applying these corrections to the position solution without the need for external processing. The NEO-6T is the preferred GPS module for the base station due to its survey-in feature, and although several of the cheaper U-blox GPS modules accept RTCM files for corrections, the NEO-6P was chosen as the rover receiver due to its high level of accuracy without correctional data. This will allow the system to have a level of redundancy in the event of system failure, it should also be noted that U-blox receivers have a time out feature for RTCM corrections, meaning that the receiver will only use correction data that is

4.2 Control System Selection

A control system is required, in which the RAW GPS data is processed into differential corrections, packaged in a RTCM format and transmitted to an in-field rover, to be applied to the rovers GPS module. For DGPS to be effective, the correction data used must be updated at least once every 15 minutes[5] to ensure accurate pseudorange compensation. Therefore, the control and communication system of the DGPS must be capable of processing and transmitting and applying the data at a sufficient rate to ensure the correction data is not timed out by the U-blox receiver as well as maintaining a high level of accuracy.

The selected NEO-6P module features several interface options, with UART, USB, SPI and I^2C compliant DDC available. The chip requires a number of support components for the active antenna. To simplify the prototype system design, an interface board was sourced, providing the USB connection capabilities, as well as the required hardware for the U-blox active antenna. Power and interface to this board is available via the USB connection, with alternative 5-volt power, I^2C and serial interface pins provided.

With the three interface options provided via the GPS periphery board, a number of suitable controllers are available, with the simplest interface option provided via the USB connection. There is no way to measure the required processing power required for the differential calculations, therefore, a method must be devised in which an estimate can be obtained. In order to estimate this required processing power, a differential GPS program was first developed on a desktop computer, allowing the data conversion and calculations to be tested, and providing an estimated processing time based on the code runtime.

However, as the differential calculations will be based purely on pseudorange corrections, any calculations should be relatively simple as opposed to the carrier range compensations, as outlined in section 2.1.3. While the pseudorange compensation will only provide an expected accuracy of a metre, it has been decided that this would be sufficient for any navigational purposes, as the resulting position of the vehicle can be augmented using rangefinder measurements from surrounding objects.

A Raspberry Pi single board computer was selected as the test controller for the system, featuring USB connections as well as digital and serial pins. This controller provides the necessary input-output connections for the GPS interface board, as well as an 800 MHz processor. This processor speed was deemed sufficient for the prototype testing, as the processor speed of differential systems when first developed in the early 90's would not have exceeded this speed, with processors only exceeding this speed during the late 90's.

4.3 GPS Communication System Design

In order to apply the corrections to the infield rover, the system must have a method of transmitting the RTCM files from the base station to the rover over long distances, with the effective range of differential corrections being up to 50 km[12] from the base station, outlined in 2.1.3. Three communication systems were considered for the system, Wi-Fi, Cellular network and radio modem, all of which provide long-range communication capabilities.

Wi-Fi can provide global transmission capabilities through internet access, though it can have a significantly shorter range of up to several hundred metres for direct peer-to-peer transmission. Providing data transfer rates of up to 300 MB per second, the system would offer sufficient bandwidth to prevent bottle necking in the transmission of differential corrections. This system would be suitable in areas where an existing Wi-Fi network was in place, as a significant investment would have to be undertaken to deploy a Wi-Fi network over a large area.

Cellular networks are widely available within New Zealand and can be used for by devices for internet access, providing a method to upload and download the differential corrections produced by the base station. This system could provide problems with coverage in remote farming areas, with many remote areas not covered, or have low signal/data transfer rates. Data transfer rates on cellular networks can be significantly affected by public usage; causing unreliable delivery times for any data transferred, and could result in decreased accuracy from the differential corrections. Using this communication method will also incur ongoing monthly data charges to the user based on the volume of data sent and received from between base station and rover.

Radio modems offer long-range communication capabilities over the 900 MHz and 2.4 GHz frequencies, with 1W 900 MHz modems capable of up to 40 Km line of sight transmission capabilities. This is within the 50km effective limit for the differential system, though significantly shorter transmission capabilities should be expected on within New Zealand agriculture, with most farms featuring rows of trees as windbreakers. According to the Radio Spectrum Management (RSM), a business unit within the Ministry of Business, Innovation and Employment (MBIE), the 900 MHz frequency is a restricted frequency due to its use for cellular networks, with unlicensed devices restricted to a power output of up to 0.0318 Watts[17]. This would significantly reduce the operational range of the DGPS, though several 900 MHz modems offer channel selection. This enables the modem to operate on higher frequencies, with the amateur 921-928 MHz frequencies licensed for up to 1-Watt transmitter power[17]. The data bandwidth of these systems is significantly lower than that of the previous two systems, with data transmission rates of up to 250 Kbps.

Thought the data transfer rate of the radio modems is significantly lower than other methods, the reliability and low cost of the system is preferable, the expected maximum travel distance for this rover from the base station should not exceed a few km within the prototype testing. Testing of the modem will be required to ensure the data transfer rates for the system are sufficient. A RFD900 radio modem was selected, featuring an adjustable frequency range 902-928 MHz at 1-Watt with data transfer rates of up to 250kbps. Data is sent and received from the modems via a serial interface and can be used to establish a multipoint network system to extend the systems communication range.

4.4 Differential System Design and Programming

Based upon the previous sub chapters, the following block diagram, Figure 4-1 Differential GPS function diagram, represents the requirements for a differential GPS system, broken into its two component parts, the base station and Rover. The RTCM differential corrections can be directly fed into the rover GPS module via the serial port on the interface board. While this would decrease the time taken for an intermediary processor to receive and forward the data from the radio modem to the GPS module, it would remove the ability for the radio modem to be used as a command interface between a remote user and the rover. Therefore, the RTCM corrections will be applied via the rover control system, instead of directly from the modem.

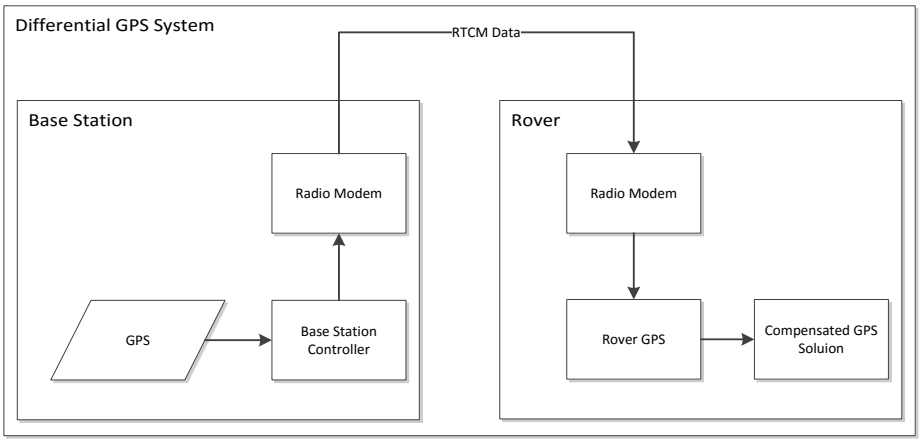


Figure 4-1 Differential GPS function diagram

In order to calculate the satellite pseudorange corrections, the position of the satellites at time of transmission must be calculated using the equations contained within 0. The variable values defining the orbital position of the satellite, are stored as hexadecimal numbers within the ephemeris subframes of the satellite transmission.

Splitting the RAW data supplied form the NEO-6P GPS module into its component variables and scaled according to Figure 9-2 within 0, provides the relevant variables to calculate the Cartesian ECEF coordinates of the specified satellite at the time of signal transmission. The true pseudorange is calculated from these calculated satellite coordinates with the base stations known coordinates according to equation 2.2 in section 2.1.4.

The resulting calculated pseudorange values can then be used to produce the resulting error from the received pseudorange according to the following formula based on equations 2.1 and 2.3 in section 2.1.4. This error is comprised from satellite, receiver and atmospheric sources, Table 2-2, with the variables from equation 4.2 described in Table 2-3 in section 2.1.4.

$$Error = PSR - R \quad \textbf{4.1}$$

$$Error = MP(t) + n(t) + c\Delta t_{sv}(t) + SA(t) + E(t) + c\Delta t_{ion}(t) + c\Delta t_{tr}(t) \quad \textbf{4.2}$$

Results from this system were unable to be verified, with the error values produced from the pseudorange calculations in the order of kilometres. Due to this high error value, a problem within the calculation method used was suspected; therefore, due to the unavailability of expert guidance within this field, the development of a DGPS was suspended. This was to ensure that other aspects of the project were not adversely affected by the increased time spent on this system, with the 2.5m CEP accuracy of the NEO-6P module deemed sufficient to continue with the project development.

Future development of this system should include a choke ring antenna to reduce possible multipath errors influencing corrections generated by the base station[12], a NEO-6T or similar GPS module should be used to provide an easy and accurate setup of the base stations position without the need for a survey to be preformed.

Chapter Five Mechanical Systems Implementation and Testing

The purpose of this research is to develop a smart system that can be implemented to deliver an autonomous vehicle for agricultural applications. To achieve this goal, a vehicle must be either designed, or modified to suit, while providing a platform capable of providing the functionality needed for the vehicle to operate autonomously. Since the main objective of this research is to develop a smart control system to operate a vehicle in an agricultural environment and several farming vehicles are available that have been proven reliable in the required operating environment, modifying an existing farming vehicle for the deployment of the smart control system becomes a feasible and economical choice.

5.1 Vehicle Considerations

Considerations to select an existing vehicle to were based on the following criteria: reliability, durability and size. It should be noted that if a suitable vehicle were not found, a vehicle would be designed and produced to meet the system requirements.

The first limiting criteria of the vehicle is its size, by ensuring the size of the vehicle was kept to a minimum, the system would be capable of providing an increased range of functionality. The small size would be particularly useful around trees, gateways and orchards. Within New Zealand, Kiwifruit orchards are abundant with the vines laced overhead into a canopy. According to Philip Martin (personal communication 30 September 2015, Appendix K) from Plant & Food Research Limited stating that typical dimensions for mature kiwifruit blocks have a canopy height of 1.8 meters with a row width of 3.5 metres. Therefore, a vehicle used in this environment would be required to have a small vertical height. While this system is not designed to specifically for Kiwifruit orchards, the use of a smaller test vehicle was deemed not to reduce the functionality of the vehicle, except in the area of towing capacity.

Farming vehicles within New Zealand have be capable of meeting the rough requirements of the local landscape, with the Quad bikes, Mules and Tractors used within the New Zealand farming industry meeting these requirements. Working within the New Zealand environment requires high mechanical reliability, with any vehicle used by farmers that provides a simple mechanical design will provide the best platform, therefore an older model of such vehicles that has had proven reliability within the industry is required. Older models also provide a familiarity within the industry, and farmers know how to us, repair and maintain these vehicles.

These older models also provide more of a challenge, with newer models including on board computers, which while offer existing interface and sensor measurements, also provide reduced reliability as these parts make trouble shooting more difficult. Therefore, a system that is as basic as possible is required to cover the widest range of vehicles that this system could be applied to, and thus a manual gear changing system should be selected.

For this platform to operate effectively within the intended agricultural environment, it must be capable navigating rough terrain, including potholes, shallow ditches and small to moderate inclines, all while providing a stable platform on which this system can control and operate. To ensure maximum stability a vehicle featuring at least four wheels should be used and be outfitted with a drive system capable of providing sufficient power to navigate within

the expected operating environment; this includes sufficient power to tow additional equipment such as sprayers and other equipment.

The vehicle should be capable of performing sustained operations for several hours; therefore, a petrol driven vehicle is preferred as it can undertake long periods of operation out in the field away from external power sources. Petrol motors also generate onboard power via an alternator, which ensures power to the control system will be maintained, though this power may be interrupted for short periods during high power draw applications, such as starter motor activation. However, petrol driven systems cannot be operated within enclosed indoor environments over long periods due to possible health risks to humans and animals.

5.1.1 Existing Farming Vehicles

Three types of vehicles commonly used within the New Zealand farming industry are quad bikes, Mule and tractors, with the size constraints limiting the tractors available for selection. There are also a number of self-driving tractor systems available on the market, these could be integrated into this system and would reduce development time.

Quad bike (Figure 5-1) and Mule (Figure 5-2) provide smaller system platforms with the advantage of a tighter turning radius over larger vehicles. The smaller vehicle size ensures increased manoeuvrability due do the reduced distance between the wheel axles. Both the mule and quad bike will act as suitable platforms for this system, though the wide availability of second hand quad bikes was the deciding factor in the vehicle selection.

Figure 5-1 HONDA TRX250TM[21]

Figure 5-2 KAWASAKI MULE™ 3010[22]

Quad bikes are abundant within the New Zealand farming industry, particularly within livestock farming and are used for the spraying of fields and orchards. Quad bikes have an advantage over mules due to their long history of use, providing users within the agricultural industry with a familiar platform to operate and maintain. This also allows users to upgrade their existing vehicles, reducing the initial system cost, which will likely increase the uptake of any system developed. Because of the manual control capabilities that this system will maintain, farmers will not have to lose this existing piece of farming equipment and if a new integrated system were obtained, the user would gain a new vehicle entirely.

There are a number of other advantages to this system in regards to user safety, with the system providing safety enhancements to the quad bike, including speed control depending on terrain conditions and cornering angle.

5.1.2 Selected Vehicle

A second hand quad bike was considered as the platform for this research, and provided a low cost vehicle frame to build on the desired system. A 1995 model, TRX-300 was purchased for this system. The motor and electrical components were still in working condition. The TRX-300 features fixed four-wheel drive with a manual gearbox, though this vehicle required extensive servicing to ensure it would be mechanically reliable. The bike features a 12-volt battery to power the on-board power systems, which is charged by an alternator when the engine is operating.

This servicing primarily involved repairing rusted out sections of the vehicle frame, with significant damage to the steering, rear deck tray, as well as many of the mounting struts. Parts serviced excluding the chassis framework included; the carburettor, front and rear brakes, rear axle, steering column and engine. This servicing ensured that all parts operated correctly and the system was structurally sound.

5.2 Steering Control Mechanism

The steering set up on the TRX-300, uses a steering pole (part 1, Figure 5-3) from the handlebars (A) to connect to the wheels via the steering linkage (part 12) to the knuckles (part 8) on the wheels. To reduce costs on the project a belt system was used instead in favour of a geared design. This was primarily due to the low speed the system would require to be running at, as well as the customization that the pulley system offered as explained in the following sections.

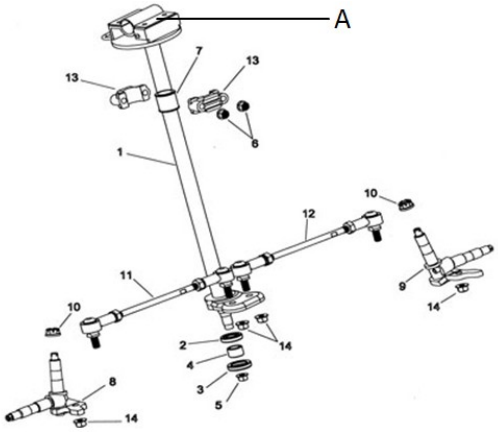


Figure 5-3 TRX-300 Steering assembly[23]

To provide adequate control over the positioning of the vehicle steering mechanism, a system must be in place to detect the position of the handlebars, and thus the angle of the vehicles wheels. To do this an encoder is required on the vehicles steering mechanism to detect the wheel angle, there are two methods of doing this. The first is using a rotary encoder on the steering column to measure the angle on the steering pole; the second is to use a linear encoder to detect how far the steering linkage moves when the handlebars are moved. Though no matter which option is chosen an absolute encoder was used, in which every readable position in a 360-degree rotation has unique value.

An incremental encoder, outputs a specific number of pulses per rotation, and while useful, they need to be homed to a specific position to be used in an application such as

steering position whenever they are powered on. This is due to incremental encoders only providing a set of pulses that when combined with a microcontroller can be used to determine a distance/angle travelled and the direction travelled, while an absolute encoder provides its position in the form of a specific value independent of being homed.

Another function that this steering system needs to provide is to enable the user to steer the vehicle manually when required. This manual steering capability will allow users to manually control the vehicle and allow the setup of the systems operational bounds within its environment, as well as to allow normal use of the vehicle when needed.

Two possible methods of controlling the steering angle that were considered; these were either by directly turning the steering pole, or by using a linear actuation system to move the steering linkage. The first method of rotating the steering pole is to use a motor to rotate the steering column via the steering column. This allows a simple motor to be coupled to move the steering via a belt or chain. This would provide a method of detecting the steering angle using an encoder that could be coupled to the motor shaft, steering column or gearing system that connects the motor and steering pole.

The second method is to use a linear actuator to alter the angle on the wheels using a steering linkage. There are a number of problems with this method; firstly, it would require mounting to the front chassis of the bike, an area that can be subjected to high amount of water, mud and other debris. The second is maintaining the vehicles manual driving capability, with a motor and gear system to turn the handlebars, a clutch or other method is needed to disconnect the motor from the steering system via an electronic command. Using a linear actuator connected to a steering linkage does not offer an easy method of disconnection via an electronic input. This means a less reliable system to control the steering system due to human error and hard to ensure that the system is reconnected correctly every time the system is used in manual mode.

This makes the linear actuator system to be the less favoured option of controlling the vehicles steering, particularly with the manual control method relying on physical disconnection by the user, as opposed to using a clutch to disconnect the motor drive.

5.2.1 Torque Measurements and Calculations

To turn the steering pole, the torque required to turn the handlebars has to be measured. The torque was measured in two ways; the first measured the maximum expected torque that would be required, this was estimated by measuring the force required to turn the handlebars, when the vehicle is stationary, on smooth concrete. For these tests, the front wheels were inflated to the recommended pressure of ≈ 3 psi as specified in the vehicles manual. While the vehicle is designed to operate in fields, it may also travel on gravel roadways and paths. Therefore, testing the torque required to turn the handlebars, with the vehicle stationary on a concrete surface should provide a minimum torque requirement of any surface the vehicle should encounter in an agricultural environment.

To measure the torque required to turn the handlebars, a steel pipe was used to provide an extended leaver arm for the handlebars. This was used to decrease the force required to generate the required torque, so that a set of portable scales could be used to

measure the force required at the end of the leaver arm to turn the wheels. Using the described method above, a measurement of 4.5kg was measured using the portable baggage scales with a leaver arm of 162 cm measured from the centre of the handlebars to the point at which the scales were attached. The resulting torque is calculated as 71.51 Nm in equation 5.1.

$$\tau = Fxd$$
$$\tau = (4.5 * 9.81) * 1.62$$
$$\tau = 71.51\ Nm\ (2sf) \quad \textbf{5.1}$$

To ensure that this system will work reliably, as any system chosen should be capable of providing this torque at 80% of its maximum output, the minimum torque required was set at 90Nm (equation 5.2).

$$\tau = \frac{71.51}{0.8} = 89.39\ Nm\ (2sf) \quad \textbf{5.2}$$

At the base of the steering pole, the strut arm connects the steering pole to the wheels via the steering linkages as seen in Figure 5-3 (parts 11, 12), and modelled in Figure 5-4.

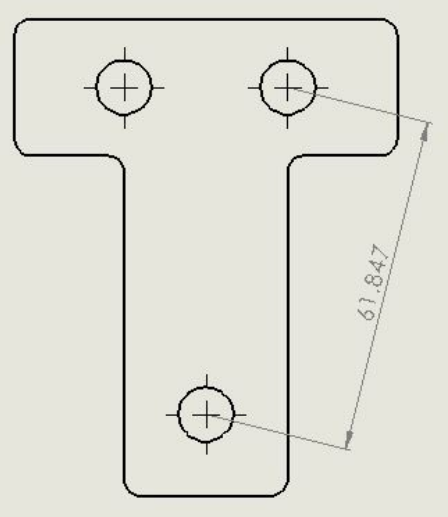


Figure 5-4 Steering Strut mount

$$x = \sqrt{15^2 + 60^2} = 61.8mm(1sf) \quad \textbf{5.3}$$

Using the dimensions of the strut arm connector above, the force on the strut arms was calculated as 1446.4 Newton in equation 5.4.

$$F = \frac{89.39}{0.0618} = 1446.4\ N \quad \textbf{5.4}$$

When a vehicle is moving, the force required to turn the wheels is greatly reduced. With testing undertaken on an asphalt surface with the bike moving at the minimum possible speed, a measurement of 3.2kg registered on the scales. This measurement was done with a steel pipe to extend the handlebars with the leaver arm distance extended to 0.95m, the torque required on the handlebars with 29.82 Nm calculated in equation 5.5.

$$\tau = (3.2 * 9.81) * 0.95$$
$$\tau = 29.82\ Nm\ (2sf) \quad \textbf{5.5}$$

As stated previously, the minimum motor torque required for the handlebars when the vehicle is moving based on an 80% output, resulting in a 37.28 Nm torque as calculated in equation 5.6. This equates to a force of 603 N shown in equation 5.7 that is applied to the wheel struts.

$$\tau = \frac{29.82}{0.8} = 37.28\ Nm\ (2sf) \quad \textbf{5.6}$$

$$F = \frac{37.28}{0.0618} = 603.2\ N(1sf) \quad \textbf{5.7}$$

5.2.1.1 Motor Selection

With the above torque and force figures available, the selected control mechanism was to turn the steering pole directly via a motor, as opposed to using a linear actuator. This was chosen primarily due to the ease of enabling/disabling a manual control system, as well as the force required to move the wheels fast enough would require a large, bulky linear actuator which would not fit within the confined space at the front of the bike.

To design the steering system, a motor must first be selected. Due to the limited funds in the project, a car windscreen wiper motor was repurposed for this. When testing this motor, the following method was used. A set of vice grips was used to grip the motor drive shaft, providing a leaver arm of 0.24m in length. Using the same set of portable scales used in the handle bar test, a measurement of 5.8 Kg was measured at the stall point of the motor, using an input voltage of 12V. This results in a calculated maximum torque rating of 13.66 Nm from equation 5.8.

$$\tau = (5.8 * 9.81) * 0.24$$
$$\tau = 13.66\ Nm\ (2sf) \quad \textbf{5.8}$$

To measure the motor current at stall, a 2.2Ω resistor was used in series with the motor. With the motor stalled a voltage drop of 7.89 volts was measured across the 2.2Ω resistor, from this a maximum current of 3.59 amperes is calculated in equation 5.9. This calculated current can be used as an expected maximum current to be drawn by the motor. The motor had a no load rotation speed of around 90 RPM.

$$I = \frac{7.89}{2.2} = 3.59\ A \quad \textbf{5.9}$$

The windscreen motor required modification before it could be used to drive this system. This was due to the motor only being designed to operate in one direction, with one of the motor terminals bring grounded to its metal body, and thus to the quad bikes chassis. Because of this, reversing the motor was impossible due to the chassis of the quad bike being grounded to the main batteries negative terminal. This grounding causes a short circuit whenever the polarity across the motor is reversed and insulating the motor from the chassis would not provide the required reliability for the motor, while maintaining the required mounting strength. To overcome this problem, the motor was disassembled and the internal terminals modified to insulate them from its metal body. This enabled the motor to be driven clockwise and anti-clockwise, by changing the polarity across the motor terminals.

5.2.1.2 Calculating Pulley ratio, pitch and thickness

Based on the available motor, a minimum gearing ratio, φ of 6.6: 1 is required between the motor and the steering pole to ensure sufficient torque is applied to position the wheels based on calculations obtained from equation 5.2 and 5.8.

$$\varphi = \frac{89.39}{13.66} \approx 6.6$$

The HTD series of timing belts were used to create a customizable gearbox that will provide the necessary functionality, while providing a low cost solution. This belt uses a rounded tooth design (see Figure 5-5) as opposed to the standard trapezoidal shape, and provides a method of accurate power transmission. Timing belts provide a method of power transmission, different from other belts due to the belts teeth that prevent slippage while turning and allows for accurate positioning, so long as the belt is correctly tensioned to prevent gear jumping.

The HTD belt type allows custom gears to be machined using CNC milling due to the rounded corners on the pulleys, which match the rounded profile left due do the cutters being unable to machine square corners on the internal angles of profile cuts.

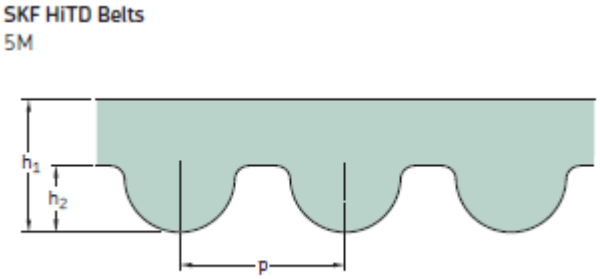


Figure 5-5 SKF HiTD Belt profile[8]

A minimum pulley size for the motor was determined to be 50mm in diameter, this is based off the size of the motor shaft and the coupling system used to join them together. A maximum pulley diameter of 120mm was selected based on the handle bar size, as well as the space available due to the position on the fuel tank with only enough space to fit a pulley of that size. Using the SKF belt selection guide, the minimum number of teeth for 5M and 8M timing belts are 16 and 22 teeth respectively[8]. To determine a belt size for the steering pole,

an estimated minimum distance of 150mm was made between the centre of the steering pole and its driving pulley, this is based on steering pole pulley size and by estimating where the system would want to be mounted.

To calculate the power that the system would be designed to transmit through the system; formula 2 from page 103 of the SKF belt selection guide was used[8].

P_d = design power, P_r = motor power rating and C_2 is the service factor of the system.

Using the measurements obtained from the stall current of the motor, a power rating of 43 Watts is obtained from equation 5.11.

$$P = V * I = 12 * 3.59 = 43.01\ Watts \quad \textbf{5.10}$$

Using this and a service factor of 1.3, which corresponds to a soft start running under 10 hours per day, a design power of 56 Watts was calculated in equation 5.11.

$$P_d = P_r C_2 = 43 * 1.3 = 56\ Watts \quad \textbf{5.11}$$

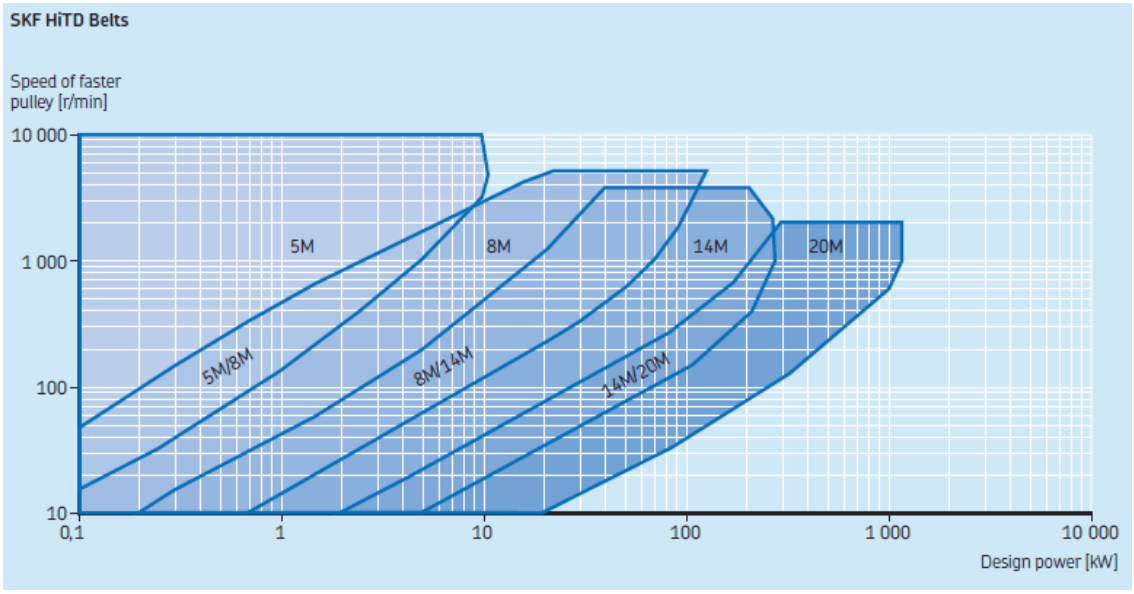


Figure 5-6 Belt cross section chart[8]

The diagram shown in Figure 5-6 was used to calculate the belt pitch; the power transmitted through the belts is equal to 56W at a maximum RPM of 60 revolutions per minute. This puts the desired belt width in the 5M/8M region, and due to the cost of the belts, 5M belts were selected. This meets the specified requirements for the power transmission, and provides customized pulley machining through the using a CNC machine with a 3mm end mill to cut the pulley teeth.

Using the centre distance chart, table 4b (shown in Figure 5-7); from the SKF power transmission manual, a pulley ratio of 2.25:1 was determined to be the best reduction for this system. This was based on the following limiting factors: the maximum steering pole pulley size of 120mm, and the minimum centre distance of 150mm. The maximum ratio available for the 5M pitch belt, while maintaining the constraint on the maximum pulley size, is a 72 tooth

driven pulley with a pitch diameter of 114.56 mm in conjunction with a 32-tooth driver pulley, with a pitch diameter of 50.93 mm.

Table 4b

Speed ratio	Driver		Driven		Belt length										
	Number of teeth	Pitch diameter	Number of teeth	Pitch diameter	350	375	400	425	450	475	500	535	565	600	635
–		mm		mm	mm										
2,11	38	60,48	80	127,32	–	–	–	–	–	–	–	115,1	130,7	148,7	166,6
2,12	34	54,11	72	114,59	–	–	–	–	–	100,4	113,4	131,5	146,9	164,7	182,5
2,13	32	50,93	68	108,23	–	–	–	–	95,7	108,7	121,6	139,5	154,8	172,6	190,3
2,15	52	82,76	112	178,25	–	–	–	–	–	–	–	–	–	–	–
2,22	36	57,30	80	127,32	–	–	–	–	–	–	–	117,2	132,9	150,9	168,9
2,25	40	63,66	90	143,24	–	–	–	–	–	–	–	–	112,9	131,4	149,7
2,25	32	50,93	72	114,59	–	–	–	–	–	102,5	115,6	133,7	149,1	167,0	184,8
2,33	48	76,39	112	178,25	–	–	–	–	–	–	–	–	–	–	–
2,35	34	54,11	80	127,32	–	–	–	–	–	–	100,8	119,3	135,0	153,1	171,1
2,37	38	60,48	90	143,24	–	–	–	–	–	–	–	–	115,0	133,5	151,8

Figure 5-7 Compressed version of Table 4b from SKF Manual[8]

With the ration selected, the centre distance is the second constrain required to complete the design with a belt length of 565 mm providing a centre distance of 149.1 mm. The belt size is a 600mm belt with a centre distance of 167mm. As the 600mm belt centre distance was significantly greater than 150mm minimum, it was decided that 0.9mm under the estimated minimum was an acceptable for this design. The physical minimum centre distance for the steering pole pulley is 102.5mm based on the specified gear combination.

With this initial reduction, a torque rating of 37.25Nm would be required on the driving gear, with the total gear ratio being 6.6:1. This requires a remaining ratio of 2.89:1 to be satisfied by the belt gearing, and can be provided by two reductions of 1.7:1. To reduce the number of unique parts in the gearbox, a 32-tooth pulley has been used for all the driver pulleys in the gearbox.

The final reduction stages used were selected as two 1.88:1, using 32 and 60 tooth pulleys with a 450mm belt. This results in an overall reduction of 7.95:1; therefore, the torque required from the motor is 11.24Nm while stationary on concrete. This is significantly less that the motors output at of 13.66Nm at 80% of maximum power.

Due to all the driving pulleys having a common tooth count of 32, the same pulley thickness can be used for all the belts. The calculated power transmission for the system does not exceed the 120-Watts minimum for 32-tooth pulley; the RPM does not exceed the minimum table value of 100. The table indicates that a 15mm belt can operate up to the 120 Watt max, and using the adjustment multiplier of 0.588 a 9mm belt can support up to 67 Watts, this exceeds the maximum power that transmitted by the system[8].

5.2.2 Initial Design and Functionality

The pulley reduction gearbox needs to provide a compact method of coupling the steering motor to the quad bikes steering pole, it must include a mounting system that ensures the steering encoder is protected from environmental conditions. The gearbox must be designed to ensure that all the belts are correctly tensioned to prevent the belts from jumping, with a mechanism to enable manual steering by disconnecting the steering motor from the drive train.

5.2.2.1 Custom Pulley Testing

Each of the selected pulley-belt combinations has a specified centre distance at which it operates, the 72-32 ratio has a centre distance of 149.1mm and the 60-32 ratio has a centre distance of 94.9mm. With these manufacturer specified values a gearbox system can be designed without a tensioning system, assuming the gearboxes pulleys remain in a fixed location.

Figure 5-8 Belt and pulleys suspended in milling machine.

To ensure the accuracy of the centre distances with the gears that were machined in house, a test was done using a manual milling machine that had a digital readout. Figure 5-8 above shows a divider head holding the one pulley, with the second pulley held in the drill chuck. With the X and Y-axis of the drill chuck zeroed concentrically to the centre of the divider head, the centres were moved apart along the X-axis while maintaining the position of the Y-axis.

The resulting distance can be seen in Figure 5-9 below, with a centre distance of 107.01mm measured. The nominal centre distance from Table 4b in Figure 5-7, for the 60:32 tooth ratio using the 450mm belt is 107.7mm, while the measured pulley centres differed from the nominal, the actual 0.69mm difference can be explained by tensioning when measuring. Thus, the 107.7mm centre distance is valid for the custom pulleys when designing the gear system, and therefore it can be assumed that all other pulleys distances will be valid with the custom pulleys.

Figure 5-9 Centre distance measurements for 32-60 pulleys using 450mm belt

5.2.2.2 Gear Box Design

Having selected the belt reductions required to increase the torque applied to the steering pole, a housing case was required, in which the gears are mounted. There are three sections to the pulley gearing system: the motor interface, the main reduction and the final reduction that connects the gearbox to the steering column. Figure 5-10 shows a colour coded model of the steering mechanism showing the manual control system (aqua), belt guides (red), and the gearing reductions(blue and green).

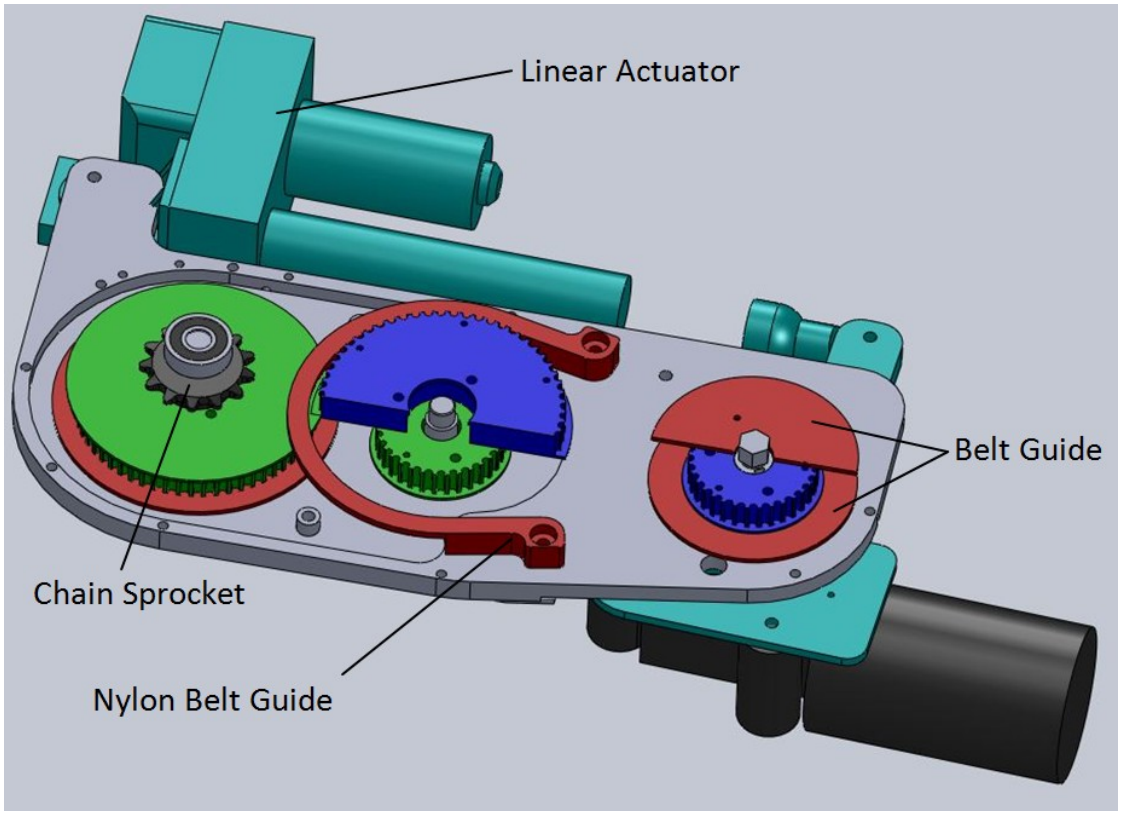


Figure 5-10 Steering Control Mechanism

The motor interface (aqua plate in Figure 5-10) was machined out of a 10mm aluminium plate, designed with raised locating lugs that were positioned to line up with the three flat mounting holes on the motor. The motor interface plate is paired with a 3mm nylon sheet to reduce any friction between itself and the aluminium surface of the pulley housing. Figure 5-11 and Figure 5-12, show the manual control mechanism (aqua) and the effect the linear actuator has on the position of the driving motor/gear when it is positioned at its actuation limits.

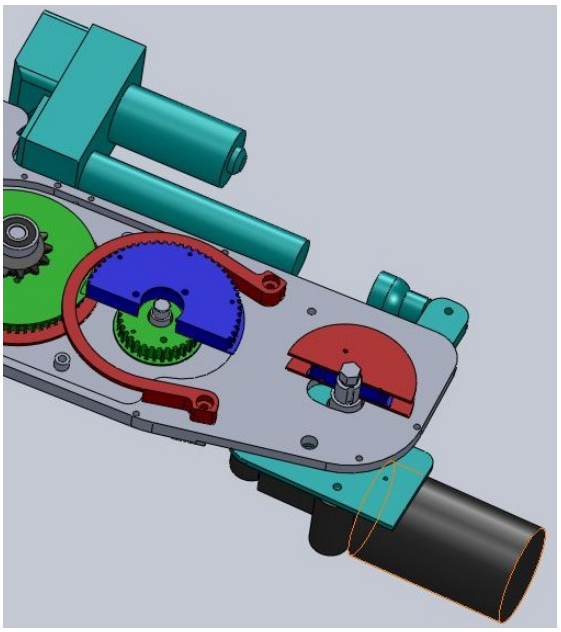


Figure 5-11 Manual Control Mechanism - Drive Belt Engaged

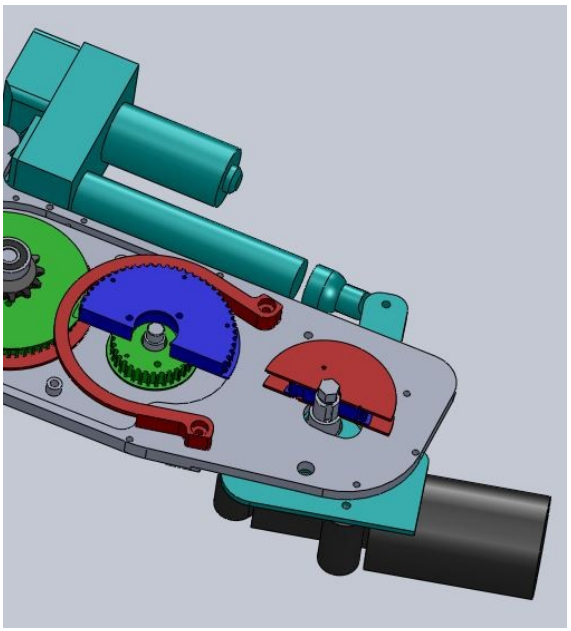


Figure 5-12 Manual Control Mechanism - Drive Belt Disengaged

The main reduction consists of two belt drives that have 32 and 60 tooth pulleys; this connects the 32-tooth pulley attached to the motor shaft by an adaptor and keyway to the first (highlighted in green and blue in Figure 5-10). This housing was designed with the bearings fitted into the pulleys, with a fixed shaft attached to the housing in to pivot on. This simplified the design by removing the need to couple a shaft to the pulleys as well as reducing the number of bearings required. To ensure the main reduction belt is correctly tensioned, the centre distances for the shafts were machined according to the centre distances specified in the SKF Guide[8], with the accuracy of these distances proven in section 5.2.2.1 above.

The final reduction connects the output of the pulley system to the pulley mounted to the steering pole. This connection was designed so that it used the existing handlebar mounting plate, Figure 5-13, to connect the 72-tooth pulley to the steering pole, while still allowing the handle bars to be connected, as they normally would be. The only difference made to the handle bar system apart from the pulley system now attached, was that the handle bars were raised by 12mm, allowing the user to use the system uninhibited by bulker motor systems that would be otherwise directly mounted to the handle bars/steering pole.

Figure 5-13 Handle Bar mounting plate

To tension the final transmission belt, the mounting system enabled the belt housing to slide away from the steering pole. In doing this the belt can be tensioned and the belt housing locked in position by tightening the mounting screws on the bottom.

To sense the steering position, an encoder was coupled to the output-driving pulley on the belt housing. Figure 5-14 shows the encoder highlighted in the red square, coupled to the base of the shaft that turns with the steering pole regardless of how the handlebars are turned. Figure 5-14 shows the final design with the chain driven design (black) that connects the output from the gearbox to the steering pole.

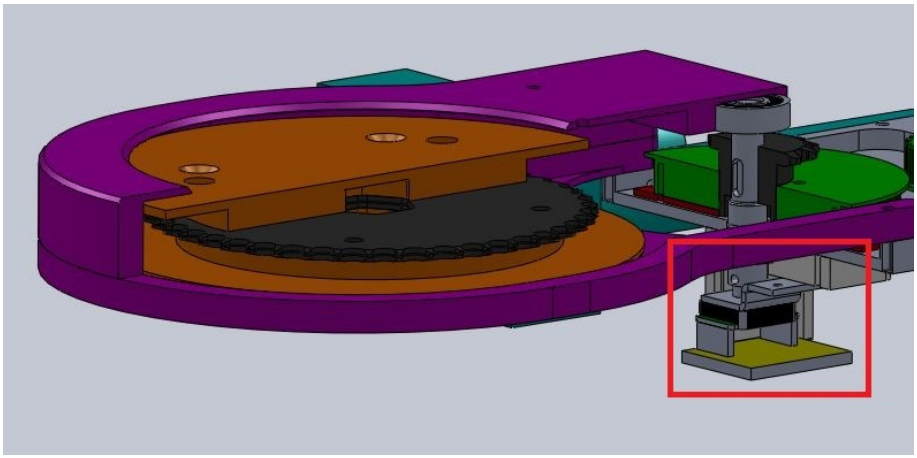


Figure 5-14 Encoder Mount on the steering mechanism

The encoder used is a Bourns absolute contacting encoder (datasheet included in Appendix B). This provides an 8-bit grey code output, featuring 128 unique positions that can be distinguished with only one bit changing at any time. This prevents incorrect values from being read, refer to section 6.2.1 for more information.

The steering mechanism has a 50-degree range of motion either side of centre, 100 degrees full range. By coupling the encoder to the output driving gear of the belt housing, this provides an increased resolution for the steering pole rotation due to the reduction ratio. The resolution is increased from 36 positions if directly coupled to the steering pole, to around 80 positions using the gearing mechanism. By coupling, the encoder to the output driver of the belt housing, it also allows the encoder to remain connected to the steering pole when in manual steering mode.

This pulley set features a special nylon guide (seen in white in Figure 5-15) to keep the belt off the driving pulley when the manual mode is triggered by ensuring the belt is kept in contact with the driven gear coloured in blue in Figure 5-10.

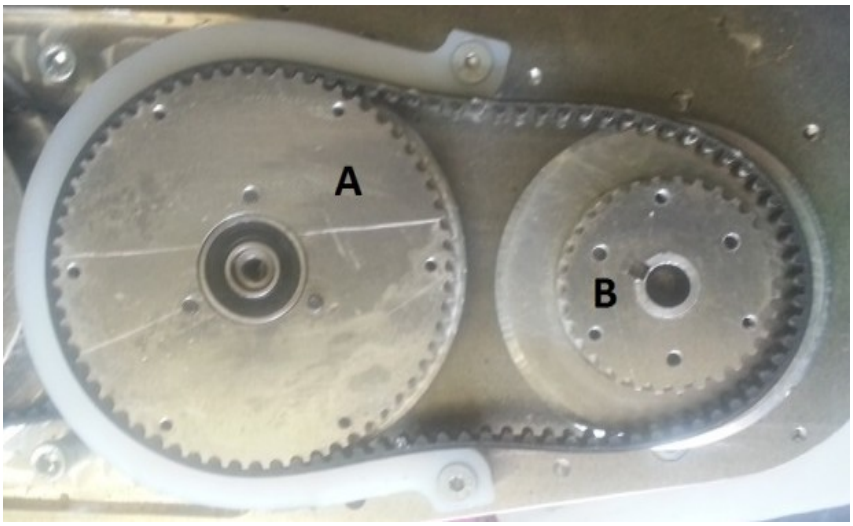


Figure 5-15 Nylon belt guide

Figure 5-15 shows the effect on the belts geometry when decreasing the centre distance between the motor pulley (B) and the driven pulley (A), while the belt is constrained to fit the driven pulley (A). Decreasing this centre distance of this mechanism pulls the timing belt off the motor pulleys teeth, allowing the driven pulley A to turn independently of the driving pulley B.

The steering system allows the user adjust the steering manually, while still providing feedback for the steering pole positioning via the absolute encoder. The steering system uses a linear actuator to move the motor driver pulley towards its companion driven pulley. This in turn changes the centre distances causing the belt to decouple from the motor driven pulley, allowing the belt and thus the steering mechanism to move freely. This decoupling is necessary, as the windscreen motor features an internal worm-wheel drive, which requires significant torque in order to drive it manually, which the user will not be capable of providing while manually operating the vehicle.

This provides a method of disconnecting the motor from the drive belt without the use of human intervention, allowing for a more reliable system without human error. Though this design works for the current system, future designs should include a clutch mechanism as the disconnection method.

5.2.3 Implementation and Testing

To mount the belt housing, a support bar added to ensure alignment between the steering pole pulley and the driving pulley from the belt housing. This belt housing support plate was attached to the bottom of the housing, and a support structure built up from three points on the chassis to provide a stable support. With the motor belt and steering pole belts, correctly tensioned, basic testing was undertaken to test that the system worked.

With the belt housing mounted, correctly aligned and the belts correctly tensioned, testing was undertaken to check the functionality of the system. This revealed several problems with the system, the first major problem bring that a previous owner damaged the steering pole, with evidence that the quad bike had rolled with damage to the brake leavers.

As shown in Figure 5-16, a significant distortion within the steering column of 3.76 degrees is visible.

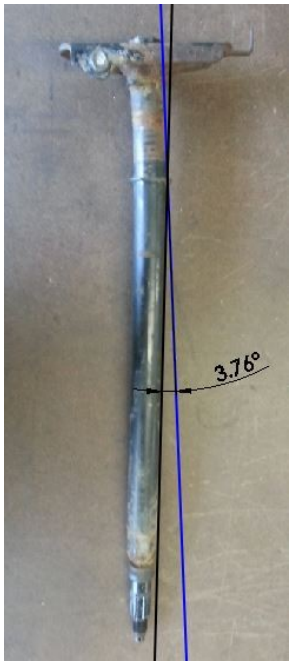


Figure 5-16 Angle of steering pole measured using Solidworks sketch

This distortion in the steering pole cause a variation in the center of rotation for the steering pole belt gear, causing the belt driving the steering pole, to tighten and slaken as the steering pole turned. Because of this slackening, the belt is able to jump over the teeth on the driven pulley which negates the purpose of the timing belt. To repair this damage, the steering column was removed, with the damaged sections cut out and replaced with new piping that was turned on a lathe and welded into position.

While this improved the circulization of the steering pole, there was still a slight variation in the rotation of the steering pole/handle bars. An investigation into building a tensioning system system to keep the steering pole belt correctly tensioned was undertaken, with a test system built. This was found to be an inefficient and impractical system, and upon consulting with the Massey University workshop technician, it was decided that a chain drive would be a more robust method of transmitting the required power to the steering mechanism. This chain drive used a 3/8 inch chain.

The advantage of using a chain drive, is that it has an intrinsic feature that causes the driven side of the chain to remain in tension, without using a static tensioning system. This prevents the chain from jumping over the teeth on the sprocket, even if the centre distance varies, unlike with the timing belt. Even though the driving and driven sprocket may not always remain at the optimal distance, due to the steering poles elliptical rotation, the chain will remain in tension on the side in which it is traveling toward the driving sprocket.

Testing the manual steering control system revielded a number of design faults; the first being that the timing belt from the driving motor would bind up, between the driven pulley and the pulley guide, shown in Figure 5-15 when the motor was disengaged from the

belt. This was caused by the gap left between the belt guide and the belt being large enough to wedge the belt in place. This was solved by decreasing the distance between the belt and the guide, and thus prevented the belt from binding.

The second and most significant problem was the linear slide design used on the motor interface mount to engage/disengage the belt from the motor. The method in which the linear actuator applyed the force to move the motor interface, caused the motor interface mount to become wedged against the guide channel of the belt housing. This prevented the steering mechanism from engaging or disengaging the motor from the belt drive without assistance from the operator. This mechanism was redesigned to work using a single pivot point which prevents any chance of the motor mount binding on the belt housing.

5.2.4 Improvements and Final Design

A number of improvements were made in the final design; these included the redesign of the manual control mechanism, replacing the belt driving for the steering pole with a chain that increasing the reduction ratio from 2.25 to 3.46, and the repositioning of the encoder into a protective housing below the belt housing.

By changing to a chain drive, the detectable positions for the steering pole increased to 125 using 97.7% of the encoder capacity. Along with this, a chain guard was included to protect the user from being injured and to prevent objects such as branches or trees from lodging in between the chain and sprocket, see the parts highlighted in purple in Figure 5-17 and Figure 5-18.

Final testing on the steering mechanism was conducted by driving the motor under the same conditions as the initial torque testing was conducted under in section 4.2.1. The steering mechanism was capable of turning the quad bikes steering system on the concrete surface, though after the belt reductions. The steering system takes 3.2 seconds to rotate from one limit to the other when tested on a concrete surface. Though this is not particularly slow, it will reduce the steering response of the vehicle, and thus the maximum speed of the quad bike needs to be limited when steering autonomously.

The average current drawn by the motor during operation was comparable to the current measured in the initial testing of the motor, with current spikes of up to 15 amperes measured through the motor driver. These motor current measurements, were within the 30 amp operating range of the chosen motor driver, see section 6.2 for details. The actuator for the manual steering control system was proven to work as designed, with a combination of internal limit switches and a current sensing to detect when the steering motor is either in a fully engaged or disengaged position.

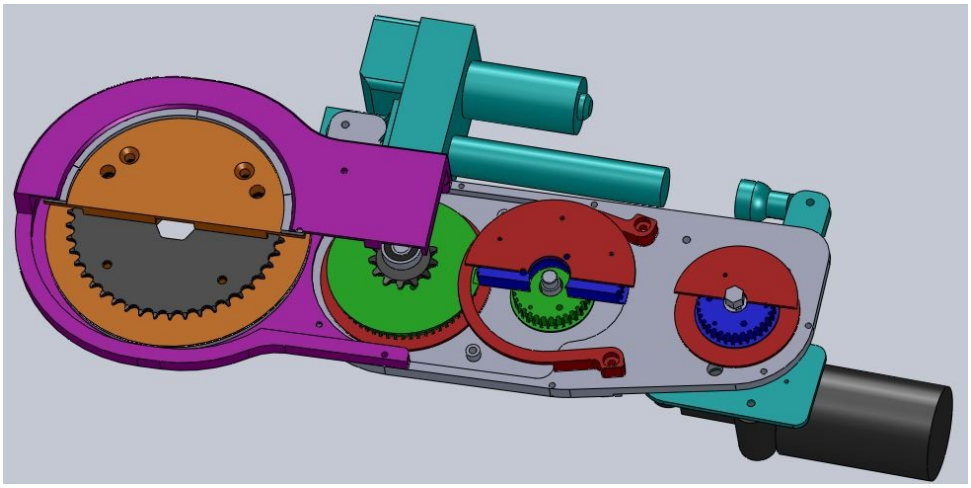


Figure 5-17 Steering Mechanism Top View

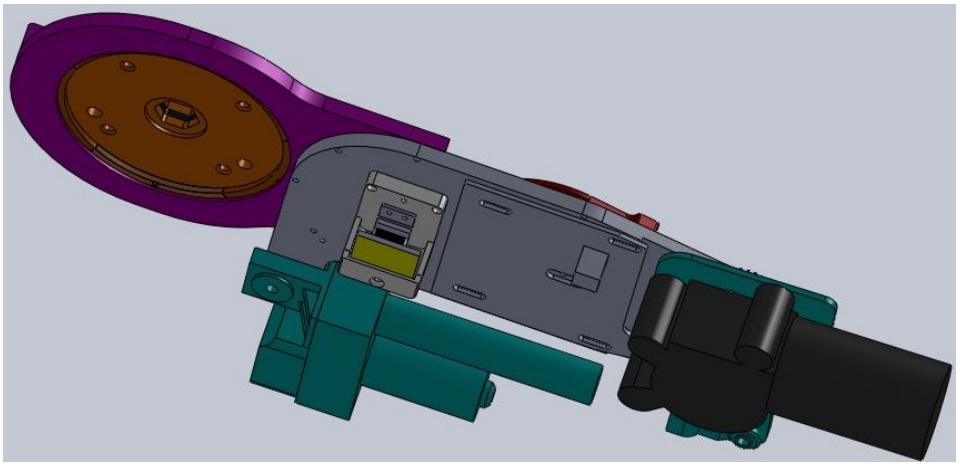


Figure 5-18 Steering Mechanism Bottom View

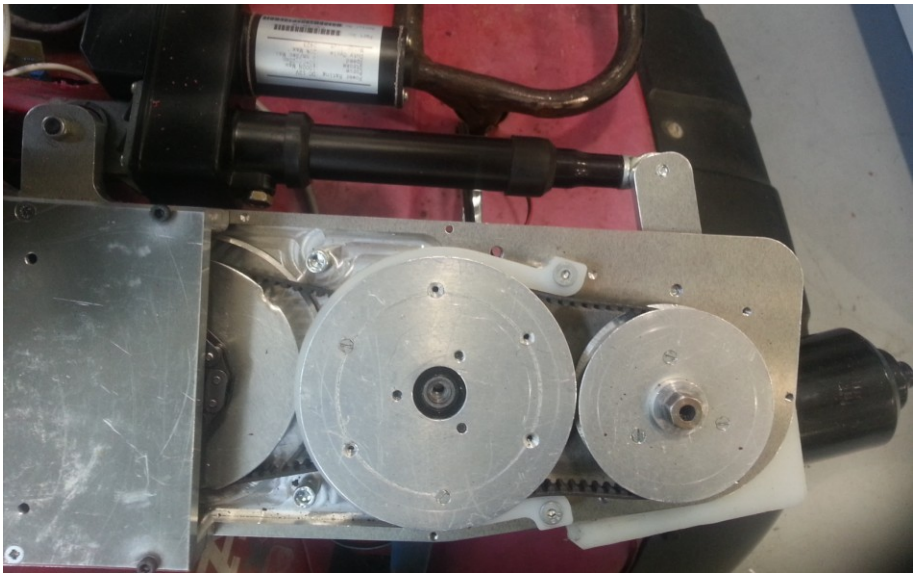


Figure 5-19 Final Steering mechanism design

5.3 Braking Control Mechanism

One of the most important features to control on the vehicle is the braking system. Without an adequate and reliable braking system, the vehicle would be too dangerous to control, with an unacceptable risk to property and people. This section investigates and outlines the mechanical systems used to control the quad bikes braking systems.

5.3.1 Existing Braking System

The quad bike has two sets of brakes, a hydraulic front brake on the front wheels, and a drum brake on the rear axle. The front brake operates via a handle located on the right side of the handlebars. When the brake leaver is actuated, a plunger is depressed providing the hydraulic pressure needed to operate the brakes, as illustrated in Figure 5-20.

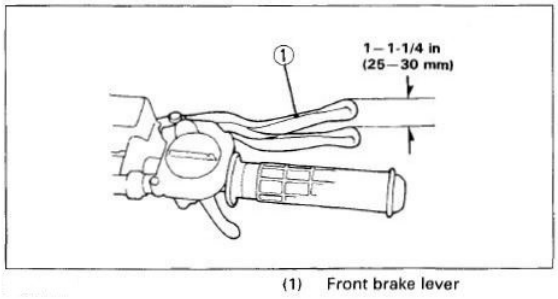


Figure 5-20 Front Hydraulic brake leaver (Appendix J)

To ensure the front brakes operated efficiently, the brakes were serviced to ensure any braking applied was effective. This involved adjusting the brake callipers on each wheel to compensate for any wear on the brake pads, as well as flushing the hydraulic system with new brake fluid.

The rear drum brake operates via the hand leaver on the left hand side of the handlebars, and features a locking leaver so that it can act as a parking brake. The rear brake leaver also provides the means in which to engage the reverse gear. The rear brake can also be operated using a foot pedal located in the right hand foot well, with both the pedal and the hand leaver are cable based and actuates a leaver on the drum brake.

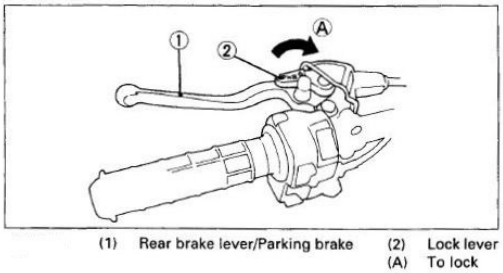


Figure 5-21 Rear Brake hand control (Appendix J)

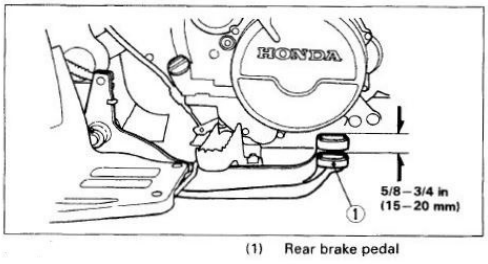


Figure 5-22 Rear Brake Foot pedal (Appendix J)

Due to the condition of the vehicle, servicing was undertaken on the rear drum brake with the rear axle removed, allowing for the replacement of the brake pads. Any corrosion on the internal brake surface as well as the rear axle was removed. During the servicing of the

rear brake, significant wear was found on the spline connecting the rear axle to the rear wheel hub on the right hand side. The spline on the wheel hub had worn to the point an interference fit was no longer possible, allowing the back wheel to rotate independent of the axle when the back wheels were lifted off the ground. Most of this wear was located on the wheel hub, though the spline on the axle did show signs of wear. Even though this wear was present, a new wheel hub provided sufficient interference to enable effective power transmission to the wheels. Replacement brake cables were fitted to ensure smooth brake actuation, and were correctly tensioned via the adjustment nuts (shown in Figure 5-23) on the leaver arm shown in Figure 5-24.

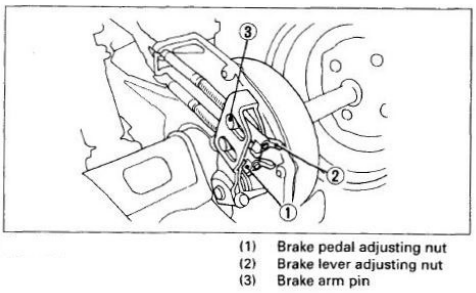


Figure 5-23 Rear Drum Brake Cable setup (Appendix J)

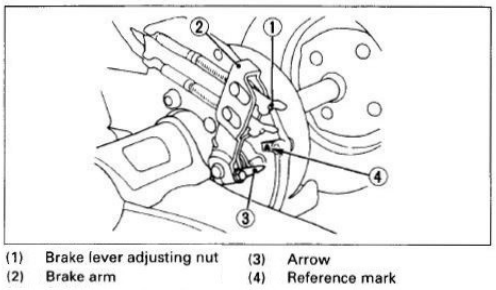


Figure 5-24 Rear Brake drum brake adjustment controls (Appendix J)

5.3.2 Brake Sensing and Actuation

Due to the limitations of the steering control mechanism regarding the turning speed of the system, the maximum speed of the vehicle is limited for autonomous navigation. Due to this limitation on the speed, an actuation method for the front hydraulic brakes was not implemented, with the rear braking mechanism determined to be sufficient to stop the vehicle. Though to ensure safety operation of the vehicle, the system needs to be capable of detecting if and when the braking is applied to the front brakes to prevent the control system from trying to maintain a set speed if either of the brakes are used.

With the front brakes using a hydraulic system for braking, two methods of detecting the actuation of the hydraulic brakes were considered; the first was to integrate a pressure transducer into the hydraulic line to detect the amount of force that is being transmitted to the brakes. This would provide an accurate and reliable method of detecting when the front brakes are being activated, while providing signal that corresponds to the braking force being applied.

The second method involved integrating a sensor into the handle to detect its position. The exact braking force required is not required for this system, the controller only needs to know if the brakes are being used and an estimated level of braking. Considering these requirements, a simple mechanism using a variable resistor can be used to measure the position of the brake handle. This method is cheap and provides an effective way for measuring the position to the accuracy required for the system.

The second method described above, was used for both the front and rear brake sensing, with both mechanisms requiring an approximation of the braking levels being applied. The front braking system is sensed by positioning the variable resistor, which is position on the

pivot point of the handle and is connected via a 3D printed arm that binds the resistor shaft to the handle.

The rear braking system is cable based, because of this it can be operated off both the foot pedal and the handgrip on the handlebars. To monitor the status of the rear brake, either a sensor would be required on both of these actuator points, or the simpler solution is to put the sensor on the brake leaver that is being actuated on the rear wheel drum brakes. This ensures that no matter which cable is being operated, the actuation on the brake drum is detected and measured by the controller.

As with the front sensing system, the rear brake position is sensed using a variable resistor configured as a voltage divider, which provides an analogue voltage signal that varies as the position of the brake arm is moved. Due to the positioning of the brake arm, trying to mount the variable resistor on the pivot point as was done for the front brake handle, proved unfeasible as space restrictions. In addition, there is a high probability of the sensor in this position being damaged by objects on the ground, particularly when the vehicle is travelling backwards. By mounting the resistor further into the chassis, and considering the protection from environmental factors, the resistor shaft was connected to the brake arm using a hinge mechanism. The completed design is shown in Figure 5-25 below.

Figure 5-25 Rear Drum brake with position sensing resistor

With the rear brake system providing two methods of cable actuation, there were two methods available to apply the brakes. The first involved attaching a actuator directly to the brake arm, while this would have provided a effective method to actuate the leaver arm, it would have been the only method with the brake arm bring unable to move independantly of the actuator. The second system was designed to use one of the brake cables to actuate the brake mechanism. This provides the same actuation as the first method, but with the added benefit allowing both cables to work at once.

The second method of actuation was used to actuate the rear brake, with the cable connecting to the foot pedal used to activate the brake. The foot pedal was converted to actuate using a linear actuator, because the foot brake is not often used to apply the brakes, with the operator always having access to the brake controls on the handlebars. In order to actuate the brakes, the force required achieve the necessary actuation must be measured.

5.3.2.1 Front Brake Testing

Figure 5-26 features a drawing showing the actuation of the front brake handle, with the grey coloured handle showing the default position of the handle when the brakes are off. The blue handle in Figure 5-26 shows the position of the handle when fully actuated as well as the two points before actuation A and B and after actuation A' and B'. A set of portable scales was used to measure the force required to operate the front brakes, resulted in a 15 Kg reading on the scales, measured at A with a distance, x of 86.6mm from the pivot point. This results in a torque requirement of 12.74 Nm calculated in equation 5.12.

$$\tau = (15 * 9.81) * 0.0866$$
$$\tau = 12.74\ Nm\ (2sf) \quad \textbf{5.12}$$

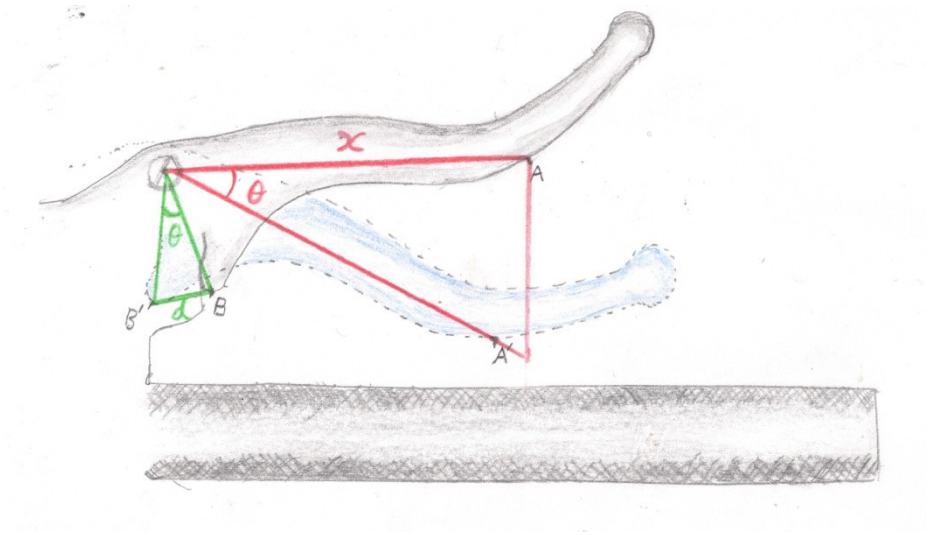


Figure 5-26 Front brake handle showing actuation

This torque results in a force of 488 Newtons calculated in equation 5.13 is applied to the hydraulic plunger using a leaver arm of 26.1mm long (pivot to point B).

$$F = \frac{12.74}{0.0261}$$
$$F = 488\ N \quad \textbf{5.13}$$

With the total motion of the existing, handle mechanism moving an angle, θ of 21.8 degrees around the pivot point, resulting in a plunger stroke of 9.7mm shown in equation 5.14.

$$d = \sin 21.8 * 26.1$$
$$d = 9.7mm \quad \textbf{5.14}$$

5.3.2.2Rear Brake Testing

The force required to actuate the rear brake shown in Figure 5-27, via the hand leaver to the limit of its travel was recorded using portable scales with a resulting measurement F_1 of 12.4 Kg (121.6N (1sf)), with a leaver arm d_1 of 106mm in length. A resultant torque required to actuate this leaver was calculated as 12.89 Nm

$$\tau = (12.4 * 9.81) * 0.106 \quad \textbf{5.15}$$

$$\tau = 12.89\ Nm\ (2sf)$$

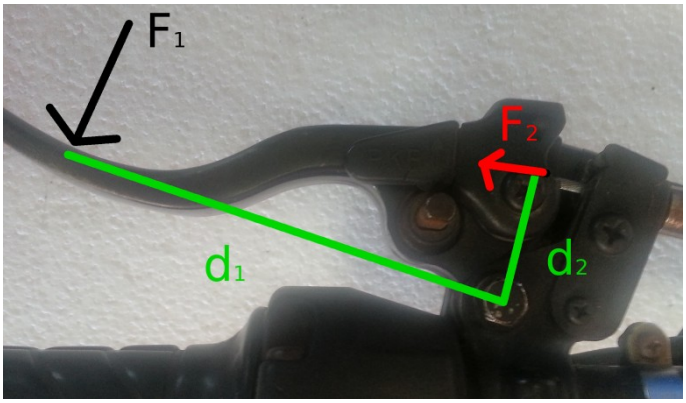


Figure 5-27 Rear brake hand control with annotations

The handle has a leaver arm with a length d_2 of 35mm to pull the rear brake cable, resulting in a calculated force F_2 of 368.4 Newtons pulling on the cable.

$$F_2 = \frac{12.89}{0.035} = 368.4\ N\ (1sf) \quad \textbf{5.16}$$

On the brake drum leaver in Figure 5-28, the front cable is anchored at a length d_3 of 98mm from the pivot point, resulting in a torque of 36.1 Nm on the leaver arm.

$$\tau = F_2 * d_1 \quad \textbf{5.17}$$
$$\tau = 368.4\ * 0.098 = 36.1\ Nm\ (1sf)$$

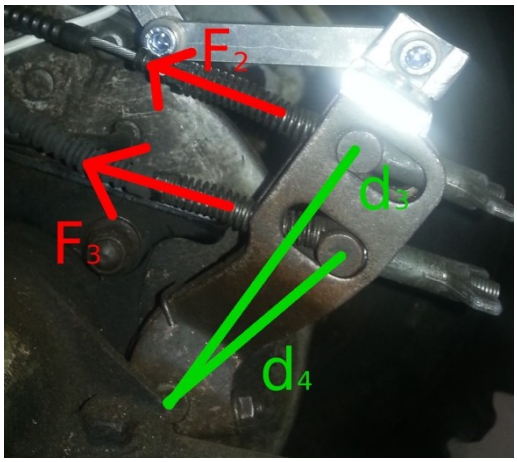


Figure 5-28 Rear Brake Drum with annotations

The cable connecting the foot pedal to the brake leaver is anchored at a length d_4 75mm from the pivot point, and to operate the foot pedal a force F_3 of 481.33 N is required on the cable as calculated in equation 5.18.

$$F_3 = \frac{36.1}{0.075} \quad \textbf{5.18}$$
$$F_3 = 481.33\ N\ (2sf)$$

The rear brake cable is anchored to the foot pedal at a distance d_5 of 84mm from the centre of rotation for the pedal arm, resulting in a calculated torque requirement of 40.43Nm from equation 5.19 to be applied to the pedal extend the brake to its limit.

$$\tau = 481.33 * 0.084$$
$$\tau = 40.43Nm \quad \textbf{5.19}$$

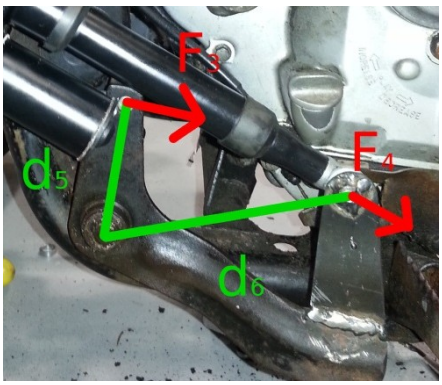


Figure 5-29 Rear brake foot pedal with annotations

The modified foot pedal shown in Figure 5-29, has an effective leaver arm length d_6 of 142mm. Therefore, a force F_4 of 284.7 Newtons is necessary from the linear actuator to extend the braking leaver to its maximum position as defined provided by the front handle grip, shown in equation 5.20.

$$F_4 = \frac{40.43}{0.142} = 284.7\ N\ (1sf) \quad \textbf{5.20}$$

5.3.3 Rear Brake Actuator System

A linear actuator has been implemented as the method of actuation for the foot pedal, which provides a self-locking mechanism so that once the position of the actuator is set it does not require to be powered to hold its position. This is particularly useful for parking systems, as it can be set as the parking brake on a sloped surface without requiring any input to maintain. The chosen actuator provided a static load capacity of 90.7 Kg (200 Lbs), with a dynamic load rating of 50.8 Kg (112 Lbs) and a custom stroke distance of 0-2 inches using two limit switches mounted on the shaft.

The linear actuator operates off a 12-volt input, with a maximum current draw of 4.8 amps and a rated travel speed of 9mm per second at 50 percent load. With such a low travel speed on the braking mechanism, the rate at which braking can be applied is limited. This is a problem if an emergency stop is required when travelling at high speeds, but due to the vehicles limited maximum speed, it should not be a significant problem for the prototype system. However, recommendations for future development are that: a system with a faster actuation time from limit to limit to be implemented on the rear braking system; implement a similar system to control the front braking system. Any changes to the braking system should still maintain a user implemented braking system for both the front and rear braking, to ensure manual control is still available.

The final implemented design offers a compact solution, positioned so that it prevents any interference to a vehicle if operating the vehicle manually. The existing foot pedal system has been modified to couple the linear actuator to the foot pedal, with the return spring being removed and the pedal shortened with a mounting plate added. The linear actuator is controlled via a microcontroller with an H-bridge motor driver chip to drive the motor, extending or retracting the actuator by changing the polarity across the motor terminals.

Figure 5-30 Linear actuator control over modified rear brake foot pedal

Figure 5-31 Linear actuator positioned to prevent interference with a user riding the bike

5.4 Gear Changing Mechanism

For the vehicle to operate autonomously, a gear changing system needs to be included on the vehicle. If the vehicle has an automatic gearbox then this is not necessary, assuming it has a method of setting drive, reverse, neutral, and park. Otherwise, if the vehicle has a manual gearbox then a gear changing mechanism needs to be implemented to allow the system to perform the necessary gear changes, including the clutch control to engage the motor and gearbox.

The TRX-300 quad bike has a manual gearbox, with a gear changing sequence in the following order; reverse, neutral, super low, then first through to the fourth and final gear as shown in Figure 5-32 below.

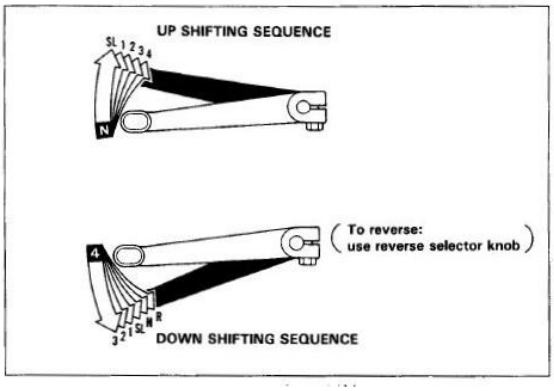


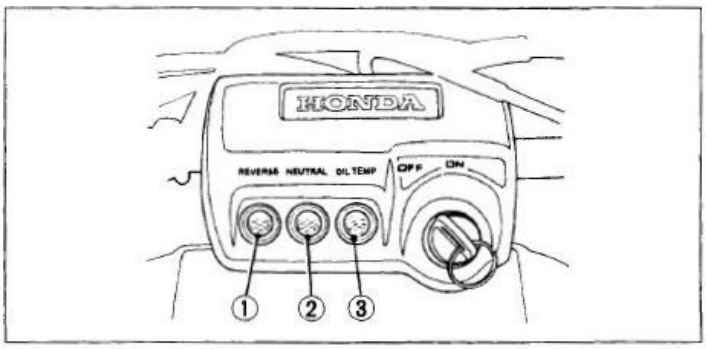
Figure 5-32 TRX-300 gear changing sequence (Appendix J)

The TRX-300 quad bike features a centripetal clutch, which when the engine reaches a certain RPM, the centripetal force acting on the clutch forces the clutch shoes into the clutch housing engaging the engine drive shaft with the transmission shaft. Because of this, the clutch will automatically engage with the gearbox once the engine has reached a speed that is capable of driving the gearbox without stalling. If the engine RPM decreases below the point at which it engages with the gearbox, the clutch will disengage and thus will prevent the engine from stalling if the engine load is too high to be sustained by the throttle input.

To provide the necessary functionality to operate the gear system, the gearbox mechanism needs to be capable of the functions: changing up and down gears, detecting/recording which gear the gearbox is currently set to and providing a fast change time that is as fast as the change time for a human operator. Finally, to maintain the manual control capably of the system, the user should be able to manually changing gears using a method that is similar to the existing gear changing foot pedal. This is particularly necessary in the event of system brake down to prevent the quad bike from becoming unusable.

To prevent accidental gear change from neutral to the reverse gear a reverse activation mechanism needs to be actuated. This is traditionally accomplished using the rear-braking handle on the left hand side of the handlebars. To do this a secondary pushpin must be pressed to couple the reverse cable to the braking leaver; this ensures the quad bike is stationary when the reverse gear is engaged. In order for the control system to be capable of engaging the reverse safety mechanism and in turn engage the reverse gear, a mechanism is required to actuate the reverse cable and is covered in section 5.5.3. Alternatively, the reverse locking mechanism could be modified to ensure that the mechanism is permanently engaged. Due to safety concerns and to prevent possible damage to the gearbox, this modification to the gear system was not undertaken.

The TRX-300 features three-indicator lamps in the centre of the handlebars, as seen in Figure 5-33. These indicator lamps provide a visual method of detecting if the gearbox is in the neutral position or if the reverse gear is engaged. It also provides a warning light for the temperature of the engine oil.



(1) Reverse indicator lamp
(2) Neutral indicator lamp
(3) Oil high temperature indicator lamp

Figure 5-33 TRX-300 Indicator panel (Appendix J)

Through examination of the wiring diagram contained with the manual for the TRX-300, it was determined that the indicator lights work using a set of switches that when active, complete the lamp circuit by grounding the negative terminals of the lamp causing them to illuminate, as shown in Figure 5-34 Indicator light activation mechanism.

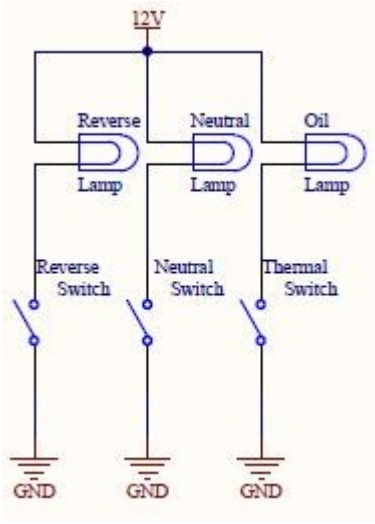


Figure 5-34 Indicator light activation mechanism

5.4.1 Design Considerations

The main objective of this system is to provide a method of changing gears on the quad bike without user intervention, while still allowing the user to change gears manually using their foot, when manual changing is required or preferred. After consulting with several of the engineering technical staff at Massey University, two designs that were considered; the first was to use a set of two spring return solenoids that when energized would either pull the gear leaver up or down. As long as the solenoids provided the necessary travel distance at the position at which they are coupled to the gear changing leaver, and can apply the required force at this position, then they would be capable of changing the gear over a very short period.

Depending on the size of the return spring used, most solenoids requiring less than a second to move from limit to limit under no load conditions. The solenoids used would require a long travelling stroke, with the gear leaver positioning the solenoids at around the centre of their travel stroke. This would allow the gear leaver to travel in both directions freely, as well as allowing the same freedom of motion if used manually, assuming the solenoids remained unpowered.

The second design featured a leaver arm on the gear pedal, which uses an actuation wheel to manipulate the arm. This design uses a motor to rotate an actuation drum to apply a force to the leaver arm mounted on the gear-changing pedal, this drum features two wheels made out of bearings, shown in Figure 5-35. These pusher wheels reduce the friction between the drum and the leaver arm. During actuation of the leaver arm, the point at which the arm and drum contact one another changes, as the angle on the leaver arm changes. The actuation drum, driven by a motor, requires a positional sensor so that a controller can know the relative position of the wheel to the leaver arm.

Manual gear changing can still be achieved by returning the actuation drum to the neutral position after each gear change, by positioning the pusher wheels on the actuation wheel at a distance that allows the leaver arm to move past the pusher wheels when in a set position. This neutral drum position shown in figures Figure 5-35 and Figure 5-36 shows the movement of the leaver arm (green) past the bearings mounted on the actuation wheel (red).

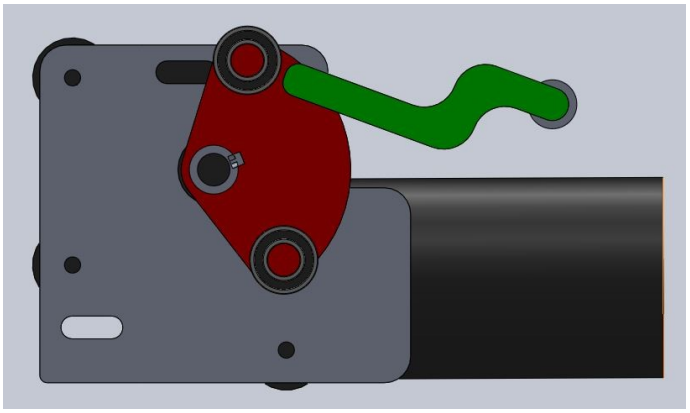


Figure 5-35 Neutral Drum position 1 clearance

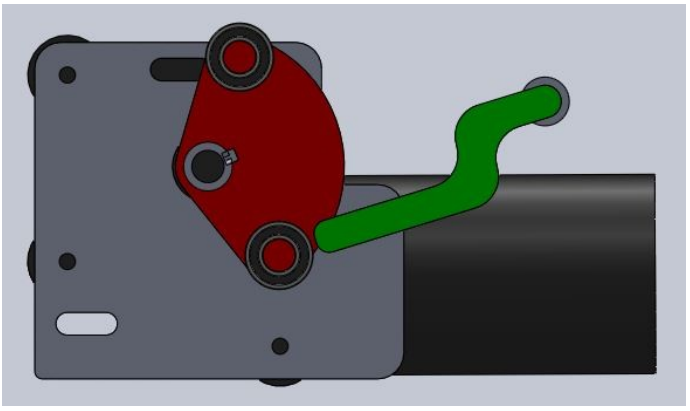


Figure 5-36 Neutral Drum position 2 clearance

Using either method will require a method to detect the gear change, the neutral and reverse gears can be detected using the indicator lights available. The TRX-300 does not feature any inbuilt method for detecting when a gear change has occurred, apart from the fore mentioned indicator lights. The solution for this problem uses limit switches to detect when the gear changing leaver has passed the position at which gear changing occurs, and thus a pulse can be sent to the controller circuit that will indicate a gear change has occurred.

5.4.2 Torque Measurements and Calculations

To measure the torque required for actuation the gear leaver and successfully change gears, a set of weights were used to measure the minimum force at which a gear change occurred. These weights were used instead of the portable scales used in previous experiments, due to the gear change arm requiring negligible force to be applied to manipulate the gear leaver once a gear change has occurred.

A 2kg, 5kg and multiple 50g sets of weights were used to apply the necessary force on the gear leaver. Using these weights, the smallest weight capable of triggering a gear change from super low to neutral was seven kilograms. With the force from the seven-kilogram weight guaranteed to trigger a change. The weights were positioned at a point that was 185mm from the pivot point, resulting in a torque of 12.7 Nm applied to the gear leaver.

$$\tau = (7.0 * 9.81) * 0.185 \quad \textbf{5.21}$$
$$\tau = 12.70\ Nm\ (2sf)$$

Due to the difficultly to measure a down change in gears, where the gear leaver must be moved up, it has been assumed that the force-required move the leaver up is similar to force required to change the gearbox to a higher gear. Thus, the minimum torque required to change gear in the gearbox was been calculated as 12.7 Nm.

To determine the torque required to operate the pusher wheel, the position at which gear change is achieved must be determined. The vertical displacement from the neutral gear leaver position and the point at which it changed was 50mm and 53mm for the up and down gear change respectively. These measurements were taken from at a chosen point with a distance of 176mm from the pivot point. This gives the angle of 16.5 degree for the up when the gear leaver in up, and 17.5 degrees for the down position.

$$\theta_{Up} = \sin^{-1}\left(\frac{50}{176}\right) = 16.5° \quad \textbf{5.22}$$

$$\theta_{Down} = \sin^{-1}\left(\frac{53}{176}\right) = 17.5° \quad \textbf{5.23}$$

Figure 5-37 shows a 3D model built in SolidWorks to test the viability of the proposed system using an image of the space behind the gear leaver. This image was scaling to allow for a 1:1 3D model of the physical mechanism and position the leaver arm and pusher wheel correctly.

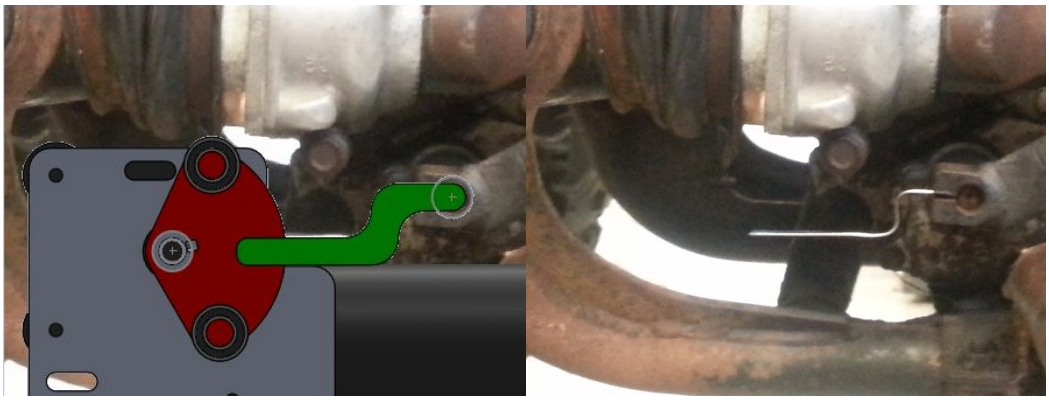


Figure 5-37 SolidWorks model (shown left) to simulate gear mechanism (original image right)

Using this, the motion of the leaver arm and pusher wheel was modelled, with the angles for the up and down gear change modelled to determine the point where the leaver arm and pusher wheel make contact. The centre distance between the wheels on the drum and the drum pivot point is 41mm, and the point at which the leaver and pusher wheel contact

was measured by setting their relative positions. When system changes down a gear, the drum and leaver arm come into contact at a point 66mm from the pivot point of the gear arm, with a 76mm length measured for changing up a gear.

From this model, the point at which the maximum force will need to be applied is where the leaver arm will be at its shortest length; in this case, it will be when the arm is 66mm in length when the system is changing down a gear. From this, a force of 192.4N will be required to act on that point to achieve a torque on the gear leaver equal to 12.7Nm.

$$F = \frac{\tau}{d} = \frac{12.7}{0.066} = 192.4N\ (1sf) \quad \textbf{5.24}$$

Therefore, from this a torque of 7.9Nm is required from the motor driving the pusher wheel, which has a leaver arm of 41mm.

$$\tau = 192.4 * 0.041 \quad \textbf{5.25}$$
$$\tau = 7.9\ Nm\ (1sf)$$

Using windscreen motor, the same method as described in section 5.2.1 was used to calculate the torque, with a final calculated stall torque of 13.66Nm. From this, it was concluded that the available motor would be more than sufficient to operate the gear changing mechanism, with a safety factor of at least 1.73. This safety factor is based off the estimated torque required to operate the gear leaver, with the actual minimum torque required to trigger a gear change between 9.26 Nm (5.1kg weight) and 12.70 Nm (7 kg weight). This torque range will compensate for any discrepancy in the positioning of the mechanism, in regards to the leaver arm distances not matching the model.

5.4.3 Integration and Testing

To ensure the system was positioned to match the Solidworks model, a guide plate was fabricated out of 2mm aluminium to ensure the centre points of the motor and gear shaft matched. With the centre points matched, a steel plate was used to mount the aluminium motor mounting plate, which was then wielded into position. A leaver arm fabricated out of 20mm steel plate and wielded to the existing leaver arm. This thickness provides an ample width for the 6mm drum wheels in case the actuator drum is not parallel to the leaver arm.

To test the functionality of the gear changing mechanism, the motor was run in both clockwise and counter clockwise directions to attempt gear change. The results of this test showed that the mechanism was capable of actuating the gear leaver. This confirmed that the earlier assumption in which the force required to actuate the gear leaver up was equal to or less than the force required for a downward actuation. However, this did reveal a problem that over rotation of the pusher wheel past its digital limits will cause the system to physically jam, within the mechanism disassembly required to return it to normal operation. This should not be a problem as long as the control system functions correctly.

Figure 5-38 A hydraulic jack was used to set the gear change point

5.5 Cable Controlled Mechanisms

Excluding the rear brake cables, the TRX-300 has three mechanisms that are controlled via cables. These are the throttle cable, choke cable and reverse engage cable. Both the throttle and choke cable connect to the carburettor, controlling the fuel supply to the engine. The reverse cable connects to the gearbox and allows the user to enable reverse gear selection from the neutral gear position. To ensure full control of the quad bike, a set of mechanisms needs to be developed to actuate these cables to any point within their extension range.

5.5.1 Throttle Cable Mechanism

The TRX-300 uses a thumb trigger to extend the throttle cable, with the cable anchored to a leaver arm coupled to the thumb trigger, with the throttle leaver mechanism shown in the Figure 5-39 below. For a computer to control the throttle, an electrical interface mechanism needs to be implemented. To design such a mechanism the range of motion needs to be determined for the throttle cable, as well as the force required to extend the cable to its full extension.

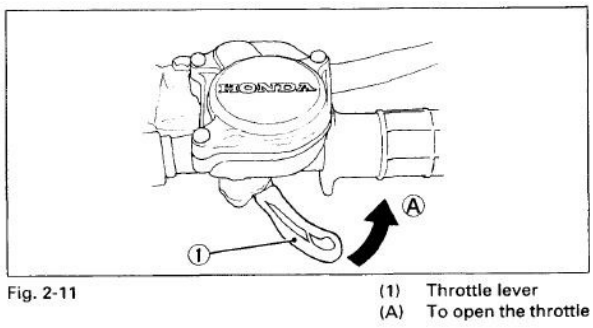


Figure 5-39 Thumb controlled throttle mechanism (Appendix J)

Figure 5-40 shows the functionality of the manual throttle mechanism to extend the throttle cable when actuating the thumb leaver shown in Figure 5-39.

Figure 5-40 Original manual throttle actuation system

A set of portable luggage scales was used to measure the force required to pull the throttle cable to its full extension, with measurements of 2.16Kg and 3.015Kg taken. Due to this variability in the measurements, an over estimate of 4kg was used for the force measurements resulting in a force of 39.24N being required to ensure reliable actuation. The throttle cable has a travel length, L, of 25.5mm shown in Figure 5-41.

Figure 5-41 Measurement of the cable extension

Two mechanisms have been considered to control the cable. The first used a linear actuator to pull the cable, though linear actuators that can provide the required force as well as a travel speed capable of adjusting the cable. Linear actuators that can meet these requirements were deemed too costly for the system compared to other options that could be available, with the added problem of locking the throttle in position if the power is cut.

The second option is based on the existing manual throttle design, which uses an arm-based system to pull the cable by rotating a leaver arm. A straight leaver arm similar to the one shown in Figure 5-40 would cause the angle at which the cable was pulled at to change, as indicated in Figure 5-42 and Figure 5-43.

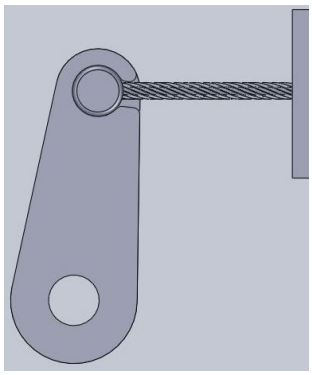


Figure 5-42 Cable leaver arm position 1

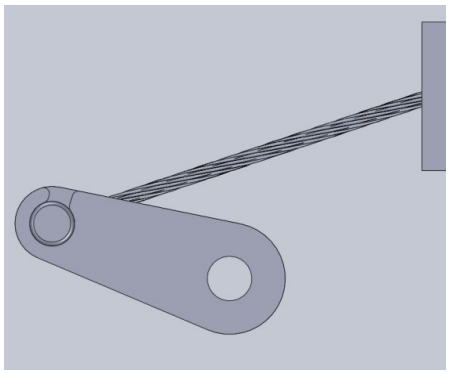


Figure 5-43 Cable leaver arm position 2

To make the system more compact, a rotary drum was used in place of a leaver arm. This provides a system that maintains the angle of the cable, with the cable angle between the cable and block shown in Figure 5-44 is the same as that in Figure 5-45 where the cable has been actuated. This prevents any possible wear on mechanism from the housing.

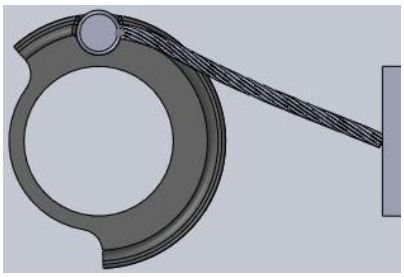


Figure 5-44 Cable drum position 1

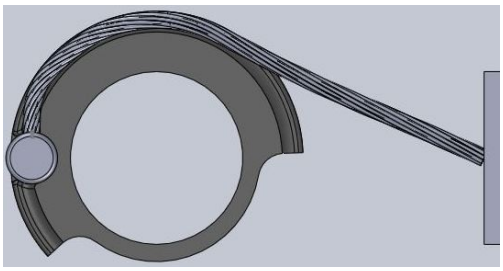


Figure 5-45 Cable drum position 2

By wrapping the cable up on the drum, the effective leaver distance remains constant while reducing the size of the mechanism by using the circumference of the drum instead of the radius to extend the cable by the required length. Using a drum based design with a diameter d of 25mm, driven by a servo with a maximum turning angle of 160 degrees and a torque rating of 1.28 Nm (2sf). To ensure this design met the requirements the angle θ, required to pull the cable to full extension needs to be confirmed, as well as the torque, τ required to extend the cable.

$$\theta = \left(\frac{L}{d*\pi}\right) * 360 \qquad \textbf{5.26}$$

$$\theta = \left(\frac{25.5}{25*\pi}\right) * 360 \; = \; 116.9° \text{ (1sf)}$$

$$\tau = F * d = 4.0 * 9.81 * 0.0125 \qquad \textbf{5.27}$$

$$\tau = 0.49\ Nm\ (2sf)$$

Equations 5.26 and 5.27 show that the selected servo is capable of reaching full extension on the throttle cable based on the angle of rotation and torque available to from the servo.

5.5.1.1 Cable Drum Design

The cable drum needed three features to be included in the design, a coupling mechanism to the servo, a coupling point for the cable, and a curved surface to wrap the cable. The throttle cable features a 5.94mm cylindrical lug on the end of the cable. This is the mounting point on the standard throttle system.

Servos have a number of different horns are available that attach to the servo drive spline, with a 10mm horn chosen to be used for this design. To couple the cable to the drum, a 6mm hole has been used to mimic the method used in the original system. Due to the size of the servo horn the diameter of the cable drum increased to 29mm to ensure clearance between the servo horn and cable lug. Recalculating the torque and angle due to the increased drum diameter shows the mechanism is still capable of operating the cable with a torque of 0.61 Nm (2sf) and a travel angle of 100.8 degrees (1sf). A 3mm channel acts as a cable guide to ensure the cable remains on the drum and thus the leaver arm distance remains constant.

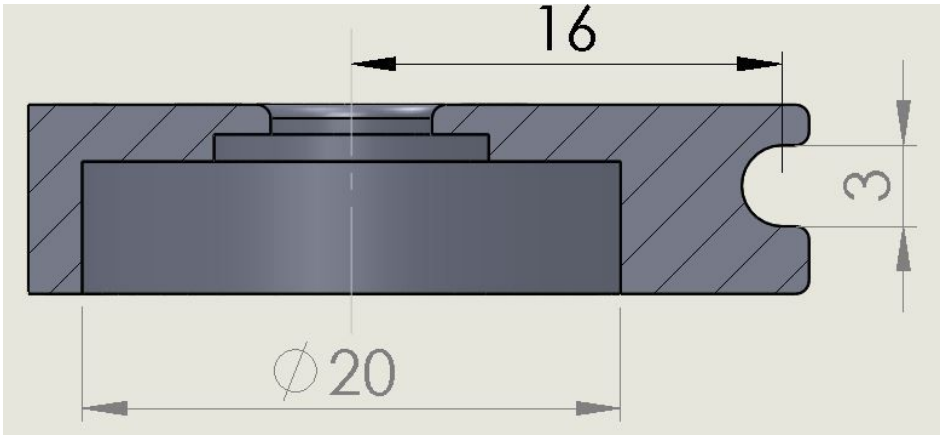


Figure 5-46 Cable channel diameter

5.5.1.2 Mechanism Housing Design

The housing mechanism was designed around the components by modelling them using 3D design software and assembling them digitally. This ensured that all the components are aligned to prevent unnecessary wear. The point at which the throttle cable attaches to the housing was aligned to be tangential to the cable drum. This keeps the wire cable parallel to the cable housing, preventing rubbing.

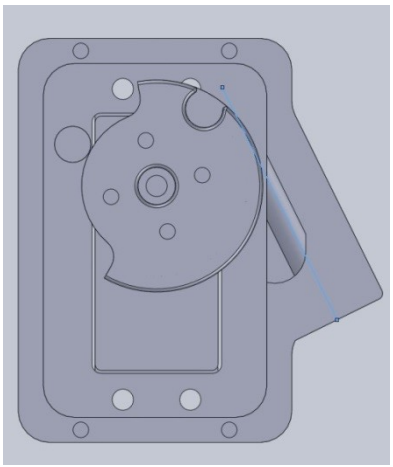


Figure 5-47 Throttle Housing Design, showing tangential relationship between cable and drum.

Figure 5-48 Servo controlled throttle cable

The resulting design of this throttle mechanism functioned with the design parameters, allowing the computer to control the throttle via the speed control board to set the position of the servo throughout the throttle cables range of travel with a positional accuracy of 0.25mm ± 0.03mm, based on the specifications of the servo used.

5.5.2 Choke Cable Mechanism

The existing choke mechanism uses a notched slide to extend the choke cable to one of three positions, off, half choke, and full choke corresponding to an extension of 7.3mm from off for half and 10.6mm for full choke.

The force required to pull the choke cable to full extension was considered negligible when compared to the throttle mechanism, with the choke assembly featuring a small return spring to ensure that tension is maintained between the choke cylinder and the carburettor. This ensures that the correct position of the choke cylinder is maintained as defined by the extension of the choke cable. The choke cable uses the same mounting lug mechanism as the throttle cable as shown in Figure 5-50, thus a similar design was used to mount it to the system. The only requirement of the choke cable mechanism is that it can be set to one of two positions, with an accuracy of within a few millimetres.

Due to the small force required to extend the choke cable, an over engineered solution was used by re-using the design of the throttle cable mechanism as described in section 5.5.1 above. An alternative system was conceived, that uses two solenoids to pull the cable to the appropriate position. By attaching, the solenoids in series and limiting their individual strokes, it is able to match the extension of the cable when the cable is extended to the half or full choke position. By trimming the stroke of the first solenoid to match the distance required to extend the cable to the half choke position, the cable is extended to the required position by energizing the first solenoid. To reach the full choke position, the second solenoid can either, be set to travel the full stroke distance from the off position to full choke, or it could be used in conjunction with the first solenoid to make up the distance from half choke to full choke.

As described in the section 5.5.1, the servo mechanism is capable of extending the cable within the required range, with an accuracy of 0.25mm ± 0.03mm. While the same basic throttle mechanism design was used for the choke cable, one modification was necessary to

make the housing compatible with the choke cable. Unlike the mounting system for the throttle cable, the choke cable does not use a threaded tip to connect to the housing; instead, it uses a notched cylinder featuring a step down, which seats the cable against a plastic housing on the slider mechanism, shown in Figure 5-49 and Figure 5-50.

Figure 5-49 Manual choke slide

Figure 5-50 Choke Cable - Mounting Point[25]

Machining a casing that includes notches to align with the notches on the cable is an impractical solution. A 3D printed ABS housing could be used to secure the cable, though this was dismissed, as it would not provide a long-term solution to ensure reliable mounting. For the choke, cable to function correctly the choke housing must prevent the cable sleeve from travelling with the inner cable when it is moved. To do this a simple clamping system was devised that uses an interference fit between two plates to stop the choke sleeve from being pulled with the choke cable. This is helped by using the flange on the cable to prevent forward movement. This is possible with the initial part of the cable sleeve having a diameter of 6mm with the flange section having a diameter of 10mm, with an interference fit designed into the housing to ensure adequate clamping.

To clamp the cable the housing was separated into two pieces, the main piece mounted on the servo and used a semicircle channel with a diameter of 6mm to align the choke cable to be tangential to the drum as shown in Figure 5-51. To ensure effective clamping was maintained, the centre point of the semicircle was raised up by 0.20mm; whereas the corresponding semicircle on the clamping lid of the mechanism did not include this offset with the resulting design, ensuring there would in an interference fit as shown in Figure 5-52.

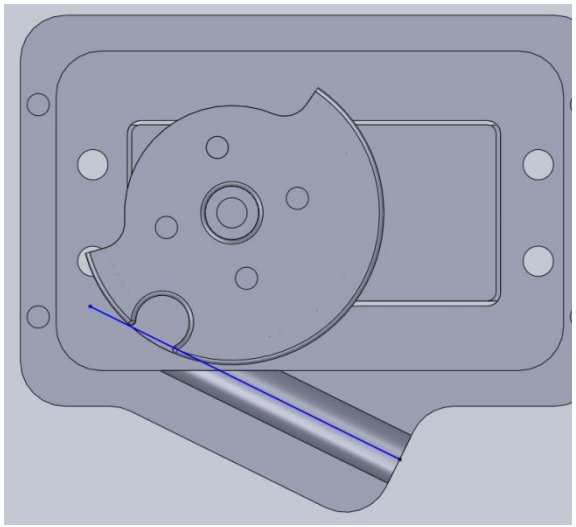


Figure 5-51 Choke Cable housing, aligning the cable to be tangential to cable drum

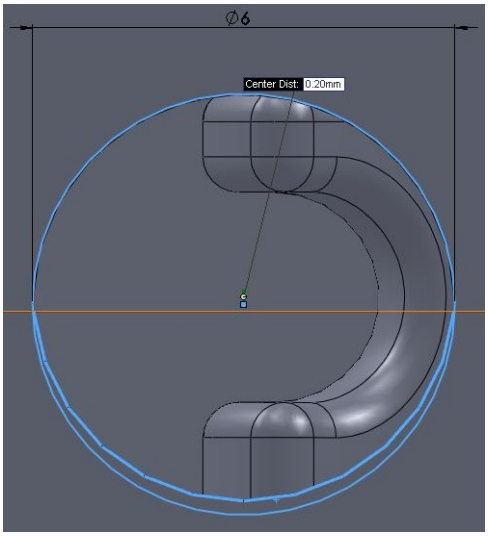


Figure 5-52 Choke Cable clamping system, two 6mm diameter semi circles with a 0.2mm interference

The resulting design of this choke actuation mechanism functioned within the design parameters, allowing the computer to accurately position the choke to either the Off, Half choke or Full choke position. The mechanism is also capable of extending the cable anywhere up to the Full choke position within an increment of 0.25mm ± 0.03mm from the off position. Testing was undertaken using a Turnigy servo tester, which successfully extended the choke cable up to the travel limit of the cable.

5.5.3 Reverse Cable Mechanism

To prevent damage to the vehicle gearbox that would result from the accidental selection of the reverse gear, the TRX-300 includes a reverse selector cable that prevents the user from engaging the reverse gear without actuation. The standard method of actuating the reverse cable uses a selector knob attached to the rear brake leaver, see Figure 5-53, which is engaged via the reverse selector knob to couple the brake leaver and reverse cable. This mechanism ensures that the brake is applied before the reverse gear is engaged. Therefore, the control system should ensure that the vehicle is stationary before the reverse gear is engaged. The reverse cable uses the same 6mm lug to couple the cable to the handle bar mechanism as used in both the throttle and choke cables.

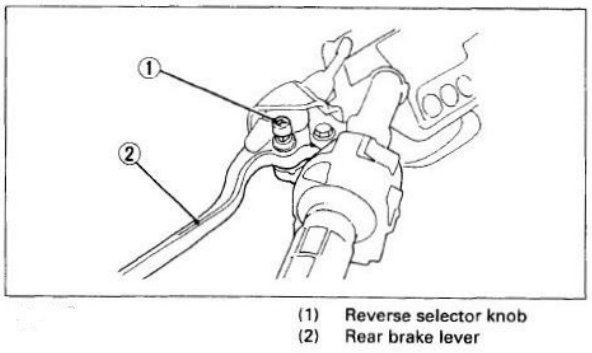


Figure 5-53 Reverse selector and cable actuation method located on the left brake control (Appendix J)

To control the reverse cable, the force required to pull the cable must be determined as well as the required extension of the cable to allow the gearbox to change from neutral into the reverse gear. A method similar to the one described to measure the rear brake in section 5.3.1, was used to measure the force required to actuate the reverse cable. To prevent the force required to actuate the rear brake from interfering with the measured force, the rear brake cable was disconnected from the handle allowing the reverse cable to be engaged without interference.

The portable scales measured a reading of 6.1Kg at a distance of 100mm from the pivot point of the brake handle, resulting in a torque of 5.98 Nm (2sf) to activate the reverse cable to a point in which the gearbox could be changed from neutral to the reverse gear.

$$\tau = 6.1 * 9.81 * 0.1 = 5.98\ Nm\ (2sf) \quad \textbf{5.28}$$
$$F = \frac{5.98}{0.04} = 149.5N$$

The reverse cable is anchored to the brake handle at a distance of 40mm from the pivot point, resulting in a force of 149.5 Newtons. Thus, 150N is the minimum force required

to extend the cable to the point in which the reverse gear can be engaged. At the point in which the reverse gear could be engaged, the extension of the reverse cable was measured at 12.5mm.

To ensure the designed mechanism operates reliably, the system is to have at least 33 percent more torque than required as a factor of safely, ensuring the reliability of the actuation mechanism by preventing maximum loading on the motor. Therefore, the mechanism must be capable of providing a force of at least 200N on the cable. Two options for actuation were explored, a solenoid or a servo mechanism, similar to the systems used to control the Throttle and Choke cables.

A solenoid has the advantage of it only requires to be powered up to move the cable the required distance, while the servo requires a digital signal to set the distance. To ensure a solenoid was capable of providing sufficient force to achieve the required extension on the cable, a large device would be required as, with solenoids having a relatively weak pulling force initially that gets stronger the closer it gets to its internal electromagnet. Solenoids with the power required to actuate this design were deemed not as cost effective when compared to a servo-operated mechanism. However, it is recommended that a solenoid mechanism be reconsidered for future prototypes.

The servo selection was based on the following factors, cost, reliability, power requirements, and torque output. To ensure reliability the servos considered would have to use metal gears and be capable of operating at 6V. Based on these factors a cost effective servo that was capable of providing 2.56 Nm (2sf) of torque at 6V was selected. Using the selected servo, the maximum radius to provide the required 200N of force is 6.4mm.

$$r = \frac{2.56}{200} = 12.8mm \quad \textbf{5.29}$$

The cable drum features a 6mm mounting point that uses the same design as the previous cable mechanisms and uses the clamping method outlined in section 5.5.2 to hold the cable in place. As the cable extension required, L is only 12.5mm, a full drum design was not required, using the circumference of the drum as the maximum cable, the drum only needs to provide an angle of wrap, θ_w of 57.3 degrees

$$\theta_w = \frac{360}{\frac{c}{L}} = \frac{360}{\frac{2\pi * 12.5}{12.5}} = 57.3° \quad \textbf{5.30}$$

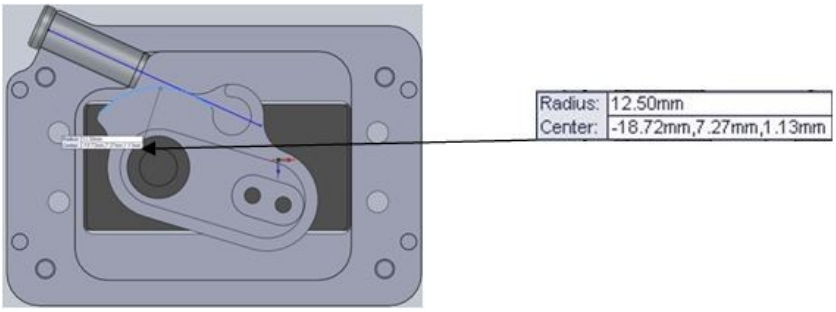


Figure 5-54 Reverse cable actuator design

The design shown in Figure 5-54 was tested using a servo tester at 6V and was capable of extending the cable to a point in which the gearbox was capable of changing into the reverse gear.

Chapter Six Electrical and Electronic Systems and Control

The primary control system is developed based on a single board computer, which will determine the speed, heading and position of the vehicle, as well as decision making in order to achieve its objectives. While the computer governs the high-level functions of the vehicle, it would not provide an efficient method of control for all the mechanical subsystems described in Chapter Five. Therefore, a series of microcontroller based modules or subsystems were developed to ensure that the individual subsystems could be effectively controlled with only basic commands being sent. This allows the computer to instruct the subsystem to perform a task and wait for a response on the status of the said task. This also allows the basic functionality of the vehicle to be maintained as the system can operate without computer commands.

6.1 Bike Management Circuits

To operate the vehicle successfully, the system must be capable of detecting and modifying the signals and controlling the vehicle. This includes power switching that would normally be achieved via physical switches, detecting visual indicator statuses, as well as monitoring engine status readings, all of which are normally undertaken by the user.

6.1.1 RPM Detection

Detecting the engine status is essential for autonomous control of the quad bike, as the system needs to control the speed of the bike as well as adjusting the selected gear depending on speed and engine load. Measuring the RPM of the engine is one way of detecting if the engine is operating, where a human operator could tell with their ears but a robot (without ears) needs other method of detection.

There are several methods to implement RPM detection including having a mechanical pickup mechanism on the camshaft in the engine, or off the gearbox using a hall effect or other such sensors. A relatively simple method used in automotive testing is having a pickup wire around one of the spark plug wires on the engine. This is done by wrapping a wire around the spark plug wire leaving one end unconnected from a circuit(capacitive coupling), or by connecting both sides of the wire to a circuit (inductive coupling).

A capacitive coupling method was selected as it only required one side anchored of the wire to be connected, and thus would be easy to wrap around the spark plug wire. Each time a spark plug fires a large voltage is sent from the induction coil to the spark plug. To protect and provide reliable spark detection to the microcontroller, the large voltages that can be induced in the pickup wire the signal must be converted to a logic level signal. This is done using a resistor and transistor to provide a 5V signal for detection with the resistor connecting the pickup wire to the base of the transistor so that the current driving the base pin on the transistor is limited. This results in a signal that oscillates at high frequency coupled with a low frequency on-off signal as seen in Figure 6-1.

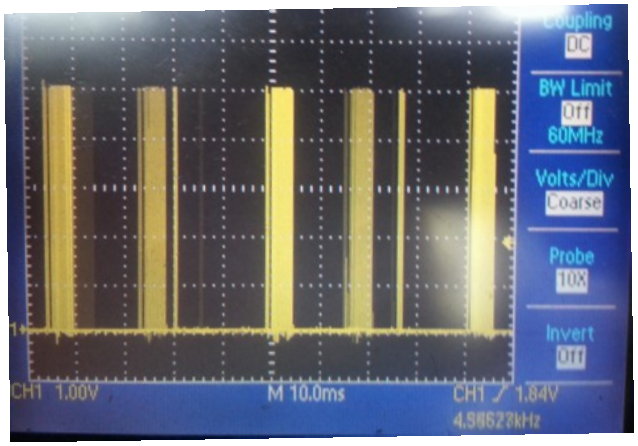


Figure 6-1 Raw 5-volt inductance signal

To smooth this signal in Figure 6-1, a capacitor was placed in parallel to the resistor on the emitter side of the transistor to smooth the high frequency signals and provide the stable signal shown in Figure 6-2.

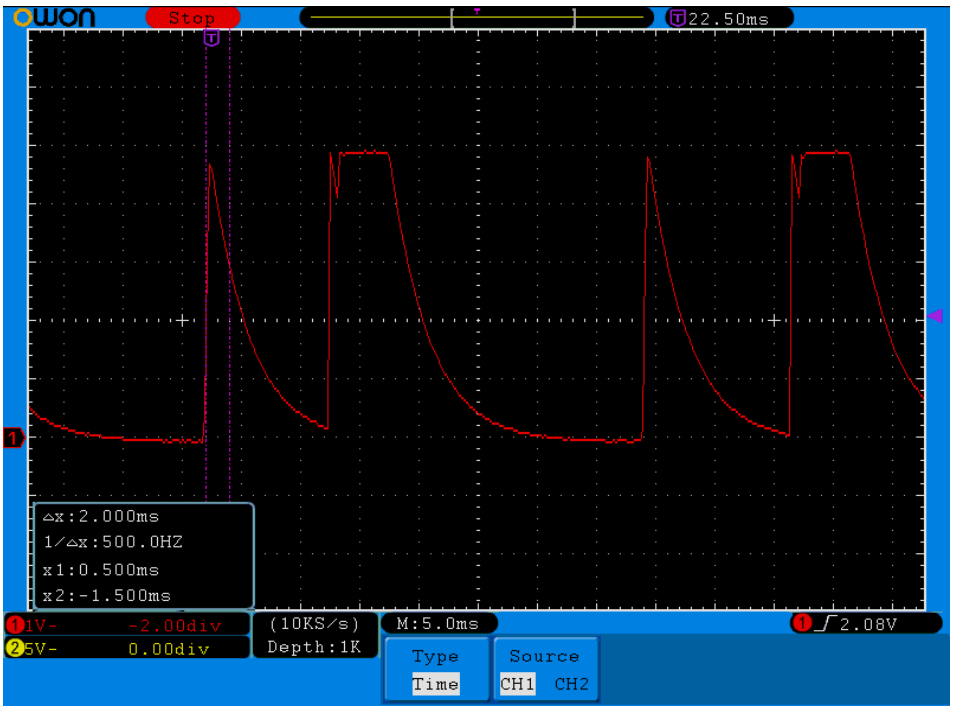


Figure 6-2 Capacitor Smoothed inductance signal, plus noise width at 3-volts

Connecting to an interrupt pin on the Arduino microcontroller allowed the number of pulses to be counted every second. After testing the integrity of the signal at higher RPM, it was discovered that the initial pulse of 2.0ms shown in Figure 6-2 above, is actually noise or a misfire signal that only occurs when the engine runs at speeds close to idle. This was confirmed after the noise was absorbed into the desired signal once the engine exceeded a fixed RPM, which was higher than the idle speed of the motor.

This is noise absorption is shown in Figure 6-3 and Figure 6-4, with the oscilloscope screen shots having the identical time scales, with the higher RPM signal captured being free of such noise.

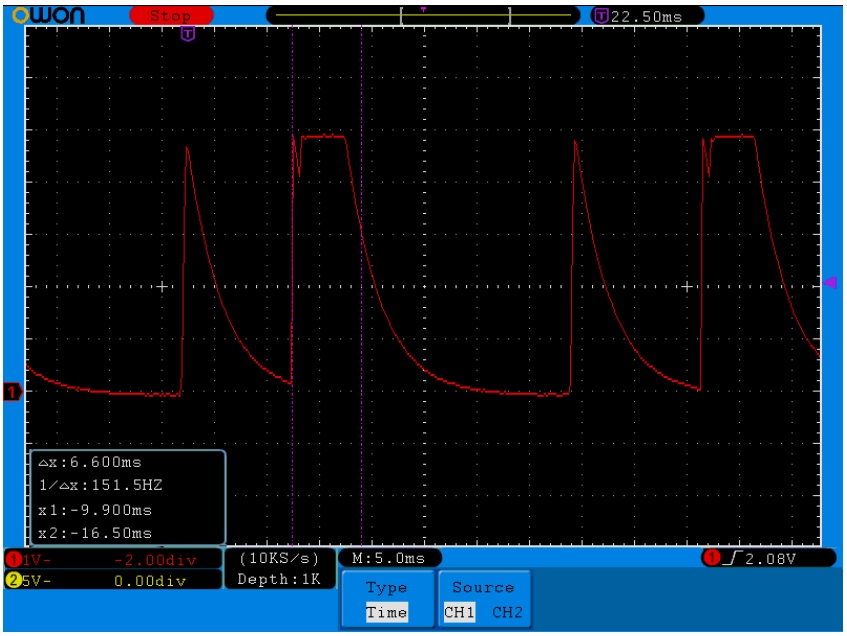


Figure 6-3 Low RPM

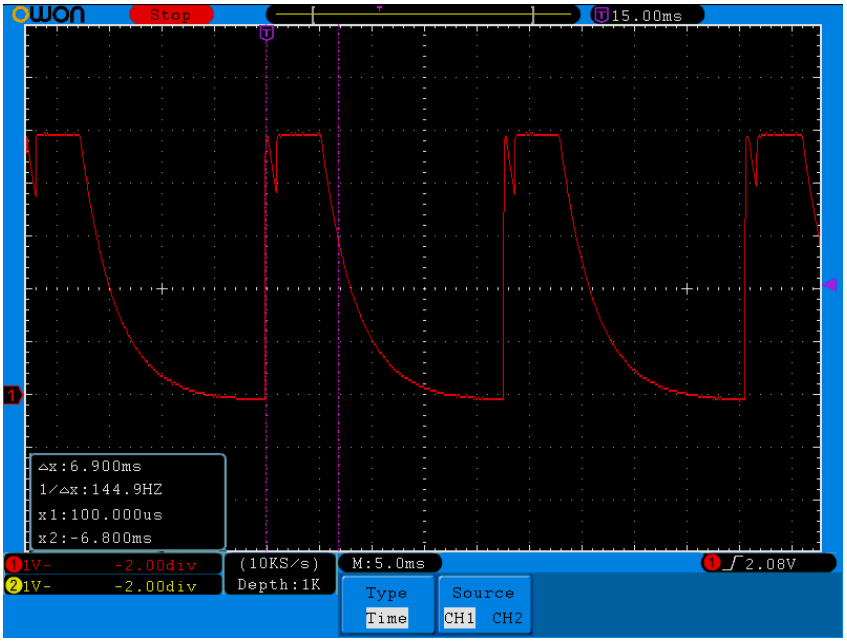


Figure 6-4 High RPM, with noise absorption.

Using an oscilloscope to measure the pulse width of the noise and the actual signal at the 3V level shows that the width of the noise at 3V is 2.0us (Figure 6-2) and the width of the signal pulse is 6.6ms (Figure 6-5).

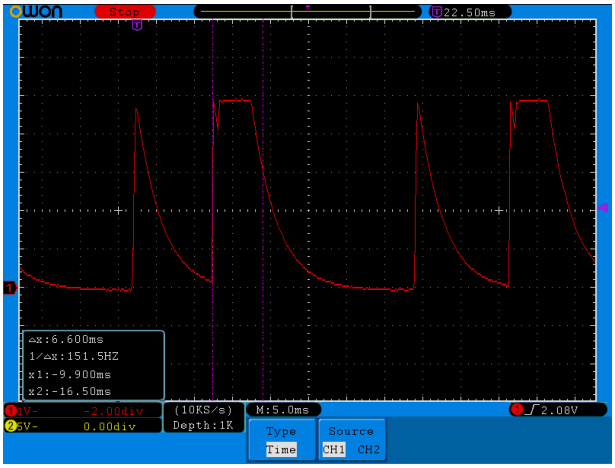


Figure 6-5 RPM Signal width at 3-Volt level

To provide a reliable signal the noise 2.0ms noise had to be removed from the output signal. To ensure this, any pulse that was shorter than 3ms (at 3V) had to be filtered out. This was achieved via an RC circuit that was implemented to ensure that any pre filter signal detected would have to be at 5-volts for greater that the 3us to ensure the 2us pulse in Figure 6-2 was absorbed, before the post-filter signal would exceed the 3V threshold. To filter the 3us pulse a RC filter was designed with an RC value of 3us with a 330nF capacitor requires a resistor value ≈ 10KΩ.

$$R = \frac{3*10^{-6}}{C} = 9.09\text{K}\Omega \qquad \textbf{6.1}$$

The resulting post RC-filter waveform shown in Figure 6-6 shows the resulting waveform with the true RPM signal represented where the waveform exceeds 3-volts as represented by the yellow line.

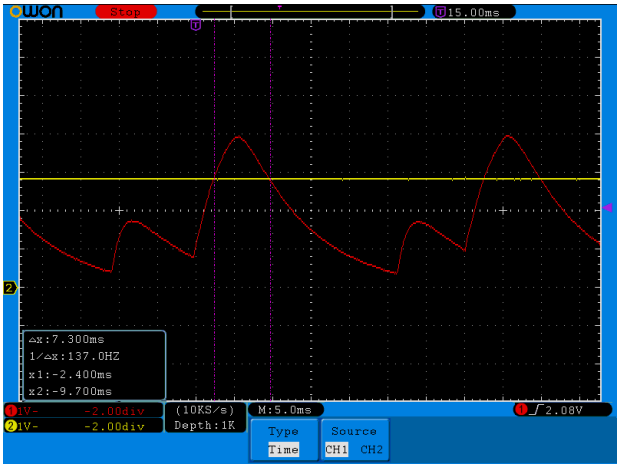


Figure 6-6 RPM signal, post RC filtering

A comparator is used to convert this filtered signal back into a clean 5V pulse that can be detected by the microcontroller, with the reference voltage on the comparator set to 3V, a clean digital pulse, devoid of unwanted noise is generated as shown below. The yellow pulse is the final output signal after the comparator and is seen below overlaid over the filtered signal.

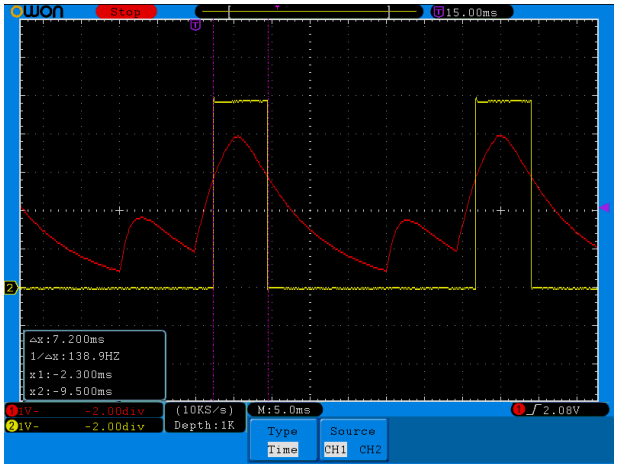


Figure 6-7 Digitized RPM signal post filtering

Figure 6-8 below shows the RPM detection circuit, with pin two on the H-Pickup header acting as the attachment point for the pickup wire, with pin 1 providing a ground pin for testing purposes. The inducted signal switches the transistor (Q5), with the voltage drop over the resistor (R17) producing the signal in Figure 6-1 which is smoothed by capacitor 1 (C1) to produce Figure 6-2. The RC filter (R18, C2), filters out the false signal noise producing Figure 6-6, and is finally digitized by the LM339 comparator (U1D) using the reference voltage provided by the variable resistor (R19) to produce the digital signal shown in yellow in Figure 6-7 above.

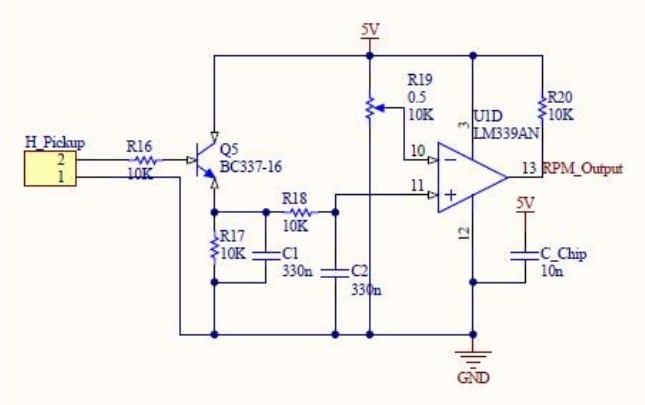


Figure 6-8 RPM Detection Circuit

6.1.2 Power Switching

Due to the vehicles age, many of the quad bikes systems such as light, horn, starter etc, are operated via buttons and switches, and were not designed to be operated via electrical signals. Therefore, a digital interface needs to be implemented to allow the computer system to control these systems.

The Lights and Engine Run control are both operated via selector switches that are toggled between multiple positions by the user. The light control has three possible outputs determined by the state of two switches. One switch determines if the lights are on or off, with the other switch determining the intensity of the lights as either high or low. This can be simplified to the three output states of off, low beam or high beam. The Engine run control

controls if the engine will operate, with three possible positions on the switch resulting in two outcomes, Engine run or engine off.

Two push buttons control the starter motor and the bikes horn, and while the horn is not an essential feature on the bike, it can be used by as a safety feature to warn people and animals of its approach, particularly around blind corners. The start button energizes a relay to activate the engines starter motor, with a recommended maximum operating time of 5 seconds as specified in the vehicles operating manual. The TRX-300 features an automatic fan control system, which controls the fan to ensure the engine is kept cool, primarily when the vehicle is stationary, preventing the engine from overheating due to the lack of airflow past its cooling fins. Unfortunately this system did not function on the acquired bike, thus a control system is required.

To ensure easy integration into any system, the control interface board was designed to accept 5-volt signals as inputs to control the 12V outputs, allowing direct interfacing from a microcontroller to occur without level shifting. By using a transistor-Mosfet arrangement, shown in Figure 6-9, a microcontroller can switch high current devices, with the chosen NT2955G Mosfet, capable of transmitting up to 12A of current at 60-volts. This setup can also be used to switch a device using a PWM input signal, allowing the controller to vary the intensity of any connected devices output. The current drawn by these devices was measured using an ammeter in series with the component with the following current draws measured; Start button current, 2.2 Amps, Fan current, 2.0 Amps, Horn current draw, 1.6 Amps and Lights circuit 2.2A

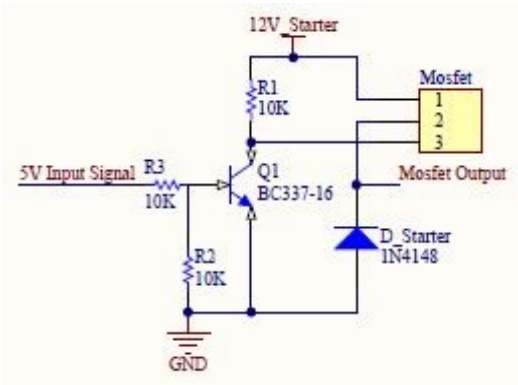


Figure 6-9 Transistor Mosfet Driver

Depending on the temperature of the engine and the surrounding environment, the speed of the fan will need to be varied when run, thus speed control system using a Mosfet instead of a relay was implemented, allowing a PWM signal to be used to control the fan speed. The horn control system was designed to accept a PWM signal to adjust the frequency and therefore the sound that the horn emits. This allows different tones to indicate different audio messages to the humans within the vicinity of the vehicle. While this horn switching preformed to specifications regarding the switching, the tone alternation did not perform as expected. The PWM control over the horns output was requires more work to be implemented in the system code. Currently the hardware is capable of performing this function.

The light controls on the bike feature a high low setting to vary the intensity of the vehicles headlights, and this feature was supported by the controls. As the original lights had been replaced by a single light that did not provide this capability, therefore the light intensity feature was not included within the controls circuit, with the control circuit only requiring a single output to set the lights as either on or off.

The Starter motor control only requires basic on-off capabilities, with the basic transistor-Mosfet circuit providing the necessary level of control and current capacity. Though it is specified that the starter motor be run for a maximum of 5 seconds, there has not been any in circuit protection to ensure that this run time is not exceeded; instead, it relies entirely on the microcontroller to control the run time. An analogue timing logic circuit could be used to provide this protection, though it was deemed unnecessary for this prototype design, but should be considered in later designs.

Control over the Engine run switch is required to turn off the engine and requires simple on-off functionality. As this switch control is not going to be oscillated, as well as not required to deliver high current, a relay was used to switch circuit control. To ensure safe operation of the system, the functionality of the engine run switch was maintained by connecting the relay in series with the physical switch. This allows the user to maintain control over the engine and can switch it off at will. To prevent the system from trying to start the motor when the engine run switch was in the off position, the relay was positioned after the run switch in the circuit, which allows a simple voltage detector to be positioned between the switch and relay.

This allows a microcontroller to check to see if 12 volts is present at the sensor, which would indicate the engine run switch is in the on position, and thus determine if the user controls will prevent a successful engine start. The voltage sensor mentioned uses a 3.3-volt Zener diode with a resistor in series, with an output voltage of 3.3-volt indicating that a voltage higher than 3.3-volts is present. This Zener arrangement was used due to the possibility of voltage spikes on the vehicles power system that could otherwise damage a microcontroller if a voltage divider was used instead. The output from this detector is either a 3.3-volt digital high voltage to the microcontroller or a grounded digital low, once the input falls below 3.3-volts

6.1.3 Temperature Sensor System

The temperature system consists of two measuring systems. The first is a simple LM335 Temperature sensor, which measures the air temperature within the air filter compartment located under the TRX-300 seat. This positioning provides the air temperature regardless of environmental conditions such as high wind chill factor or direct sunlight.

The second temperature measuring system uses a MLX90614 Infrared (IR) thermometer (datasheet in Appendix C) that provides a non-contact method to measure the surface temperature of the engine. This IR sensor is capable of measuring temperatures ranging from -70 to 382.2 degrees Celsius and features two data interfaces. The infrared temperature system allows high temperature measurements to be obtained without using a Thermocouple, which would require an amplification circuit to provide readable voltages to a microcontroller.

For temperature monitoring, the first interface is an I2C interface that provides a 12-bit output value and a temperature division value of 0.02 degrees Celsius. The I2C interface also allows multiple sensors to be addressed using the same two data lines, allowing up to 100 devices measured by the microcontroller on the single data line pair. This provides an easily expandable temperature monitoring system with no extra IO pins being used. While interface method is an advantage for systems that require multiple sensors, only one sensor is required for this system. The I2C interface will also use two interrupt pins on the ATmega2560 microcontroller, which is a third of the total interrupt pins available.

The second interface is a PWM (Pulse Width Modulation) output. This system only provides a 10-bit output value with a temperature division value of 0.14 degrees Celsius. While this interface provides a reduced measurement range of only -20 to 120°C, it still provides sufficient operational range to measure the surface temperature of the engine. Although this interface would normally require the use of interrupts or halting the program to measure the pulse width, it can be simplified with a filter to convert the Pulse width to an analogue voltage, which can be measured via one of the ATmega2560 analogue pins. The temperature response from the sensor output has a reduced response rate due to the pulse width to an analogue voltage conversion, though any latency within this output voltage will not affect the operation of the auto-choke or cooling fan, due to the low priority assigned to update this temperature reading.

An interface board was designed, shown in Figure 6-10, to convert and present the measured temperature reading to the microcontroller as analogue voltages, powered off a 5-volt input. Featuring two, RC filters to smooth the PWM output of the infrared sensor as well as an input header for a cable mounted LM335 sensor. The board requires a four-wire cable to provide power and transmit the analogue voltages representing the engine and air temperature.

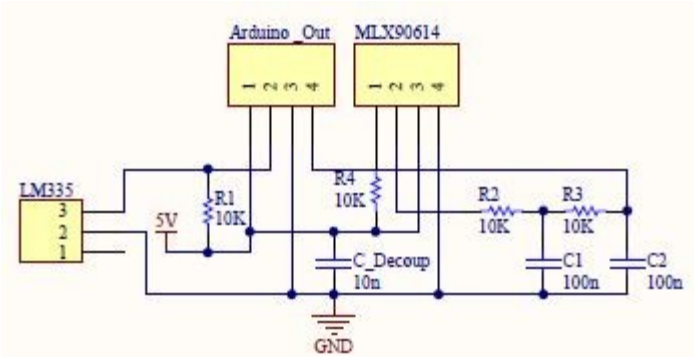


Figure 6-10 Temperature Sensor Board

The LM335 sensor within the interface circuit acts as a voltage divider circuit, with voltage drop across the sensor represented by the conversion of 10mV/°K, which converts to degrees Celsius using equation 6.2 (from datasheet in Appendix C).

$$TLM_{°C} = (V_{in} * 0.01) + 273 \quad \textbf{6.2}$$

The equation governing the smoothed voltage signal from the MLX90614 infrared sensor is more complex, with the 1/8th of the PWM period always high at the start as well as

1/8th of the PWM period as a low at the end of the pulse. Thus, the temperature measured by the infrared sensor is only represented by half the signals period.

$$TIR_{°C} = \frac{10V_{in} - 6.25}{0.179} - 20 \quad \textbf{6.3}$$

6.1.4 Indicator Lamp Detection

There are three indicator lamps located between the handlebars indicating when the Neutral gear is engaged, the Reverse gear is engaged and a high engine oil temperature. These lights illuminate when the corresponding switch, as shown in Figure 6-11, is activated, connecting the lamp to ground and completing the circuit. Because of this circuit arrangement, 12-volts will show across the open switch, which can be used to detect the illumination of the light and therefore the status of the system represented.

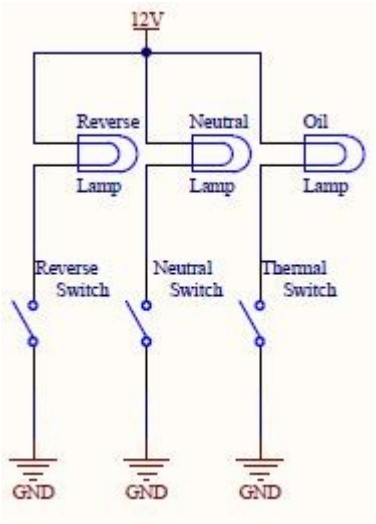


Figure 6-11 Indicator Light circuit

A simple circuit was designed to indicate the status of the individual light circuits, by providing a microcontroller with logic level voltage signals, which correspond with the status of the circuit. The circuit shown in Figure 6-12, uses the ground signal provided from the bikes switching circuit to ground the signal line to the microcontroller outputs, with a 5-volt pull up resistor used to provide a 5-volt signal when the switch is open. A diode has been included to prevent the 12-volt signal from the lamps from damaging the microcontroller circuit when the switch is open. This diode results in a 0.7-volt voltage measured on the output line when the switch is closed. This is within the 0.9-volt logic low threshold of the Atmega328 (Appendix D, page 313) and Atmega2560 (Appendix E, page 355) microcontrollers used within this project.

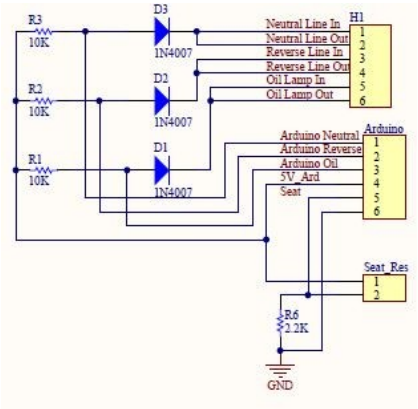


Figure 6-12 Lamp status detection circuit

The resulting logic-level output signals for this device are logic high when the lamp is off, logic low when the lamp is illuminated, therefore the status signal for the neutral gear will be logic low when the neutral gear is selected, and high when any other gear is selected.

6.2 Steering Control System

To operate autonomously the vehicle must be capable controlling the angle of the vehicles wheels and thus the direction in which it will travel. To do this it must be able to manipulate the steering angle while also being able to detect the position of the steering assembly.

6.2.1 Steering angle detection

To ensure reliable detection of the steering position, an encoder needs to be attached to the steering mechanism, for this there are two kinds of encoders available. They are either incremental or absolute encoders.

The incremental providing two pulses that result in a speed and the direction it is turning. This type of encoder can be used to provide an accurate reading of the position of a shaft by combining the number of pulses received with its turning direction, which provides a control system with an accurate position based its last known position. Incremental encoders provide a cheaper device for detecting the rotation of a shaft when compared to absolute encoders of the same quality.

Though this type of encoder has one major drawback, which is it must know the position at which it started to determine the angle at which the shaft is at, otherwise it will only know how far the shaft has rotated since power up. This is why incremental encoders need to be 'homed' to a known position on power up. Once the encoder has reached a known position, the control system can then count the number of increments in a specific direction that the shaft has turned relative to its home point, though this cannot account for any increments that could possibly be missed due to a busy system.

This is where absolute encoders surpass the incremental because they work on the principle that every position, that it is capable of detecting within a full rotation, has a unique binary identification code. Therefore, a unique position for the shafts position is available when the device is powered up. This position reading comes in the form of a binary code that is derived from a grey code sequence, which ensures that no more than one bit in the binary code changes from one position to another. This grey code is used to prevent errors caused by

slight misalignments between contacts, which can cause momentary contacts that can cause incorrect readings as the shaft rotates[26].

The mechanical steering system was designed to allow the user to control the steering manually, with a linear actuator providing an automated method to disconnect the motor from the drive train. The mechanical system is designed to ensure the encoder maintains coupled with the steering pole, regardless of the steering motor being engaged. This allows the steering controller board to provide continuous monitoring over the steering position. The Bourns ACE-128 encoder was selected to provide a 128-bit resolution on the position of the steering pole, which provides a resolution of 0.8 degrees over the 100 degree turning arc of the steering pole (Datasheet available in Appendix B).

This encoder was selected as it provided 128-bit resolution at a low cost compared to other encoders, though unlike the other encoders it does not provide the output format in a convenient format such as serial, I2C, etc. Instead, this encoder provides 10 interface pins; two are common pins where one of them must be connected to ground (GND) as the encoder works by connecting the other eight pins to this common ground to output the current position.

For a microcontroller to read the output of this system, a set of pull up resistors is required on each of the eight output pins. The recommended encoder configuration is demonstrated in Figure 6-13, with the values of P_1 to P_8 indicating bits for the grey code position of the encoder.

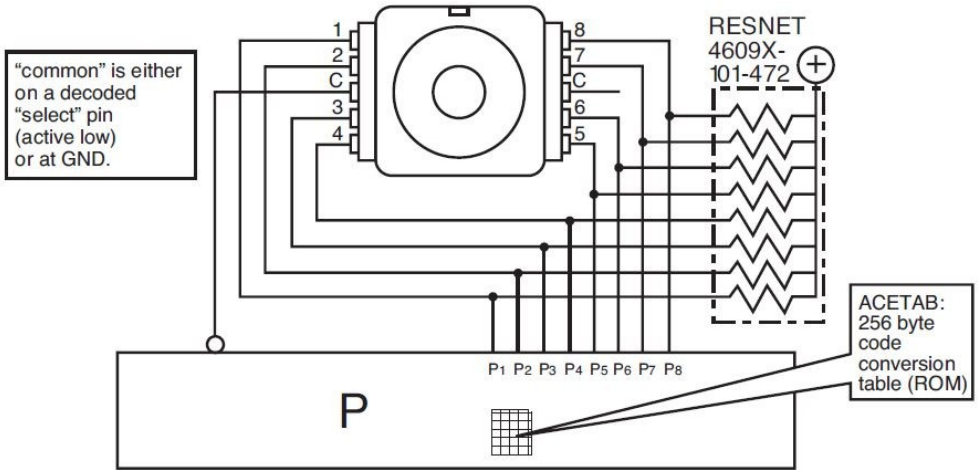


Figure 6-13 Bourns ACE-128 encoder control diagram from datasheet in Appendix B

Because the encoder works by grounding the output terminals, there is minimum voltage required to drive the output pins, though a maximum rating of DC 10-volt or 0.1 Watts is specified. This allows a microcontroller to read the digital values off the pins directly when a compatible voltage is used to power the encoder.

To provide a linear representation of the encoder output, the grey code output must be converted using a conversion table. The grey code output is converted using the conversion table shown in Appendix B coded into the ATmega328 microcontroller, resulting in a linear output for positions 0-127.

An interface board shown in Figure 6-14 and Figure 6-15 was designed to connect directly to the ACE-128 encoder and contained the pull up resistors necessary. This requires a ground and 5V input supply with eight output lines for each of the pins P_1 – P_8. Designed using

Altuim designer, resulted in the following board design, which was produced with female headers, allowing the board to connect to the encoder using the existing header pins.

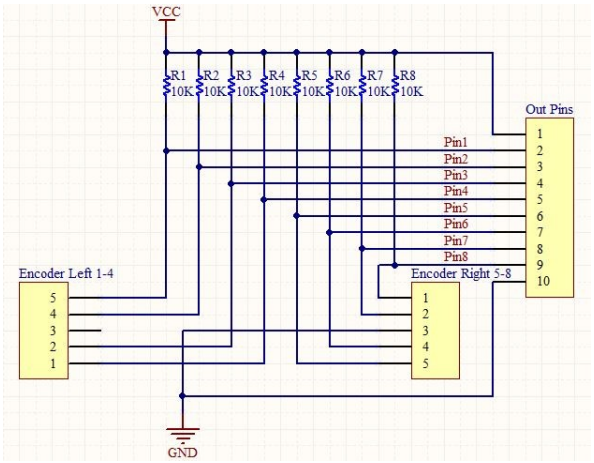


Figure 6-14 Bourns ACE-128 interface schematic

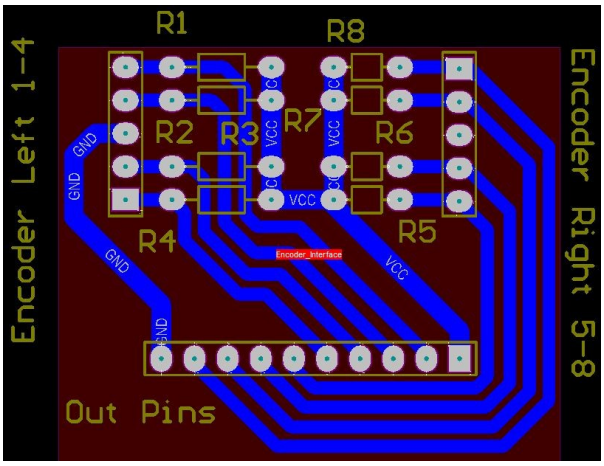


Figure 6-15 Bourns ACE-128 interface PCB

6.2.2 Motor Controller

The steering system uses two DC motors, a motor to drive the steering gearbox, and a linear actuator that allows the user to either engage or disengage the driving motor. Both motors are required to be reversible. Therefore, an H-bridge motor driver configuration is required to control these motors. To select a motor driver, the maximum current draw of the motor is required to prevent overheating and damage that could be caused by exceeding the drivers current rating. As outlined in section 5.2.1.1, 3.59 amperes at 12 volts was the maximum current drawn by the motor at stall, thus any motor driver must be capable of handling this amount of current with an acceptable safety margin.

A 30-amp motor driver board from Pololu.com was used. The VNH2SP30 motor driver (driver datasheet available in Appendix F, table 12) has an operating voltage range of 5.5 – 16 volts with a continuous current rating of 14 amps (30 amps peak), this exceeds the required specifications measured for this motor. Operating at 5V logic levels, the driver requires a 5-volt input to power the driver chip, with the output determined by the truth table (Table 6-1).

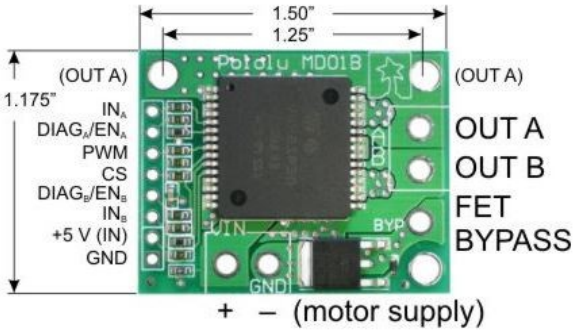


Figure 6-16 Pololu VNH2SP30 Motor Driver Carrier MD01B Board[27]

IN_A	IN_B	$DIAG_A/EN_A$	$DIAG_B/EN_B$	OUT_A	OUT_B	CS	Operating mode
1	1	1	1	H	H	High Imp.	Brake to V_{CC}
1	0	1	1	H	L	$I_{SENSE}=I_{OUT}/K$	Clockwise (CW)
0	1	1	1	L	H	$I_{SENSE}=I_{OUT}/K$	Counterclockwise (CCW)
0	0	1	1	L	L	High Imp.	Brake to GND

Table 6-1 VNH2SP30 Truth table under normal operating conditions (Appendix F)

The IN_A and IN_B inputs determine the motor direction and the speed of the motor determined by the PWM signal with a frequency of up to 20 kHz. The motor current can be measured on the CS pin as an analogue voltage, which correlates to approximately 0.13 volts per amp at the output of the device. To prevent damage to the motor and motor driver system, the driver includes built-in protection against reverse-voltage, over-voltage, under-voltage, over-temperature, and over-current. Any errors caused due to the protection features result in the chip shutting down, with the cause of the error indicated on the DIAG pins by grounding the pull up pins (Appendix F, Table 15).

Testing of the VNH2SP30 motor driver was completed using the selected steering system motor, using a signal generator to provide the required PWM input along with a bench top power supply that provided the necessary inputs. The results of the testing proved the viability of the motor controller by successfully driving the motor with the motor driver becoming warm to the touch when the motor was stalled, showing peak current draw of 16 Amps on start-up measured on an oscilloscope.

The Pololu website provided a circuit diagram for the set up on the VNH2SP30 motor driver board as shown in Figure 6-17. This allowed the board to be recreated in-house with the motor driver chips sourced from china for US $2.19 per chip. Using this design, custom motor driver boards were designed that included an on-board microcontroller cheaper than the US $34.95 boards that were available from Pololu.

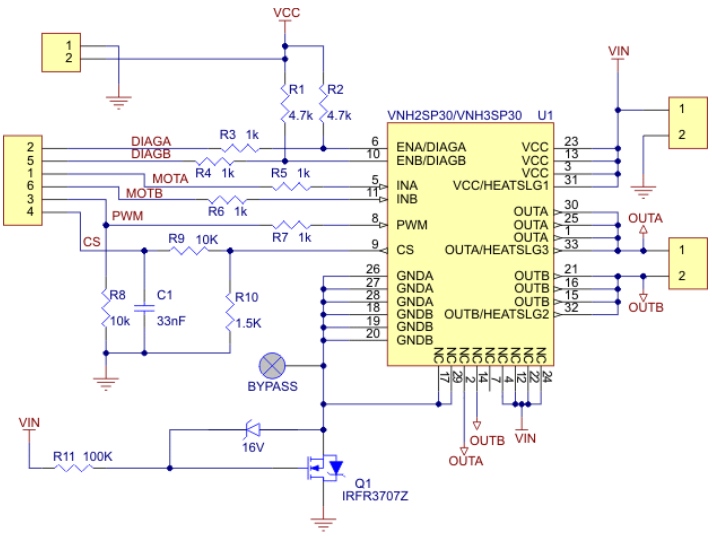


Figure 6-17 Schematic of the Pololu High Current Motor Driver Carrier [27]

The linear actuator used for the manual steering control has a rated maximum current draw of 4.8 amps. Therefore, the VNH2SP30 motor driver is sufficient to operate the linear actuator controlling the manual steering mechanism.

6.2.3 Microcontroller Selection

Any microcontroller chosen to control the mechanical steering system needs to be capable of interfacing with the main control system, through either a serial connection or I2C. The microcontroller must also be capable of reading eight signal pins on the encoder, which will provide signal of either 5 volts or GND to sense the 8-bit position of the encoder. To control the motor drivers at least four pins are required, two of which control the direction of the motor, a third for the pin capable of producing a PWM signal to control the speed of the motor and a fourth pin capable of reading the analogue voltage on the current sensing pin. If possible, the diagnostic pins should be available for reading, though not required for this system due to the over specification of the motor drivers being unlikely to cause problems due to overload.

From the specified components, the microcontroller is required to have two pins for serial communication (Transmit TX, Receive RX), 10 input pins for reading voltage levels with at least two of these required to read analogue signals with all the signals ranging between 0-5 volts. Finally six digital pins are required to control the direction and speed of the motor drivers, two of which need to be PWM enabled. Due to previous experience in other projects, an Atmel 328p microcontroller was selected to control the steering system basing the control board an Arduino design. This microcontroller provides 14 digital and six analogue pins, with a clock speed of 16MHz. Of these digital pins, two can be used for serial communications, six can provide PWM outputs and two can be used as interrupt pins. With an operating voltage of 5 volts, its analogue pins accept voltages with the range of 0-5 volts and can measure these in increments of 4.88mV using a 10-bit analogue to digital converter.

The Atmel 328p microcontroller provides all the necessary pins required to operate the system, as well as providing the capability of reprogramming the microcontroller via the serial connection.

6.2.4 Controller Design

The basic design for the Atmel 328p microcontroller board was based on the Arduino Uno R3 schematics. Using this existing and proven design ensures that all the required components were included. This was primarily in regards to the crystal clock design, decoupling capacitors and the protection present on the reset pin as well as all of their subsequent values were correct.

One of the advantages of using the Uno based system, is that an Arduino Uno R3 board that has had the Atmel 328p chip removed, can be used to reprogram the microcontroller over the serial connection. By connecting the TX, RX, Reset and GND pins from the microcontroller board to the corresponding pins on the Uno board a USB serial connection to the programming software can be established, allowing for in circuit programming and debugging. To enable serial communication between the microcontrollers in the system, these three of pins are required, TX, RX and GND. By designing the steering controller to include the Reset pin, the system can be reprogrammed with ease.

The layout for the microcontroller communication cable is shown in Figure 6-18. This layout facilitates simpler PCB design by directly connecting to the Atmega328 microcontroller pins as shown in Figure 6-18. To ensure reliable communication between the modules in the system, it is necessary to prevent data corruption that can occur during transmission due to external noise corrupting the data. There are a number of methods available to ensure data integrity including the use of a cyclic redundancy check, CRC, to encode the data and reveal any errors incurred during data transmission.

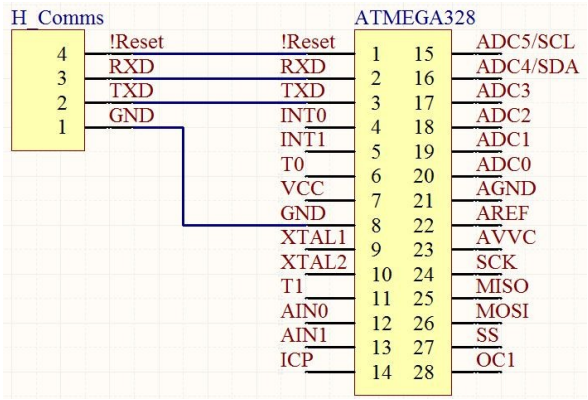


Figure 6-18 Communication Cable pin layout

To reduce the possibility of errors caused due to signal noise interfering with serial data transmitted on the communication line, the transmit and receive channels are shielded by twisted ground wires using Cat5 network cable.

The Cat5 cable features four sets of two pair wires, with one of the pair in each set of wired designed to be grounded. This provides the four signal wires required by the system to be shielded. Pairing the Reset line with a ground line prevents the possibility of unintentional resetting of the microcontroller due to voltage drops caused by signal noise. However, it is

unlikely that the Reset line could be pulled below the 0.9-volt threshold for the 2.5 microseconds required to reset the Atmel 328p microcontroller (Appendix D, page 318).

The steering controller board requires a 5-volt supply to power the microcontroller, encoder and motor drivers. As outlined in section 0, the quad bike has an on board 12-volt supply from the battery. This is reduced to 5 volts using a linear regulator rated to provide 1.0 amperes to the system. A 10-pin connector connects the encoder interface board to the steering controller board.

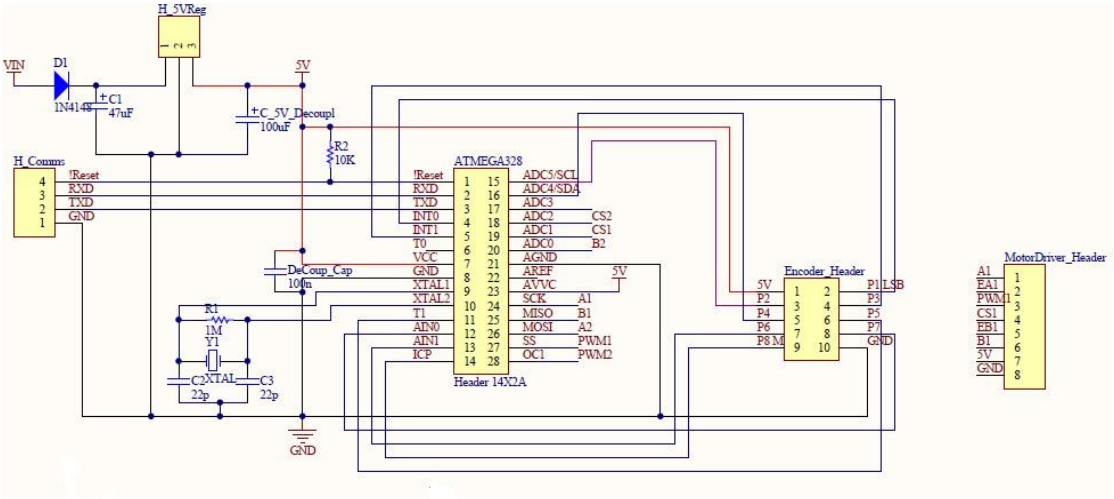


Figure 6-19 Steering controller Schematic

The Pololu VNH2SP30 motor driver board is connects to the steering controller via an eight pin header, providing the required inputs to control the steering motor using the A, B, PWM and current sense pins. The linear actuator motor driver is included on the steering controller board using the design specified in Figure 6-17 under section 6.2.2, and includes a PCB design similar to the Pololu design, ensuring that the heat dissipation on this board is similar to that of the Pololu board.

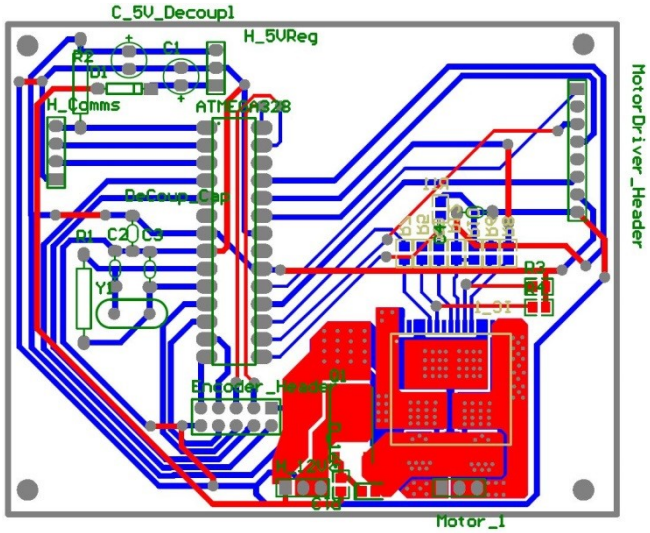


Figure 6-20 Steering Controller PCB

A second motor driver has been included in this design to provide control over the linear actuator that determines if the steering motor is coupled to the gearbox. The linear actuator has adjustable internal limit switches that increase the impedance of the motor using diodes. This allows the impedance to be raised for a single direction on the motor, preventing the motor from driving past these two limit points. These diodes allow the polarity to be reversed to drive the motor in the opposite direction. Using this, the microcontroller can sense when one of these limits is reached via the current sense pin on the motor driver and stop the motor at its limit.

6.3 Speed Control System

The speed control module requires three components to operate, a throttle control method to increase the speed of the vehicle, a system to detect the speed of the vehicle and a system to monitor and apply braking to reduce the vehicles speed. To interface the speed controller with the main interface controller, the same serial communication system as outlined in section 6.2.3 has been used.

The braking system uses two variable resistor systems to monitor the position of the brake controls, providing a variable voltage signal with a possible output range of 0-Vcc volts. With a 5-volt supply, the output range of the resistors operates within the specified analogue input range for the Atmel 328p microcontroller, allowing for direct reading of the analogue signal without pre-processing of the signal. A linear actuator outlined in section 5.3.3, actuates the rear braking mechanism, with its position controlled by the microcontroller and a VNH2SP30 motor driver system as outlined in section 6.2.2.

6.3.1 Speed Detection

The simplest and most effective method of detecting rotational movement of the wheels on the bike is to use an encoder, and unlike the steering mechanism, this encoder is only required to provide a speed and direction, both of which can be provided using a simple incremental encoder. Refer to section 5.2 for more information about encoders.

The TRX-300 quad bike features a cable driven speedometer, driven from the vehicles drive train. The bike has a fixed four-wheel drive system, with all four wheels driven by the motor. Therefore, as one wheel is driven, all four rotate. Though all four wheels rotate at the same time, they can rotate at different speeds due the differential gear boxes located on the front and rear axles. This allows the wheels on the outside of the vehicles turn radius to rotate faster than the wheel on the inside.

Because of this fixed four-wheel drive setup, placing an encoder on a single wheel could incur a speed error when turning due to the differential, thus the cable mechanism from the gearbox was selected as the detection point to measure the speed of the bike, as it would provide reading with greater accuracy during turning than a single wheel. This also provides a simple and out of the way area to position the encoder, that is protected from obstacles that may damage the encoder. The encoder can be either connected directly to the output shaft of the gearbox or using the existing cable, which will allow for flexible positioning of the encoder. As the speedometer cable was broken on the acquired bike, the encoder was mounted directly to the gearbox to save costs as seen in Figure 6-21.

Figure 6-21 Encoder mount on drive train

6.3.2 Board Design

The speed controller board requires input connections for the speed encoder and analogue signals from the front and rear brake sensors as well as the analogue signal from the manual throttle control. Output connections are needed to actuate the servo within the throttle control mechanism outlined in 5.5.1 and to control the linear actuator used within the rear braking mechanism outlined in 5.3.3.

Like the steering controller board, an Atmel 328p was used as the modules microcontroller using the same communication system as shown in Figure 6-18. This system required the two interrupt pins available on the microcontroller to detect input signals from the drive train encoder. Two analogue input pins provide a measurable signal for the braking systems and a third pin used to measure the analogue input from the manual throttle mechanism, that uses a variable resistor to measure the position of the original throttle control.

The final requirement of this controller board in the integration of a VNH2SP30 motor driver outlined in 6.2.2, that requires two digital pins for direction control, a PWM enabled pin for speed control and a analogue pin to read the motor drivers current sense pin. This board is capable of additional functionality with an onboard LED to indicate when the system is in manual mode as well as additional analogue and digital pins.

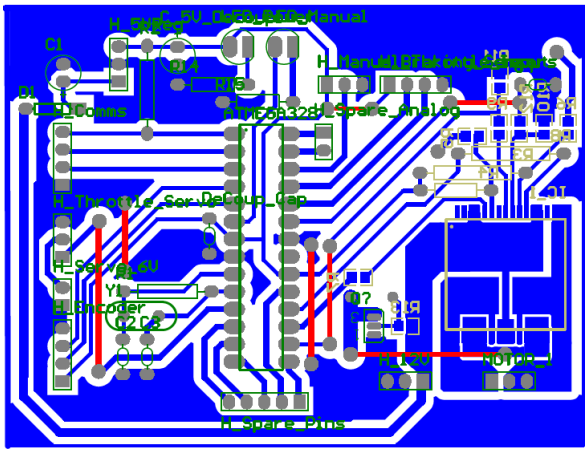


Figure 6-22 Speed Controller PCB

6.4 Gear Changing Controller System

The TRX-300 features a manual gearbox, which requires the actuation of a foot pedal to change the gear, with display lights between the handlebars to indicate if either the Reverse or the Neutral gear is selected. To actuate the designed gear changing mechanism outlined in section 5.4, the control board is required to engage the motor to rotate the actuator drum, with the maximum rotation less than 100 degrees. This angle is small enough to allow a variable resistor attached to the output drive shaft of the motor, and provides positional feedback on the actuator arm. This resistor feedback effectively converts the system to a large servo, allowing the controller to control and position the actuation drum within a few degrees of precision.

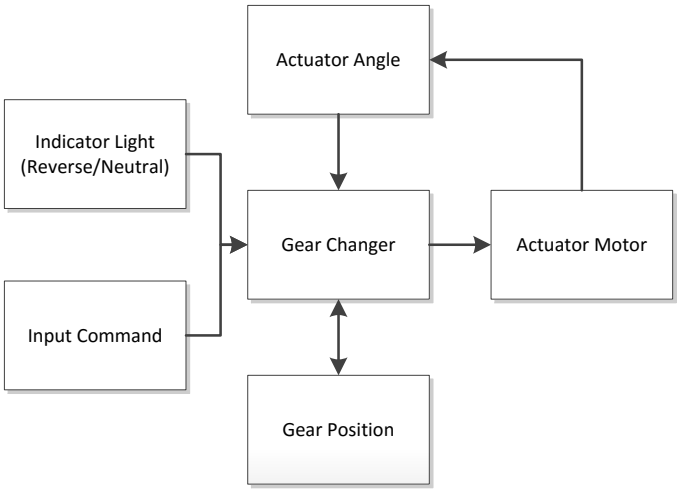


Figure 6-23 Gear controller overview

Using the indicator lights system outlined in chapter 6.1.4, a gear change to or from neutral and reverse gear can be detected, though there is no existing method of detecting any other gear change. By positioning the gear leaver at the point at which a gear change occurs, a signal can be generated when the position of the leaver exceeds this point. To detect these positions, set of mechanical micro switches are mounted on the gear mechanism, which are activated when the leaver arm passes point at which gear change occurs. These micro switches provide a method of detecting gear changes irrespective of the gear being selected, as well as providing detection to manual gear changes by a user.

For the gear changing system to operate successfully, the following functionalities are required change up, change down, provide a signal to indicate a gear change has occurred, and the control mechanism must return the actuation drum to the neutral position after each gear change. By returning the gear drum to the neutral actuation position, it ensures that the user has unimpeded control to change gear manually at any time. Having the actuation drum in the neutral position has the undesired effect of increasing the angle in which the gear drum has to rotate before contact is made with the leaver arm. However, this is only true if the system was constantly changing gears in one direction.

To ensure smooth and reliable gear changing, the system will require gear-changing profiles, which will adjust the point at which gear changing occurs based on speed, RPM and engine load. It is possible to reduce the time taken to change gears by ‘Pre-loading’ the

actuation drum, where the drum is positioned to be in contact with the leaver arm. This reduces the actuation angle of the drum by up to half and thus decreases the time required to perform a gear change.

Two options were considered to control the gear changing mechanism. The first is a purely analogue control system that would respond to high low inputs along signal lines and would indicate gear change via two channels that would need to be monitored. The second is a digital control system that would use a programmable microcontroller to monitor and control the speed, direction a position of the motor. This system would communicate via serial interface to send and receive commands, querying the controller for gear status could be done on command.

6.4.1 Analogue Control System

The mechanism performs a gear change by rotating actuation drum to specific positions, which remain constant throughout any gear changes, i.e. does not change depending on which gear is engaged. The neutral position for the actuation drum can be expressed as a range within its actuation path instead of a specific position. Therefore, as long as the drum is within this range then manual gear change will be possible. Four positions are required to be detectable: preload up, preload down and the upper and lower limits for the neutral position.

The position of the drum is represented as an analogue voltage within a range of zero to 5-volts, and the limit positions can be equated to a voltage within this range. These positions are equal to the voltage measured from the drum resistor when the actuation drum is positioned at these limit points. A series of variable resistors provide an accurate and easily adjustable method of matching the voltage at these positions, and act a reference values for the circuit.

The operation of the mechanism can be simplified to a state machine, where it is either gear changing up/down or in the neutral position, see Figure 6-24.

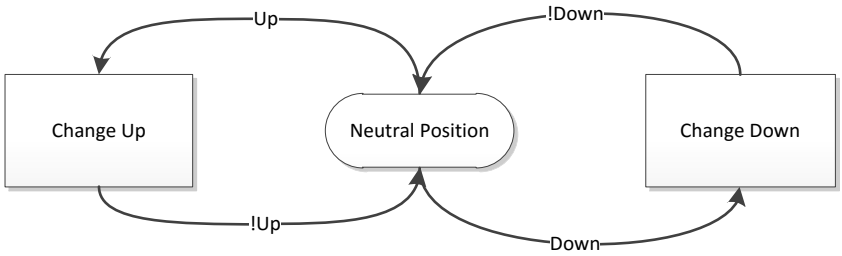


Figure 6-24 Analogue gear change state machine

Converting these states to a set of logic statements provides the digital signals inputs required to operate the VNH2SP30 motor driver based on the logic table, Table 6-1 in section 6.2.2. A series of four comparators are used to simplify the logic statements in regards to the position of the actuation drum and its relationship to the preset limit positions, with the comparators set to produce logic high when these positions were met as described by the following conditions.

Comparator 1 (Comp1) - logic high when the drum voltage is greater than the preload position for a down gear change, and is used to set the position in which the actuation drum makes contact with the gear changing leaver arm.

Comparator 2 (Comp2) - logic high when the drum voltage is less than the preload position for a up gear change, and used to set the position in which the actuation drum makes contact with the gear changing leaver arm.

Comparator 3 (Comp3) - logic high when the drum voltage is below the lower limit of the neutral position.

Comparator 4 (Comp4) - logic high when the drum voltage is above the upper limit of the neutral position.

To provide the necessary command signals to operate the circuit, the following inputs are required: up, down and preload. The circuit providing the outputs for the interrupt signals from the micro switches, status lights for the neutral and reverse gear indicator lights. These signals can be combined into logic statements that enable the control of the direction of the motor, which will determine the gear change and return to a neutral position with the resulting truth table shown in Table 6-2.

(! = not, & = and, # = or)

Inputs Signals = Up, Down, Preload, Comp1, Comp2, Comp3, Comp4.

Output Signals = UC, DC.

Change Up (UC)

$$UC = \big((Up \,\&\, !\,Preload) \,\#\, (Up \,\&\, Preload \,\&\, Comp2)\big) \,\&\, !\,Down \quad \textbf{6.4}$$

Change Down (DC)

$$DC = \big((Down \,\&\, !\,Preload) \#(Down \,\&\, Preload \,\&\, Comp1)\big) \,\&\, !\,Up \quad \textbf{6.5}$$

To ensure the drum is returned to the neutral position the following logic was developed based on the up and down change outputs.

Neutral Down (ND)

$$ND = !\,Up \,\&\, !\,Down \,\&\, Comp3 \,\&\, !\,Comp4 \quad \textbf{6.6}$$

Neutral Up (NU)

$$NU = !\,Up \,\&\, !\,Down \,\&\, !\,Comp3 \,\&\, Comp4 \quad \textbf{6.7}$$

Combining these statements results in two-logic statements that control the direction of the actuation drum, with a counter-clockwise rotation of the drum resulting in the mechanism changing up a gear. Using Table 6-1 from section 6.2.2, a high on A with a low on B would result in a clockwise rotation, with the logic statement resulting in either A or B high

with the other low. Therefore, the following logic statements will control the direction of the motor/actuator drum.

$$A = !\,Up\ \&\ \begin{pmatrix}(Down\ \&\ !\,Preload)\#(Down\ \&\ Preload\ \&\ Comp1)\\ \#\ (!\,Down\ \ \&\ Comp3\ \&\ !\,Comp4)\end{pmatrix} \quad \textbf{6.8}$$

$$B = !\,Down\ \&\ \begin{pmatrix}(Up\ \&\ !\,Preload)\ \#\ (Up\ \&\ Preload\ \&\ Comp2)\\ \#\ (\,!\,Up\ \&\ !\,Comp3\ \&\ Comp4)\end{pmatrix} \quad \textbf{6.9}$$

These statements are mutually exclusive and work according to the following conditions. The motor rotates in a clockwise (A) direction only when the Up signal is low as well as one of the following conditions is met:

1) Down is high, preload low.

   Normal down gear change routine.

2) Down is high, preload is high, Comp1 is high.

   Preload routine, and with gear change occurs once preload is low.

3) Down is low, Comp3 is high, Comp4 is low.

   Neutral return resulting when there is no input to the system and the actuation drum registers as below the neutral lower limit.

The motor rotates in a counter-clockwise(B) direction only when the Down signal is low as well as one of the following conditions:

1) Up is high, preload low.
   Normal up gear change routine.
2) Up is high, preload is high, Comp2 is high.
   Preload routine, and with gear change occurring once preload is low.
3) Up is low, Comp3 is low, Comp4 is high.
   Neutral return resulting when there is no input to the system and the actuation drum registers as above the neutral upper limit.

Input	Condition 1	Condition 2	Condition 3	Condition 4	Condition 5	Condition 6
Up	0	0	0	1	1	0
Down	1	1	0	0	0	0
Preload	0	1	0	0	1	0
Comp1	0	1	0	0	0	0
Comp2	0	0	0	0	1	0
Comp3	0	0	1	0	0	0
Comp4	0	0	0	0	0	1
Output						
A	1	1	1	0	0	0
B	0	0	0	1	1	1

Table 6-2 Gear Changing Logic Table

Building this circuit using standard logic integrated packages (IC) would result in a cluttered circuit that would be time consuming to design and produce, with the added components decreasing the overall reliability of the circuit. To simplify the board design and increase the circuit reliability, a programmable logic device (PLD) provides the logic function for the circuit. The ATF16V8B PLD (datasheet in Appendix G) operates at 5-volt logic featuring eight input pins, with a further eight input/output (I/O) pins and allows highly complex logic functions to be programmed.

6.4.1.1PLD System Design

Unlike a standard logic IC design, the PLD requires a clock input to refresh its outputs, and thus the control system requires either a clock input line or a clock source on the circuit. For this system, a simple oscillator circuit was built using a Schmitt triggered NAND gate to produce a PWM signal with a frequency of 40 kHz and duty cycle of 50%. This frequency ensures that any change in 10 kHz PWM signal driving the motor driver is captured by the PLD to appear on the output. The PLD logic controls the direction of the motor with the motor run at full speed during gear change, and at a reduced speed for preloading and neutral return to prevent over rotation of the drum, that could result in an unwanted oscillation due to the hysteresis on the neutral control.

The logic statements for the PLD were developed, programmed and tested using the Wincupl programming software. Figure 6-25 shows the final code, with the PLD designed to operate as a state machine that automatically returned once an interrupt signal is detected

from either micro switch.

```
/* *************** INPUT PINS *********************/

PIN 1    =  CLK;        /*                                  */
PIN 2    =  Up;         /*                                  */
PIN 3    =  Down;       /*                                  */
PIN 4    =  Preload1;   /*                                  */
PIN 5    =  NeutralPin;     /*  PWM Signal                      */
PIN 6    =  Comp1;      /*                                  */
PIN 7    =  Comp2;      /*                                  */
PIN 8    =  Comp3;      /*                                  */
PIN 9    =  !Comp4;         /*                              */
PIN 11   =  !OE;        /*                                  */
Pin 12   =  UpMicro;        /*                              */
Pin 13   =  DownMicro;  /*                                  */

/* *************** OUTPUT PINS *********************/
PIN 15    = X; /*                                  */
PIN 16    = Y; /*                                  */
PIN 17   =  UpSignal;   /*                                  */
PIN 18   =  DownSignal;     /*                              */
PIN 19   =  InterSig;   /*                                  */

FIELD controller = [X,Y];

$DEFINE Ready 'b'00
$DEFINE Neutral_Return 'b'01
$DEFINE Null1 'b'10
$DEFINE Null2 'b'11

]SEQUENCE controller {
   PRESENT Ready
     IF InterSig NEXT Neutral_Return;
     DEFAULT NEXT Ready;
   PRESENT Neutral_Return
     IF !Comp3&!Comp4 NEXT Ready;
     DEFAULT NEXT Neutral_Return;
   PRESENT Skipping
     DEFAULT NEXT Ready;
   PRESENT Shooting
     DEFAULT NEXT Ready;
-}

InterSig =  UpMicro # DownMicro # controller:Neutral_Return;
DownSignal = (!Up&((Down&!Preload1)#(Down&Preload1&Comp1)))
             #(NeutralPin&Comp3&!Comp4&(controller:Neutral_Return#(!Up&!Down)));
UpSignal = (!Down&((Up&!Preload1)#(Up&Preload1&Comp2)))
           #(NeutralPin&!Comp3&Comp4&(controller:Neutral_Return#(!Up&!Down)));
```

Figure 6-25 Wincupl Code

Using this state machine design, a signal is provided to the parent control system informing the status of the control mechanism, the 'Ready' state and the 'Neutral_Return' state. The 'Ready' state is the default state for the system, which waits for an input signal to determine the direction of the gear change. Once a gear change has triggered, as detected via the micro switches interrupt signal, the system switches to the 'Neutral_Return' state. Once in this state the controller will automatically return the drum to a point within the neutral that satisfies both the Comparator 3 and 4 conditions.

The PLD logic was simulated in Wincupl as shown in Figure 6-26 below, with the input signals represented by the Green signals, and the simulated outputs represented by the Blue. This simulation was designed to test a gear change triggered by the up signal, with the signals from comparator 1 is ignored as they have no effect on the Up output signal. The Neutral pin operates with a constant PWM input as represented by the signal that alternates every clock cycle.

To test the functionality of the PLD logic, the Preload, error detection and the neutral return mechanism. Cycles 1-2 shows the error checking with both the Up and Down inputs resulting in a low on the directional outputs. The down input is registered as a low on clock cycle 3 with the system output reflecting this change with the UpSignal resulting in a High value. The preload signal is high until cycle 10, which results in the UpSignal pulled low over cycles 8-9, due to Comparator 2 (Comp2) input switching to a low. This simulates the actuation drum reaching the up preload position and stopping until the preload input is removed.

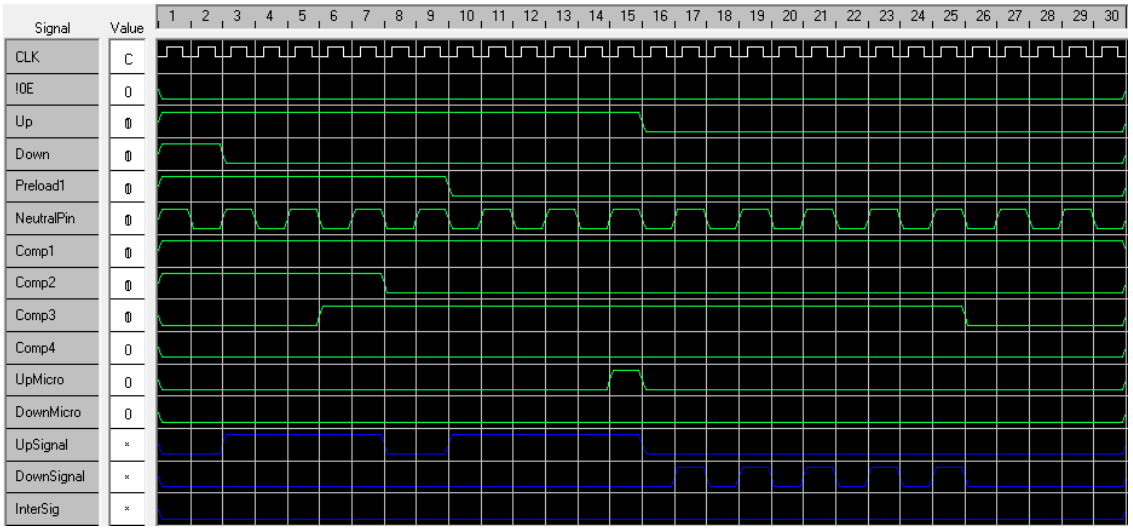


Figure 6-26 PLD Logic Simulation

A gear change was simulated by providing a pulse on the UpMicro input, which when detected by the parent microcontroller that would set the Up input signal to ground, triggering a state change from the 'Ready' state to the 'NeutralReturn' state, pulling the UpSignal low. By entering the 'NeutralReturn' state the system pulses the DownSignal output according to the NeutralPin input so long as the input signal from comparator 3 (Comp3) remains high. Once the comparator 3 input returns to a low state, the actuation drum has been returned to a position above the lower neutral limit.

6.4.1.2Analogue System Testing

To test the PLD controller with the mechanical system, the circuit was built on a breadboard, with the input/output results tested with an oscilloscope. The motor remained disconnected from the circuit, with the outputs from the motor driver monitored instead. A hand adjusted variable resistor was used in place of the motor shaft resistor to provide the feedback to the system to determine the position of the motor shaft. To set the resistor reference positions for the comparators, the motor drum was manually moved to the

individual set points with the voltages at those set points measured. The resulting reference voltages were generated using variable resistors to trim the voltage levels to match those measured on the motor drum resistor.

The results of the breadboard testing matched the simulated results above, with the output response of the PLD driving the output of the motor driver. After this initial bench testing, the controller was tested using the mechanical gear changing mechanism mounted on the bike. The reference set points were re-tuned to ensure accurate limits. The PLD controller mechanism operated with similar results to the simulated and bench top test, though there were some instances where the PLD became unresponsive and required resetting by shutting down the power to the system. This could be due to incorrect logic, a fault within the PLD itself, or cause by possible voltage spikes from the motor that were unable to be filtered by the linear regulator powering the logic system.

6.4.2 Microcontroller Control System

Similar to the analogue system in section 6.4.1, a microcontroller-based system would provide similar control with respect to the control logic but with the added advantage of monitoring and controlling the gear system independently of a parent controller. Unlike the analogue system that requires a parent microcontroller to monitor the system for gear change interrupts, this microcontroller provides a dedicated monitoring and control system, thus reducing the load on the main system controller. This ensures all interrupt signals are captured, such as gear change interrupts, as well as provides a method of requesting the status of the system, such as the current gear engaged and can ensure that the motor/ motor driver is not damaged by monitoring the current draw of the system.

To enable easy setup and adjustment of the mechanisms limit points as outlined in section 6.4.1, four variable resistors are used that provide analogue reference positions of the actuation drum for the preload and neutral return limits. As in the analogue circuit, three outputs are required to control the motor via the motor driver, two for the direction, and a PWM enabled output to control the motor speed. Ten inputs are required. Two digital interrupt enabled inputs are used to detect the micro switches and two more to detect logic level inputs from the gear indicator circuit. Four analogue inputs are required to read the drum positions set via the variable resistors, as well as an analogue input to detect the current position of the actuator arm and an analogue input to detect the motor driver current.

A serial communication interface us used to communicate between the microcontroller and the main interface controller circuit, with an Atmega328 microcontroller used as specified in section 6.2.3. The circuit operates at a voltage of 5V, produced from the 12V input using a LM7805 1 amp linear regulator, and the circuit also includes decoupling capacitors for the circuit and microcontroller. A diode provides reverse voltage protection for the circuit.

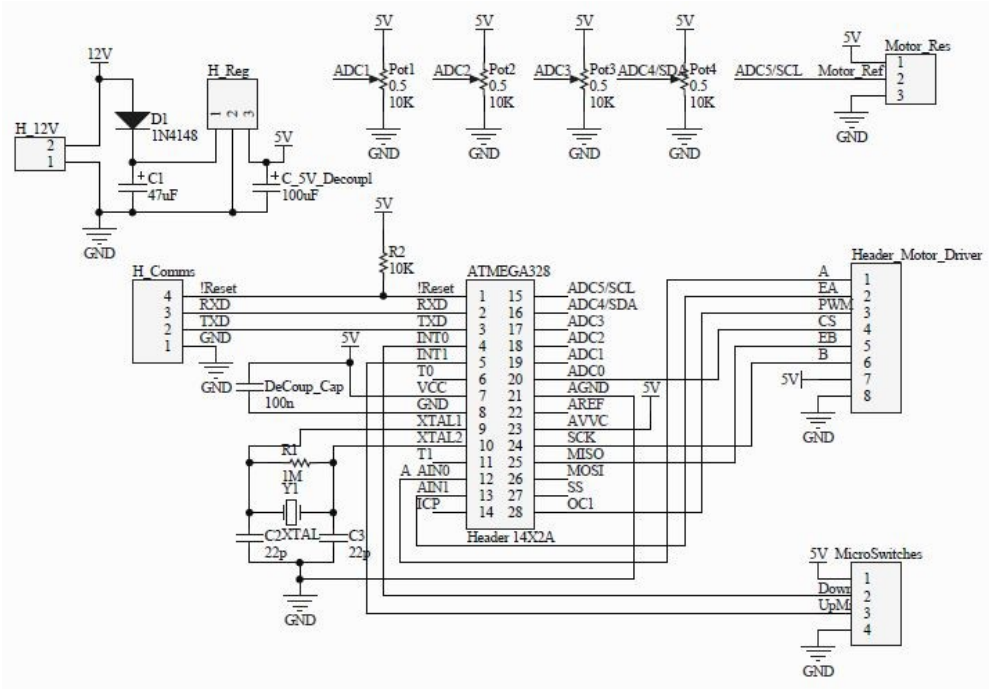


Figure 6-27 Gear Changing Microcontroller Schematic

6.4.3 Systems Testing

The microcontroller control system was selected as the control system for the gear changing mechanism, primarily due to the increased functionality the system provided over the analogue system. Testing was undertaken by ensuring that the mechanism was capable of achieving the designed functionality as specified above. The preload, neutral return systems were tested and preformed as expected with only minor modifications to the programmed motor speed required to prevent over rotation of the actuation drum. Filtering functionality was added to the code to prevent signal bouncing on the micro switches, which was detected by the controller as multiple gear changes.

6.5 Module Interface System

A Raspberry Pi Model B single board computer (Figure 6-28) is used to provide high-level control over the individual systems on the quad bike, pertaining to the path finding and object avoidance control. Although this is a powerful computing device, it does not provide the necessary I/O capabilities to provide adequate control over the bikes mechanical control systems. Therefore, an intermediary system is required to oversee the control mechanism on the bike, and provide a single interface in which the Raspberry Pi can send and receive commands.

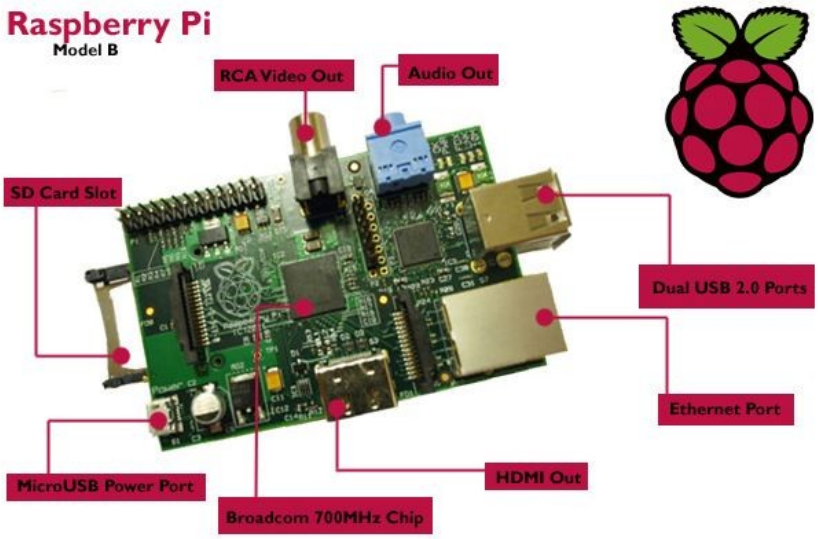


Figure 6-28 Raspberry Pi Model B Single Board Computer[28]

As covered in the previous sections of this chapter, there are three systems used to control the Steering, Speed and Gear changing for the vehicle. All of these systems use a serial interface as their communication system to send and receive commands. A compass system is required to provide the controller with a vehicle heading at times when the GPS system is unable, this could be due to loss of signal caused by trees or buildings, or when the system is turned on and stored in a building.

I/O pins are required to interact with the Power switching board outlined in section 6.1.2, which controls the lights, horn, engine fan, and starter motor. Two of which require PWM enabled pins to vary the intensity of the horn and the speed of the engine fan, with a further, two to switch the lights and starter motor. An interrupt enabled pin is also required to detect the pulses generated by the RPM detection circuit, outlined in section 6.1.1, which when counted, provides an RPM value for the motor. The statuses of several existing manual input switches require monitoring; including the Horn, starter button, light switches, and engine run switch status. Two analogue input pins are required to sense the temperature readings from the Temperature sensor board (Section 6.1.3), which are represented as analogue voltages.

To ensure compatibility and ease of programming, a microcontroller from the ATmega family was chosen to act as the subsystem controller, interfacing the vehicles modules with the Raspberry Pi controller. Within the ATmega microcontroller family, there are two microcontrollers capable of providing the necessary number of serial interfaces and IO interfaces. These are the Atmel AT91SAM3X8E and the Atmel ATmega2560, which have similar capabilities. Comparing these two microcontrollers reveals a single major difference in performance/compatibility with only the ATmega2560 providing 5-volt tolerant pins, with a 3.3-volt maximum input/output for the AT91SAM3X8E. Therefore, voltage level converters for the 5-volt data outputs that the most components use are required.

The CMPS10 digital compass provides the necessary compass data for the system, and features three communication interfaces to read the compass heading, I2C, Serial and PWM. This compass outputs a compass heading ranging from 0-359.9 degrees, and features a 16-bit

processor to provide a tilt compensated heading, which is designed to remove any errors caused by the relative angle of the PCB to the ground. The module can also provide the raw tilt data used by the onboard processor, and the module operates at a range of voltages between 3.3 - 5 volts, with a nominal current draw of 25mA. This compass module is also capable of providing the inclination of the board via either the serial or I2C interface. Serial command information and product details are contained within the datasheet located in Appendix H of this thesis.

Figure 6-29 CMPS10 Compass Module (Appendix H)

The ATmega2560 microcontroller was chosen to provide the interface between the Raspberry Pi controller and the quad bikes various subsystems, providing 54 digital pins, 15 of which provide PWM output, 16 analogue input pins and features four serial interfaces. To simplify the development of the interface, an existing support board, the Arduino Mega 2560 (Figure 6-30) was purchased, which provides a USB connections as well as a power connector that accepts a voltage range of 7-12 volts, with a maximum range of 6-20 Volts. The USB connection provides a simple method of connecting the ATmega2560 controller interface to any computer, including the Raspberry Pi. This USB interface uses one the serial interfaces as well as providing power to the microcontroller through the USB connection.

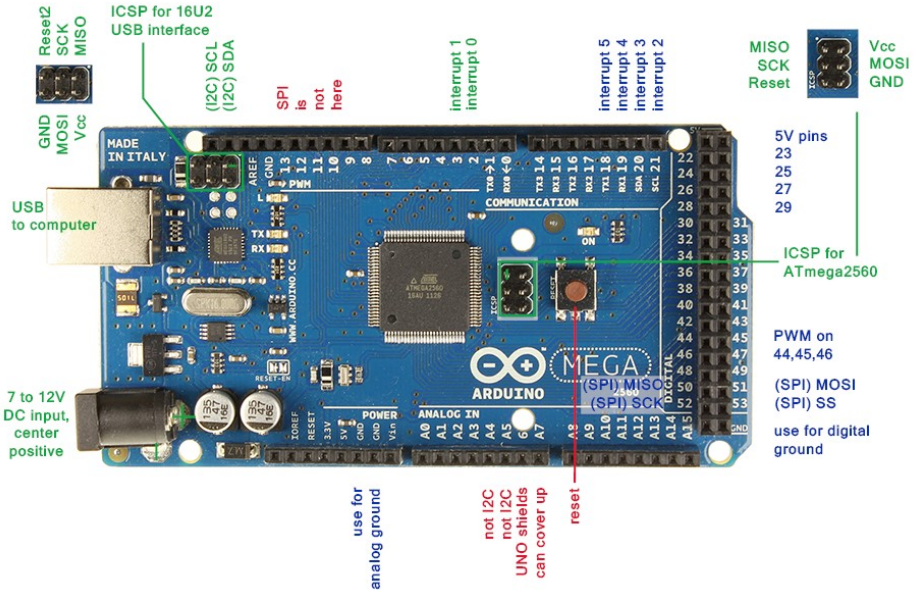


Figure 6-30 Arduino Mega 2560 Board[29]

The three remaining serial interfaces provide the connections to the Gear, Steering and Speed controllers, with the cables connecting each of these controllers. These cables

required three pins to interface with the ATmega2560 microcontroller TX, RX and a digital pin to enable a reset command. A ground connection is also required on the cable to shield the data transfer lines as well as provide direct path for a common ground connection.

To connect the CMPS10 compass, an I2C connection was considered, though this connection was not used due this device being the only component to use this connection method and would result in two of the six-interrupt pins being assigned to the I2C connection. The PWM method was disregarded as it only provides the heading information. Therefore, the serial connection was the only method remaining in which to receive the required information from the compass module. With all the hardware serial connections in use, a software serial connection was used, with eighteen pins on the ATmega2560 capable of providing the two pins necessary for the software serial connection. As mentioned above, the ATmega2560 features six interrupt pins one of which is used to detect the pulses generated from the engine RPM detection.

6.5.1 Atmega2560 Interface Board

A simple interface board for the Arduino Mega 2650 was designed to act as a shield plug-in for the existing Arduino development board, with the 5-volt power required for the Temperature sensors, Compass and Power switching circuits provided by the development boards on board voltage regulator. In the unlikely event that the current draw from this regulator is exceeded, causing a voltage drop, the ATmega2560 microcontroller will remain fully powered via the USB connection.

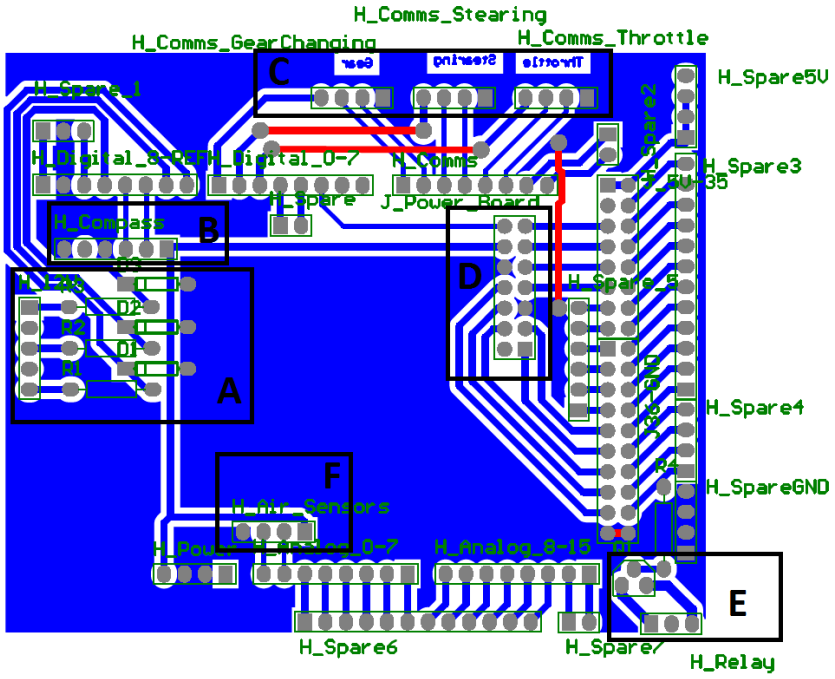


Figure 6-31 PCB layout for the Atmega2560 Interface board

A. Zener-resistor input that provides a logic level high-low output for input voltages that exceed 5-volt logic levels.
B. Header interface for the CMPS10 compass module.
C. Serial interface for the modules on the system, Gear, Steering and Throttle.

D. Header connector for the interface board outlined in sections 6.1.1 and 6.1.2.
E. The switching transistor controlling the engine run control relay
F. The temperature sensing interface reads the analogue voltage from the sensor outlined in section 6.1.3.

Three inputs are included in the design, which to detect the 12-volt inputs from the bikes switches, using a Zener-resistor combination and provide a logic level signal to the microcontroller, marked as A in Figure 6-31. The Zener-resistor set up was used as opposed to a voltage divider to ensure that a 3.3V signal was always provided by the Zener, preventing the possibility of voltage spikes that could be transmitted through the voltage divider and damaging the microcontroller. A transistor switch could have also been used to achieve the same results.

As outlined in section 6.1.2, the run switch relay must be toggled by the microcontroller to allow the engine to start or stop. The run relay must be energized to for the engine to run, the physical run switch must also be engaged as well, therefore a pick up system is required to detect when the run switch has not been physically disabled by the user. The components for the run control circuit are shown under the marked area of E on Figure 6-31, along with a circuit diagram in Figure 6-32.

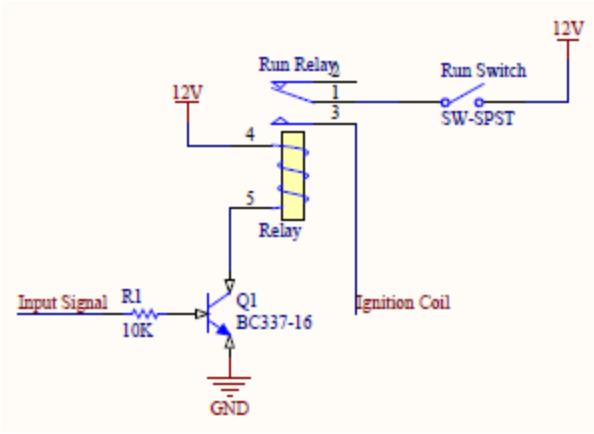


Figure 6-32 Engine run control circuit

The serial communication interfaces include connections for the transmission and receive lines, a ground line to provide a common ground between the modules and a reset line. This reset connected to a digital output pin with a pull-up resistor to allow the system to reset the attached module electronically.

The header pins highlighted in the area D on Figure 6-31, connect to the bikes power switching and RPM detection board outlined in sections 6.1.1 and 6.1.2. The area B provides a connection for the CMPS10 compass module shown in Figure 6-29 and the final area F, provides power and signal detection for the temperature sensors outlined in section 6.1.3.

6.6 User Interface for Raspberry Pi System

An interface is required allow the users to easily and effectively interact with the Raspberry Pi controller. For this interface, a number of mechanisms were considered including a soft key interface, a keyboard or touchscreen. The interface requires a screen to display

maps, position and vehicle data, as well as a method of interacting with this data which can be achieved via soft keys interface, where the function of each key is displayed on screen and changes based on the functions available in that menu.

Though a soft key interface could provide limited method of entering text, by using a ABC function similar to that used by cell phones in the 90's, this could be overcome by using a water proof keyboard with either a wireless or USB interface. Using waterproof keyboard would provide a comprehensive input interface to access the data displayed on the interface, and allow navigation of menus and a simple method of text input. However, any keyboard would need to be secured to the system interface, particularly as wireless solutions can be lost or batteries can be depleted, leaving the user with no method of interfacing with the system. The final system considered is a touch screen interface that combines the functionality of a display screen with a mouse; a keyboard can also be implemented in software using an on screen keyboard.

6.6.1 Touch Screen interface

A touch screen interface was selected as the primary interface system between the operator and Raspberry Pi system. A number of options are available regarding the size and type of screen to be selected, as well as the hardware that would act as an interface between the Raspberry Pi and the display. There are a number of touch screen interfaces available on the market, including screens that interface via a standard USB connection. To ensure compatibility between the screen and Raspberry Pi software, a hardware interface developed for the Raspberry Pi was used, which provides a LVDS connection to a touch screen using a HDMI and USB connection. This system can also be used on any device that provides a HDMI and USB output, and therefore can still be used if the Raspberry Pi controller were to be replaced.

Two types of touch screens are that were considered for this interface, both offer similar functionality of a traditional touch screen, i.e. it detects where your finger touches the screen, but they detect this touch using different methods. These two types are known as either capacitive or resistive models.

6.6.1.1 Capacitive Touch Screen

The first type is a the capacitive touch screen, which is commonly used in consumer products such as smart phones and tablets and provides multi-touch capabilities which allows the screen to detect simultaneous points of contact on the screen at once. While this is a useful feature on smart phones, the same cannot be said for most industrial applications, with the multi-touch feature primarily being used for gesture-based applications such as zooming. Capacitive touch screens use an electrical field to detect the presence of a conductive material, and due to the conductivity of the human body, a change in the screens electric field occurs when the user touches the screen, which is detected by the screen and used as in input for the device[30].

A capacitive touch screen would provide the necessary functionality required for the user interface, though the multi-touch functionality is not required for a basic interface. There are a number of problems associated with using the capacitive touch screen relating to the durability and usability within the intended outdoor operating environment. One of these

problems relates to the surface wear on the screen, with contact from dirty fingers leading to scratches that can impede the visibility of the display, this can be reduced by adding a screen protector, but this will only protect the screen from scratches and not from possible impacts.

Cracks across the screens surface can occur due to impacts on the screen surface from objects within the operating environment, though the likely hood of this could be reduced by adding a Perspex sheet to cover the screen. This Perspex cover would protect the screen from concentrated impacts by spreading the force over a larger area, a gel like media between the screen and Perspex may increase the force that could be applied. Adding this cover would likely reduce the effectiveness of screen in detecting the presence and therefore the position of a person's finger, therefore testing would be required to find the maximum possible thickness in which a finger could interface. This protective cover would also provide a method of waterproofing the interface to prevent water damage.

The system requires a conductive material such as skin to detect the pointer position on the screen. Therefore, while the capacitive screen is suitable as an input and display system for the touch screen interface, some testing is required to see whether they can be overcome, the drawbacks described above, particularly in relation to the waterproofing and screen protection.

6.6.1.2Resistive Touch Screen

Unlike the capacitive touch screen, which responds to the presence of a conductive material on its surface, resistive touch screens responds to pressure and thus does not require conductive material. Resistive touchscreens consist of two layers, featuring a coating on one side. These layers are assembled with a small gap between them. When a pressure is applied to the surface of the screen, the two layers come into contact, passing a voltage signal, which is processed as a position on the screens surface. Because of the pressure based detection method used by this touchscreen display, any object can be used to apply the necessary pressure, meaning that anything from a gloved finger to a piece of wood can be used to interact with the device.

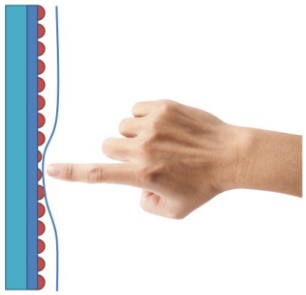


Figure 6-33 Finger interacting with a resistive touchscreen[31]

With the flexibility provided by the pressure based input system , a protective cover can easily be implemented, with the only restriction being that it provides the necessary flexibility to transmit the required pressure to the screen. To ensure the best protection from environmental damage, any Perspex sheet covering the screen should be as thick as possible, to reduce impact forces, while remaining flexible enough to prevent an excessive force being required from the user to activate the screen. A simple piece of plastic sheeting is all that is

required to waterproof this screen, as this will not significantly reduce the pressure transmitted though the material.

6.6.1.3Interface Screen Size

To ensure ease of use, the size of the touchscreen should be sufficient that the user can easily select options presented on the screen, as well as reading any text displayed at a distance of at least one meter. To meet these specifications, the screen must be sufficiently large enough to fit all the required information on the screen and displayed at a size that is both a readable and is simple to select.

Testing was conducted using tablets and phones available at local electrical stores to get an estimated minimum screen size though looking at readable text size, as well as the planning of possible interface designs on 5, 7 10 and 12 inch screens. These sizes are based on screen sizes compatible with the Raspberry Pi interface board on the Chalkboard electronics online store[34]. From this testing a minimum screen size of 7 inches was selected with a preferred size of 10 inches or more, leaving 10 and 12-inch models available.

6.6.1.4Selected Touchscreen system

Based on the screens available with the HDMI-LVDS converter from the Chalkboard electronics store, a 10-inch LCD Capacitive touchscreen was selected, featuring a display size of 1280*800 pixels, the screen provides sufficient space to display large, easy to read interface. The largest resistive screen available was 7 inch, and was deemed not provided to the necessary screen capacity to allow the virtual interface buttons and text to be displayed at a sufficient size.

Figure 6-34 Chalkboard Electronics 10-inch LCD capacitive touchscreen[32]

At the time, that this product was chosen, this was the best product that would meet the system requirements, though at the time of this thesis, a number of other products had been developed including a number of Resistive touchscreen overlays. These overlays provide

touchscreen feedback over standard display screens. Therefore for future development it is recommended that either a 10-inch or greater resistive touchscreen display be used for the system, using either an integrated system or an overlay system as described above.

6.6.2 Power Systems

To ensure reliability and stability in the primary controller, the Raspberry Pi and its peripheral systems, including screen, wireless communications, and GPS sensors must be powered from a reliable power source. Any power system will have to convert the 12-volts onboard power supply present on the quad bike and be capable of converting power beyond the output if the engines alternator. The power system should also handle voltage spikes and drops and not lose power to the controller during peak operating times, including when the starter motor is activated.

6.6.2.1 Control System Backup Power System

For the system to operate reliably, it must have a reliable power supply to ensure uninterrupted operation. The Raspberry Pi model B single board computer operates at a voltage of 5V, and a recommended that the power supply capable of providing a minimum of 700mA continuous current[18].

To ensure reliable operation of the Raspberry Pi and the peripheral components such as the GPS module, Wi-Fi card, Arduino Mega2560 interface and the Touch screen display, a power supply capable of supplying up to two amps is required. This was calculated from the recommended power requirements of the individual devices. The ATV has a 12-volt battery onboard to provide power for lights, starter motor and horn. When the motor is running, an onboard alternator provides a charging voltage to the battery that has been measured at a voltage of 13.4-volts. Therefore, any power converter must be capable of accepting an input voltage of greater than 15-volts, allowing a 10% safety factor on the voltage input.

6.6.2.1.1 Power Converter Selection

Two methods have been considered to convert the 12-volt supply to 5-volts using either a linear regulator or a switch mode supply. The linear regulator operates by adjusting the effective series resistance of the regulator using the internal feedback voltage. It does this by adjusting the resistance of a transistor to act as an adjustable voltage divider circuit[9, 35]. These have large inefficiencies at large voltage conversions, with a higher device efficiency when the voltage in, V_{in}, is close to voltage out, V_{out}. Another limiting factor of linear regulators is that there must be a minimum voltage difference between V_{in} and V_{out} to maintain a stable output, though there are low dropout regulators that can operate with smaller minimum voltage differences than standard regulators[9]. In using figures above for the system developed, a linear regulator would have a maximum efficiency, E of 37.3%. This assumes a voltage supply of 13.4V from the alternator while motor is operating.

$$E = \frac{5}{13.4} = 37.3\% \ (1sf) \tag{6.10}$$

Assuming a current draw is two amps, this would result in power dissipation (P_d) of 16.8 Watts.

$$P_d = (13.4 - 5.0) * 2.0$$
$$P_d = 16.8W \tag{6.11}$$

Linear regulators do have advantages and are preferred in applications where simple and low cost solutions are required, as well as low noise, low dropout and they have an internal feedback loop which allow for a faster response to transient voltages than switching mode power supplies[9]. For these reasons, linear regulators were used on the Speed, Steering and Gear control boards, though the power draw on these boards is only a few hundred milliamps.

The second method uses a switch mode supply, which switches the transistor either fully on or off to produce an average voltage over the switching period. This produces a square pulse wave output that is smoothed using an inductor and capacitor resulting in a ripple of a few millivolts. Due to the transistor switching fully on and off, the voltage drop across the switching transistor is low due to when the transistor is on and conducting current, the voltage drop across its power path is minimal. When the transistor is off and blocking high voltage, there is almost no current through the transistor due its high resistance. Because of this method this switching method switch mode power converters usually has greater than 90% efficiency[9].

To ensure the reliability of the Raspberry Pi a reliable power source needs to be selected. While a linear regulator would be a low cost solution, the system does not operate off mains power and thus is limited to onboard generation methods provided by the vehicles alternator. Due to the inefficiencies of the linear supply, 37% efficient as calculated above, it is undesirable when compared to the switch mode system. While the switch mode is more expensive in standard industrial applications, low power reliable switch modes are available for use in aerial drones, in a price range comparable with linear regulators.

Based on the findings above, a Turnigy UBEC switch mode was selected to power the Raspberry Pi and its peripheral components. This switch mode supply, features a selectable voltage output of either 5 or 6V, providing a 3A nominal current output with a maximum output of 5A. It also operates with a voltage input range of 5.5v-23v, which covers the specified supply voltage of up to 15-volts.

Figure 6-35 Turnigy UBEC Switch mode[36]

To test the reliability of the selected switch mode supply, the system was run for an extended period powering the Raspberry Pi, using the bikes battery as the input supply. This revealed that the Turnigy UBEC switch mode supply was capable of running the Raspberry Pi,

even under heavy loads, with the Pi using 100% of its available processing power. During testing the GPS module, Wi-Fi card, Arduino Mega interface and the Touch screen display were all powered off the UBEC supply. The UBEC did get warm during this testing, but maintained a constant 5V output.

One problem revealed during this testing was the Raspberry Pi losing power when the vehicle's starter motor was activated. This can be attributed to the voltage across the battery being pulled low due to the high current draw of the starter motor. This could be attributed to the battery used to run the vehicle being the incorrect battery type for the bike, though the battery used was the only one available that fit the project budget. The solution developed for this problem, was to implement a secondary battery to as a backup for the UBEC to create an uninterruptible power supply (UPS).

6.6.2.1.2 Uninterruptible Power Supply (UPS) for Raspberry Pi

To ensure uninterrupted power to the Raspberry Pi control system, a backup power source needs to be provided. This subsystem needs to be charged off the vehicles main power circuit, but prevent any components from the main power circuit from drawing power from the backup power source. The Raspberry Pi UPS should also provide a status signal to the Pi regarding the vehicles main power status, so that action can be taken depending on the availability of this source. This would allow the Pi to perform a safe shutdown upon detection that the main power was interrupted. This interruption in main power could be caused by high power draw from system components during operation and can be caused by the activation of the vehicles starter motor or by the main power being switched off.

If the voltage from the main power supply drops, the backup battery should supply the required power to run the switch mode supply. An expected source of power failure is the operation of the starter motor, which is set to run for a maximum of two seconds. As soon as this power failure is detected by the Pi, it checks every second, for a total of three seconds to see if the power has been restored before shutting down. If the power is restored within this three-second window the Pi will not shutdown and resume normal operations. This allows time for the voltage levels to return to normal before shutting down to prevent any unnecessary system shutdowns.

The UPS needs to run the switch mode for a set amount of time after main power has been lost to allow the Pi to safely shutdown. In the event that the Pi needs more time than the fixed time available, due to busy system or data saving, then it should be able to override the automated power off sequence until the Pi is ready to shutdown.

To achieve the required functionality specified above, a simple system was designed and built. This is opposed to purchasing an existing, more expensive system that would be otherwise comparatively complicated and may not offer the same features required for the system.

Initial testing to measure the voltage drop on the main power supply during motor start shows that from an initial battery voltage of 12.6 volts. The battery voltage dropped to

7.74 volts when the starter motor was activated and rose to 11.9 volts over 0.7 seconds after the initial voltage drop. During this testing, it took 1.4 seconds for the motor to start with the voltage across the battery increasing to 13.4 volts charging from the alternator.

6.6.2.1.2.1 UPS System Design

Once the requirements of the UPS system were evaluated, it could be broken down into smaller systems. These included a backup battery, a charging method for the battery backup, a power switching method, timing circuit, main power detection, and a power off override to prevent power switching off.

A sealed lead acid battery was chosen as the backup battery. These batteries while heavy, have high capacity and discharge currents, while proving rugged and reliable, both of which are important qualities for outdoor use in an agricultural environment. To charge the backup battery while preventing the main supply from draining the backup battery, a power diode was used in series between the positive input from mains and the positive terminal of the battery. This keeps the battery charged to the following voltage level $V_{main} = V_{Battery} + V_{Diode}$, which means that the voltage across the backup battery will always be the main supply voltage minus the voltage drop across the diode. Therefore, when vehicle motor is operating the output of the alternator will keep the battery charged to 12.8-volts.

$$V_{Bat} = V_{alt} - V_{Diode} \qquad \textbf{6.12}$$
$$V_{Bat} = 13.4 - 0.6 = 12.8V$$

While many UPS systems use a microcontroller to monitor and ensure an uninterrupted supply to the system, the circuit designed does it using an analogue timing configuration. This timing method is achieved using a simple Resistor-Capacitor timing circuit, featuring a variable resistor to allow for adjustments in the discharge curve. A diode is included to prevent the capacitor, C1 in Figure 6-36, from discharging back through the main circuit after the input power has failed. The discharge voltage curve is monitored by a LM339 Comparator, which compares the discharge curve to a reference voltage.

The reference voltage is generated from the voltage drop across two diodes operated in a forward bias mode, in conjunction with resistor to limit the current. This gives a reference voltage of 1.2 ± 0.1 volts, which is generated using the backup battery as the voltage source. The comparator also draws its powered from the backup battery. This ensures the comparator remains powered after the mains power has been disconnected. The output of the comparator controls a power MOSFET. This MOSFET switches the power to the switch mode supply.

An override circuit is included to charge the timing circuit form the backup battery. This prevents the capacitor voltage form discharging below the 1.2-volt threshold, preventing the system from powering down. The Raspberry Pi has a set of general-purpose input-output pins (GPIO) that operate 3.3-volt logic level. Therefore, the override circuit must be capable of being driven at this voltage level. This system uses a transistor charge pump to provide 12 volts to charge the timing circuit, and maintain the timing of the analogue circuit, using a 3.3V input from the Raspberry Pi. Main power sensing is achieved using a Zener diode in series with a resistor. This provides a voltage drop of 2.5 volts across the diode that can be detected 3.3-volt Raspberry Pi input pins.

6.6.2.1.2.2 UPS System Testing

To test the circuit designed, individual components of the system was tested to ensure they functioning as expected. During this testing, a problem with the timing circuit was detected, with the circuit maintaining the expected timing. This was caused by the timing capacitor discharging back into the bikes main power circuit. This was due to the diode designed to stop the reverse current flow being damaged during circuit power up. This 1N4148 diode was damaged due to the timing capacitor's current draw, which exceeded the 1N4148s maximum current rating of 450mA (Datasheet within 0).

Due to the extremely high instantaneous current, the capacitor draws from the system on charging, a small resistor, R_Limit in Figure 6-36, was inserted in series in between the diode and the timing circuit. The positioning of this resistor limits the current to the timing capacitor, without affecting the resistance of the timing circuit. Calculating the resistance based on the voltage input from the main power source should not exceed 15 volts, and the current through the 1N4148 Diode should not exceed 450mA, a resistor value of 47Ω was selected. This results in a maximum current of 320mA through the diode.

$$I = \frac{V}{R} = \frac{15}{47} \approx 320mA \quad \textbf{6.13}$$

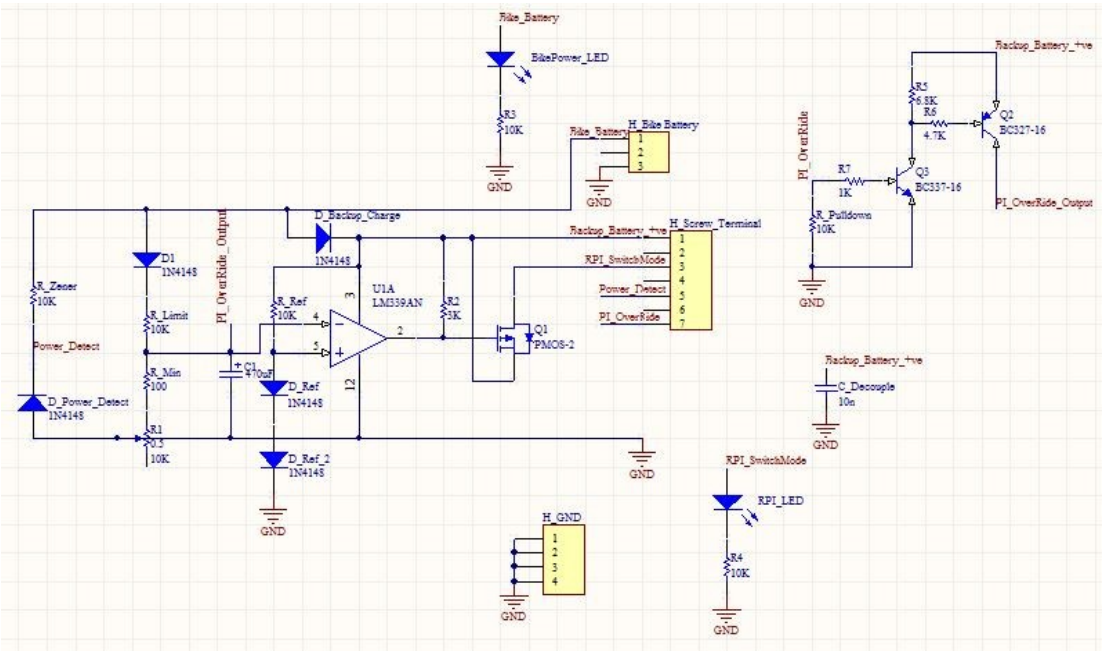


Figure 6-36 UPS Circuit Schematic

During testing, the Raspberry Pi would lose power as the input power source was removed, with the power from the UBEC insufficient keep the system running during this switch. The cause of this stemmed from the selected backup battery, and its ability to supply the instantaneous current required supplement from the main system the supply to the UBEC. This problem was solved by increasing the size of the sealed lead acid battery from a 12V 1.2Ah with an initial current of less than 0.36 amperes, to the next available battery which had a 12V 9Ahr rating with an initial current of less than 2.7 amperes.

Figure 6-37 UPS Board

A possibly solution could have been to add a series of capacitors in parallel with the battery, which would increase the effective instantaneous current capacity, as the battery was capable of running the Raspberry Pi. This was not implemented, as there was a larger battery available on site and provided a simpler option, though for cost saving measures, the capacitor option should be investigated in future designs.

Chapter Seven Software Development

This chapter covers the software developed for this system primarily consisting of hardware programming for the systems control modules, along with the user interface and path finding algorithm implemented on the Raspberry Pi controller.

7.1 Steering Control System

The purpose of the steering controller is to turn the steering mechanism to a specified position and control the linear actuator that engages the steering motor. To control the steering position, the position must be read from the 8-bit encoder with connections to digital and analogue pins outlined in section 5.2.4.

7.1.1 Encoder Position detection

The position of the encoder shaft is represented by a grey code number from 0-254, with each pin apart from the first representing a value from of 2^x following the pattern of 1, 2, 4, 8.... 64, 128. Reading each pin, a position number can be generated, i.e. 00101110 = 44 with the most significant bit (MSB) on the left. This grey code number must be converted to an absolute position using a look up table within the datasheet. Due to the nature of the grey code encoding, not every possible number within the 0-255 range registers as a position, with only 128 positions detectable.

The maximum speed of the steering motor will not produce more than 10 changes in position per second. Therefore, having this function run every time the controller completed a loop would reduce the response time for the system. The measured encoder position is stored as a global variable, allowing other routines to access the position without the need to call the read encoder function. Every 50ms, a polling flag is set by timer that causes the system to read the current encoder position and updates the global encoder position. If the new position is different from the previous, a serial message is sent to update the main controller, with the message formatted as follows; 'E:XXX' with the XXX representing the numeric position of the encoder from 0-127.

To simplify the encoder output, the centre point of the encoder is positioned to match the centre point of the steering system, with the gear reduction on the mechanism ensuring that every position is unique. Due to the complexity associated with aligning the encoder position and the steering system, an offset value is used to shift the output of the encoder so that the 64th position is present when the steering system is centred.

7.1.2 Motor Control logic

To adjust the vehicles heading, the controller must be capable of turning the steering mechanism to a set position. This functionality was achieved using a simple proportional-integral (PI) controller, which calculates the PWM duty cycle that controls the motor speed. The motor speed is calculated independently of the direction of travel, with the difference between the current position and the set position calculated as an absolute number, the PWM value is then calculated using the following formula, with a maximum and minimum value of 100 and 20 respectively.

$$Output(t) = P * Error(t) + I \int_0^t Error\,(t) * dt \quad \textbf{7.1}$$

To operate effectively, the control system requires tuning which was undertaken on a concrete surface, with final tuning constants of P= 4 and I = 0.05. The integral error is capped to the minimum and maximum PWM values to prevent over oscillations that could occur with large position changes. To prevent the integral value from growing out of control, the PI control calculation function is only called when an input variable changes, this is typically caused by the current encoder position differing from the encoder position used during the previous calculation.

While travelling to the desired position, the system operates within a while loop, which constantly updates the encoder position to check for any changes. Serial commands are also checked to provide a method of exiting this loop in case any issues arise. The current draw of the steering motor is monitored to detect any faults with the system, with this value transmitted to the serial controller. The function loop can self end if an abort command is received via the serial communications or once the steering has reached within one position of the desired position.

A second control system is used for the linear actuator, which engages the steering motor with the motor direction dependant on if the motor is being engaged or disengaged. This routine runs as a loop with the motor operated in a preset direction under a constant 80 percent duty cycle, and performs serial communication checks ensure that an abort command can be received. This control loop can exit under two conditions; the first is if an abort command is received via the serial communication system and the second is if the current draw measured from the motor exceeds 7.0 amperes as calculated by the CheckCurrent routine. This current limit is triggered by the actuator shaft reaching either of the preset actuator limits, which triggers a reverse diode to prevent the actuator from operating in that direction.

7.1.2.1 Motor Control Functions

To control the motor, five basic functions have created, CheckCurrent, MotorCW, MotorCCW, SetPWM and Brake. As the name suggests, the CheckCurrent routine measures the voltage value on the current sense pin, and converts it to a current by multiplying the measured value, which can range from 0-1024 by 0.0375. This multiplier is determined by the following calculation; with the CS pin outputting 0.13 volts per amp of output current and the ATmega328 providing analogue voltage increments of 4.88mV.

$$Multiplier = \frac{0.00488}{0.13} = 0.0375\ (3sf) \quad \textbf{7.2}$$

The MotorCW, MotorCCW and Brake routines function in the same manner, with the outputs to the motor governed by the truth table presented in table 7-1.

IN_A	IN_B	$DIAG_A/EN_A$	$DIAG_B/EN_B$	OUT_A	OUT_B	CS	Operating mode
1	1	1	1	H	H	High Imp.	Brake to V_{CC}
	0				L	$I_{SENSE} = I_{OUT}/K$	Clockwise (CW)
0	1			L	H		Counterclockwise (CCW)
	0				L	High Imp.	Brake to GND

7-1 Truth table for VNH2SP30 normal operating conditions (Appendix F)

The MotorCW function is outlined in Figure 7-1, with the function requiring two input variables that select the motor that will be run as well as the PWM percent that will be applied to the motor input. Because the function controls multiple motors, the directional input pins for each motor are specified as a set of three character variables, M for motor, A/B for motor Driver Pin and integer number to denote the specified motor. The brake function operates in a similar manner with both the A and B pins pulled low, causing the driver to break to ground and the PWM output set to low.

```
void MotorCW(int Motor, int PWMVal)
{
  if (Motor == 1)
  {
    MotorA = MA1;
    MotorB = MB1;
  }
  else
  {
    MotorA = MA2;
    MotorB = MB2;
  }
  digitalWrite(MotorA,HIGH);
  digitalWrite(MotorB,LOW);
  SetPWM(Motor,PWMVal);
}
```

Figure 7-1 Motor control function

The SetPWM function is the final routine used within the motor control class, which provides the PWM output for the motor driver using the timer one PWM libraries. The PWM output produced a 20Hz signal. With the frequency set on system start up, the timer also provides the encoder check flag interrupt. This routine accepts an integer value from 0-100, representing the desired PWM duty cycle, which is converted to an integer value between 0-1023 to be passed to the PWM timer function. The output pin is selected via an input variable representing either motor one or two and is used to select the desired motor.

7.1.3 Serial Communication Commands

There are three commands available to the main controller, with all other coded commands only provided for debugging purposes. These three commands are:

Read - returns the current stored encoder position.

Goto - requires an integer number from 1-125 otherwise an invalid position error is sent via the serial interface. This command will actuate the steering system until the encoder output matches the specified integer position.

ManOn/ManOff - sets the manual control mechanism.

Due to the self-locking system outlined in section 4.2, the control system is not required to maintain a fixed position, which can only be changed through the powering of the motor. However, this is only true if the manual control system is not engaged. The final two commands (ManOn/ManOff) control the manual mode linear actuator. This causes the system to either retract/disengage or extend/engage. Disengaging the steering motor allows the user to turn the handle controls freely, where as extending to engage the motor pulley and timing belt preventing the user from manually turning the handlebars.

7.2 Speed Control System

The speed control system actively adjusts the throttle and braking systems to maintain a preset speed value with the current speed measured via an encoder mounted on the vehicles drive system. As outlined in section 4.3.2, the rear brakes are controlled via a linear actuator, using the same model of linear actuator used in the manual steering control. To control the speed, it must first be detectable the output of the speed encoder providing 5-volt pulses during the revolution, with the encoder used providing 200 pulses per revolution as outlined in section 5.3.1.

To measure the speed of the vehicle, two interrupts were attached to the encoder input channels, that are triggered by the rising voltage signal of the encoder pulses and increment a global 'Direction' count variable within the interrupt service routine (ISR). A Timer is used to retrieve the value of the count at a set time interval of 10 times per second, allowing a number of counts per second to be extrapolated. By measuring the distance travelled by the vehicle per count, a speed in metres per second is calculated and used as the basis for the speed control routine.

The speed is controlled using two systems, a servo controlled actuator increasing the throttle. This increases the speed depending on the selected gear and a linear actuator system that applies the rear break. A simple PI controller can provide the necessary control for the vehicles speed.

To ensure the safe operation of this vehicle, the rate of acceleration that the controller is able to provide needs to be limited to prevent sudden acceleration. As an added safety feature, the control system disables the throttle system if either the front or rear brakes are applied. This feature allows an operator to stop the vehicle simply and safely and is effective independent whether the vehicle is operating manually or autonomously. Further development of this feature is required; with the addition of a hill start routine required, which will detect the angle of the vehicles and provide a throttle breaking combination.

7.2.1 Serial Commands

There are three main commands that can be sent to the speed controller. These commands control the Braking, Throttle and Manual Status. Both the Braking and throttle commands require a number input that ranges from 0-100 and 0-136 respectively in conjunction with their respective command designators. Each command begins with a unique designator, with a B representing braking and a T representing a throttle command. The Braking command requires an input percentage value that the system will adjust the position of the linear to match, with the maximum and minimum positions determined by hard coded positions measured by the brake position sensors outlined in section 4.3.2.

These hard coded positions can be adjusted via serial commands, with the commands words Get, Set and Ct provided to set and transmit the values for the rear brake maximum, the rear brake off and the front brake off. While these positional limits can be adjusted via the serial commands, the controller does not feature any storage system, thus the limits will be reset to the default hard coded positions on start up. Therefore, on start up the limits must be reset using the Set command from the positional values stored in the main controller.

In order to update these position the limits must be recorded before powering off with the Get command, which will return the current stored values for the specified variable. The system also features the Ct command, which will set the position of the limits based on the current reading on the sensor. This Ct command is particularly useful when the system is first set up or altered, as it only requires the user to manipulate the hand controls to their maximum positions.

The final configurable variable is the throttle scale factor, that is responsible for scaling down the input throttle commands by a scale factor and was primarily used to dampen the sensitivity of the throttles manual control mechanism.

7.3 Gear Control System

The gear-changing controller relies on commands received via the serial interface to signal gear changes, as it has no direct access to sensors monitoring engine status or vehicle speed. Limit switches are bound to external interrupt pins to detect a rising voltage signal caused by an actuation of the gear changing leaver. When an interrupt is detected, a count variable is incremented to reflect the current gear, with the indicator lights checked to confirm reverse and neutral gears.

Due to the omission of switch debouncing in the control circuit design, multiple interrupts were triggered by a single gear change, resulting in unreliable detection. Debouncing was instead implemented via an in-code solution, with a millisecond delay followed by a voltage level check, allowing any debouncing to subside. For the external interrupts to increment the current gear position, the initial position of the gear must be known, thus on start-up the system will enter 'home' mode, actuating until the neutral gear is detected. This homing mode is capable of being called at any time via a serial command. The neutral and reverse gears can be detected via an output from the Indicator lamp detection circuit outlined in section 6.1.4.

To reduce the gear change time limited by the actuation speed of the gear leaver, the controller features a 'Preload' function that operated by positioning the actuation leaver to be in contact with the leaver arm as described in section 6.4.1. This preload function is set via a serial command and works according to the following manner shown in Figure 7-2.

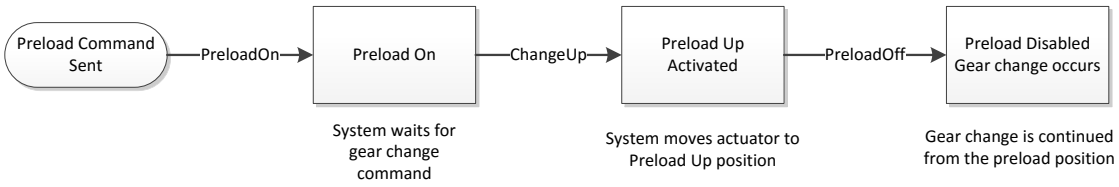


Figure 7-2 Preload function diagram

While this preload function is not necessary for a gear change, it reduces the average time for a gear change by up to half. On detection of a gear change via the external interrupts, the system updates the internal gear variable and transmits the current gear position via the serial connection using the "G:" code identifier followed by the new gear position represented as a single character, R, N or numeric 1-5.

7.4 ATmega Module Interface System

The ATmega controller integrates and controls all of the input-output devices used on the vehicle, receiving and acting on data from the individual modules as well as commands received via the Raspberry Pi controller. This controller is primarily responsible for controlling the instructions sent to the control modules, to control the speed, direction and gear selected based on the instructions received from the Raspberry Pi control system. Figure 7-3 shows the interactions between the modules and ATmega controller.

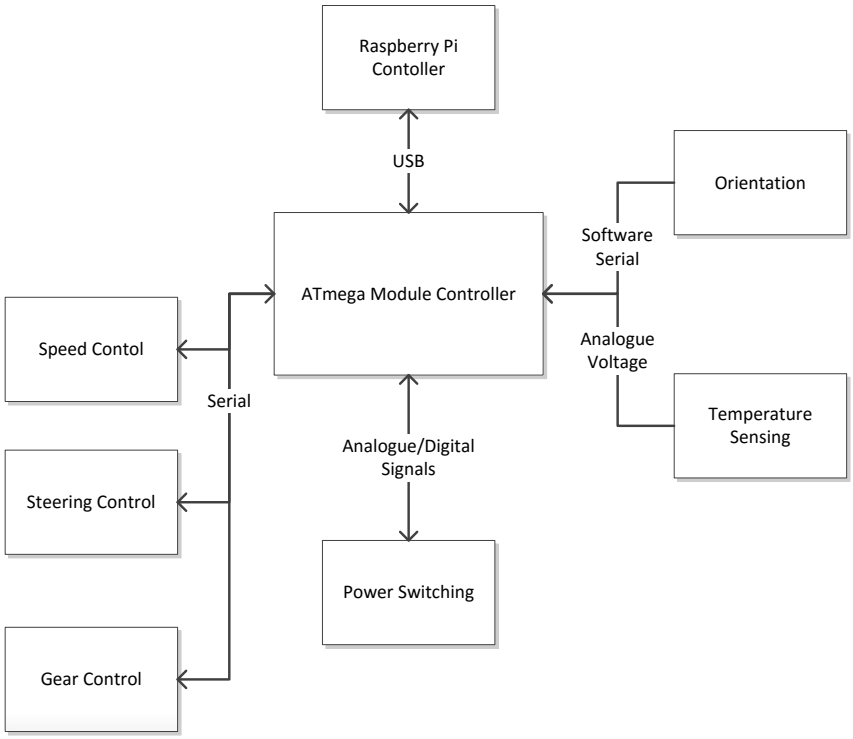


Figure 7-3 ATmega Controller interaction diagram

During normal operations, the Raspberry Pi controller is responsible for determining a path to the specified destination. Therefore, the Pi controller will send simplified commands

such as speed and heading values to the ATmega, where it will adjust the vehicles speed, gear and steering angle based on a set of rules that will govern how these individual modules will interact with one another.

Examples of this include speed-gear interactions and the speed-steering control, with the speed that the vehicle is capable of achieving at any time, determined by not only the throttle/break actuation of the speed controller, but the selected gear which depends on the current speed and RPM of the engine. The second example relates the speeds of the vehicle as it is limited by the angle of the steering system, which will alter the cornering speed of the vehicle depending on the steering angle, as well as the horizontal inclination of the vehicle to the ground.

The ATmega controller interfaces with the CMPS10 digital compass via a virtual serial interface, outlined in section 6.5, with the required data requested via a serial command to provide a compass bearing as well as the pitch and roll for the unit. This pitch and roll is used as a safety feature to control the speed and alert a home base if the vehicle has rolled. Due to time constrains and delays on the project, the integration between the modules to control the selected gear based on speed and the speed based on the turning angle was unable to be implemented.

7.5 Raspberry Pi User Interface

The user interface is designed to provide the user with a simple method of interacting with the system during vehicle operation, by only providing the useful information to the user. Data available to the user includes the time, GPS position, speed, RPM, Temperature (air and engine), selected gear and the compass data (bearing, pitch and roll). This prototype user interface shown in Figure 7-4 is designed to display two tab pages, the first features the data mentioned above, providing the user with any relevant data while driving the vehicle and includes the engine starter button. The second display tab is used for building and displaying the maps that the system will use for navigation, allowing the user to map the vehicles operational boundaries and any obstacles within the environment.

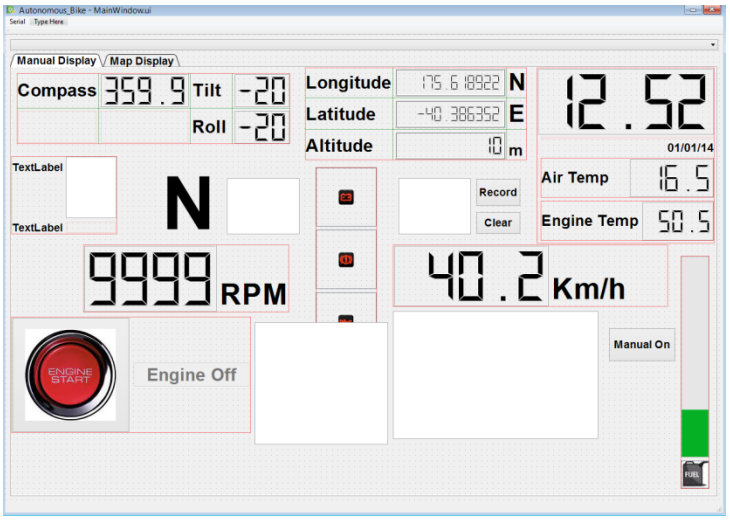


Figure 7-4 Program User Interface

Figure 7-4 features the user interface implemented using the Qt Creator program to produce a visual interface for the system, with the background code written in Microsoft Visual Studio using the C++ language. The underlying code controlling the serial communication for the control-interface system worked. Currently the system requires to be reinitialized on occasion in order to receive and transmit data to and from the ATmega controller. Further work obviously is needed to improve the reliability, especially as the map display or map generation system is fully implemented.

7.6 Path-finding Control Module

In order to self navigate; the system requires a method of determining a traversable path connecting the start and destination positions, avoiding any obstacles within its path. As outlined in section 2.2.2, a path-finding algorithm must be implemented using a map to store any data available for its surrounding environment. Based on the existing research and methodologies for autonomous vehicles, a grid based map was selected in which to store the data, with an A* search algorithm selected to path-find using this map type.

As discussed in [2], grid based maps can result in large data files, which could adversely affect the performance of the system. Thus to keep data overheads small, the maps would cover only the operational area for the specific field/zone that it is operating in. If the size of this zone exceeds a specified area, the resulting map could be broken into multiple segments, where they would be combined for global path finding to find the optimal navigational route. These segments would be used individually for local object avoidance path planning, following the path generated from the global planning.

In order to use the area map, the system must be able to locate its position and orientation in relation to the map. Therefore, the map must have a datum that specifying the GPS coordinates of a point on the map, this point with the known grid size can be used to position the vehicle on the map based upon its measured GPS coordinates. In order to test the viability of this mapping system, a test program was developed in Microsoft Visual Studio featuring a map display, allowing the visualization of the A* search algorithm. The map data for this test system is stored as a text file, using characters to represent the grid squares, with each line representing a grid line in the Y direction. Four characters represent the grid squares, A, B, E and S, with A indicating the vehicle can enter the grid, B if the square is blocked, with S and E representing the Start and End grid position respectively.

Figure 7-5 shows the test program, displaying the path finding process completed on the 50 by 50 cell map with Table 7-2 acting as the maps legend. The light blue cells in Figure 7-5 indicate that the cell was inspected during the algorithms searching process, with the shortest path from start to end highlighted in dark blue.

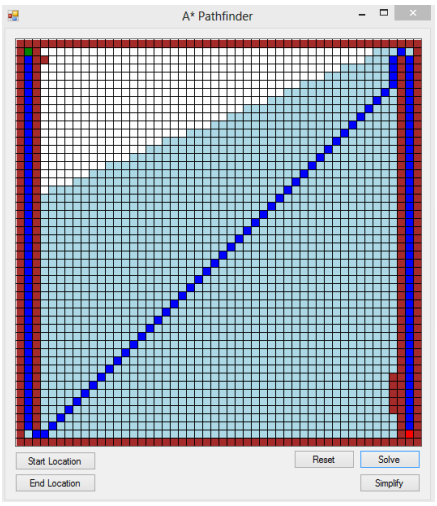


Figure 7-5 A* path finding test program

Cell Color	Definition
Crimson	Impassable
White	Enterable
Light Blue	Checked by algorithm
Blue	Generated Path
Green	Start Point
Red	Destination Point

Table 7-2 A* Map Legend

While this basic algorithm has been successfully in determining a viable path joining the start position to the destination, it requires several additions to ensure that the vehicle could successfully navigate this part. In order to determine a navigable path for the vehicle, several factors have to be incorporated into the algorithm, including details about the vehicle itself. These details need to cover the size, dimensions and turning radius of the vehicle. Incorporating these details into the algorithm will ensure that the vehicle will be capable of navigating the plotted course, ensuring the vehicle will have the required clearance to travel around obstacles, as well as the manoeuvrability to travel around any corners along the vehicles path.

Further work is required to incorporate these vehicle features into the path-finding module, with some preliminary integration of the vehicles dimensions having been implemented, though further testing is required to ensure correct functionality. Modifications to include the vehicles turning radius has not yet been undertaken, which must include multipoint turning to navigate through enclosed spaces. A simplification feature must be included to produce several waypoints that the system will use instead of following as set path, and will result in a simplified navigational system. Other features to be implemented should include a multipath feature that will check for simpler paths if the current path will be difficult to navigate due to the use of multipoint turns.

Chapter Eight System Testing, Implementation and Results

During the development of the systems outlined in Chapter Five and Chapter Seven, testing was undertaken on the basic functionality of the components present to verify the design of the individual modules. This chapter will only cover the testing undertaken to test the functionality of the individual system components, and their ability to communicate and act as a whole.

To ensure the manual control features on the quad bike system operated via commands sent from the system controller, testing was undertaken with commands sent to the ATmega module controller. This feature was tested using two means of transmitting the manual control commands; via a windows serial interface and the Raspberry Pi touchscreen user interface implemented on the bike. This testing requires both the throttle and steering controllers to enter and maintain manual mode, with the vehicles throttle controlled via the thumb trigger and the steering motor disconnecting via the linear actuator. Further testing was undertaken with the Raspberry Pi interface to test the interfaces ability to communicate with the vehicle controller and receive GPS data via the USB connection.

The inverse of this must also occur, where the throttle control is returned to the control of the system and the steering motor actuator must be re-engaged, with sufficient belt tension to prevent the belt from slipping when driving the steering mechanism. Repeated testing found that the system was capable of engaging the steering motor, with only minor adjustments required to the limit controls of the system. Testing of the steering system proved the mechanism was capable of positioning the vehicles wheels to any of the 125 detectable positions between the mechanical limits of the vehicles steering. The rate of turn for the mechanical system was limited by the maximum speed of the motor, with only a minor decrease of 0.4 seconds for the limit-to-limit travel time on a grass surface when compared to concrete testing. The measured stationary actuation time on grass is 2.8 seconds comparable to a no-load time of 2.4 seconds, indicating that torque was not the major limiting factor for this system.

The braking system developed featured a disable throttle function that was activated to set the throttle control to its lower limit once braking was detected for either the front or rear braking systems. Performance during in field testing showed further improvement is required, as the rear brake system often did not return below the programmed threshold, which could cause the throttle controls to be disabled. Increasing the limit at which the throttle cut off was implemented could result in an increase in reliability. However, further investigation is needed to prevent the reduction in detection levels with low level braking.

An accurate detection of the vehicles speed has not been obtained from the vehicle controller due to the unreliability of the Raspberry Pi's serial interface, with a number of the data packets containing information regarding the wheels rotations, as measured by the gearbox encoder being dropped by the system. This has prevented accurate measurements to correlate the vehicles travel distance per encoder tick from being recorded.

The gear changing system performs as designed with the system capable of changing to any of the gearboxes gear positions based on commands received by the controller system.

Manual gear changing capabilities are maintained during operation, with the ability of manual and automatic gear changing modes. Automatic gear changing has not yet been implemented due to the inaccuracies surrounding the vehicles speed; hindering the measurement of the vehicles speed and engine RPM during gear change. Without these two variables, the development of gear changing profiles was not possible, though basic gear changing was implemented for reverse and first gear as the vehicle is in a stationary state during these changes, thus the vehicles speed is not a relevant variable. To develop the gear change profile for the vehicle, the vehicles RPM, speed, throttle position and current gear must be recorded over time during manual vehicle operation. This will allow a set of rudimentary gear changing conditions to be developed with the acceleration profiles for each gear, though this can be improved through the addition of engine load calculations.

Testing on the user interface display focused on the ease of interaction with the displayed buttons and the readability of the on screen information. The touchscreen interface displays several intractable buttons that control the manual controls for the user, including the engine start and manual control surface activation. The touch capabilities of the interface was test through the repeated pressing of the displayed buttons during different conditions with the vehicle stationary, moving slowly over near flat surface and moving at walking speed over uneven terrain.

Testing was not undertaken at higher speeds as it would be deemed unsafe for the user to be interfacing with this system during these times unless the vehicle is operating autonomously. Testing of the on screen interface revealed indicated an increase in button size for the controls was necessary, with the existing button size providing unreliable detection during all test conditions. Based on the touchscreen interface testing, the system should lock the screen controls when the user is manually operating the vehicle over a specified speed. This would prevent user distraction induced by the user's interaction with the screen, reducing the likely hood of an accident caused by the user's inattention of the surrounding environment.

Autonomous navigation and path finding requires accurate location and speed control. Due to budget restrictions, the developed platform only provides rough location information with the low cost GPS module. In addition, the ultrasonic sensor is only able to provide obstacle detection within a maximum range of seven metres.

Chapter Nine Discussion, Conclusion and Recommendations

This research aimed to develop a base vehicle for an autonomous farming system using an existing farming vehicle as the platform. With the systems design focused around modular components, implementation across multiple vehicle types is possible with minimal alterations required. This required the development of independent control modules to operate the manual controls present on vehicles that would allow it to operate/navigate independently of a human operator.

The modules developed within this research were able to provide the required functionality and adequately control the vehicle steering, speed and gearbox systems. Testing undertaken on the system proved the modules were able to perform as expected. The modules and the proposed system built a platform that, with further development, could allow for fully autonomous control over the vehicles speed and direction.

The development of a vehicle management system proved to be a positive attribute of the system that networked the independent modules. This provided a central communication and control interface, path finding and object avoidance interface for the Raspberry Pi controller.

The touchscreen display interface added a versatile feature to the system. It provided a user interface for the vehicles higher-level control system, to provide details on the vehicles status as well as the capability to display maps generated by the object avoidance system. This also displays the position and orientation of the vehicle on its internal operating environment maps, based on the vehicles GPS coordinates and compass heading.

This research proposed and developed a generic vehicle control system that is capable of controlling the various components present on a quad bike, allowing a computer control system to operate the vehicle without the need for human intervention. While successful in the development of a basic vehicle control system, further development of the systems programming is required for a fully autonomous control of the system. This is particularly pertains to the automatic adjustment of the systems outputs based on the current value of selected systems variables.

To ensure the system operates in a safe and reliable manner, the individual system modules should automatically communicate with one another through the vehicle controller to alter the vehicles speed. This speed alteration should be based on the vehicles turning radius and the pitch and roll of the vehicle, reducing the risk of unintentional rolling of the vehicle due to high-speed manoeuvres across various terrain profiles.

The preliminary object avoidance system developed generates a traversable path to the vehicles destination based on a pre-generated map of the operating environment. With further improvement on path planning based on the feedback of an object detection module, the vehicle could operate autonomously and automatically detect new objects within its environment.

Future development of this system should focus around the development of the control, path finding and object detection systems. Due to limited project funding, several of the control systems are not capable of operating at speeds that ensure optimal responsiveness to system inputs. Thus, it is recommended that the following systems should be redesigned to improve effectiveness and reliability of the specified modules within this project.

The current vehicle steering system uses a linear actuator to tension the motor timing belt, allowing the user to operate the vehicles steering. This design uses externally mounted moving parts to disconnect the motor from the steering gearbox. Long-term use of this design could result in breakdowns due to wear caused by actuation when exposed to the elements, thus the system should be redesigned to feature an enclosed clutch system eliminating exposed moving parts within the design.

Future development within the speed control system should focus around faster actuation time from limit to limit on the rear braking system, with a similar system implemented to control the front braking system. Any changes to the braking system should still maintain a user implemented braking system for both the front and rear braking, to ensure manual control is still available. While not as critical as the braking system, improvements to the throttle control mechanism should be investigated though the use of a stepper motor actuator. Using a stepper motor could improve the control resolution, with benefit of automatically returning to the minimum position when depowered.

References

[1] H. Cheng, *Autonomous intelligent vehicles : theory, algorithms, and implementation*. London ; New York: Springer, 2011.

[2] U. Özgüner, T. Acarman, and K. A. Redmill, *Autonomous ground vehicles*. Boston: Artech House, 2011.

[3] A. El-Rabbany, *Introduction to GPS : the Global Positioning System*: Boston, MA : Artech House, c2006, 2006.

[4] R. T. S. T. P. Callister, "The factors which have resulted in migrant workers being 'Essential' Workers on New Zealand dairy farms," presented at the Labour, Employment and Work Conference 2010, 2010.

[5] L. G. S. T. Sundar Raman, Inc. (2005, Performance Evaluation of Global Differential GPS (GDGPS) for Single Frequency C/A Code Receivers. Available: http://www.gdgps.net/system-desc/references.html

[6] J.-M. Zogg. (2009). *GPS - Essentials of Satellite Navigation*.

[7] NAVSTAR. (1996). *NAVSTAR GPS USER EQUIPMENT INTRODUCTION*.

[8] S. Group. (2011). *SKF Power transmission belts*.

[9] H. J. Zhang, "Basic Concepts of Linear Regulator and Switching Mode Power Supplies," in *Linear Technology*, ed, 2013, p. 16.

[10] U. S. Government, "GLOBAL POSITIONING SYSTEM STANDARD POSITIONING SERVICE SIGNAL SPECIFICATION," U. S. A. Force, Ed., 2nd ed, 1995, p. 51.

[11] M. R. Mosavi, M. Sadeghian, and S. Saeidi, "Increasing DGPS Navigation Accuracy using Kalman Filter Tuned by Genetic Algorithm," *International Journal of Computer Science Issues,* p. 246, 2011.

[12] T. G. Jay Farrell, "Differential GPS Reference Station Algorithm—Design and Analysis," *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY,* vol. 8, p. 13, 2000.

[13] T. K. M. H. Saka, C. Ozsamli and R. M. Alkan, "Sub-Metre Accuracy for Stand Alone GPS Positioning in Hydrographic Surveying," *Journal of Navigation,* vol. 57, pp. 135-144, January 2004 2004.

[14] N.-N. T. Zheng, Shming; Hong Cheng; Qing Li; Lai, Guanpi; Wang, Fei-Yue, "Toward Intelligent Driver-Assistance and Safety Warning Systems," *IEEE Intelligent Systems,* vol. 19, pp. 8-11, 2004.

[15] C. D. McGillem and T. S. Rappaport, "A Beacon Navigation Method for Autonomous Vehicles," *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY,* vol. 38, p. 8, 1989.

[16] S. J. Hae Don Chon, Heejae Jung, Sang Won An, "Using RFID for Accurate Positioning," *Journal of Global Positioning Systems,* vol. 3, p. 8, 2004.

[17] G. I. S. New Zealand Gazette Office, Department of Internal Affairs "Radiocommunications Regulations (General User Radio Licence for Short Range Devices) Notice 2015," 2015.

[18] R. P. Foundation, "Raspberry Pi Model Specifications - Raspberry Pi Documentation," pp. This section contains documentation with technical information about the Raspberry Pi hardware, including official add-ons and the Pi itself., 2015.

[19] W. S. N. Zealand. (2015). *Quad bike safety | Worksafe*. Available: http://www.business.govt.nz/worksafe/information-guidance/national-programmes/quad-bike-safety

[20] L. I. N. Zealand. (2016). *PositioNZ - Real Time Service*. Available: http://www.linz.govt.nz/data/geodetic-services/positionz/positionz-real-time-service

[21] (2015). *TRX250TM | Honda Motorcycles NZ*. Available: http://www.hondamotorbikes.co.nz/motorcycles/farm-2-and-4-wheel/quad-bikes-and-atvs/trx250tm/

[22] L. L. C. Atv Source. (2015). *ATV Source - Manufacturers - Kawasaki - Mules 2003 - Mule 3010*. Available: http://www.atvsource.com/manufacturers/kawasaki/2003/mule_3010.htm
[23] E. P. Online. *TRX-300 Steering system photo*. Available: http://www.etonparts.com/steering_system_rascal.htm
[24] G. Science. (2015). *Earth's Magnetic Field/Declination around New Zealand*. Available: http://www.gns.cri.nz/Home/Our-Science/Earth-Science/Earth-s-Magnetic-Field/Declination-around-New-Zealand/Figure-1
[25] C. M. Spares. (2016). Available: http://www.cmsnl.com/products/cable-comp-choke_17950hm5671/#.VqBqP1K51oM
[26] P. E. Black. (2014). *Gray code*. Available: http://xlinux.nist.gov/dads//HTML/graycode.html
[27] Pololu. (2015). *Pololu - VNH2SP30 Motor Driver Carrier MD01B*. Available: https://www.pololu.com/product/706
[28] V. LLC. (2016). *Raspberry Pi Model B Revision 2 (512 MB) - Virtuabotix LLC*. Available: https://www.virtuabotix.com/product/raspberry-pi-model-revision-2-512-mb/
[29] Nantonos. (2012). *Arduino Mega2560 R3 pinouts photo*. Available: http://forum.arduino.cc/index.php?topic=125908.0
[30] Y. Lancet. (2012). *What Are The Differences Between Capacitive & Resistive Touchscreens?* Available: http://www.makeuseof.com/tag/differences-capacitive-resistive-touchscreens-si/
[31] Y. Lancet. (2012). *Resistive Touchscreen*. Available: http://www.makeuseof.com/tag/differences-capacitive-resistive-touchscreens-si/
[32] T. Technetronics. *10" LCD LVDS bundle with capacitive touchscreen and ambient light sensor (RaspberryPi)*. Available: http://www.tenettech.com/product/3110/10-lcd-lvds-bundle-with-capacitive-touchscreen-and-ambient-light-sensor-raspberrypi
[33] F. F. O. N. Zealand. *New Zealand National Policy: Immigration and Labour*. Available: http://www.fedfarm.org.nz/advocacy/National-Policy/Immigration-and-Labour.asp
[34] C. Electronics. (2014). Available: http://www.chalk-elec.com/
[35] M. Burris. (2015). *Types of Voltage Regulators*. Available: http://components.about.com/od/Components/a/Types-Of-Voltage-Regulators.htm
[36] Hobbyking.com. *TURNIGY 3A UBEC w/ Noise Reduction*. Available: http://www.hobbyking.com/hobbyking/store/__4319__TURNIGY_3A_UBEC_w_Noise_Reduction.html

Appendices

Appendix A GPS Formula

M_0	Mean Anomaly at Reference Time
Δn	Mean Motion Difference from Computed Value
e	Eccentricity
$(A)^{1/2}$	Square Root of the Semi-Major Axis
$(OMEGA)_0$	Longitude of Ascending Node of Orbit Plane at Weekly Epoch
i_0	Inclination Angle at Reference Time
ω	Argument of Perigee
OMEGADOT	Rate of Right Ascension
IDOT	Rate of Inclination Angle
C_{uc}	Amplitude of the Cosine Harmonic Correction Term to the Argument of Latitude
C_{us}	Amplitude of the Sine Harmonic Correction Term to the Argument of Latitude
C_{rc}	Amplitude of the Cosine Harmonic Correction Term to the Orbit Radius
C_{rs}	Amplitude of the Sine Harmonic Correction Term to the Orbit Radius
C_{ic}	Amplitude of the Cosine Harmonic Correction Term to the Angle of Inclination
C_{is}	Amplitude of the Sine Harmonic Correction Term to the Angle of Inclination
t_{oe}	Reference Time Ephemeris
IODE	Issue of Data (Ephemeris)

Figure 9-1 Ephemeris Parameters Definitions[10]

Parameter	No. of Bits	Scale Factor (LSB)	Effective Range***	Units
IODE	8			(see text)
C_{rs}	16*	2^{-5}		meters
Δn	16*	2^{-43}		semi-circles/sec
M_0	32*	2^{-31}		semi-circles
C_{uc}	16*	2^{-29}		radians
e	32	2^{-33}	0.03	dimensionless
C_{us}	16*	2^{-29}		radians
$(A)^{1/2}$	32	2^{-19}		$meters^{1/2}$
t_{oe}	16	2^{4}	604,784	seconds
C_{ic}	16*	2^{-29}		radians
$(OMEGA)_0$	32*	2^{-31}		semi-circles
C_{is}	16*	2^{-29}		radians
i_0	32*	2^{-31}		semi-circles
C_{rc}	16*	2^{-5}		meters
ω	32*	2^{-31}		semi-circles
OMEGADOT	24*	2^{-43}		semi-circles/sec
IDOT	14*	2^{-43}		semi-circles/sec

* Parameters so indicated are two's complement, with the sign bit (+ or -) occupying the MSB;
** See Figure 2-8 for complete bit allocation in subframe;
*** Unless otherwise indicated in this column, effective range is the maximum range attainable with indicated bit allocation and scale factor.

Figure 9-2 Ephemeris Parameters[10]

$A = \left(\sqrt{A}\right)^2$	Semi-major axis
$n_0 = \sqrt{\frac{\mu}{A^3}}$	Computed mean motion - rad/sec
$t_k = t - t_{oe}$*	Time from ephemeris reference epoch
$n = n_0 + \Delta n$	Corrected mean motion
$M_k = M_0 + nt_k$	Mean anomaly
$M_k = E_k - e \sin E_k$	Kepler's equation for eccentric anomaly (may be solved by iteration) - radians
$\nu_k = \tan^{-1}\left\{\frac{\sin\nu_k}{\cos\nu_k}\right\} = \tan^{-1}\left\{\frac{\sqrt{1 - e^2}\ \sin E_k\ /\ (1 - e\cos E_k)}{(\cos E_k - e)\ /\ (1 - e\cos E_k)}\right\}$	True anomaly
$E_k = \cos^{-1}\left\{\frac{e + \cos\nu_k}{1 + e\cos\nu_k}\right\}$	Eccentric anomaly
$\Phi_k = \nu_k + \omega$	Argument of latitude
	Second Harmonic Perturbations
$\delta u_k = C_{us} \sin 2\Phi_k + C_{uc} \cos 2\Phi_k$	Argument of latitude correction
$\delta r_k = C_{rc} \cos 2\Phi_k + C_{rs} \sin 2\Phi_k$	Radius correction
$\delta i_k = C_{ic} \cos 2\Phi_k + C_{is} \sin 2\Phi_k$	Correction to inclination
$u_k = \Phi_k + \delta u_k$	Corrected argument of latitude
$r_k = A(1 - e\cos E_k) + \delta r_k$	Corrected radius
$i_k = i_0 + \delta i_k + (\mathrm{IDOT})\ t_k$	Corrected inclination
$x_k' = r_k \cos u_k$ $y_k' = r_k \sin u_k$	Positions in orbital plane
$\Omega_k = \Omega_0 + (\dot{\Omega} - \dot{\Omega}_e)\ t_k - \dot{\Omega}_e\ t_{oe}$	Corrected longitude of ascending node
$x_k = x_k' \cos\Omega_k - y_k' \cos i_k \sin\Omega_k$ $y_k = x_k' \sin\Omega_k + y_k' \cos i_k \cos\Omega_k$ $z_k = y_k' \sin i_k$	Earth-Centered, Earth-Fixed coordinates

* t is GPS system time at time of transmission, i.e., GPS time corrected for transit time (range/speed of light). Furthermore, t_k shall be the actual total time difference between the time t and the epoch time t_{oe}, and must account for beginning or end of week crossovers. That is, if t_k is greater than 302,400 seconds, subtract 604,800 seconds from t_k. If t_k is less than -302,400 seconds, add 604,800 seconds to t_k.

Figure 9-3 Element of Coordinate Systems[10]

Appendix B Bourns ACE-128 - Datasheet

Features

- Absolute encoder / absolute code output
- Digital output
- Sturdy construction
- Bushing mount
- Available with PC board mounting bracket (optional)
- *RoHS compliant

BOURNS®

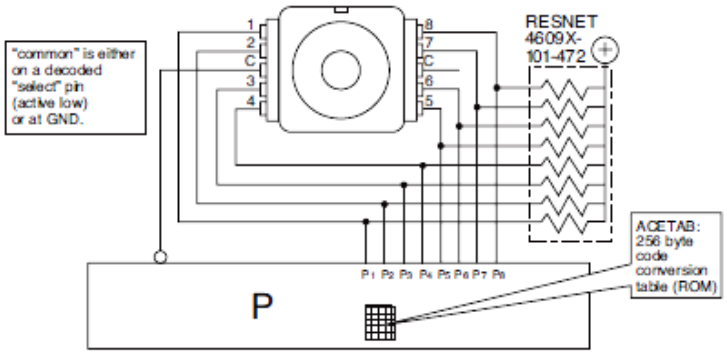
EAW - Absolute Contacting Encoder (ACE™)

General Information

Until now, the choice of an absolute encoder meant an expensive, and larger-sized product. Through the use of combinatorial mathematics, the absolute code pattern of the Bourns® Absolute Contacting Encoder (ACE™) is placed on a single track for a very economical, energy-efficient and compact product. Bourns® ACE™ provides an absolute digital output that will also retain its last position in the event of a power failure.

An intelligent alternative to incremental encoders and potentiometers, the Bourns® ACE™ is ideally suited for many industrial, automotive, medical and consumer product applications.

Recommended Control Diagram for ACE-128



Electrical Characteristics

Output	8-bit code with 128 absolute states
Closed Circuit Resistance	5 ohms maximum
Open Circuit Resistance	100 K ohms minimum
Contact Rating	10 milliamp @ 10 VDC or 0.1 watt maximum
Insulation Resistance (500 VDC)	1,000 megohms minimum
Dielectric Withstanding Voltage (MIL-STD-202 Method 301)	
Sea Level	1,000 VAC minimum
Electrical Travel	Continuous
Contact Bounce (60 RPM)	2.7 milliseconds maximum*
RPM (Operating)	120 maximum

Environmental Characteristics

Operating Temperature Range	-40 °C to +85 °C (-40 °F to +185 °F)
Storage Temperature Range	-40 °C to +85 °C (-40 °F to +185 °F)
Humidity	MIL-STD-202, Method 103B, Condition B
Vibration	15 G
Contact Bounce	0.1 millisecond maximum
Shock	50 G
Contact Bounce	0.1 millisecond maximum
Rotational Life	50,000 shaft revolutions minimum
IP Rating	IP 40

Mechanical Characteristics

Mechanical Angle	360 ° Continuous
Running Torque	0.5 to 1.5 N-cm (0.75 to 2.50 oz-in.)
Mounting Torque	79 N-cm (7 lb.-in.) maximum
Shaft Side Load (Static)	4.5 kg (10 lbs.) minimum
Weight	Approximately 14 gms. (0.50 oz.)
Terminals	Printed circuit board terminals
Soldering Condition	
Manual Soldering	96.5Sn/3.0Ag/0.5Cu solid wire or no-clean rosin cored wire 370 °C (700 °F) max. for 3 seconds
Wave Soldering	96.5Sn/3.0Ag/0.5Cu solder with no-clean flux 260 °C (500 °F) max. for 5 seconds
Wash processes	Not recommended
Marking	Manufacturer's name and trademark, part number, and date code.
Hardware	One lockwasher and one mounting nut are shipped with each encoder, except where noted in the part number.
Packaging	45 pcs./tray

*High probability of missing quadrature codes with maximum bounce.

*RoHS Directive 2002/95/EC Jan. 27, 2003 including annex and RoHS Recast 2011/65/EU June 8, 2011.
Specifications are subject to change without notice.
Customers should verify actual device performance in their specific applications.

EAW - Absolute Contacting Encoder (ACE™)

BOURNS®

Pin Output Code For ACE-128

Bit/Pin correlation: b7 b6 b5 b4 b3 b2 b1 b0 = p8 p7 p6 p5 p4 p3 p2 p1
A binary "1" denotes an "open" switch and a binary "0" denotes a "closed" switch.
Positions 0-127 are seen by a clockwise rotation of the shaft.

Position	p8	p7	p6	p5	p4	p3	p2	p1	Decimal Output
0	0	1	1	1	1	1	1	1	127
1	0	0	1	1	1	1	1	1	63
2	0	0	1	1	1	1	1	0	62
3	0	0	1	1	1	0	1	0	58
4	0	0	1	1	1	0	0	0	56
5	1	0	1	1	1	0	0	0	184
6	1	0	0	1	1	0	0	0	152
7	0	0	0	1	1	0	0	0	24
8	0	0	0	0	1	0	0	0	8
9	0	1	0	0	1	0	0	0	72
10	0	1	0	0	1	0	0	1	73
11	0	1	0	0	1	1	0	1	77
12	0	1	0	0	1	1	1	1	79
13	0	0	0	0	1	1	1	1	15
14	0	0	1	0	1	1	1	1	47
15	1	0	1	0	1	1	1	1	175
16	1	0	1	1	1	1	1	1	191
17	1	0	0	1	1	1	1	1	159
18	0	0	0	1	1	1	1	1	31
19	0	0	0	1	1	1	0	1	29
20	0	0	0	1	1	1	0	0	28
21	0	1	0	1	1	1	0	0	92
22	0	1	0	0	1	1	0	0	76
23	0	0	0	0	1	1	0	0	12
24	0	0	0	0	0	1	0	0	4
25	0	0	1	0	0	1	0	0	36
26	1	0	1	0	0	1	0	0	164
27	1	0	1	0	0	1	1	0	166
28	1	0	1	0	0	1	1	1	167
29	1	0	0	0	0	1	1	1	135
30	1	0	0	1	0	1	1	1	151
31	1	1	0	1	0	1	1	1	215
32	1	1	0	1	1	1	1	1	223
33	1	1	0	0	1	1	1	1	207
34	1	0	0	0	1	1	1	1	143
35	1	0	0	0	1	1	1	0	142
36	0	0	0	0	1	1	1	0	14
37	0	0	1	0	1	1	1	0	46
38	0	0	1	0	0	1	1	0	38
39	0	0	0	0	0	1	1	0	6
40	0	0	0	0	0	0	1	0	2
41	0	0	0	1	0	0	1	0	18
42	0	1	0	1	0	0	1	0	82
43	0	1	0	1	0	0	1	1	83
44	1	1	0	1	0	0	1	1	211
45	1	1	0	0	0	0	1	1	195
46	1	1	0	0	1	0	1	1	203
47	1	1	1	0	1	0	1	1	235
48	1	1	1	0	1	1	1	1	239
49	1	1	1	0	0	1	1	1	231
50	1	1	0	0	0	1	1	1	199
51	0	1	0	0	0	1	1	1	71
52	0	0	0	0	0	1	1	1	7
53	0	0	0	1	0	1	1	1	23
54	0	0	0	1	0	0	1	1	19
55	0	0	0	0	0	0	1	1	3
56	0	0	0	0	0	0	0	1	1
57	0	0	0	0	1	0	0	1	9
58	0	0	1	0	1	0	0	1	41
59	1	0	1	0	1	0	0	1	169
60	1	1	1	0	1	0	0	1	233
61	1	1	1	0	0	0	0	1	225
62	1	1	1	0	0	1	0	1	229
63	1	1	1	1	0	1	0	1	245

Position	p8	p7	p6	p5	p4	p3	p2	p1	Decimal Output
64	1	1	1	1	0	1	1	1	247
65	1	1	1	1	0	0	1	1	243
66	1	1	1	0	0	0	1	1	227
67	1	0	1	0	0	0	1	1	163
68	1	0	0	0	0	0	1	1	131
69	1	0	0	0	1	0	1	1	139
70	1	0	0	0	1	0	0	1	137
71	1	0	0	0	0	0	0	1	129
72	1	0	0	0	0	0	0	0	128
73	1	0	0	0	0	1	0	0	132
74	1	0	0	1	0	1	0	0	148
75	1	1	0	1	0	1	0	0	212
76	1	1	1	1	0	1	0	0	244
77	1	1	1	1	0	0	0	0	240
78	1	1	1	1	0	0	1	0	242
79	1	1	1	1	1	0	1	0	250
80	1	1	1	1	1	0	1	1	251
81	1	1	1	1	1	0	0	1	249
82	1	1	1	1	0	0	0	1	241
83	1	1	0	1	0	0	0	1	209
84	1	1	0	0	0	0	0	1	193
85	1	1	0	0	0	1	0	1	197
86	1	1	0	0	0	1	0	0	196
87	1	1	0	0	0	0	0	0	192
88	0	1	0	0	0	0	0	0	64
89	0	1	0	0	0	0	1	0	66
90	0	1	0	0	1	0	1	0	74
91	0	1	1	0	1	0	1	0	106
92	0	1	1	1	1	0	1	0	122
93	0	1	1	1	1	0	0	0	120
94	0	1	1	1	1	0	0	1	121
95	0	1	1	1	1	1	0	1	125
96	1	1	1	1	1	1	0	1	253
97	1	1	1	1	1	1	0	0	252
98	1	1	1	1	1	0	0	0	248
99	1	1	1	0	1	0	0	0	232
100	1	1	1	0	0	0	0	0	224
101	1	1	1	0	0	0	1	0	226
102	0	1	1	0	0	0	1	0	98
103	0	1	1	0	0	0	0	0	96
104	0	0	1	0	0	0	0	0	32
105	0	0	1	0	0	0	0	1	33
106	0	0	1	0	0	1	0	1	37
107	0	0	1	1	0	1	0	1	53
108	0	0	1	1	1	1	0	1	61
109	0	0	1	1	1	1	0	0	60
110	1	0	1	1	1	1	0	0	188
111	1	0	1	1	1	1	1	0	190
112	1	1	1	1	1	1	1	0	254
113	0	1	1	1	1	1	1	0	126
114	0	1	1	1	1	1	0	0	124
115	0	1	1	1	0	1	0	0	116
116	0	1	1	1	0	0	0	0	112
117	0	1	1	1	0	0	0	1	113
118	0	0	1	1	0	0	0	1	49
119	0	0	1	1	0	0	0	0	48
120	0	0	0	1	0	0	0	0	16
121	1	0	0	1	0	0	0	0	144
122	1	0	0	1	0	0	1	0	146
123	1	0	0	1	1	0	1	0	154
124	1	0	0	1	1	1	1	0	158
125	0	0	0	1	1	1	1	0	30
126	0	1	0	1	1	1	1	0	94
127	0	1	0	1	1	1	1	1	95

Appendix C MLX90614 Infra Red Thermometer - Datasheet

Melexis
Microelectronic Integrated Systems

MLX90614 family

Single and Dual Zone
Infra Red Thermometer in TO-39

Features and Benefits

- Small size, low cost
- Easy to integrate
- Factory calibrated in wide temperature range:
  -40…+125°C for sensor temperature and
  -70…+380°C for object temperature.
- High accuracy of 0.5°C over wide temperature range (0…+50°C for both Ta and To)
- High (medical) accuracy calibration
- Measurement resolution of 0.02°C
- Single and dual zone versions
- SMBus compatible digital interface
- Customizable PWM output for continuous reading
- Available in 3V and 5V versions
- Simple adaptation for 8…16V applications
- Sleep mode for reduced power consumption
- Different package options for applications and measurements versatility
- Automotive grade

Applications Examples

- High precision non-contact temperature measurements
- Thermal Comfort sensor for Mobile Air Conditioning control system
- Temperature sensing element for residential, commercial and industrial building air conditioning
- Windshield defogging
- Automotive blind angle detection
- Industrial temperature control of moving parts
- Temperature control in printers and copiers
- Home appliances with temperature control
- Healthcare
- Livestock monitoring
- Movement detection
- Multiple zone temperature control – up to 127 sensors can be read via common 2 wires
- Thermal relay / alert
- Body temperature measurement

Ordering Information

Part No.	Temperature Code	Package Code	- Option Code - X X X (1) (2) (3)	Standard part	Packing form
MLX90614	E (-40°C...85°C) K (-40°C…125°C)	SF (TO-39)		-000	-TU

(1) Supply Voltage/ Accuracy
A - 5V
B - 3V
C - Reserved
D - 3V medical accuracy

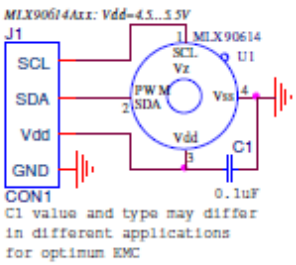
(2) Number of thermopiles:
A – single zone
B – dual zone
C – gradient compensated*

(3) Package options:
A – Standard package
B – Reserved
C – 35° FOV
D/E – Reserved
F – 10° FOV
G – Reserved
H – 12° FOV (refractive lens)
I – 5° FOV

Example:
MLX90614ESF-BAA-000-TU

*** : See page 2**

1 Functional diagram



MLX90614 connection to SMBus

Figure 1: Typical application schematics

2 General Description

The MLX90614 is an Infra Red thermometer for non contact temperature measurements. Both the IR sensitive thermopile detector chip and the signal conditioning ASSP are integrated in the same TO-39 can.

Thanks to its low noise amplifier, 17-bit ADC and powerful DSP unit, a high accuracy and resolution of the thermometer is achieved.

The thermometer comes factory calibrated with a digital PWM and SMBus (System Management Bus) output.

As a standard, the 10-bit PWM is configured to continuously transmit the measured temperature in range of -20…120°C, with an output resolution of 0.14°C.
The factory default POR setting is SMBus.

8.5.1 Single PWM format

In single PWM output mode the settings for PWM1 data only are used. The temperature reading can be calculated from the signal timing as:

$$T_{OUT} = \left(\frac{2t_2}{T} \times \left(T_{O\_MAX} - T_{O\_MIN}\right)\right) + T_{O\_MIN}$$

where Tmin and Tmax are the corresponding rescale coefficients in EEPROM for the selected temperature output (Ta, object temperature range is valid for both Tobj1 and Tobj2 as specified in the previous table) and T is the PWM period. Tout is T_{O1}, T_{O2} or T_a according to Config Register [5:4] settings.
The different time intervals $t_1 \ldots t_4$ have following meaning:

t_1: Start buffer. During this time the signal is always high. t_1 = 0.125s x T (where T is the PWM period, please refer to Figure 14).

t_2: Valid Data Output Band, 0…1/2T. PWM output data resolution is 10 bit.

t_3: Error band – information for fatal error in EEPROM (double error detected, not correctable).
t_3 = 0.25s x T. Therefore a PWM pulse train with a duty cycle of 0.875 will indicate a fatal error in EEPROM (for single PWM format). FE means Fatal Error.

Example:

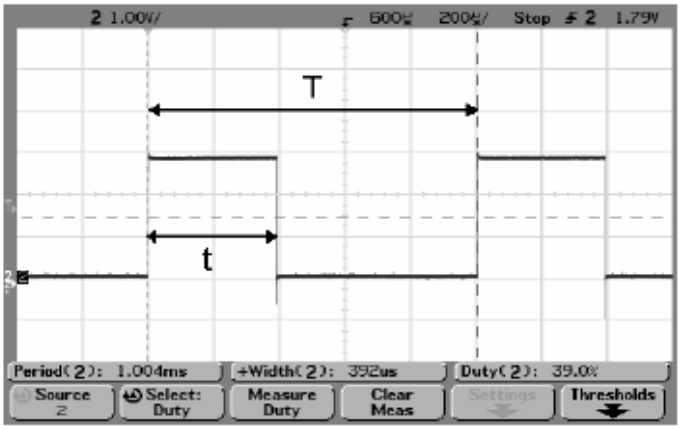


Figure 15: PWM example single mode

$$T_{O\_MIN} = 0°C \rightarrow T_{O\_MIN}(EEPROM, 0x01) = 100 \times \left(T_{O\_MIN} + 273.15\right) = 27315d = 0x6AB3$$

$$T_{O\_MAX} = 50°C \rightarrow T_{O\_MAX}(EEPROM, 0x00) = 100 \times \left(T_{O\_MAX} + 273.15\right) = 32315d = 0x7E3B$$

Captured PWM period is T = 1004µs
Captured high duration is t = 392 µs
Calculated duty cycle is:

$$D = \frac{t}{T} = \frac{392}{1004} = 0.3904 \text{ or } 39.04\%$$

The temperature is calculated as follows:

$$T_O = 2 \times (0.3904 - 0.125) \times (50 - 0) + 0 = 2 \times 0.2654 \times 50 = 26.54°C$$

Appendix D Atmega328 - Datasheet

28. Electrical Characteristics

28.1 Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on any Pin except $\overline{\text{RESET}}$ with respect to Ground	-0.5V to V_{CC}+0.5V
Voltage on $\overline{\text{RESET}}$ with respect to Ground	-0.5V to +13.0V
Maximum Operating Voltage	6.0V
DC Current per I/O Pin	40.0 mA
DC Current V_{CC} and GND Pins	200.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

28.2 DC Characteristics

T_A = -40°C to 85°C, V_{CC} = 1.8V to 5.5V (unless otherwise noted)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
V_{IL}	Input Low Voltage, except XTAL1 and $\overline{\text{RESET}}$ pin	V_{CC} = 1.8V - 2.4V V_{CC} = 2.4V - 5.5V	-0.5 -0.5		$0.2V_{CC}$(1) $0.3V_{CC}$(1)	V
V_{IH}	Input High Voltage, except XTAL1 and $\overline{\text{RESET}}$ pins	V_{CC} = 1.8V - 2.4V V_{CC} = 2.4V - 5.5V	$0.7V_{CC}$(2) $0.6V_{CC}$(2)		V_{CC} + 0.5 V_{CC} + 0.5	V
V_{IL1}	Input Low Voltage, XTAL1 pin	V_{CC} = 1.8V - 5.5V	-0.5		$0.1V_{CC}$(1)	V
V_{IH1}	Input High Voltage, XTAL1 pin	V_{CC} = 1.8V - 2.4V V_{CC} = 2.4V - 5.5V	$0.8V_{CC}$(2) $0.7V_{CC}$(2)		V_{CC} + 0.5 V_{CC} + 0.5	V
V_{IL2}	Input Low Voltage, $\overline{\text{RESET}}$ pin	V_{CC} = 1.8V - 5.5V	-0.5		$0.1V_{CC}$(1)	V
V_{IH2}	Input High Voltage, $\overline{\text{RESET}}$ pin	V_{CC} = 1.8V - 5.5V	$0.9V_{CC}$(2)		V_{CC} + 0.5	V
V_{IL3}	Input Low Voltage, $\overline{\text{RESET}}$ pin as I/O	V_{CC} = 1.8V - 2.4V V_{CC} = 2.4V - 5.5V	-0.5 -0.5		$0.2V_{CC}$(1) $0.3V_{CC}$(1)	V
V_{IH3}	Input High Voltage, $\overline{\text{RESET}}$ pin as I/O	V_{CC} = 1.8V - 2.4V V_{CC} = 2.4V - 5.5V	$0.7V_{CC}$(2) $0.6V_{CC}$(2)		V_{CC} + 0.5 V_{CC} + 0.5	V
V_{OL}	Output Low Voltage(3) except RESET pin	I_{OL} = 20 mA, V_{CC} = 5V I_{OL} = 10 mA, V_{CC} = 3V			0.9 0.6	V
V_{OH}	Output High Voltage(4) except Reset pin	I_{OH} = -20 mA, V_{CC} = 5V I_{OH} = -10 mA, V_{CC} = 3V	4.2 2.3			V
I_{IL}	Input Leakage Current I/O Pin	V_{CC} = 5.5V, pin low (absolute value)			1	µA
I_{IH}	Input Leakage Current I/O Pin	V_{CC} = 5.5V, pin high (absolute value)			1	µA

28.5 System and Reset Characteristics

Table 28-3. Reset, Brown-out and Internal Voltage Characteristics(1)

Symbol	Parameter		Min	Typ	Max	Units
V_{POT}	Power-on Reset Threshold Voltage (rising)		1.1	1.4	1.6	V
	Power-on Reset Threshold Voltage (falling)(2)		0.6	1.3	1.6	V
SR_{ON}	Power-on Slope Rate		0.01		10	V/ms
V_{RST}	$\overline{\text{RESET}}$ Pin Threshold Voltage		0.2 V_{CC}		0.9 V_{CC}	V
t_{RST}	Minimum pulse width on $\overline{\text{RESET}}$ Pin				2.5	µs
V_{HYST}	Brown-out Detector Hysteresis			50		mV
t_{BOD}	Min Pulse Width on Brown-out Reset			2		µs
V_{BG}	Bandgap reference voltage	V_{CC}=2.7 T_A=25°C	1.0	1.1	1.2	V
t_{BG}	Bandgap reference start-up time	V_{CC}=2.7 T_A=25°C		40	70	µs
I_{BG}	Bandgap reference current consumption	V_{CC}=2.7 T_A=25°C		10		µA

Notes: 1. Values are guidelines only.
2. The Power-on Reset will not work unless the supply voltage has been below V_{POT} (falling)

Table 28-4. BODLEVEL Fuse Coding(1)

BODLEVEL 2:0 Fuses	Min V_{BOT}	Typ V_{BOT}	Max V_{BOT}	Units
111	BOD Disabled			
110	1.7	1.8	2.0	V
101	2.5	2.7	2.9	
100	4.1	4.3	4.5	
011	Reserved			
010				
001				
000				

Notes: 1. V_{BOT} may be below nominal minimum operating voltage for some devices. For devices where this is the case, the device is tested down to V_{CC} = V_{BOT} during the production test. This guarantees that a Brown-Out Reset will occur before V_{CC} drops to a voltage where correct operation of the microcontroller is no longer guaranteed. The test is performed using BODLEVEL = 110, 101 and 100.

Appendix E Atmega2560 - Datasheet

31. Electrical Characteristics

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on any Pin except $\overline{\text{RESET}}$ with respect to Ground	-0.5V to V_{CC}+0.5V
Voltage on $\overline{\text{RESET}}$ with respect to Ground	-0.5V to +13.0V
Maximum Operating Voltage	6.0V
DC Current per I/O Pin	40.0mA
DC Current V_{CC} and GND Pins	200.0mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

31.1 DC Characteristics

T_A = -40°C to 85°C, V_{CC} = 1.8V to 5.5V (unless otherwise noted)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
V_{IL}	Input Low Voltage, Except XTAL1 and Reset pin	V_{CC} = 1.8V - 2.4V V_{CC} = 2.4V - 5.5V	-0.5 -0.5		$0.2V_{CC}$(1) $0.3V_{CC}$(1)	V
V_{IL1}	Input Low Voltage, XTAL1 pin	V_{CC} = 1.8V - 5.5V	-0.5		$0.1V_{CC}$(1)	
V_{IL2}	Input Low Voltage, RESET pin	V_{CC} = 1.8V - 5.5V	-0.5		$0.1V_{CC}$(1)	
V_{IH}	Input High Voltage, Except XTAL1 and RESET pins	V_{CC} = 1.8V - 2.4V V_{CC} = 2.4V - 5.5V	$0.7V_{CC}$(2) $0.6V_{CC}$(2)		V_{CC} + 0.5 V_{CC} + 0.5	
V_{IH1}	Input High Voltage, XTAL1 pin	V_{CC} = 1.8V - 2.4V V_{CC} = 2.4V - 5.5V	$0.8V_{CC}$(2) $0.7V_{CC}$(2)		V_{CC} + 0.5 V_{CC} + 0.5	
V_{IH2}	Input High Voltage, RESET pin	V_{CC} = 1.8V - 5.5V	$0.9V_{CC}$(2)		V_{CC} + 0.5	
V_{OL}	Output Low Voltage(3), Except RESET pin	I_{OL} = 20mA, V_{CC} = 5V I_{OL} = 10mA, V_{CC} = 3V			0.9 0.6	
V_{OH}	Output High Voltage(4), Except RESET pin	I_{OH} = -20mA, V_{CC} = 5V I_{OH} = -10mA, V_{CC} = 3V	4.2 2.3			
I_{IL}	Input Leakage Current I/O Pin	V_{CC} = 5.5V, pin low (absolute value)			1	µA
I_{IH}	Input Leakage Current I/O Pin	V_{CC} = 5.5V, pin high (absolute value)			1	
R_{RST}	Reset Pull-up Resistor		30		60	kΩ
R_{PU}	I/O Pin Pull-up Resistor		20		50	

Appendix F VNH2SP30-E Motor Driver - Datasheet

ELECTRICAL CHARACTERISTICS (continued)

Table 10. PWM

Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit
V_{pwl}	PWM Low Level Voltage				1.25	V
I_{pwl}	PWM Pin Current	V_{pw}=1.25V	1			µA
V_{pwh}	PWM High Level Voltage		3.25			V
I_{pwh}	PWM Pin Current	V_{pw}=3.25V			10	µA
$V_{pwhhyst}$	PWM Hysteresis Voltage		0.5			V
V_{pwcl}	PWM Clamp Voltage	I_{pw} = 1 mA I_{pw} = -1 mA	V_{CC}+0.3 -6.0	V_{CC}+0.7 -4.5	V_{CC}+1.0 -3.0	V V
C_{INPWM}	PWM Pin Input Capacitance	V_{IN} =2.5V			25	pF

Table 11. Switching (V_{CC}=13V, R_{LOAD}=0.87Ω, unless otherwise specified)

Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit
f	PWM Frequency		0		20	kHz
$t_{d(on)}$	Turn-on Delay Time	Input rise time < 1µs (see fig. 8)			250	µs
$t_{d(off)}$	Turn-off Delay Time	Input rise time < 1µs (see fig. 8)			250	µs
t_r	Rise Time	(see fig. 7)		1	1.6	µs
t_f	Fall Time	(see fig. 7)		1.2	2.4	µs
t_{DEL}	Delay Time During Change of Operating Mode	(see fig. 6)	300	600	1800	µs
t_{rr}	High Side Free Wheeling Diode Reverse Recovery Time	(see fig. 9)		110		ns
$t_{off(min)}$ (1)	PWM Minimum off time	9V<V_{CC}<16V; T_j=25°C; L=250µH I_{OUT}=15A			6	µs

Note: 1. During PWM operation, to avoid false Short to Battery detection, the PWM signal must be low for a time longer than 6µs.

Table 12. Protection And Diagnostic

Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit
V_{USD}	Undervoltage Shut-down Undervoltage Reset			 4.7	5.5	V V
V_{OV}	Overvoltage Shut-down		16	19	22	V
I_{LIM}	High-Side Current Limitation		30	50	70	A
V_{CLP}	Total Clamp Voltage (V_{CC} to GND)	I_{OUT}=15A	43	48	54	V
T_{TSD}	Thermal Shut-down Temperature	V_{IN} = 3.25 V	150	175	200	°C
T_{TR}	Thermal Reset Temperature		135			°C
T_{HYST}	Thermal Hysteresis		7	15		°C

ELECTRICAL CHARACTERISTICS (continued)

Table 13. Current Sense (9V<V_{CC}<16V)

Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit
K_1	I_{OUT}/I_{SENSE}	I_{OUT}=30A; R_{SENSE}=1.5kΩ T_j= - 40 to 150°C	9665	11370	13075	
K_2	I_{OUT}/I_{SENSE}	I_{OUT}=8A; R_{SENSE}=1.5kΩ T_j= - 40 to 150°C	9096	11370	13644	
dK_1 / K_1 (*)	Analog sense current drift	I_{OUT}=30A; R_{SENSE}=1.5kΩ T_j= - 40 to 150°C	-8		+8	%
dK_2 / K_2 (*)	Analog sense current drift	I_{OUT} >8A; R_{SENSE}=1.5kΩ T_j= - 40 to 150°C	-10		+10	%
I_{SENSEO}	Analog Sense Leakage Current	I_{OUT}=0A; V_{SENSE}=0V; T_j= - 40 to 150°C	0		65	µA

Note:(*) Analog sense current drift is deviation of factor K for a given device over (-40°C to 150°C and 9V<V_{CC}<16V) with respect to it's value measured at T_j=25°C, V_{CC}=13V.

WAVEFORMS AND TRUTH TABLE

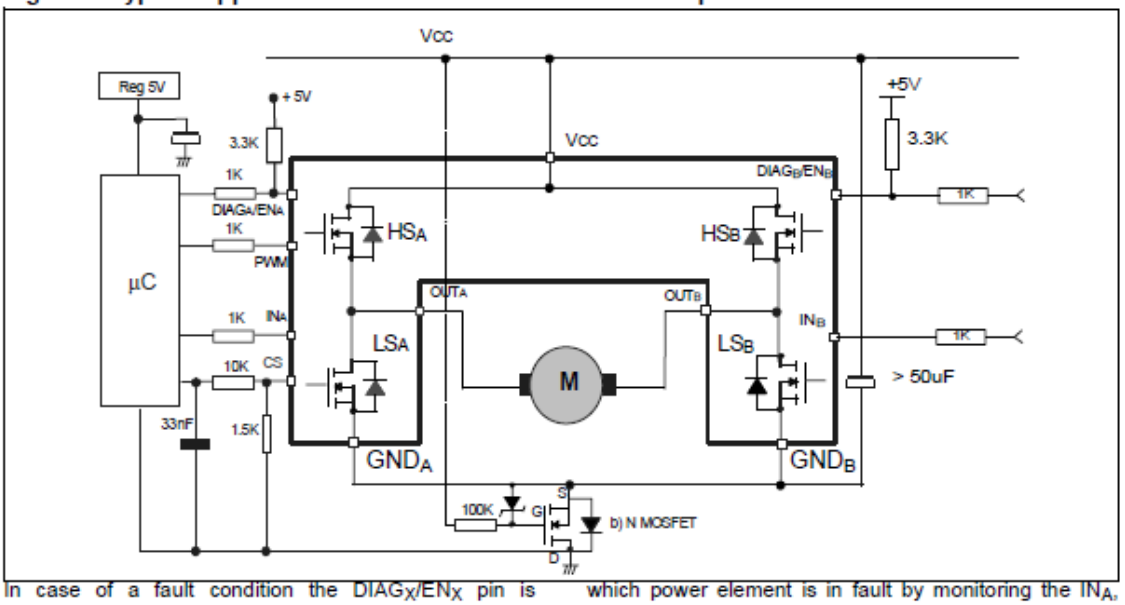
Table 14. Truth Table In Normal Operating Conditions

In normal operating conditions the DIAGX/ENX pin is considered as an input pin by the device. This pin must be externally pulled high.

PWM pin usage: in all cases, a "0" on the PWM pin will turn-off both LS_A and LS_B switches. When PWM rises back to "1", LS_A or LS_B turn on again depending on the input pin state.

IN_A	IN_B	$DIAG_A/EN_A$	$DIAG_B/EN_B$	OUT_A	OUT_B	CS	Operating mode
1	1	1	1	H	H	High Imp.	Brake to V_{CC}
1	0	1	1	H	L	$I_{SENSE}=I_{OUT}/K$	Clockwise (CW)
0	1	1	1	L	H	$I_{SENSE}=I_{OUT}/K$	Counterclockwise (CCW)
0	0	1	1	L	L	High Imp.	Brake to GND

Figure 5. Typical Application Circuit For Dc To 20khz PWM OperationShort Circuit Protection



In case of a fault condition the $DIAG_X/EN_X$ pin is considered as an output pin by the device.
The fault conditions are:
- overtemperature on one or both high sides (for example if a short to ground occurs as it could be the case described in line 1 and 2 in the table below);
- short to battery condition on the output (saturation detection on the Low-Side Power MOSFET).

Possible origins of fault conditions may be:
OUT_A is shorted to ground ---> overtemperature detection on high side A.
OUT_A is shorted to V_{CC} ---> Low-Side Power MOSFET saturation detection.
When a fault condition is detected, the user can know which power element is in fault by monitoring the IN_A, IN_B, $DIAG_A/EN_A$ and $DIAG_B/EN_B$ pins.
In any case, when a fault is detected, the faulty leg of the bridge is latched off. To turn-on the respective output (OUT_X) again, the input signal must rise from low to high level.

Table 15. Truth Table In Fault Conditions (Detected On OUT_A)

IN_A	IN_B	$DIAG_A/EN_A$	$DIAG_B/EN_B$	OUT_A	OUT_B	CS
1	1	0	1	OPEN	H	High Imp.
1	0	0	1	OPEN	L	High Imp.
0	1	0	1	OPEN	H	I_{OUTB}/K
0	0	0	1	OPEN	L	High Imp.
X	X	0	0	OPEN	OPEN	High Imp.
X	1	0	1	OPEN	H	I_{OUTB}/K
X	0	0	1	OPEN	L	High Imp.
		Fault Information		Protection Action		

Note: 2. Notice that saturation detection on the low side power Mosfet is possible only if the impedance of the short-circuit from the output to the battery is less than 100mohm when the device is supplied with a battery voltage of 13.5V.

Appendix G ATF16V8B Programmable Logic Device - Datasheet

Features

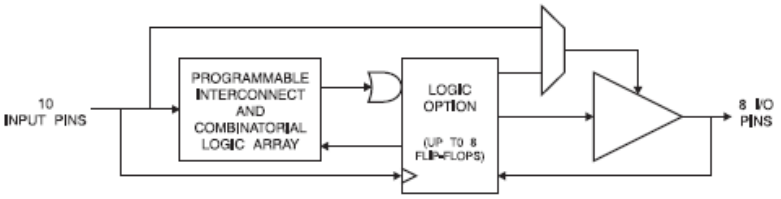
- Industry Standard Architecture
  - Emulates Many 20-Pin PALs®
  - Low Cost Easy-to-Use Software Tools
- High Speed Electrically Erasable Programmable Logic Devices
  - 7.5 ns Maximum Pin-to-Pin Delay
- Several Power Saving Options

Device	I_{CC}, Stand-By	I_{CC}, Active
ATF16V8B	50 mA	55 mA
ATF16V8BQ	35 mA	40 mA
ATF16V8BQL	5 mA	20 mA

- CMOS and TTL Compatible Inputs and Outputs
  - Input and I/O Pull-Up Resistors
- Advanced Flash Technology
  - Reprogrammable
  - 100% Tested
- High Reliability CMOS Process
  - 20 Year Data Retention
  - 100 Erase/Write Cycles
  - 2,000V ESD Protection
  - 200 mA Latchup Immunity
- Commercial, and Industrial Temperature Ranges
- Dual-in-Line and Surface Mount Packages in Standard Pinouts

High Performance Flash PLD

Block Diagram



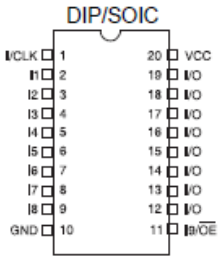
Description

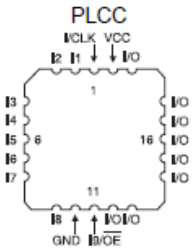
The ATF16V8B is a high performance CMOS (Electrically Erasable) Programmable Logic Device (PLD) which utilizes Atmel's proven electrically erasable Flash memory technology. Speeds down to 7.5 ns are offered. All speed ranges are specified over the full 5V ± 10% range for industrial temperature ranges, and 5V ± 5% for commercial temperature ranges.

(continued)

Pin Configurations

Pin Name	Function
CLK	Clock
I	Logic Inputs
I/O	Bidirectional Buffers
$\overline{OE}$	Output Enable
VCC	+5V Supply





Top view

0364C

Description (Continued)

Several low power options allow selection of the best solution for various types of power-limited applications. Each of these options significantly reduces total system power and enhances system reliability.

The ATF16V8Bs incorporate a superset of the generic architectures, which allows direct replacement of the 16R8 family and most 20-pin combinatorial PLDs. Eight outputs are each allocated eight product terms. Three different modes of operation, configured automatically with software, allow highly complex logic functions to be realized.

Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0V to +7.0V (1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0V to +14.0V (1)
Programming Voltage with Respect to Ground	-2.0V to +14.0V (1)

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6V dc, which may undershoot to -2.0V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC} + 0.75V dc, which may overshoot to 7.0V for pulses of less than 20 ns.

DC and AC Operating Conditions

	Commercial	Industrial
Operating Temperature (Case)	0°C - 70°C	-40°C - 85°C
V_{CC} Power Supply	5V ± 5%	5V ± 10%

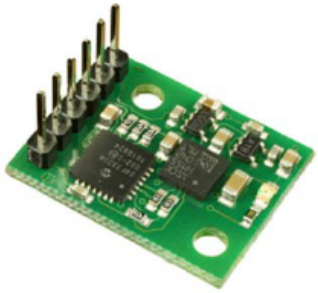
DC Characteristics

Symbol	Parameter	Condition			Min	Typ	Max	Units
I_{IL}	Input or I/O Low Leakage Current	$0 \le V_{IN} \le V_{IL}$ (MAX)				-35	-100	µA
I_{IH}	Input or I/O High Leakage Current	$3.5 \le V_{IN} \le V_{CC}$					10	µA
I_{CC}	Power Supply Current, Standby	V_{CC} = MAX, V_{IN} = MAX, Outputs Open	B-7, -10	Com.		55	85	mA
				Ind.		55	95	mA
			B-15, -25	Com.		50	75	mA
				Ind.		50	80	mA
			BQ-10	Com.		35	55	mA
			BQL-15, -25	Com.		5	10	mA
				Ind.		5	15	mA
I_{CC2}	Clocked Power Supply Current	V_{CC} = MAX, Outputs Open	BQL-15, -25	Com.		1		mA/MHz(2)
				Ind.		1		mA/MHz(2)
I_{CC3}	Clocked Power Supply Current	V_{CC} = MAX, Outputs Open, f = 15 MHz	B-7, -10	Com.		60	90	mA
				Ind.		60	100	mA
			B-15, -25	Com.		55	85	mA
				Ind.		55	95	mA
			BQ-10	Com.		40	55	mA
			BQL-15, -25	Com.		20	35	mA
				Ind.		20	40	mA
I_{OS}(1)	Output Short Circuit Current	V_{OUT} = 0.5V					-130	mA
V_{IL}	Input Low Voltage				-0.5		0.8	V
V_{IH}	Input High Voltage				2.0		V_{CC} + 0.75	V
V_{OL}	Output High Voltage	$V_{IN} = V_{IH}$ or V_{IL}, V_{CC} = MIN	I_{OL} = -24 mA Com., Ind.				0.5	V
V_{OH}	Output High Voltage	$V_{IN} = V_{IH}$ or V_{IL}, V_{CC} = MIN	I_{OH} = -4.0 mA		2.4			V

Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.
2. Low frequency only. See Supply Current versus Input Frequency curves.

Appendix H CMPS10, Tilt Compensated Compass Module - Datasheet

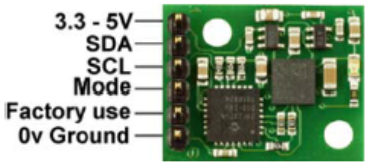
CMPS10 - Tilt Compensated Compass Module



Introduction
The CMPS10 module is a tilt compensated compass. Employing a 3-axis magnetometer and a 3-axis accelerometer and a powerful 16-bit processor, the CMPS10 has been designed to remove the errors caused by tilting of the PCB. The CMPS10 produces a result of 0-3599 representing 0-359.9 or 0 to 255. The output of the three sensors measuring x, y and z components of the magnetic field, together with the pitch and roll are used to calculate the bearing, each of these components are also made available in there raw form. We have also written examples of using the CMPS10 module with a wide range of popular controllers. The CMPS10 module requires a power supply at 3.3 - 5v and draws a nominal 25mA of current. There are three ways of getting the bearing from the module. A serial interface, an I2C interface or a PWM output.

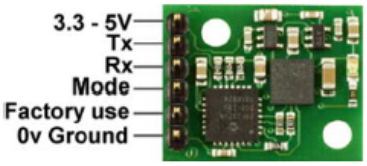
Mode selection
For data on each mode please click the mode heading. Note the CMPS10 looks at the mode selection pins at power-up only.

I2C mode



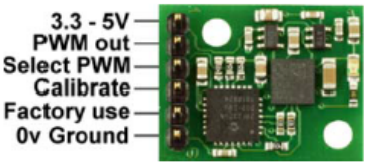
To enter the I2C mode of operation leave the mode pin unconnected

Serial mode



To enter the serial mode of operation connect the mode pin to ground

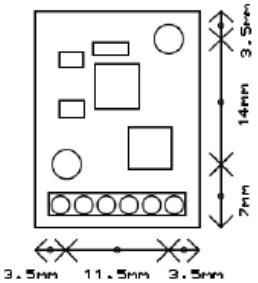
PWM mode



To enter the PWM mode of operation connect the select PWM pin to ground

Data update frequency
Updates of the tilt compensated heading occur at 75hz with the data is filtered by means of a 45 sample buffer, this means a complete refresh of the buffer is achieved every 640ms. Raw data from the magnetometer and accelerometer is available every 13.3ms.

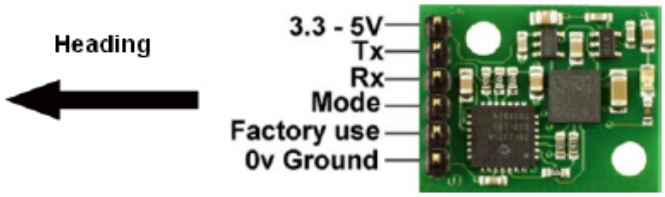
PCB Drilling Plan
The following diagram shows the CMPS10 PCB mounting hole positions.



We have examples of using the Compass module with a wide range of popular controllers.

Serial mode

Connections



To use the serial mode of operation the mode pin must be connected to ground.

Communication settings
The Serial mode operates over a link with a default baud rate of 9600 bps (no parity, 2 stop bits) and 3.3v-5v signal levels. This is not RS232. Do not connect RS232 to the module, the high RS232 voltages will irreversibly damage the module.

Commands
Below is a table describing commands that can be sent to the CMPS10 and the data it will respond with.

Commands for Serial

Command	Name	Bytes returned	Returned data description
0x11	GET VERSION	1	Software version
0x12	GET ANGLE 8 BIT	1	Angle as a single byte 0-255
0x13	GET ANGLE 16 BIT	2	Angle as two bytes, high byte first 0-3600
0x14	GET PITCH	1	Pitch angle +/- 0-85°
0x15	GET ROLL	1	Roll angle +/- 0-85°
0x21	GET MAG RAW	6	Raw magnetic data, 16 bit signed: X high, X low, Y high, Y low, Z high, Z low
0x22	GET ACCEL RAW	6	Raw accelerometer data, 16 bit signed: X high, X low, Y high, Y low, Z high, Z low
0x23	GET ALL	4	angle high, angle low (0-3600), pitch (+/- 0-85), roll (+/- 0-85)
0x31	CALIBRATE EN1	1	returns ok (0x55)
0x45	CALIBRATE EN2	1	returns ok (0x55)
0x5A	CALIBRATE EN3	1	returns ok (0x55)
0x5E	CALIBRATE	1	returns ok (0x55)
0x6A	RESTORE 1	1	returns ok (0x55)
0x7C	RESTORE 2	1	returns ok (0x55)
0x81	RESTORE 3	1	returns ok (0x55)
0xA0	BAUD 19200	1	returns ok (0x55)
0xA1	BAUD 38400	1	returns ok (0x55)

Calibration the CMPS10
I would recommend evaluating the CMPS10 performance first before implementing this function. Its purpose is to remove offsets caused by constant magnetic sources around the CMPS10. First of all you need to determine North and align the CMPS10 with it, then write a sequence of 3 commands in the correct order with a small delay between bytes, 100ms will be more than adequate. The sequence to enter calibration mode is 0x31,0x45,0x5A, then calibrate the first point by sending 0x5E to the command register, this should also light the LED. The Compass should then be rotated 90° and 0x5E sent to the command register again, repeat for two further 90° rotations and the calibration completes and the LED turns off. Please make sure that the CMPS10 is not located near to ferrous objects as this will distort the magnetic field and induce errors in the reading.

Restore of factory calibration of the CMPS10
To perform a restore of the factory calibration write a sequence of 3 commands in the correct order with a small delay between bytes, 100ms will be more than adequate. The sequence is 0x6A,0x7C,0x81.

Changing the baud rate
While the CMPS10 operates at a default serial bus baud rate of 9600 you may wish to change this. There are two other baud rates that can be used, for 19200 just send 0xA0 or alternatively for 38400 send 0xA1. Please note that the CMPS10 will always default to its 9600kbps rate after power cycling and after setting a new baud rate the ok response (0x55) will be sent at the newly selected speed.

Appendix I 1N4148 Diode - Datasheet

High-speed diodes 1N4148; 1N4448

FEATURES

- Hermetically sealed leaded glass SOD27 (DO-35) package
- High switching speed: max. 4 ns
- General application
- Continuous reverse voltage: max. 75 V
- Repetitive peak reverse voltage: max. 75 V
- Repetitive peak forward current: max. 450 mA.

APPLICATIONS

- High-speed switching.

DESCRIPTION

The 1N4148 and 1N4448 are high-speed switching diodes fabricated in planar technology, and encapsulated in hermetically sealed leaded glass SOD27 (DO-35) packages.

The diodes are type branded.

Fig.1 Simplified outline (SOD27; DO-35) and symbol.

LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{RRM}	repetitive peak reverse voltage		–	75	V
V_R	continuous reverse voltage		–	75	V
I_F	continuous forward current	see Fig.2; note 1	–	200	mA
I_{FRM}	repetitive peak forward current		–	450	mA
I_{FSM}	non-repetitive peak forward current	square wave; T_j = 25 °C prior to surge; see Fig.4			
		t = 1 µs	–	4	A
		t = 1 ms	–	1	A
		t = 1 s	–	0.5	A
P_{tot}	total power dissipation	T_{amb} = 25 °C; note 1	–	500	mW
T_{stg}	storage temperature		−65	+200	°C
T_j	junction temperature		–	200	°C

Note

1. Device mounted on an FR4 printed circuit-board; lead length 10 mm.

Appendix J TRX-300 Manual

The following pages were taken from the operating manual for a 1990 TRX-300 Fourtrax quadbike.

Throttle Lever

The throttle lever (Fig. 2-11) is next to the right handgrip, and it is operated by the thumb. Pressing the lever opens the throttle. When pressure is released, spring tension automatically closes the throttle.

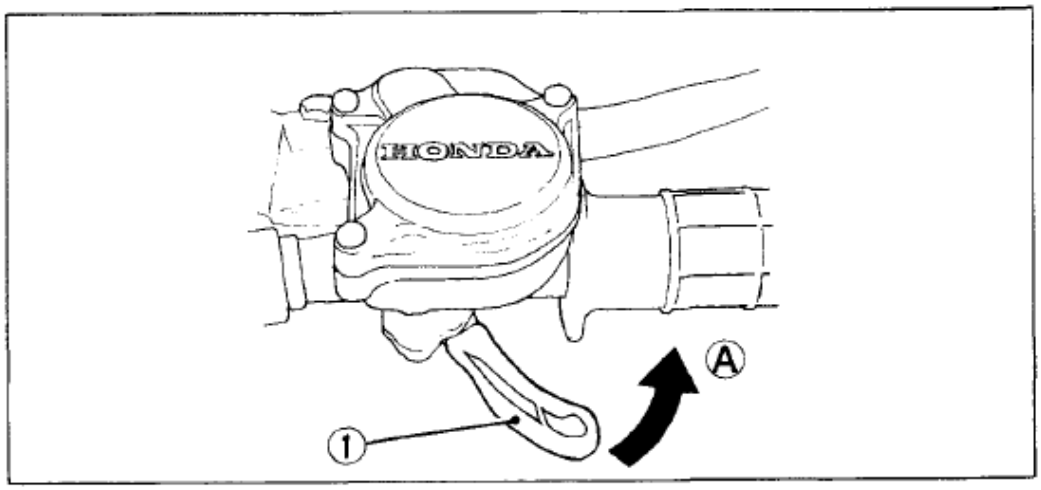


Fig. 2-11

(1) Throttle lever
(A) To open the throttle

If the oil high temperature indicator lamp comes on while you are riding, immediately bring the vehicle to a stop, turn the engine off and let it cool.

CAUTION:

* **Failure to stop the engine immediately after the oil high temperature indicator lamp comes on will adversely affect the service life of the engine.**

* **Do not carry or store articles on the front bumper. This may restrict air flow through the engine, causing it to overheat.**

If the oil high temperature indicator lamp comes on often, have your FOURTRAX inspected by your Honda dealer. To provide more air flow under adverse conditions, a heavy duty cooling fan is available as an optional part.

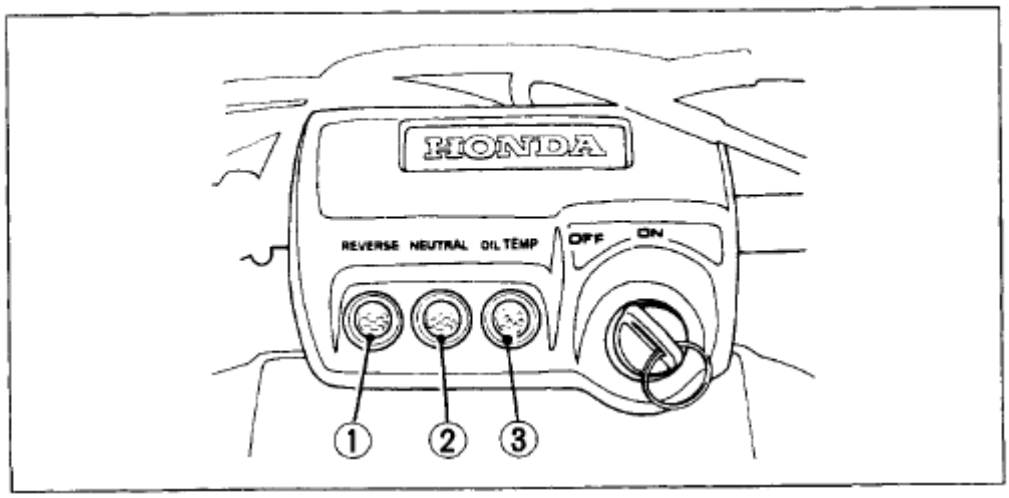


Fig. 2-13

(1) Reverse indicator lamp
(2) Neutral indicator lamp
(3) Oil high temperature indicator lamp

GEARSHIFT CONTROLS

Gearshift Pedal

The gearshift pedal is near the left footpeg. One full stroke of the pedal shifts the transmission to the next higher or lower gear in the shifting sequence (Fig. 2-14). The pedal automatically returns to the horizontal position when released.

Your FOURTRAX has five forward gears (SL, 1, 2, 3 and 4). To upshift to a higher gear, put the toe of your boot under the gearshift pedal and raise the pedal one full stroke. To downshift, step on the gearshift pedal and depress the pedal one full stroke. (For further operational instructions, see pages 59—60.)

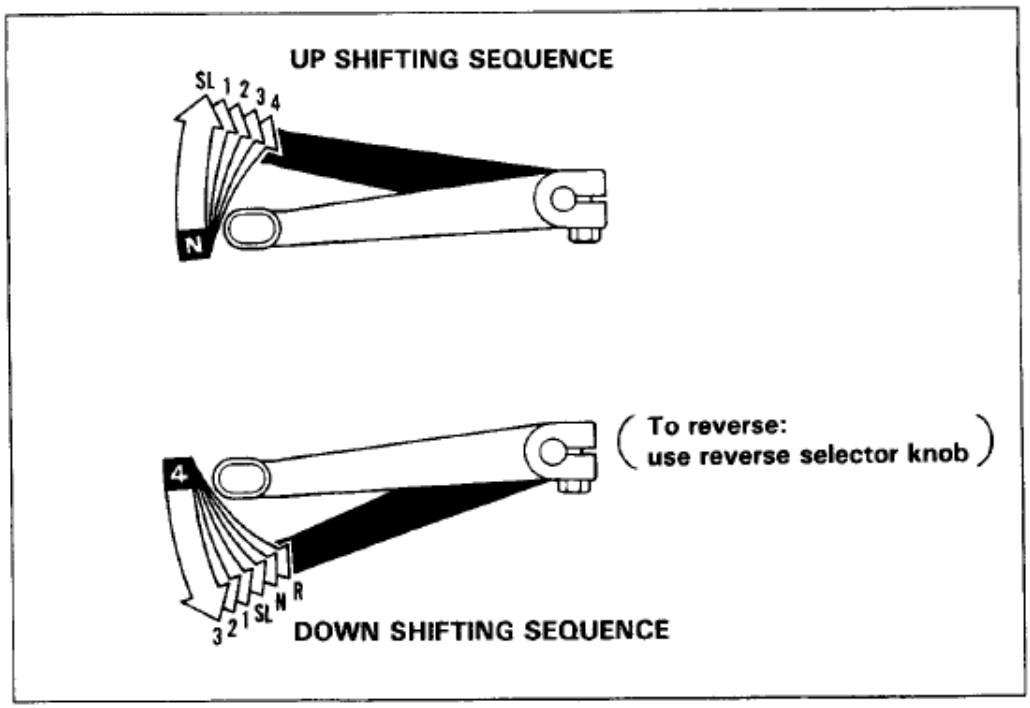


Fig. 2-14 Gearshifting sequence

Reverse Selector Knob

The reverse selector knob (Fig. 2-15) is on the rear brake lever.

To shift into reverse, first bring the FOURTRAX to a complete stop and make sure the transmission is in neutral. While pushing the reverse selector knob (1) in, squeeze the rear brake lever (2), then depress the gearshift pedal.

CAUTION:
Bring the FOURTRAX to a complete stop before shifting the transmission into reverse. If the transmission is shifted into reverse while the vehicle is moving, the transmission will be damaged.

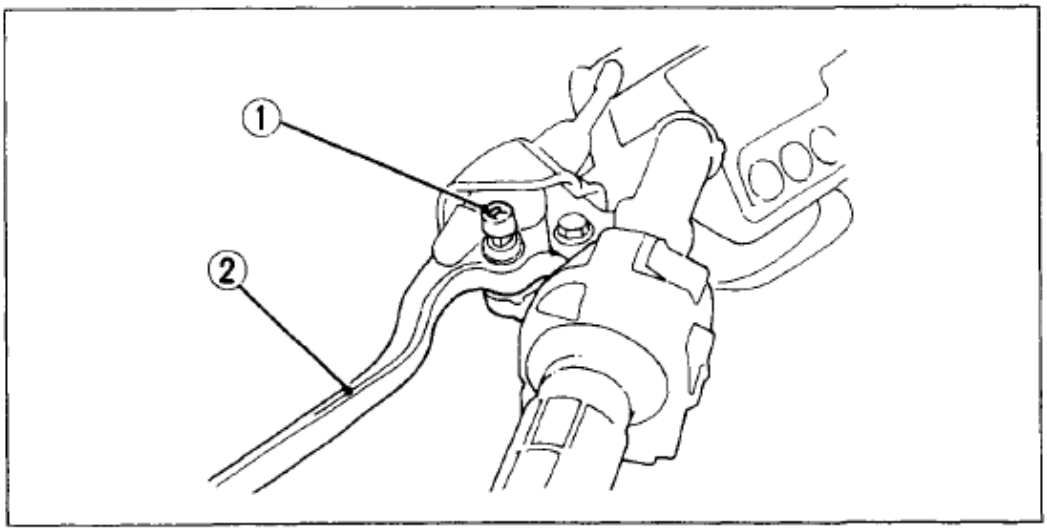


Fig. 2-15

(1) Reverse selector knob
(2) Rear brake lever

BRAKE CONTROLS

Front Brakes

The front brakes are operated by squeezing the front brake lever, near the right handgrip.

Rear Brake

The rear brake is operated either by depressing the brake pedal or by squeezing the rear brake lever (Fig. 2-16).

Parking Brake

The rear brake lever has a lock which allows it to be used as a parking brake.

To set the parking brake, squeeze the rear brake lever (1) and lock it with the lock lever (2). Always set the parking brake when parking and before starting the engine.
To unlock the parking brake, squeeze the rear brake lever (1) until the lock lever releases.

NOTE:
Using the parking brake in freezing weather may cause the brakes to freeze in the locked position.

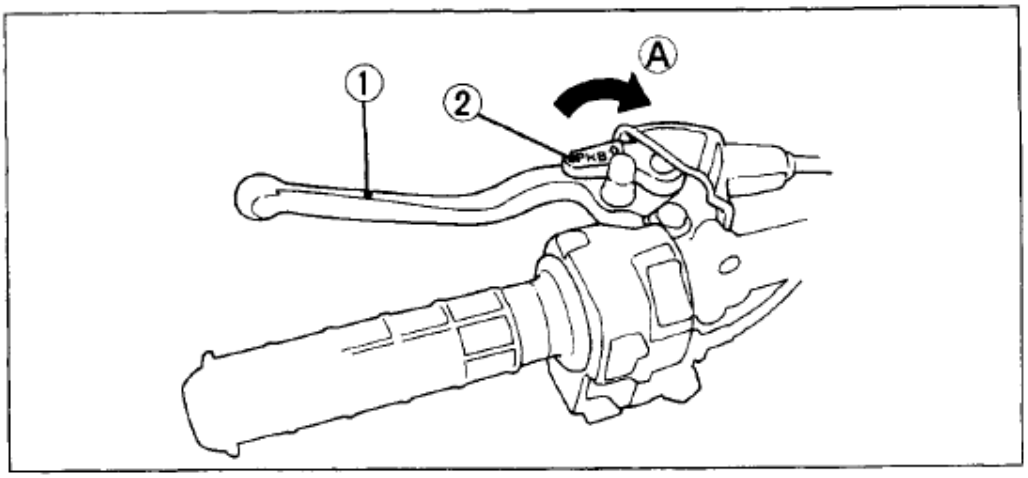


Fig. 2-16 (1) Rear brake lever/Parking brake (2) Lock lever (A) To lock

BRAKES

FRONT BRAKES

This FOURTRAX has hydraulic front drum brakes on both front wheels. Each day before riding you should check the brake lever free play and inspect the system for fluid leaks. The brake fluid level should be checked frequently, and the brake shoe linings should be inspected periodically.

To check lever free play and shoe lining clearance:

1. Measure the distance the brake lever (Fig. 4-21) moves before the brakes start to take hold. Free play, measured at the tip of the front brake lever (1) should be within 1—1-1/4 in (25—30 mm).

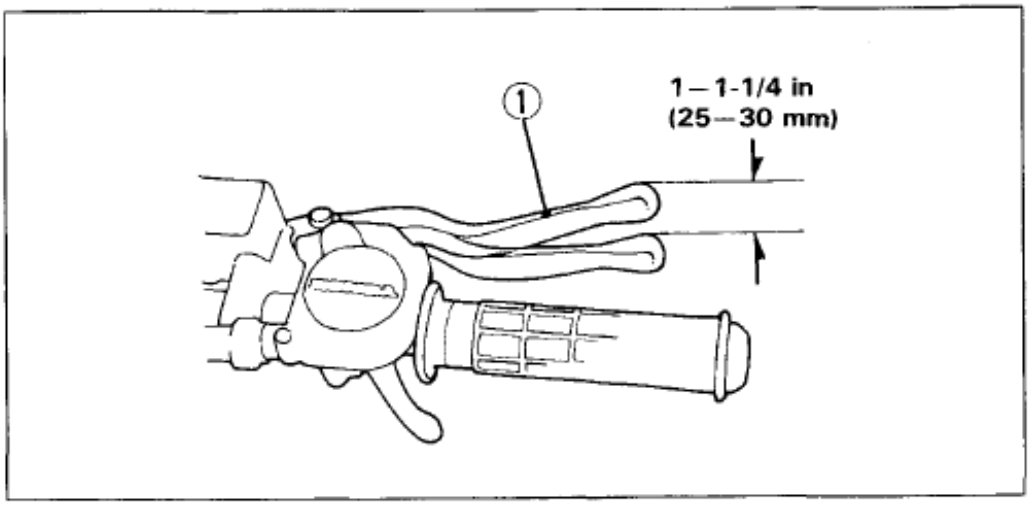


Fig. 4-21 (1) Front brake lever

REAR BRAKE

This FOURTRAX has a single mechanical drum brake on the rear axle housing. Each day before riding you should check the operation of both the rear brake pedal and the rear brake lever. You should also periodically check the brake shoes for wear.

To check and adjust brake pedal free play:

Measure the distance the rear brake pedal moves before the brake starts to take hold. Free play, measured at the end of the pedal (Fig. 4-25), should be 5/8—3/4 in (15—20 mm).

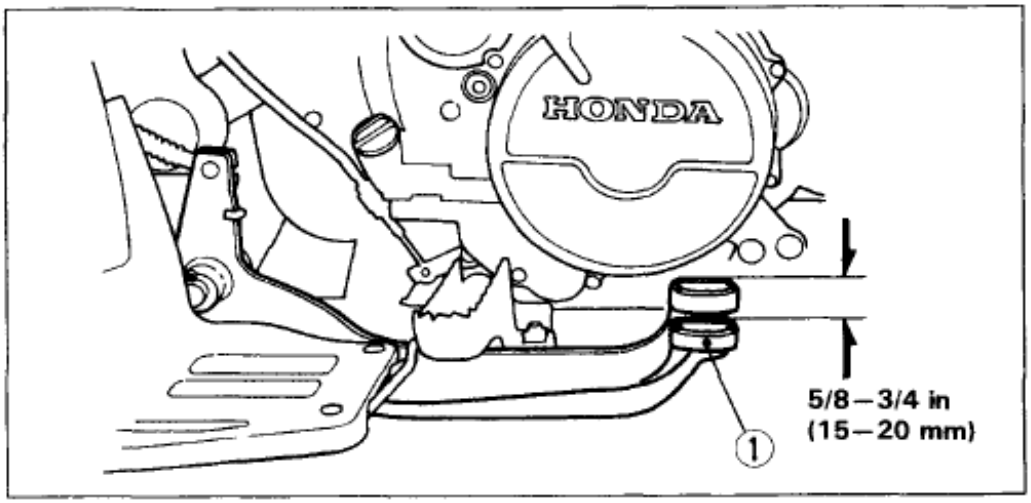


Fig. 4-25 (1) Rear brake pedal

To adjust the free play, turn the brake pedal adjusting nut (Fig. 4-26), located on the brake operating rod at the rear of the frame.

NOTE:
Make sure the cut-out on the adjusting nut is properly seated on the brake arm pin (3).

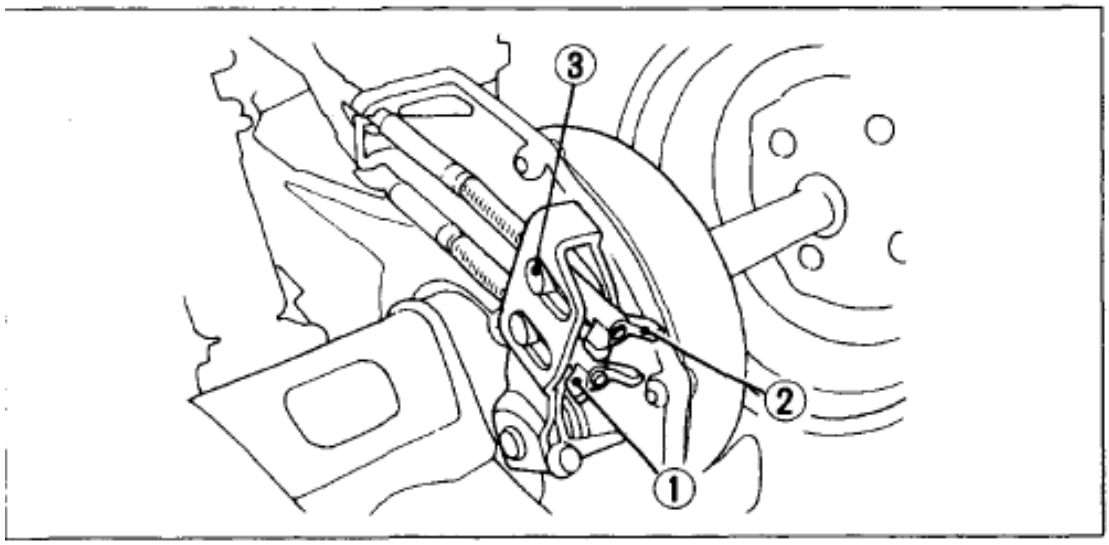


Fig. 4-26

(1) Brake pedal adjusting nut
(2) Brake lever adjusting nut
(3) Brake arm pin

To check rear brake shoe wear:

When the brake is applied, an arrow (3) attached to the brake arm (2) moves toward a reference mark (4) on the brake panel (see Fig. 4-28).

If the arrow aligns with the reference mark on full application of the brake, the brake shoes must be replaced. See your authorized Honda dealer for this service.

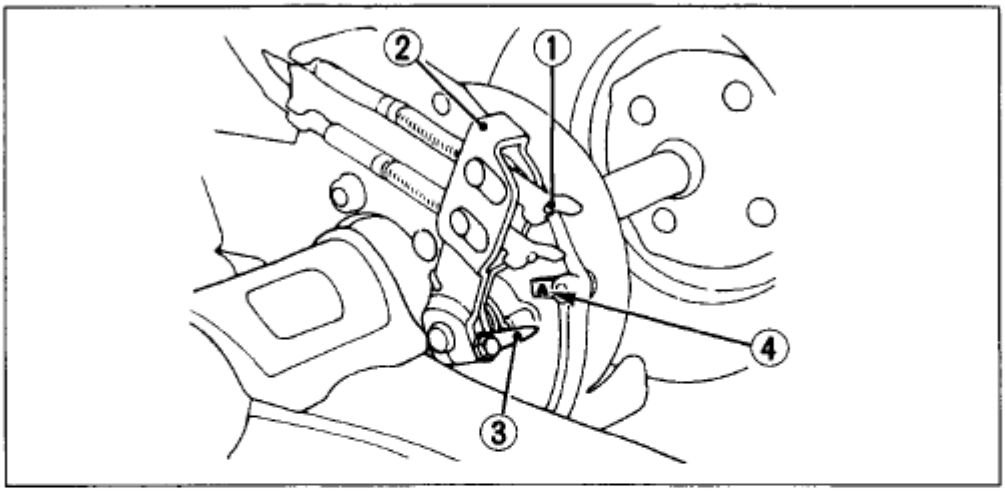


Fig. 4-28
(1) Brake lever adjusting nut
(2) Brake arm
(3) Arrow
(4) Reference mark

Other Checks:

Check the condition of the brake cables. If there are kinks or signs of wear that could cause sticking or failure, have the cables replaced by an authorized Honda dealer. Make sure the brake arm, spring, and fasteners are in good condition.

Appendix K Personal Communications - Email

Gmail sjocorpe . <sjocorpe@gmail.com>

FW: Masters Thesis Help
2 messages

Maree Debenham <Maree.Debenham@plantandfood.co.nz> Wed, Sep 30, 2015 at 3:11 PM
To: "sjocorpe ." <sjocorpe@gmail.com>

Hi Sam

Hope this helps.

Maree

From: Philip Martin
Sent: Wednesday, 30 September 2015 2:03 p.m.
To: Maree Debenham
Subject: RE: Masters Thesis Help

Hi Maree,

Kiwifruit row widths and height vary quite a bit depending when the block was established and what was fashionable at the time, but typical dimensions for a mature block would be the following:

- Canopy height – 1.8 metres above ground
- Row width – 3.5m (strip-male layout i.e. alternate rows of female and male vines)
- Vine spacing – 6m (one plant per bay i.e between posts that support canopy) or 3m (two plants per bay aka double-planted)

Hope that helps.

Regards,

Phil.

From: Maree Debenham