WIRELESSLY CONTROLLED TIMING BELT BASED CLIMBING PLATFORM

By

Yanping (Doris) Liu

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ENGINEERING

In the School
of
Engineering Science

SIMON FRASER UNIVERSITY

Summer 2010

Approval

Name: **Yanping Liu**

Degree: **Master of Engineering**

Title of Thesis: **Wirelessly Controlled Timing Belt Based Climbing Platform**

Examining Committee:

Chair: _____

Dr. Jie Liang, P.Eng.
Assistant Professor, Engineering Science, SFU

_____

Dr. Carlo Menon, P.Eng.
Senior Supervisor
Assistant Professor, Engineering Science, SFU

_____

Dr. Rodney Vaughan
Supervisor
Professor, Engineering Science, SFU

_____

Dr. Ash Parameswaran, P.Eng.
Internal Examiner
Professor, Engineering Science, SFU

Date Defended/Approved: August 24 2010

Declaration of
Partial Copyright Licence

Abstract

This project report presents the design of a wireless control system for a timing belt based platform (timing belt robot). Small climbing robots are of particular interest in the exploration of outer space where they can be used to perform inspection and observation. An embedded system controlled timing belt robot with four tracks was designed which is able to process information from all of its sensors, has the ability to adjust to different environments, and is also controllable through a wireless communication link. This project report presents the robot's control hardware design, software design, wireless communication system design, prototype construction, remote computer Graphical User Interface design and robot control algorithms.

Keywords: control, embedded system, wireless, robot

Acknowledgements

I would like to thank my senior supervisor, Dr. Carlo Menon, who provided such a great topic and has aided me throughout the course of this project. I would also like to thank my supervisor, Dr. Rodney Vaughan, for helping me so much on the wireless communication system.

I then extend my acknowledgements to Dr. Shahram Payandeh, Dr. Mehrdad Moallem, Dr. Ash Parameswaran, Dr. Paul Ho and Dr. Lesley Shannon for the inspiring lessons you gave.

Moreover, I would like to express my appreciation to all current and previous members of the MENRVA group at SFU, especially Jeff Krahn, Amir Sadeghi and Yasong Li for making parts and ordering components.

I also would like to thank my managers and colleagues at Schneider Electric for their support and understanding.

Table of Contents

List of Figures and Tables

List of Abbreviation and Acronyms

ADC	Analog to Digital Converter
API	Application Programming Interface
CAN	Controller Area Network
DC	Direct Current
DSP	Digital Signal Processor
GPIO	General Purpose Input/Output
GUI	Graphical User Interface
JTAG	Joint Test Action Group
I2C	Inter-Integrated-Circuit
IC	integrated circuit
IEEE	Institute of Electrical and Electronics Engineers
IR	Infrared
ISM	Industrial, Scientific and Medical (radio spectrum)
LCD	Liquid Crystal Display
LED	Light-Emitting Diode
MCU	Microcontroller Unit
NI	National Instruments
PC	Personal Computer
PCB	Printed Circuit Board
PID	Proportional Integral-Derivative
PWM	Pulse Width Modulation

QFP	Plastic Quad Flat Pack
RF	Radio Frequency
SCI	Serial Communications Interface
SMD	Surface Mount Device
SPI	Serial Peripheral Interface
TI	Texas Instruments
UART	Universal Asynchronous Receiver-Transmitter

1: INTRODUCTION

Interest in studying climbing robots is growing rapidly within the scientific and industrial communities. Wall climbing robots are able to perform inspections, observations, repairs, and act as personal assistants. Their small size allows them to enter areas that are inaccessible to humans. The use of wall climbing robots is especially desired in the exploration of outer space, where they are envisioned to be used for surface exploration, construction and maintenance of manned stations and satellites. Figure 1-1 below shows the NASA Mars Rover 2009 which is approximately the size of a car. The transporting of robots into space is the highest factor in the cost of robotic space exploration. Our goal is to research and design a miniaturized robot with regard to both size and weight; furthermore, the robot was designed to be able to detect walls, transfer from the ground to a wall and climb up a wall.

Figure 1-1: NASA Mars Rover 2009 [1]

1.1 Robot Platform

The climbing timing belt robot with four tracks was designed within Dr. Menon's lab and is composed of three parts, the front module, rear module, and a robotic link connecting the front and rear modules, as shown in Figure 1-2. Please refer to Figure 1-4, Illustrated Timing Belt Robot Top View, for names of tracks and modules.

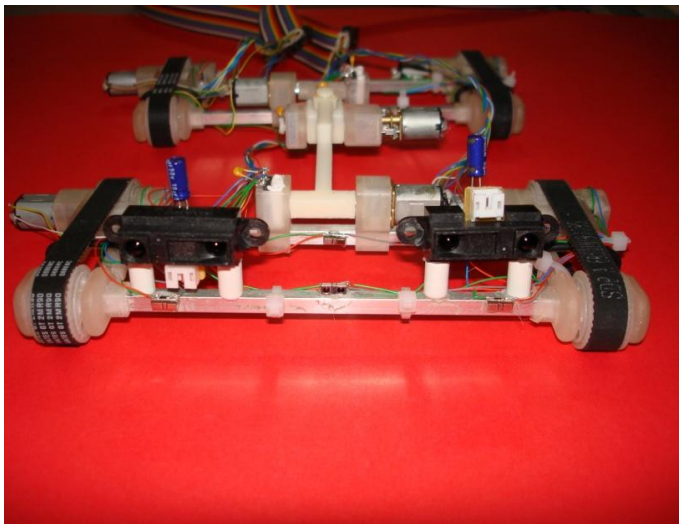


Figure 1-2: Timing Belt Robot

The moving parts of the climbing robot consist of seven motors, four of which are connected to the tracks and three of which control the joints connecting the front and rear modules. There is also a free moving steering joint. Diagrams of the timing belt robot are shown in Figures 1-3 and 1-4.

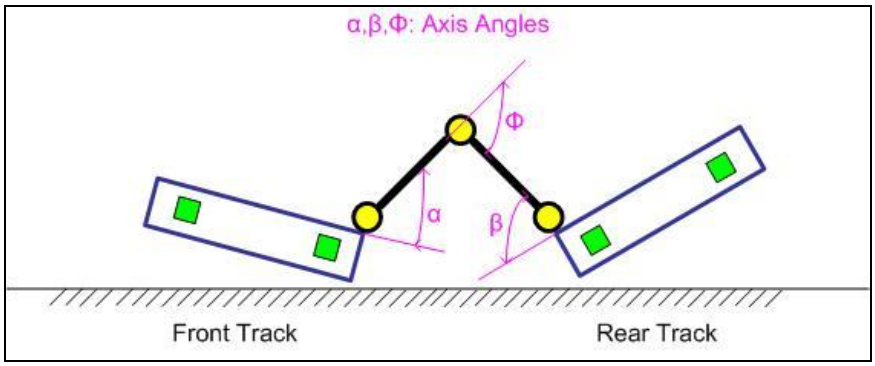


Figure 1-3: Diagram of the Timing Belt Robot Showing the Motor Controlled Joints

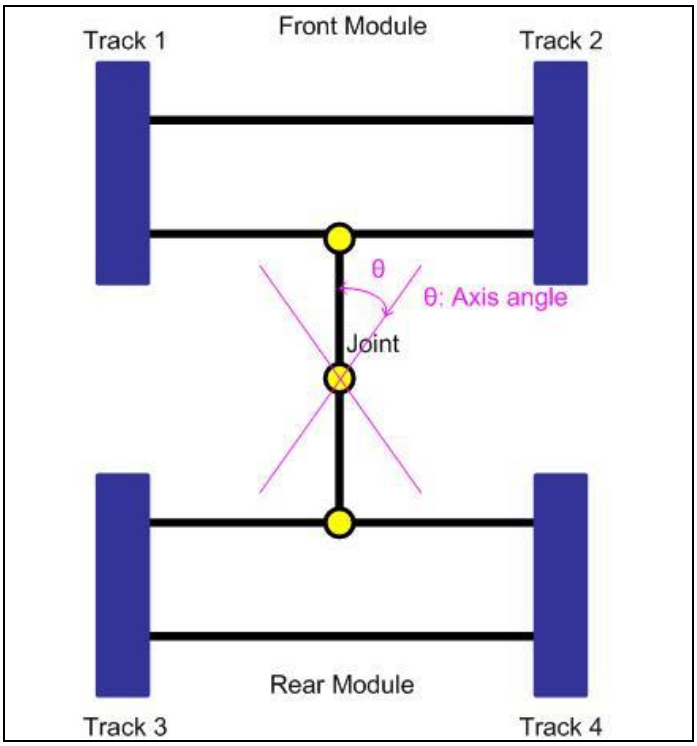


Figure 1-4: Illustrated Timing Belt Robot Top View

For the intended application, the climbing robot should not only have the ability to follow commands through wireless communication, it should also have the ability to judge and control itself. If a wall is detected, the robot should be able to adjust itself so that it is perpendicular to the wall, move forward to contact the wall and begin climbing.

The wireless communication module selected for this system has an embedded microcontroller that can pass wireless communication commands to the Digital Signal Processor (DSP) on the robot via serial communication, as well as giving it the ability to be programmed through a wireless link.

1.2 Project Scope

The development of the wirelessly controlled climbing timing belt or tracked robot required a multi-disciplinary effort involving mechanical design, sensor selection, DSP hardware design, RF hardware design, prototype construction, control algorithm design, DSP embedded software design, wireless software design, and PC user interface design. My role in this project was to select the Digital Signal Processor (DSP) and wireless modules, place sensors in suitable locations as well as build a prototype board, design both control algorithms and the software.

A successful climbing robot should be able to handle various tasks. The scope of this M.Eng project is therefore focused on designing a robot with a powerful DSP so that it can intelligently process information from all sensors in order to easily adapt to different environmental conditions. With this in mind, the robot should be lightweight with a reliable control system, and it should also be controlled wirelessly by a PC. For this M.Eng project, the robot is designed and developed so that it has the following key features:

- Both the timing belt motors and the motors in the link connecting the two robotic modules can turn forward and reverse.

- All the motors are controlled by Pulse Width Modulation (PWM) so that their rotational speed can be controlled and adjusted separately.
- When a wall is detected, while moving forward, the robot can change its trajectory so that it is perpendicular to the wall.
- The robotic platform is able to transfer from horizontal to vertical surfaces by using active control of the motors connecting the front and rear modules of the robot.

1.3 Project Report Outline

This M.Eng project involved microcontroller selection, a short range wireless communication system hardware selection, hardware design and prototype construction. It also includes the software design for a Texas Instruments (TI) Digital Signal Processor (DSP) TMS320F2808, ZigBee wireless communication system Co-ordinator, ZigBee wireless communication system End Device, and LabVIEW PC user interface.

This project report is organized as follows:

- Section two provides an overview of the robot, presents system block diagrams, and analyzes the sensors used in the robot platform.
- Section three presents the system design of the robot including the hardware design, wireless communication system design, control

algorithm design, control software and LabVIEW Graphical User Interface (GUI) design respectively.

- Section four provides signal strength measurement for the ZigBee wireless communication system.
- Section five presents the conclusion of all preliminary results.
- Section six indicates the future work necessary to achieve a fully reliable climbing timing belt robot.

2: PROJECT OVERVIEW

This section describes the climbing timing belt robot platform, gives an overview of the entire system, and analyzes the various sensors used in the robot platform.

2.1 Two Modular Combined Timing Belt Robot Platform

The timing belt robot consists of four timing belt tracks each driven by one DC gear motor. Three additional geared motors are used to control the robotic link with a motor mounted on each of the two bars used to connect the two modules. A third motor in the middle of the robotic link controls the angle, Φ, (see Figure 1-3) between the two connecting bars. The joint is free to steer at higher angles and can easily transfer from horizontal to vertical and vice-versa. The robot is shown in Figure 2-1 and the numbers indicate the positions of the wide range distance measuring Infrared (IR) sensors, proximity sensors, and potentiometers used to measure the displacement (see Table 2-1).

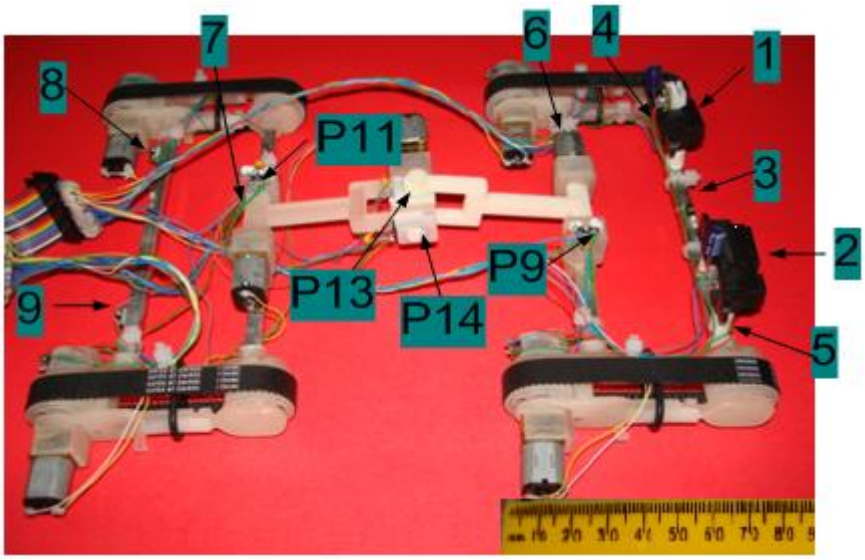


Figure 2-1: Climbing Timing Belt Robot

Table 2-1: List of Sensors Used On the Robot

Number on Fig2-1	Description	Specifications	Model Number
1,2	Wide range distance measuring Infrared (IR) sensor made by Sharp	10cm to 80cm 0V to 3.3V	GP2Y0A21YK
3,4,5,6,7,8,9	Surface mount high efficiency infrared proximity sensor	0cm to 6cm 0V to 3.3V	HSDL-9100
P9,P11,P13, P14	Rotary potentiometer	10kΩ	EVW-AE4001B14

In order to precisely control the speed and direction of the timing belts, the motors controlling the timing belts in each module are also attached to a pair of rotary potentiometers which are not labelled in Figure 2-1. Figure 2-2 below shows the potentiometer holders on the drive modules [2]. Since the rotary potentiometer has a dead zone, a single rotary potentiometer is unable to detect a full 360° rotation because there is no output in the potentiometer's dead zone.

In order to fully monitor each timing belt, a pair of potentiometers is used to achieve full 360^{o} monitoring of the turning of each gear motor.

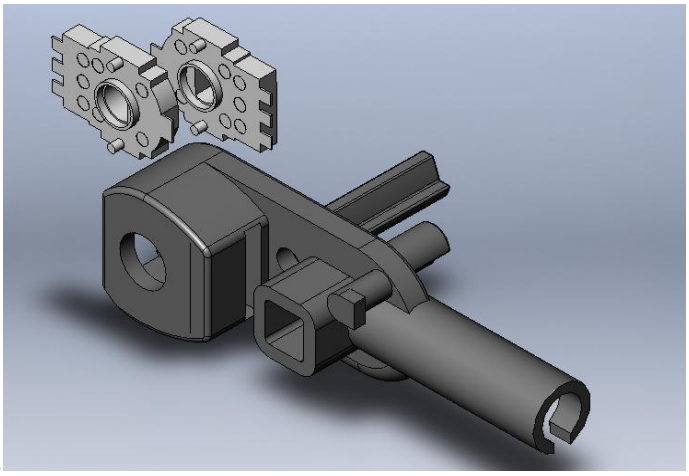


Figure 2-2: Potentiometer Holders on the Timing Belt Modules

The complete device and sensor map for the timing belt robot platform showing the positions of devices and sensors is displayed in Figure 2-3 and this map was used to define the communication protocol and program software. Seven DC gear motors, nine proximity sensors and twelve potentiometers are numbered in this figure. The timing belt robot senses the environment around itself by monitoring its nine proximity sensors. The robot is able to move forward, left or right by controlling the four track motors, and it is also able to tilt its tracks using the three joint motors. Utilizing the twelve potentiometers, the robot is able to receive feedback from the motor rotation to determine speed and distance travelled and control steering.

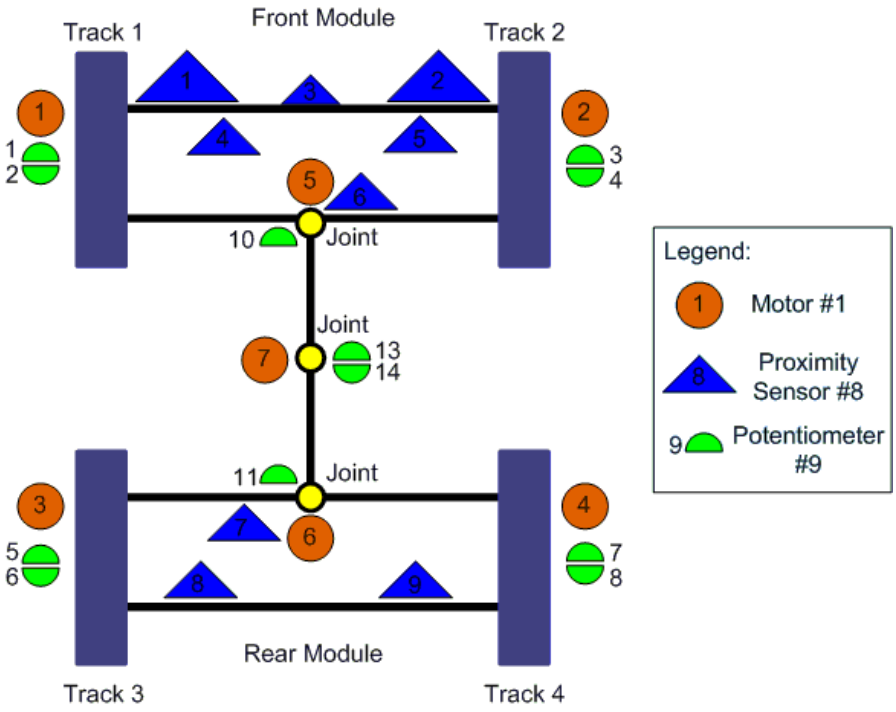


Figure 2-3: Robot Device and Sensor Map

2.2 System Overview

The TMS320F2808 microcontroller reads commands received from the user via the LabVIEW Graphical User Interface (GUI), and obtains sensor readings from the Analog-to-Digital Converter (ADC). The microcontroller controls the motors on the robot and causes them to rotate continuously or step by step. Proportional Integral-Derivative (PID) controllers are used to control the robot joint folding angles, set timing belt speeds on the rear module so that the front and rear modules (see in Figure 2-3) are be parallel at all times, and adjusts the robot so that it is perpendicular to a wall prior to climbing. The speed of each motor is controlled by PWM, so the robot can move at different velocities. When the timing belt robot receives a command from a user instructing it to move

forward and a wall is detected in its path, the robot will adjust itself so that it is perpendicular to the wall. When the robot is approximately 25cm from a wall, an alarm is sounded and when the robot reaches the wall, it begins to lift the front module in order to climb the wall.

Figure 2-4 shows the LabVIEW GUI and ZigBee Co-ordinator system block diagram. The ZigBee Co-ordinator is the root of the network tree, and it starts the network originally. There is one Co-ordinator in each ZigBee network. The Co-ordinator board, shown within the dashed line in Figure 2-4, which consists of a Jennic JN5139 ZigBee module with built- in microcontroller, LCD display, switches and buttons. A SMA antenna is used by the JN5139 ZigBee module for wireless communication. The user is able to send commands through the LabVIEW GUI in order to control the climbing timing belt robot either step by step or continuously. The LabVIEW GUI initializes the serial port in the computer, and then communication is established between both the computer and the Co-ordinator. LabVIEW provides the driver for serial communication. The user must select the correct serial port before running the LabVIEW GUI as this is not done automatically by LabVIEW. When the ZigBee Co-ordinator receives commands from the controlling PC, it broadcasts them through the wireless network. The Co-ordinator also has the ability to scan for End Devices, and allow them to join the network. If an End Device has successfully joined the network, the LCD on the Co-ordinator board will display the End Device name. An End Device in a ZigBee network has limited functionality, and it can communicate with the Co-ordinator.

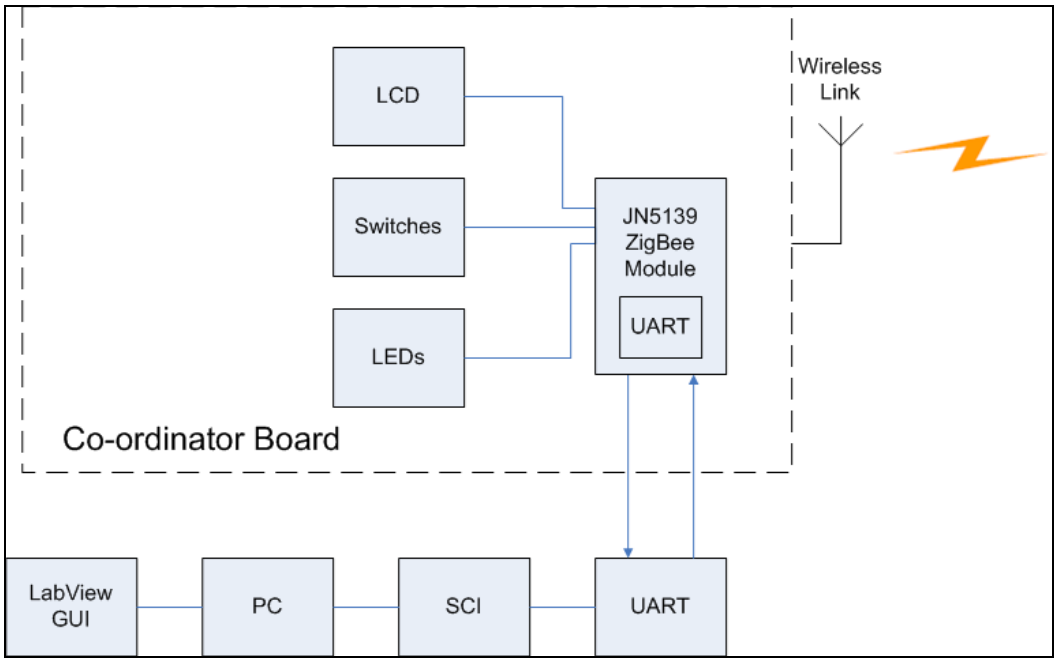


Figure 2-4: Block Diagram of GUI and ZigBee Co-ordinator

Figure 2-5 shows the TI TMS320F2808 DSP and ZigBee End Device system block diagram. Inside the larger dashed box denoted by a dashed line in Figure 2-5 is the TMS320F2808 and all its modules, while inside the smaller box, also denoted by a dashed line, is the ZigBee JN5139 module acting as an End device. After an End Device joins the wireless network, it is able to receive and send messages through the network and passes messages to the TI DSP via serial communication. A small UART board is used to adapt the voltage difference between the JN5139 ZigBee module and the Serial Communications Interface (SCI) connector coming from the DSP. The voltage coming out from JN5139 ZigBee module is 3.3V, and the voltage coming from DSP SCI connector is ±12V. Inside the large box in Figure 2-5 are the ADC, SCI, PWM, TIMER, and General Purpose Input/Output (GPIO) drivers as well as the TMS320F2808 hardware peripherals drivers that were programmed. The PID controller is the

core of the control system. A simple real-time operating technique is used to enable the system to react in a real-time manner, and is also used to coordinate different tasks. In order to reduce the weight of the robot, a Jennic JN5139 ZigBee module with a built-in ceramic antenna was used in the design. In the future, the control board can be redesigned to be much smaller, but the ZigBee module will remain the same. PWM is used to control the motors on the robot and is done through an H-Bridge driver. LEDs are used to indicate the status of the power supply, robot and to indicate that the program is running. A buzzer sounds an alarm when the timing belt robot detects a wall and is preparing to climb.

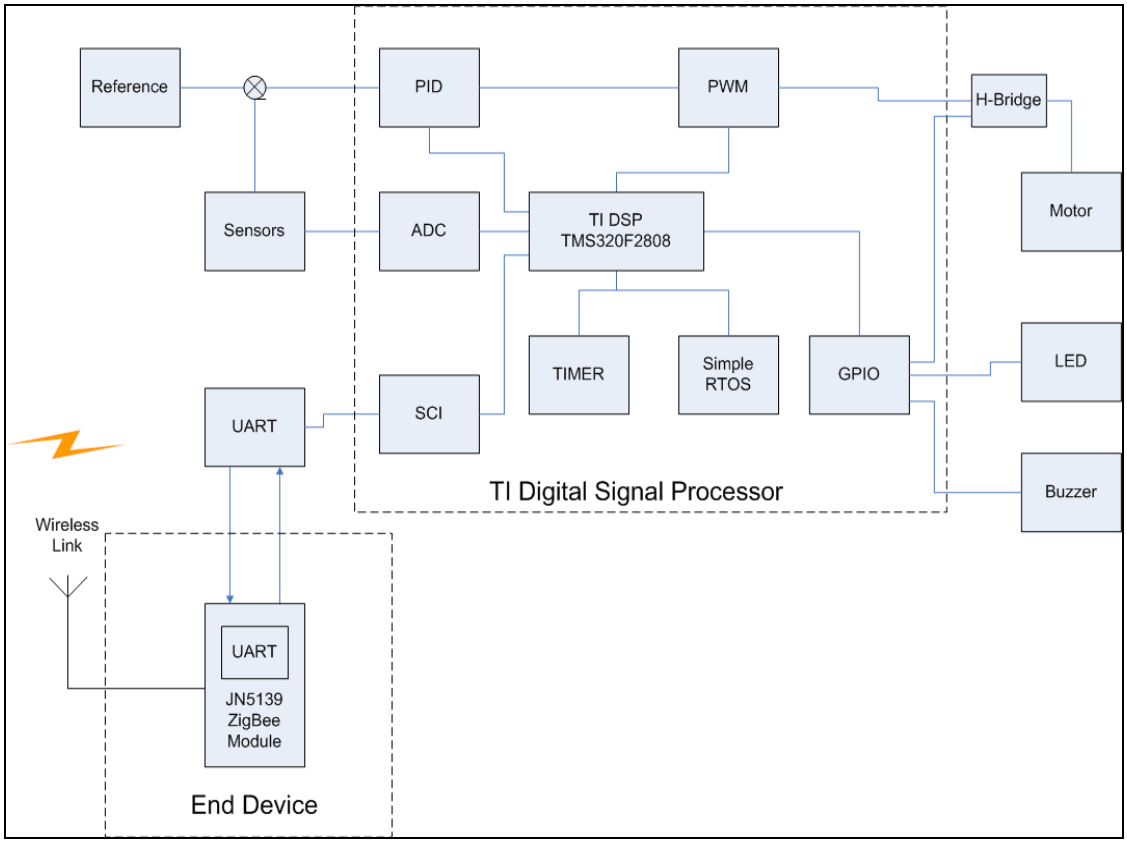


Figure 2-5: Block Diagram of TI DSP and ZigBee End Device

2.3 Sensors Used On the Robot

Sensors presented in Table 2-1 are discussed in the following sections.

2.3.1 GP2Y0A21YK Infrared Sensor Made by Sharp

Two GP2Y0A21YK wide range distances measuring Infrared (IR) sensors made by Sharp were mounted on the robot to detect obstacles. The sensors are placed on the robot with one of them flipped up-side down and both turned slightly away from each other in order to increase the separation of the LEDs and decrease the amount of mutual interference between both of them. The distance between two infrared emitters is 9.5cm. When both sensors read the same voltage, the robot is perpendicular to a wall or obstacle. Figure 2-6 shows the Sharp GP2Y0A21YK sensors on the robot.

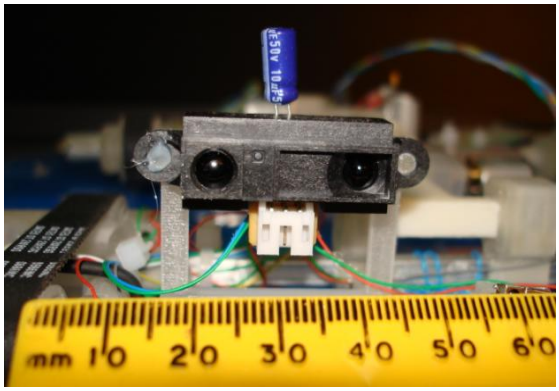


Figure 2-6: Sharp GP2Y0A21YK Sensor on the Robot

In order to calibrate the sensors, the distance between the robot and the wall was changed and the sensor output voltage was measured. Also, different materials were used for the wall in order to determine the sensitivty of the sensors with different obstacles. The relationship between wall distance and the

sensor output voltage is illustrated in Figure 2-7. The measured output voltage verses the inverse distance between the robot and the wall was plotted in order to achieve a curve that is nearly linear in the region of interest. Substituting different materials for the wall had negligible effects on the sensor output voltage.

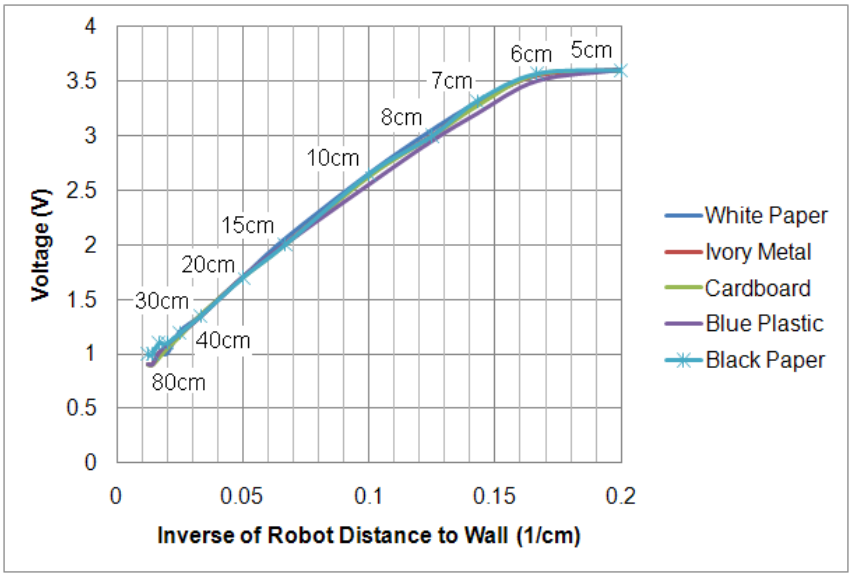


Figure 2-7: Sharp Infrared Sensor Output Voltage vs. Inverse of Distance to Wall

The relationship between the sensor output voltage and the inverse of the distance between the robot and the wall is linear when the distance is between 10cm to 80cm. The timing belt robot was tested mostly on ivory-coloured metal, so a linear function was fit to the data for the ivory-coloured metal and is presented in Figure 2-8 below. The straight line shown in black best fits the model, and has an R^2 value of 0.9982 indicating a close approximation since an R^2 value of 1 indicates a perfect match to the data.

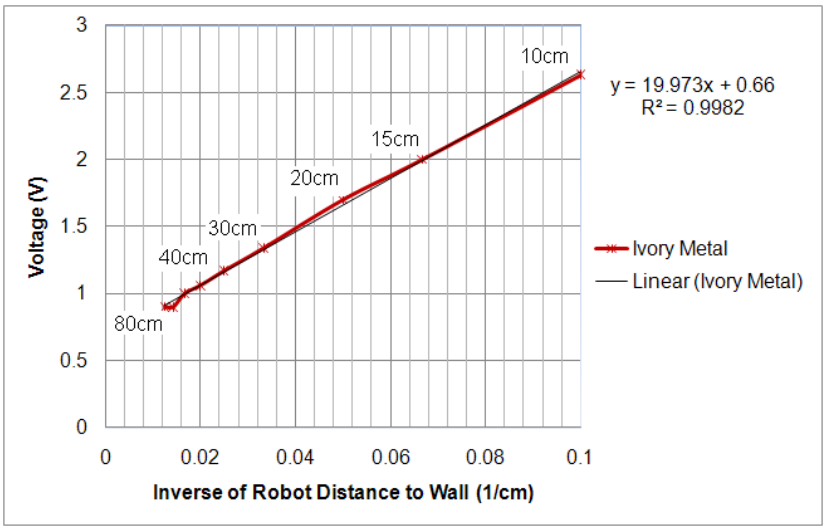


Figure 2-8: Linear Relationship between Output Voltage and Inverse Number of Distance for Ivory Metal

From Figure 2-8, the output voltage variation with the inverse of the distance is shown below in equation 2-1,

$$y = 19.973x + 0.66 \quad \text{.....(2-1)}$$

where *y* is the sensor output in volts, and *x* is the inverse of robot distance to wall in 1/cm. The R^2 value indicating the accuracy of the fit is shown in equation 2-2,

$$R^2 = 0.9982 \quad \text{.....(2-2)}$$

When the sensor output is 3.3V, the ADC value is at a maximum of 4095 (12-bit ADC), so the relationship between the distance from the robot to the wall and the ADC value can be determined. The relationship between the distance from the robot to the wall and the corresponding ADC value is shown in equation 2-3,

$$d = \frac{24784.677}{ADC - 819} \quad \text{.....(2-3)}$$

where *d* is the distance in cm, and *ADC* is the microcontroller ADC's value which has a range from 0 to 4095.

2.3.2 Avago HSDL-9100 Proximity Sensor

The HSDL-9100 is a reflective sensor with an integrated high efficiency infrared emitter and photodiode housed within a small SMD package. The optical proximity sensor is housed in a specially designed metal-shield to ensure excellent optical isolation resulting in low optical cross-talk [4]. The range of the sensor is from 0 to 6cm. Figure 2-9 illustrated the HSDL-9100 proximity sensor. This sensor is specifically optimized for size, performance and ease of design.

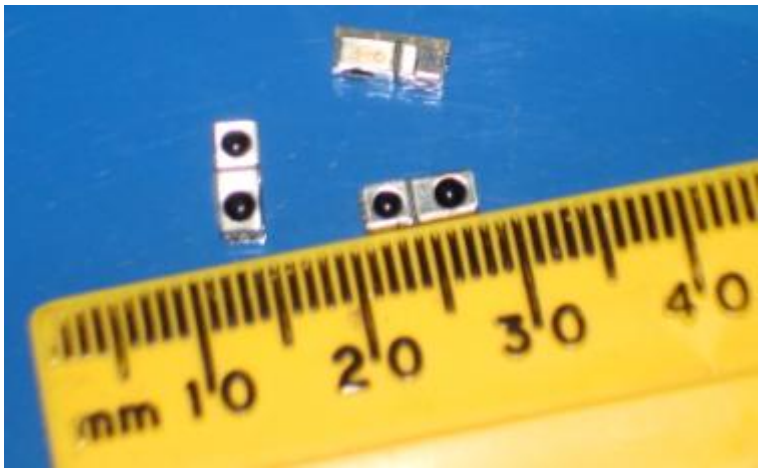


Figure 2-9: Avago HSDL-9100 Proximity sensor

There is one HSDL-9100 proximity sensor placed between the two forward facing Sharp GP2Y0A21YK sensors and it faces forward similar to both of the Sharp sensors. This proximity sensor is used in synergy with the Sharp sensors to detect a wall or obstacle since the Sharp sensor works the best at a distance between 10cm and 80cm from an obstacle. When the distance is within the range of the proximity sensor, the Sharp sensor output is ignored.

The measured relationship between the HSDL-9100 proximity sensor output voltage and the distance to the wall is shown in Figure 2-10. Different materials were used in the measurement.

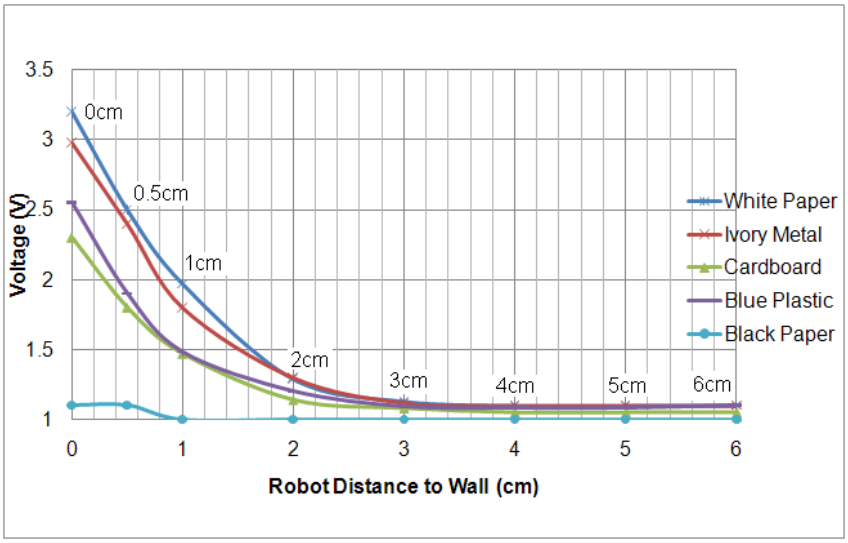


Figure 2-10: HSDL-9100 Output Voltage vs. Robot Distance to Wall

The Sharp sensor ADC reading is at the maximum of 4095 (12-bit ADC) when the sensor is between 3cm to 7cm away from the wall because at this distance the sensor output voltage is greater or equal to 3.3V. From Figure 2-10 above, we can see that the proximity sensor output voltage starts to change when it is 6cm away from the wall, and works the best when it is 3cm from the wall. The Sharp sensors and the middle proximity sensor work together to obtain an accurate distance from the wall and when the distance is between 7cm and 80cm, the Sharp sensor output is used. There is a dead zone which can not be measured between 6cm and 7cm distances because the Sharp sensor outputs the maximum ADC value at this range. The HSDL-9100 proximity sensor's output cannot be used at this distance. When the distance from wall is less than 6cm,

the proximity sensor's output is used to calculate the distance. The sensor reading flowchart is shown in Figure 2-11.

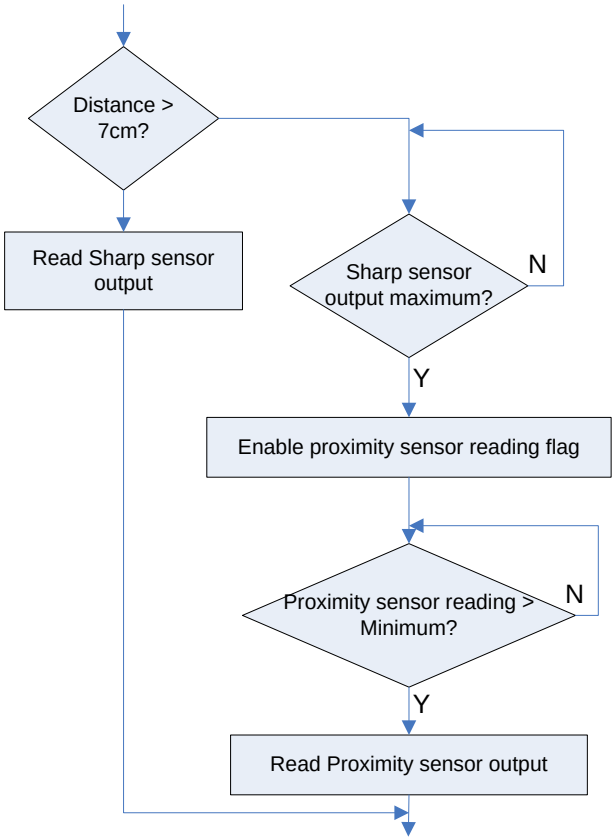


Figure 2-11: Sharp and Proximity Sensor Reading Flowchart

In Figure 2-1, shown previously, numbers 4, 5 and 6 indicate three proximity sensors placed in the front module of the robot, and numbers 7, 8 and 9 are three proximity sensors in the rear module of the robot. A diagram of the robot with proximity sensors locations on its body is illustrated in Figure 2-12 below.

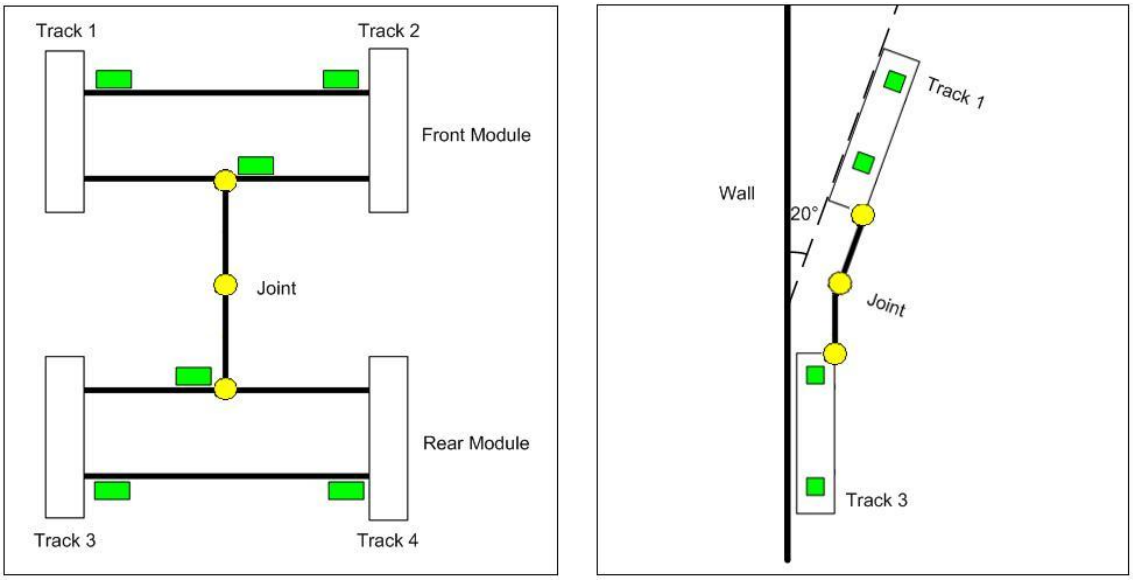


Figure 2-12: Robot with Proximity Sensors

The proximity sensors are all mounted facing the surface the robot is on in order to detect the distance between robot body and the surface. Three sensors, working in unison, are able to define a surface and determine if the robot is detaching from a wall or transferring from one surface to another.

3: SYSTEM DESIGN

This section is described in the following order: hardware design, ZigBee wireless communication system, communication protocol design, control algorithm design, control software and LabVIEW user interface design.

3.1 Hardware Design of the Climbing Timing Belt Robot

The hardware design is implemented on the TMDSDOCK2808. Four L293D H-Bridge driver chips are used to drive the motors, and one 8:1 multiplexer chip connected to the ADC is used to monitor the eight rotary potentiometers on the tracks. The ZigBee JN5139 module is connected to the microcontroller through a serial communications interface (SCI). The system prototype board built for this project is shown in Figure 3-1. For dimensions of this board, please refer to Appendix C.

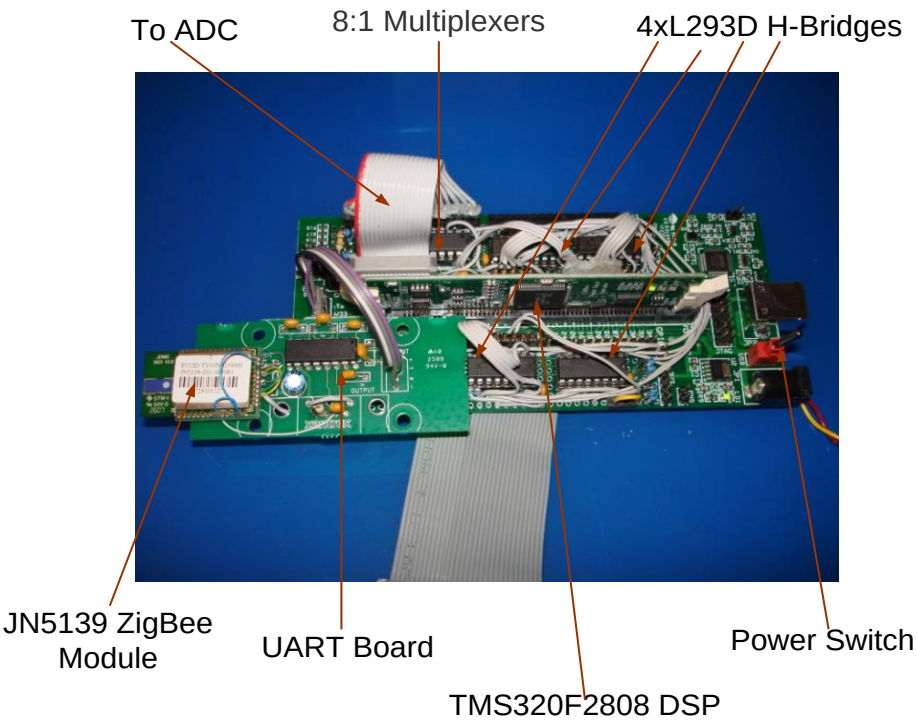


Figure 3-1: System Prototype Board

3.1.1 Motor Control

The L293D H-Bridge driver from Texas Instruments is used to drive all seven motors on the robot in order to turn them either clockwise or counter clockwise. The L293D is designed to provide bidirectional drive current of up to 600mA at voltage from 4.5 V to 36 V.

Because the motors on the tracks need to run bi-directionally at different speeds, they need to be controlled by PWM which, according to the H-Bridge driver datasheet, can be set at the frequency of 3 kHz. To minimize the PWM channel number, the motors are controlled by both PWM and General Purpose Input/Output (GPIO). Figure 3-2 shows the logic diagram with two L293D H-Bridge driver chips being used to control the four motors which in turn, control the tracks.

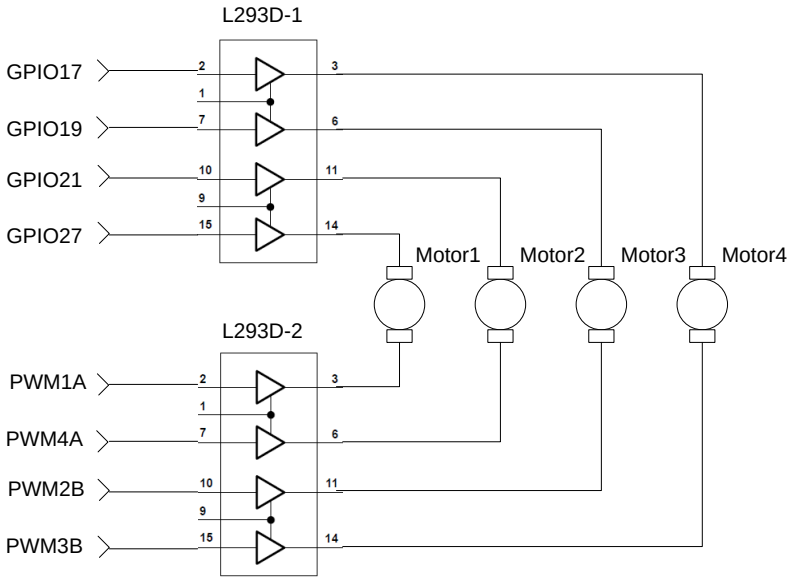


Figure 3-2: Logic Diagram of Motor Control Circuit

When the tracks need to rotate in a forward direction, the GPIO outputs low voltage level, and the motor speed can be controlled by the PWM duty cycle. Figure 3-3 is the predicted ideal voltage waveforms at the two terminals of a motor turning forward.

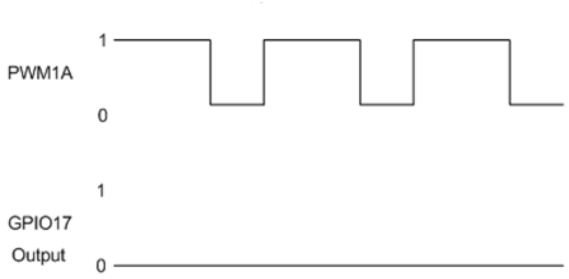
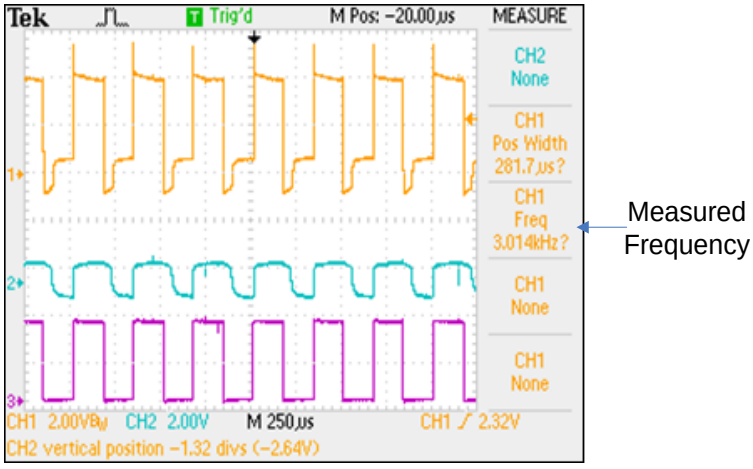


Figure 3-3: Predicted Voltage Waveforms for Forward Motor Rotation

Figures 3-4 shows the actual measured voltage waveforms at the terminals of a single timing belt control motor. An oscilloscope was used to measure the voltage waveforms with a setting of 2 Volts per division along the

y-axis. When the track motor were turning forwards, two motor terminal voltages were measured and shown in channel 1 (orange color) and channel 2 (blue color). The motor PWM terminal (orange) has a duty cycle of 50% and a frequency of 3 kHz (The oscilloscope horizontal scale was set to 250µS per division). When compared to Figure 3-3, the measured waveforms are distorted when the motor is turning. In Figure 3-4, the pink waveform is the PWM output voltage measured at the TI TMS320F2808 microcontroller output. The PWM at the microcontroller output is a nice square wave, but it becomes distorted at the motor terminals when the motor is rotating.



CH1: orange, 2V, 250µs, Motor PWM terminal voltage

CH2: blue, 2V, 250µs, Motor GPIO terminal voltage(low)

CH3: purple, 2V, 250µs, Microcontroller PWM output voltage

Figure 3-4: Waveform Measured at Motors Two Terminals for Motor Turning Forward

When a motor runs in the reverse direction, the GPIO outputs high voltage levels, and the motor is then controlled by the voltage difference between the two terminals. If the motor needs to turn at the same speed but in the reverse direction, the PWM should output a reversed logic level. Figure 3-5 shows the

predicted voltage waveforms at the motor terminals when the motor is turning in the reverse direction. When the motor rotates in the reverse direction, the GPIO output must be high. When the PWM output is at a low voltage level, the motor will turn in the reverse direction. On the other hand, if the PWM is high and the two motor terminals are at same voltage level the motor will stop.

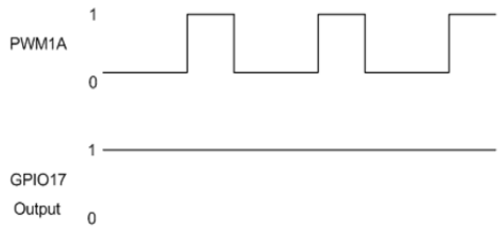
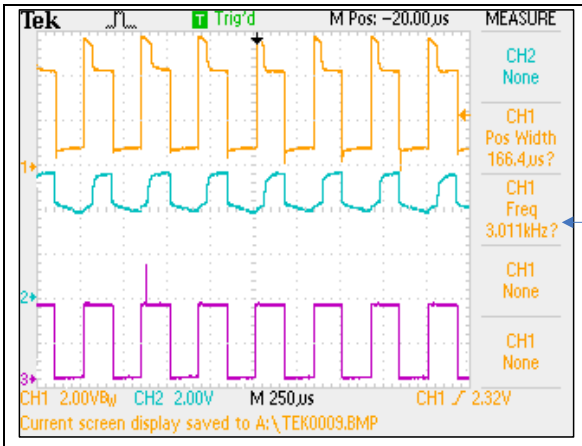


Figure 3-5: Predicted Voltage Waveforms for Reverse Motor Rotation

Please note that in Figures 3-3 and 3-5, both motors are controlled at the same speed. The motor PWM needs to output reverse logic in order for a motor to turn in the reverse direction.

Figure 3-6 shows the actual voltage measurement taken using an oscilloscope of one of the timing belt control motor terminals with the motor turning in the reverse direction. When the track motor rotation was reversed, the terminal voltages of two motors were measured and are shown in channel 1 (orange color) and channel 2 (blue color). The motor PWM terminal (orange) has a duty cycle of 50% and a frequency of 3 kHz. In Figure 3-6, the pink waveform indicates the PWM output voltage measured at the microcontroller output. We can see the PWM at the microcontroller output is a nice square wave, but becomes distorted at the motor terminals when the motor is rotating.



CH1: orange, 2V, 250µs, Motor PWM terminal voltage

CH2: blue, 2V, 250µs, Motor GPIO terminal voltage(high)

CH3: purple, 2V, 250µs, Microcontroller PWM output voltage

Figure 3-6: Measured Waveforms at Motors Two Terminals for Reverse Motor Rotation

Motor rotational speeds can be controlled by changing the PWM duty cycle. Each motor is controlled separately so that the belts on the timing belt robot can turn at different speeds to obtain a better trajectory. When the robot is turning, its motors are able to rotate at different speed for different time durations. Figure 3-7 displays the measured waveforms when the robot was turning left and it can be seen that the four motors are rotating at different speed with different time durations. The PWM control for timing belts 2 and 4 is in reverse logic when motor is reversely rotating and the waveforms are shown in blue and green respectively in Figure 3-7.

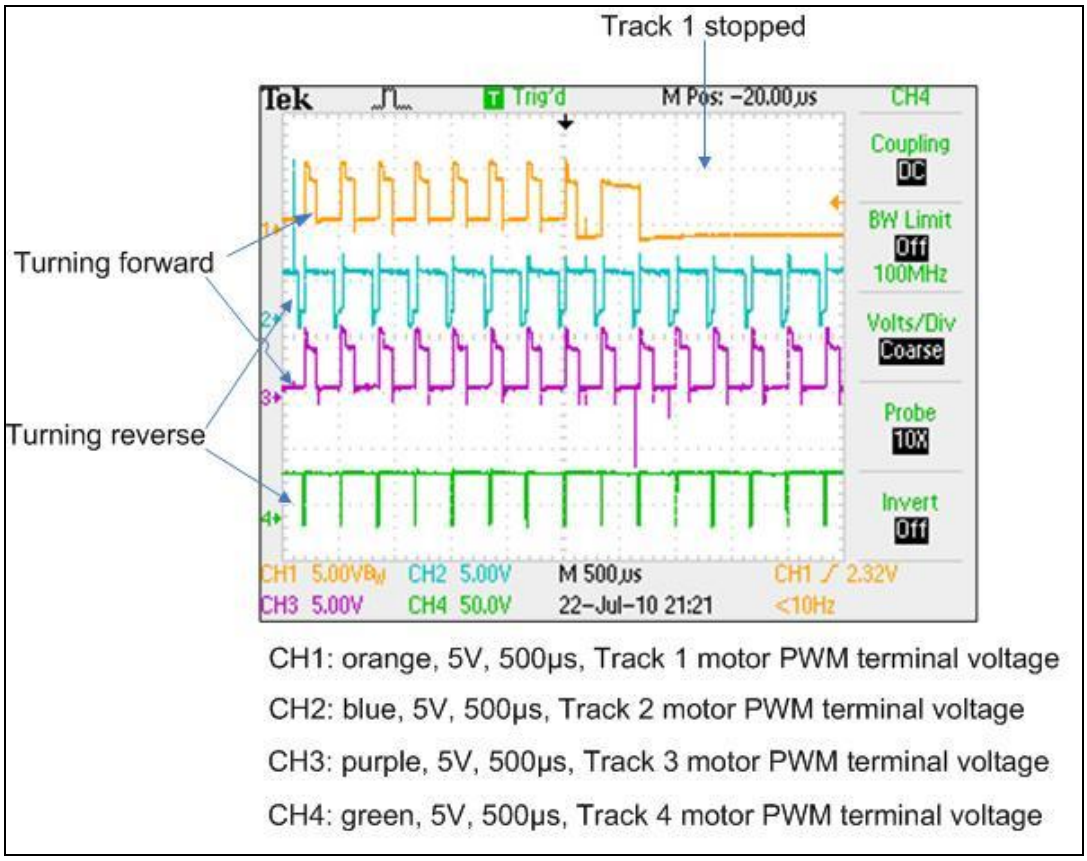


Figure 3-7: Measured Motor PWM Terminal Voltages When Robot was Turning Left

3.1.2 Analog and Digital Ground

There are seven motors on the robot, so it is important to separate the analog and digital grounds. Since the system is built on a prototype board, the separation of these two grounds is very important for the reduction of noise in the digital signals which relay sensor readings to the microcontroller. Generally, digital grounds tend to be noisier than analog grounds due to the switching noise generated in digital chips when the digital state is changed. Noise is reduced maximally when the digital and analog grounds are carefully separated and connected together at only a single point.

3.2 ZigBee Wireless Communication System Design

3.2.1 ZigBee Wireless Communication System Design Overview

The selected wireless communication system utilizes ZigBee, a worldwide open standard for wireless radio networks in the monitoring and control fields based on the IEEE 802.15.4 standard. ZigBee operates in the industrial, scientific and medical (ISM) radio bands; around 868 MHz in Europe, 915 MHz in countries such as USA and Australia, and 2.4 GHz in most jurisdictions worldwide. For details, please refer to my previous report titled “Bluetooth, ZigBee, Wi-Fi and WiMax for Wireless Communications.”

The wireless module which met all of the requirements was the Jennic 802.15.4 SMD Module JN5139, which has a built-in ceramic antenna. It was selected because of its small light weight module combined with long battery life and secure networking. The required distance between the End Device and Co-ordinator in this project was well within the ZigBee system’s range of approximately one kilometre. For more details about the advantages of ZigBee, please refer to my previous article entitled “ENSC 891 Directed Studies – Practical Short Range Wireless Communications Module Final Report” [32]. See Figure 3-8 for the Jennic 802.15.4 SMD Module used in the system.

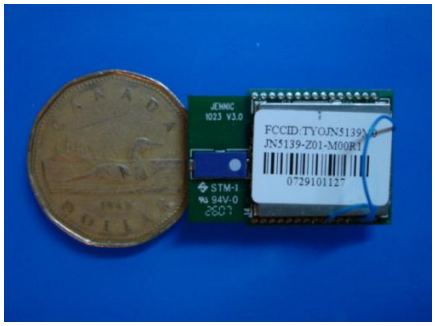


Figure 3-8: Jennic 802.15.4 SMD Module

The ZigBee wireless system was used to transfer user commands from the remote computer to the robot control system as well as the robot's status and parameters back to the remote computer. Star topology was implemented and tested. The system can connect up to four End Devices with one Co-ordinator. In this project, the Star topology was chosen since there is distinct possibility that more timing belt robots may possibly join the network. The network setup for the robot control system is illustrated in Figure 3-9 below. This system has the potential to connect and control up to four End Devices.

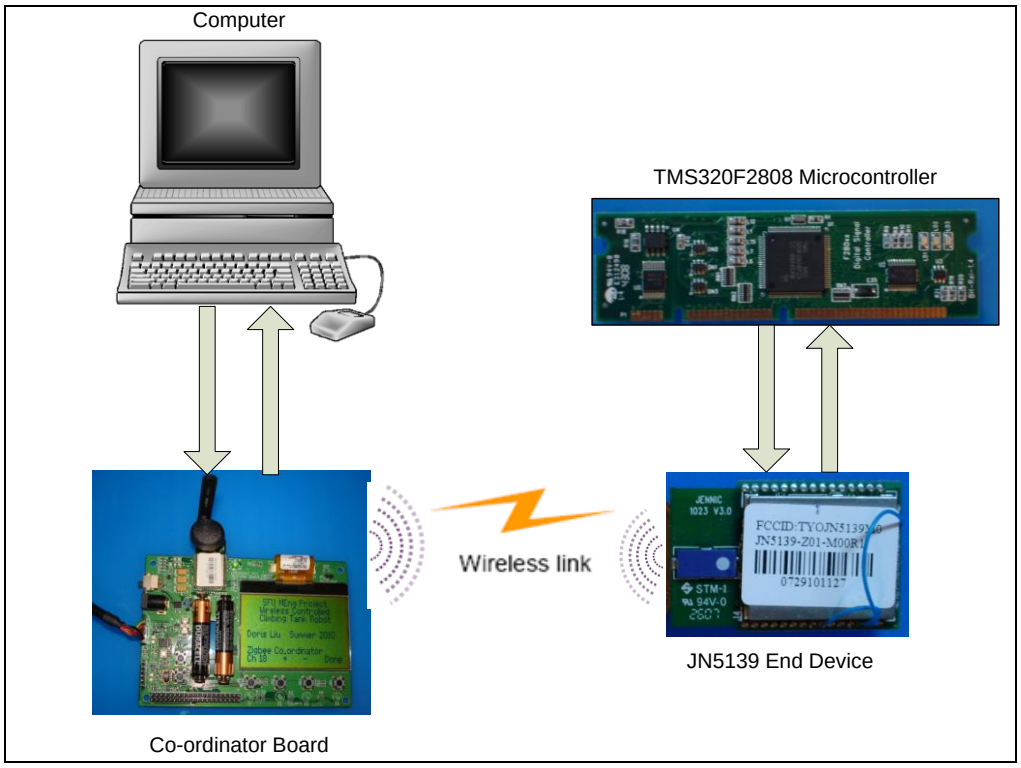


Figure 3-9: Wireless Network

The maximum transmission power for the Jennic JN5139 ZigBee module is 2.5dBm [7] which is very high among ZigBee vendors. The 802.15.4 standard requires radios to have a minimum output power of -3dBm or 0.5mW. Other radios on the market today generally have power outputs of between 0 and 3dBm or 1 and 2mW. The ZigBee wireless communication system network block diagram is shown in Figure 3-10.

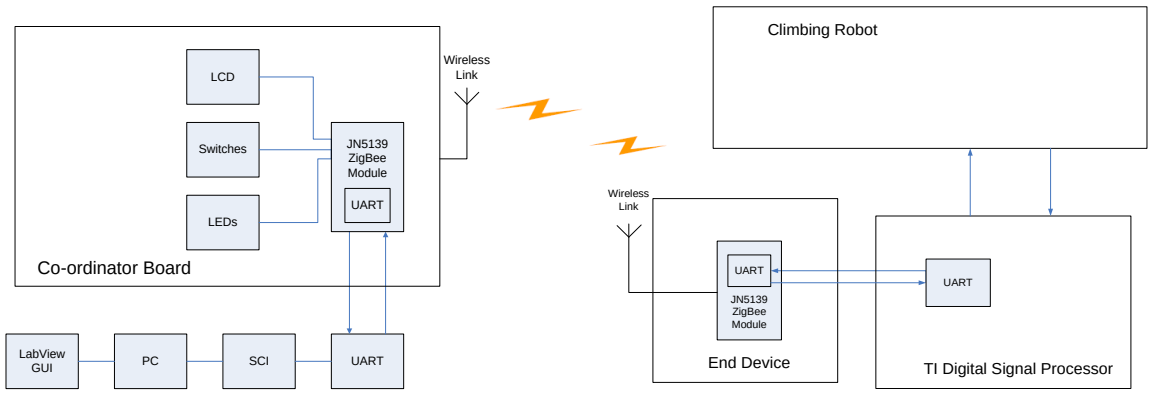


Figure 3-10: ZigBee Wireless System Network Block Diagram

3.2.2 ZigBee Communication Software Design

The code development for the climbing robot is based on the Jennic Home Sensor Demo System source code which may be downloaded from the Jennic website. The demo code is only designed for wirelessly transmitting data from the End Device to the Co-ordinator, but the climbing robot network needs to be able to send control commands and receive sensor signals. This was done by adding the vTransmitDataPacket() function and detailed code. In order to communicate with the robot's microcontroller, serial communication was added so that the End Device can pass commands to the robot and vice versa.

The Co-ordinator and End Device network setup process is based on the Jennic demo system with the source code for the Co-ordinator and End Device defined as coordinator.c and enddevice.c respectively. The ZigBee software has an event driven structure and the Co-ordinator software flowchart is illustrated in Figure 3-11.

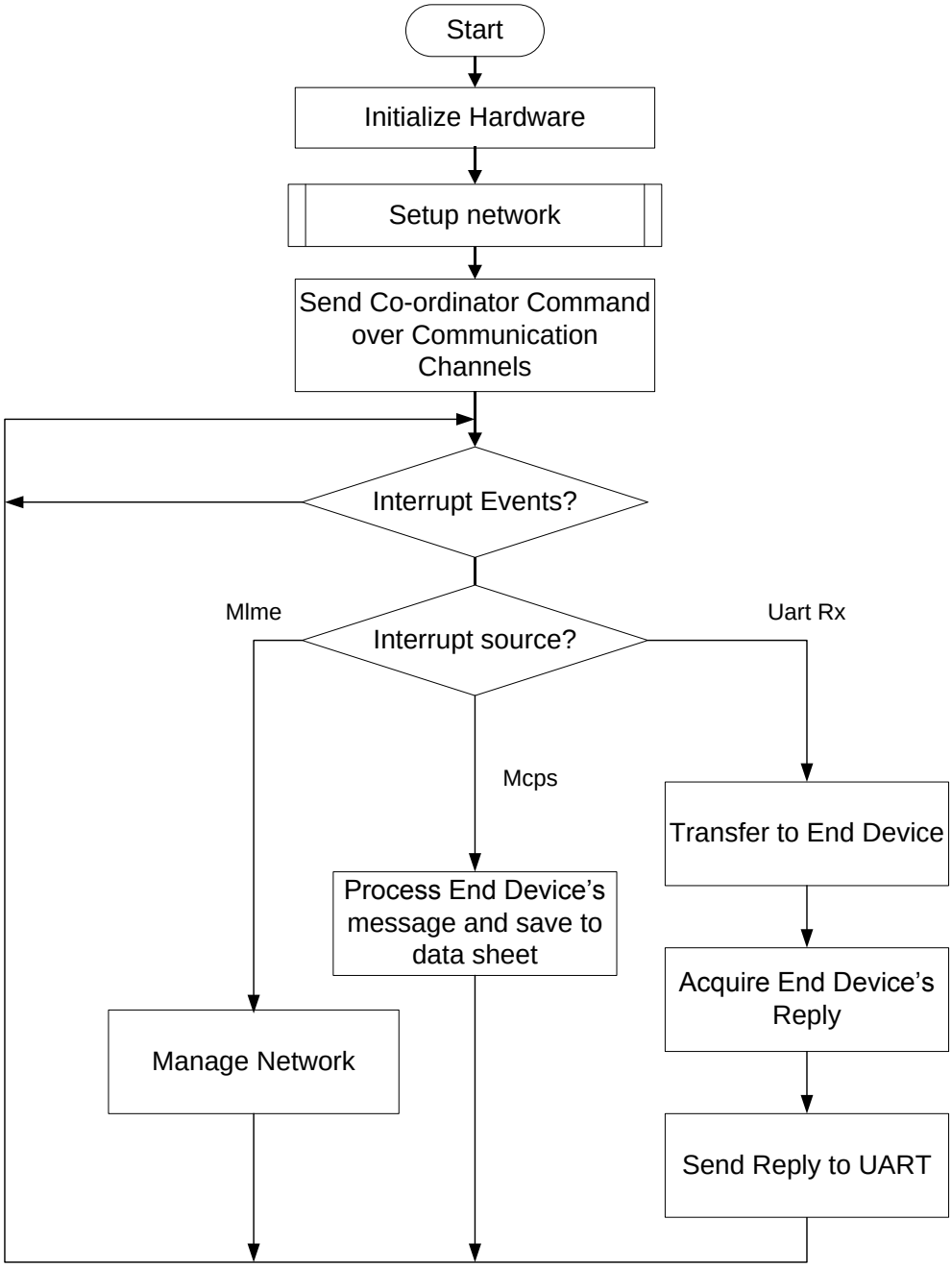


Figure 3-11: Co-ordinator Flowchart

The End Device software flowchart is illustrated in Figure 3-12.

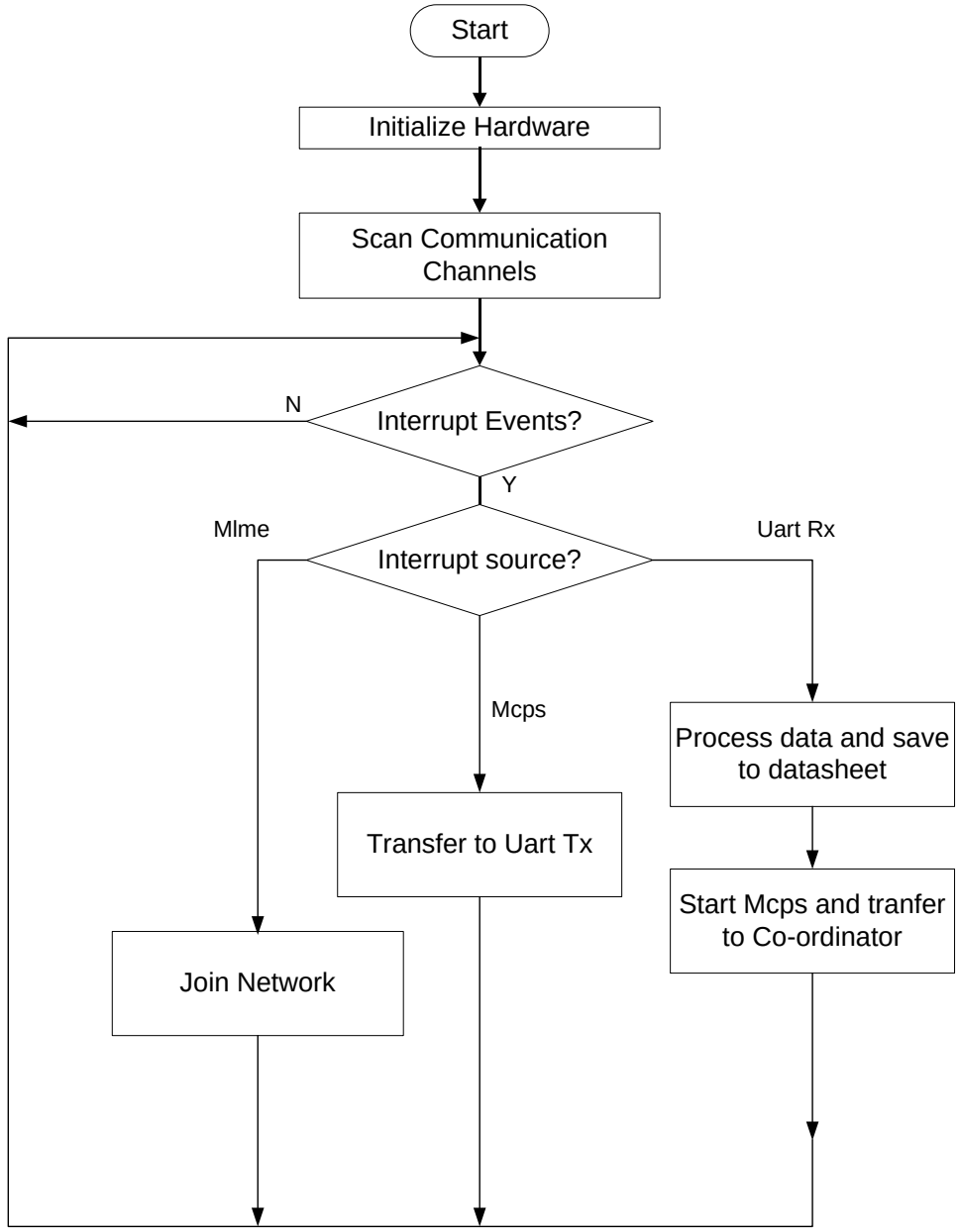


Figure 3-12: End Device Flowchart

Signal strength for a Jennic ZigBee module is from -10dBm to -95dBm [7]. Jennic provides a data structure with one data value for the Received Signal Strength Indicator (RSSI) with a range from 255 to 0 corresponding to a signal strength of -10dBm to -95dBm with a linear relationship. The signal strength can be read from the RSSI and then scaled to a dBm value. A display is added on the

Co-ordinator board, and it is able to show the signal strength for up to four End Devices.

3.2.3 Communication Protocol

To decrease communication traffic, a simple data packet is used in the system. A single control data byte is used for sending and receiving information and a byte is used for checking the communication network setup. There is the ability to add more control bytes to the system if needed. The detailed definition of the control byte for step control motors and reading sensors is listed in Figure 3-13 below. The control byte is used to set motors direction, PWM duty cycle and running time.

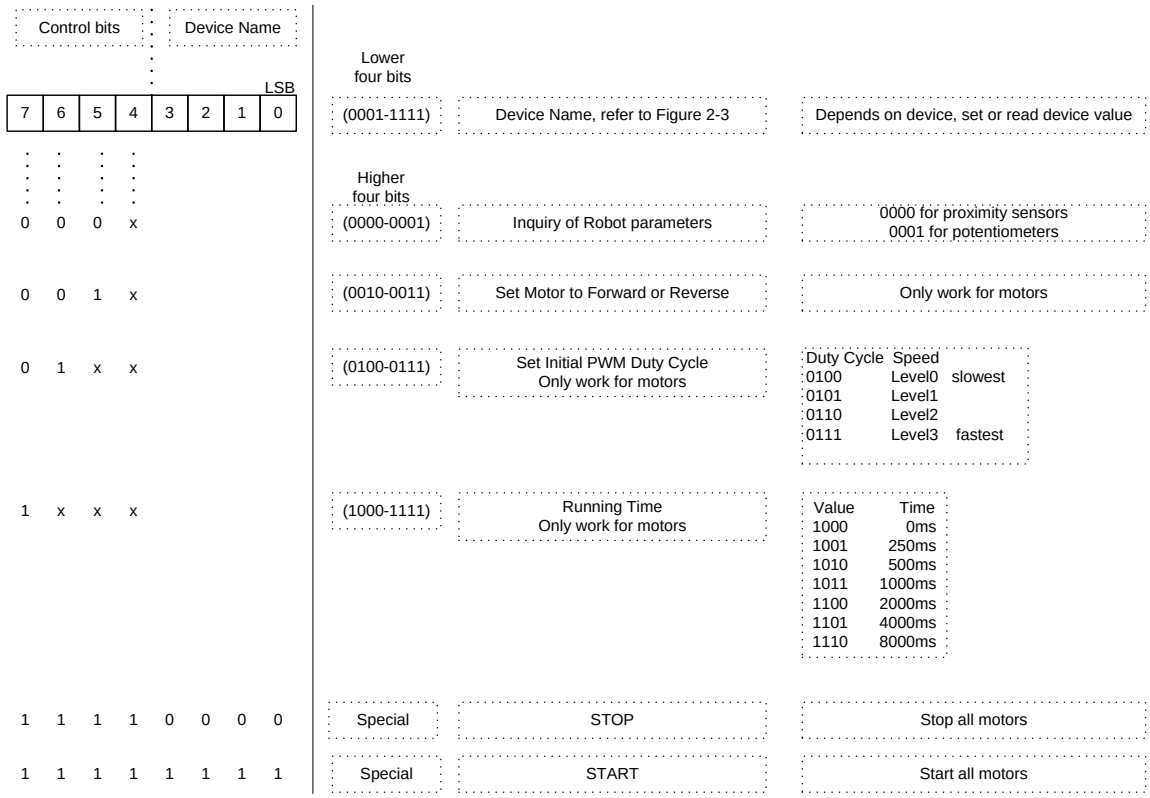


Figure 3-13: Communication Protocol Definition

The special event control byte was defined so that the motors could run continuously, or communicate commands, calibration, preload etc. to the timing belt robot. The special events control byte definition is listed in Figure 3-14.

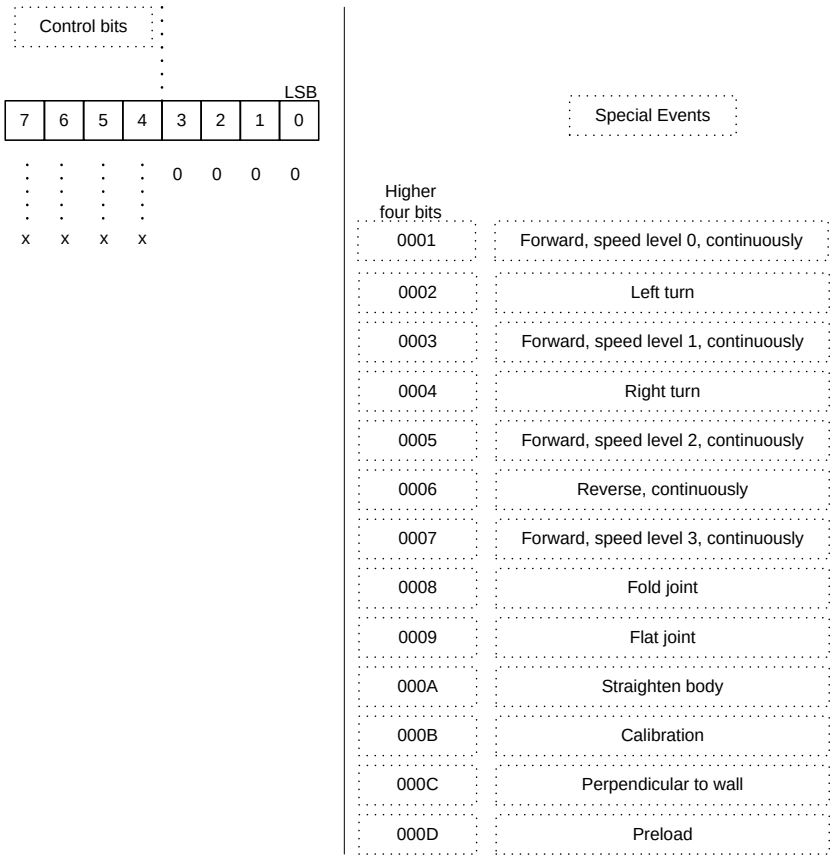


Figure 3-14: Special Events Control Byte Definition

3.3 Control Algorithm Design

The front module of the timing belt robot can turn a maximum of 45 degrees in respect to the rear module because of the horizontal joint. Figure 3-15 illustrates the movement of the joint. The timing belt motors were initially designed to turn in only one direction, leaving the robot with limited mobility. When steering, only timing belts on one side would turn while the timing belts on the other side stopped. This meant the robot could easily get stuck and because of this, a reverse motor control was added in order to for the robot to turn smoothly in any direction.

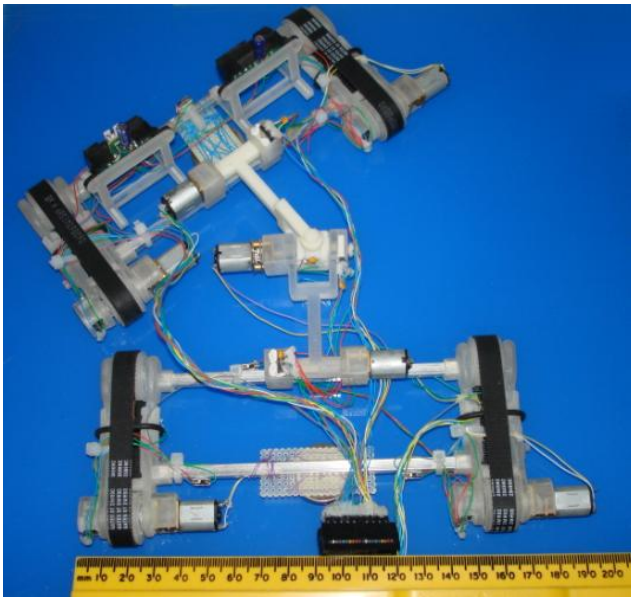


Figure 3-15: Maximum Angle of Steering

3.3.1 Front Module and Rear Module Parallel Control

When the timing belt robot is moving in the forward direction, one track may sometimes rotate slower than the others even when all the PWMs are set to the same duty cycle. If this condition occurs, the timing belt robot begins to twist and turn involuntarily. In order to combat this tendency, there is a single potentiometer mounted in the middle of the turning joint to monitor the steering angle. When the front module of the timing belt robot is parallel to the rear module, the potentiometer output voltage is 2.46V. The output voltage increases to 2.74V when the front module is 45 degree to the left as shown in Figure 3-15. The output voltage decreases to 2.10V when the front module is at an angle of 45 degree to the right.

To control the timing belt robot velocity, only the speed of the two front motors is set. A PID controller was added to control the speed of the two rear motor in order to maintain a parallel orientation when required. The centre

reference point in the PID controller code is the middle joint steering potentiometer output voltage of 2.46V, and the output from the PID controller is a PWM to individually control the rear two motor speeds.

With the timing belt robot initial position set at the maximum steering angle of 45 degrees to the left, as shown in Figure 3-15, the Move Forward command was sent to the robot, and the corresponding waveform was recorded and is displayed in Figure 3-16. It can be seen that the rear right module motor PWM duty cycle (in purple) is much greater than that of the left motors PWM duty cycle (in blue). This corresponds to the left motor turning at a greater speed than the right motor in order to straighten the robot's body.

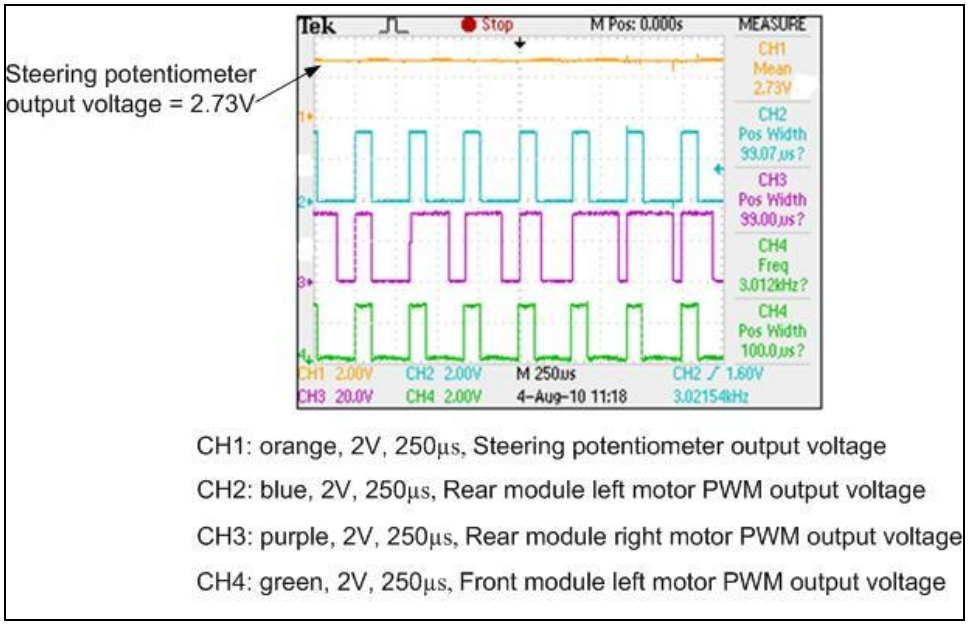


Figure 3-16: Measured Motor PWM Waveform at Straighten When Front Module is 45 Degree to the Left

3.3.2 Timing Belt Robot Turning Control

The robot turn control was designed to enable the robot to turn 360 degrees continuously with a small turning radius. The rear module is controlled

so that it remains parallel to the front module at all times. This enables the robot to readily adjust itself to be perpendicular to a wall and also to be able to travel in a straight line.

In the first design, all the motors on the robot tracks were able to move only in one direction. The front timing belts are able to steer a maximum of 45 degrees to the left or right, so the turning radius, r, is approximately 30cm from the rotational centre to the inner tracks. When the robot detects a wall, the distance must be large enough to allow the robot to adjusting itself to be perpendicular to the Wall. If too close to a wall or obstacle, the robot has to reverse direction in order to create the space needed to orient itself perpendicular to the wall. Figure 3-17 illustrates the robot turning.

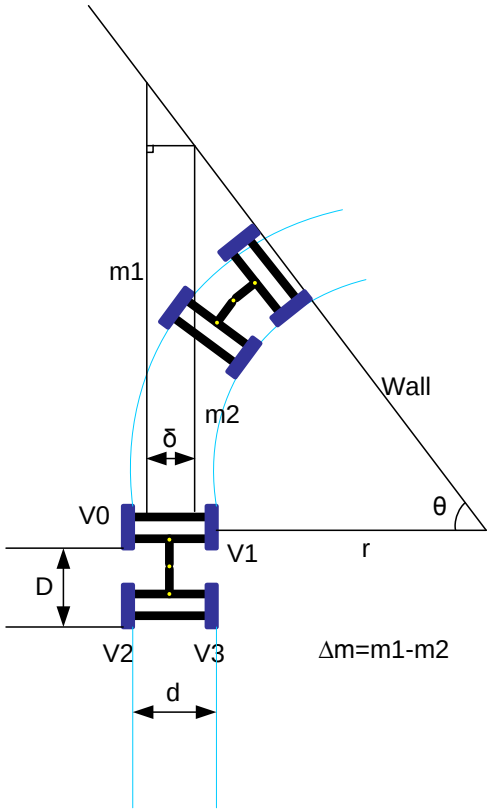


Figure 3-17: Robot Adjusts itself to be Perpendicular to a Wall

After motor reverse control was added and the turning control algorithm was improved, the robot was then able to turn 360 degrees on the spot. This was achieved by turning only one front motor, while the other front motor was stopped. The PID controller is then used to adjust the rear module so it is parallel to front module while the robot is turning. Due to the motor maximum rotation speed limitation, keeping only one front motor rotating and the other stopped allows the rear module to follow the front module. If one front motor rotates forward and the other rotates in the reverse direction, the rear module is unable to keep up and the robot body becomes twisted. A diagram of the robot turning left 90 degree is displayed in Figure 3-18. The turning centre is shown as a red dot and the turning trajectory is also shown.

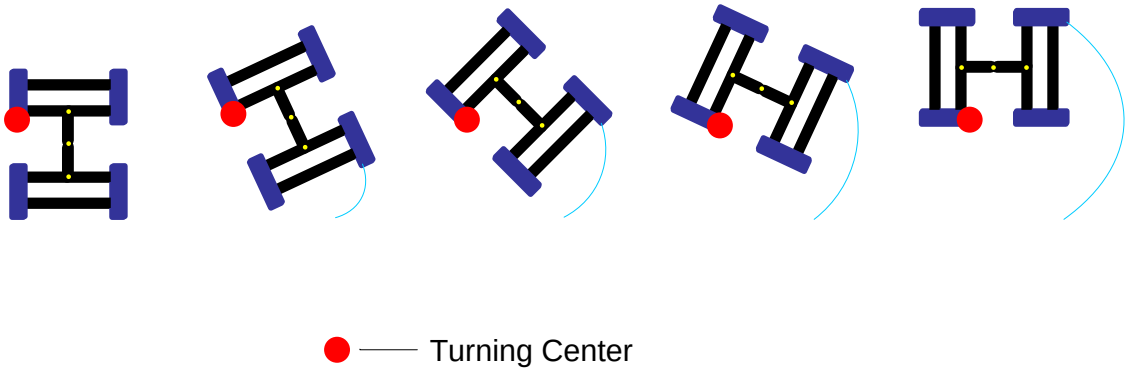


Figure 3-18: Robot Turning Left 90 Degree Illustration

The waveform was measured with an oscilloscope while the robot was turning left and is shown in Figure 3-19. From Figure 3-19, it can be seen that the left motor of the front module was stopped, and that the right motor was turning at a predetermined speed. The left motor on the rear module was turning in the reverse direction, which is shown on channel 3 (in purple). At the same time, the

right motor on the rear module was turning forward (channel 4 in green), with greater speed than the right motor on the front module because the rear right track was turning with a larger radius. This can be seen in Figure 3-19 as the green waveform has a greater PWM duty cycle than the blue one and both control motors rotating in the forward direction.

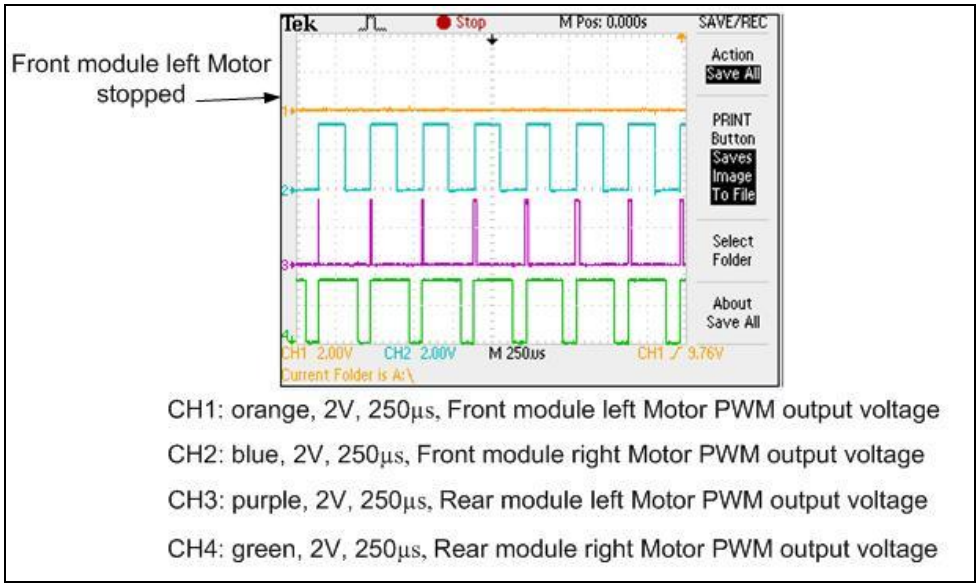


Figure 3-19: Waveform for Robot Turning Left

3.3.3 Perpendicular to Wall Adjustment Control

The two Sharp IR sensors were used to detect walls and obstacles. If the distance between the robot and a wall is within 80cm, the Sharp sensor output voltage will be greater than 0.9V. The algorithm was designed so the robot will start to adjust itself so that it can be perpendicular to a wall when the distance from the wall is approximately 40cm. This corresponds to a sensor output voltage of approximately 1.19V. The technique used to detect whether or not the robot is perpendicular to the wall simply checks if both Sharp sensors output the

same voltage. When both sensors output the same voltage, the robot is perpendicular to the wall.

A PID controller is used to control the timing belt robot's front timing belts. The reference used by the controller is when there is no difference between the two Sharp sensor output voltages. The controller outputs two PWMs to each of the robot's two front timing belt motors. A higher output voltage from the left Sharp sensor in comparison to the right sensor corresponds to the robot's left side being closer to the wall. The PWM duty cycle of the right motor should be increased in order to increase the speed of the right motor, while the PWM duty cycle of the left motor should be decreased. See Figure 3-20 for a diagram.

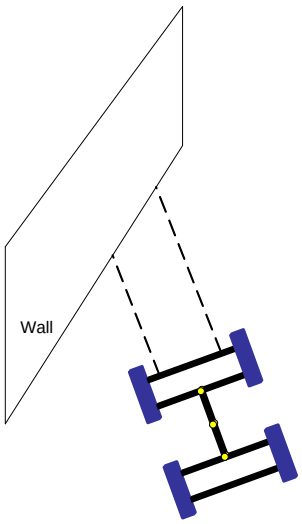
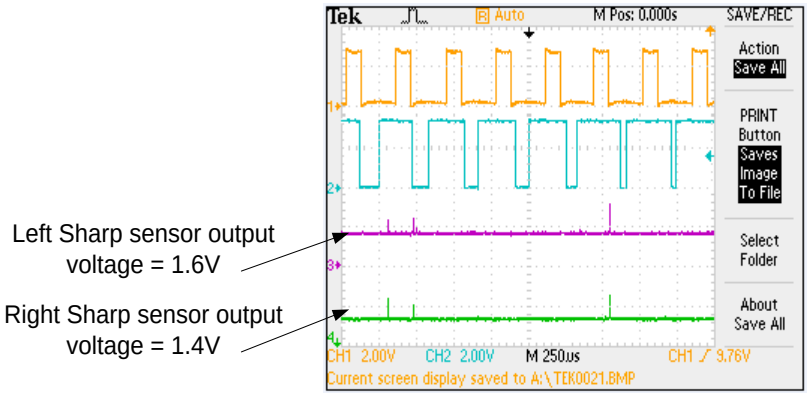


Figure 3-20: Timing Belt Robot Perpendicular to Wall Detection

An oscilloscope was used to measure the waveform as is shown in Figure 3-21. From the graph, we can see that the left Sharp sensor output voltage is greater than the right Sharp sensor output voltage. This corresponds to the robot's left side being closer to the wall as illustrated in Figure 3-20 above.

Channel one, corresponding to the left motor of the front module, shows that the PWM duty cycle is reduced while channel two corresponds to the right motor of the front module and the PWM duty cycle is increased to increase the motor speed. As can be seen in the figure, the PWM duty cycle of channel one (in orange) is less than the PWM duty cycle of channel two (in blue), indicating that the right motor rotates at a higher speed.



CH1: orange, 2V, 250µs, Front module left Motor PWM output voltage

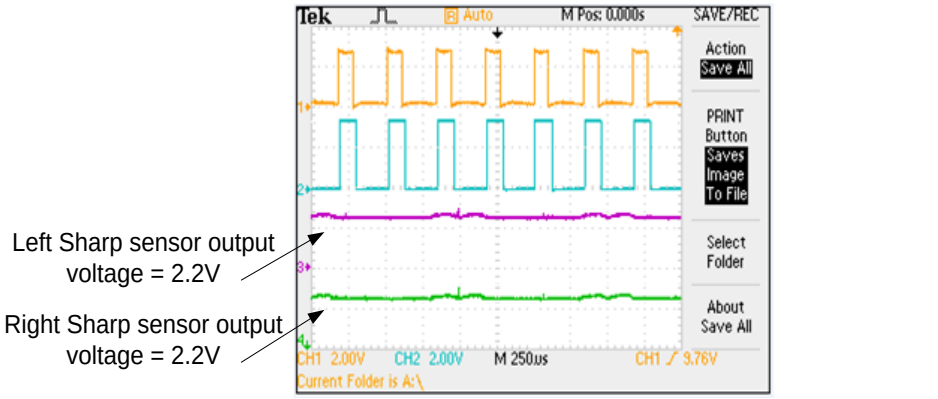
CH2: blue, 2V, 250µs, Front module right Motor PWM output voltage

CH3: purple, 2V, 250µs, Left Sharp sensor output voltage

CH4: green, 2V, 250µs, Right Sharp sensor output voltage

Figure 3-21: Perpendicular Adjustment Waveform When Robot's Left Side is Closer to Wall

Figure 3-22 shows the measured waveform after the robot had completed the adjustments necessary to adjust itself to be perpendicular to a wall. In the captured waveform, both Sharp sensors output the same voltage, 2.2V, indicating that the timing belt robot body was perpendicular to the wall. The two front motors are shown to both be running at the same speed since the PWM duty cycles for channel one (in orange) and channel two (in blue) are identical.



CH1: orange, 2V, 250µs, Front module left Motor PWM output voltage

CH2: blue, 2V, 250µs, Front module right Motor PWM output voltage

CH3: purple, 2V, 250µs, Left Sharp sensor output voltage

CH4: green, 2V, 250µs, Right Sharp sensor output voltage

Figure 3-22: Waveform after Perpendicular Adjustment

3.3.4 Horizontal to Vertical Surface Transition

The timing belt robot incorporates three rotary joints in its design. These joints are capable of folding and unfolding to a specified angle and each joint is monitored by a rotary potentiometer. Each potentiometer output voltage value is measured by the ADC portion of the microcontroller, and PID controllers are used to control the joint motors so that the joint can fold or unfold to a specific angle.

When the robot detects a wall, it first adjusts itself to be perpendicular to the wall. The robot then starts to fold its middle joint as it approaches the wall. When the Sharp sensor's output voltage is 3.3V indicating the distance between robot and wall is 7cm, the robot will stop the movement of the two front tracks and fold the joint to approximately 90 degrees. The robot then moves forward towards the wall, and starts to climb by lifting the front module when the proximity

sensor output indicates that the front module is within 1cm of the wall. The step by step transition from a horizontal to vertical surface is illustrated in Figure 3-23. The six proximity sensors, which are facing the surface the robot is in contact with, are used to determine the distance between the robot body and the surface underneath.

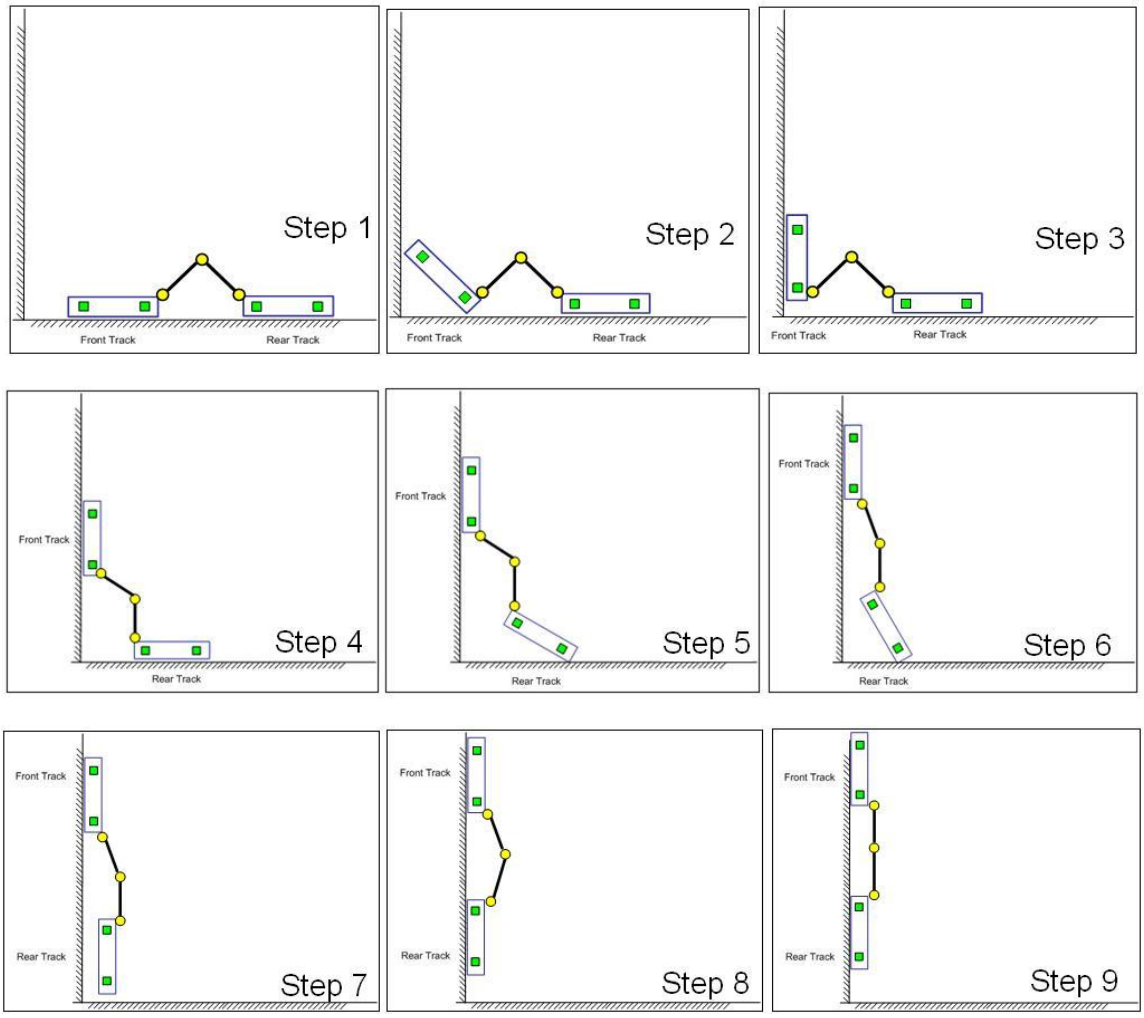


Figure 3-23: Steps for Horizontal to Vertical Surface Transition

3.4 Control Software Design of the Climbing Timing Belt Robot

3.4.1 PID Controller and Implementation

PID stands for proportional, integral and derivative which are the three terms of input adjustments. The PID controller ideal parallel form is written as [24]:

$$\mathrm{MV(t)} = K_p e(t) + K_i \int_0^t e(\tau)\,d\tau + K_d \frac{de}{dt}(t) \quad \text{.....(3-1)}$$

where K_p is the proportional gain, K_i is the integral gain and K_d is the derivative gain. K_p, K_i and K_d are the three tuning parameters used to decide the response to system disturbances.

The block diagram of a PID controller used to control motor speed in this project is shown in Figure 3-24.

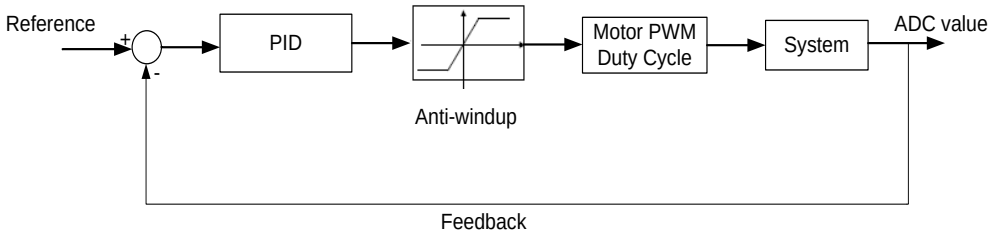


Figure 3-24: BLock Diagram of a PID Controller Used to Control Motor Speed

The discrete PID controller is implemented in C programming language in theTMS320F2808 as shown in the following Figure 3-25. This PID controller can be called multiple times by changing the PID parameters.

```
//Proportional gain calculation
Up = ( ( PidCtrl->Error * (int32)PidCtrl->Kp )
        >> PidCtrl->K );

//Integral gain calculation
Ui = ( ( PidCtrl->ErrorAccum * ( int32 )( PidCtrl->Ki ) )
      >> PidCtrl->K );

//Derivative gain calculation
Ud = ( ( ( int32 )PidCtrl->ErrorDelta * ( int32 )( PidCtrl->Kd ) )
      >> PidCtrl->K );

//Calculate the new PID controller output
PidCtrl->U = Up + Ui + Ud;

//Clamp the output value at its specified max or min
if( PidCtrl->U > PidCtrl->Umax )
{
    PidCtrl->U = PidCtrl->Umax;
}
else if( PidCtrl->U < PidCtrl->Umin )
{
    PidCtrl->U = PidCtrl->Umin;
}
```

Figure 3-25: Discrete PID Controller in Robot Control System

3.4.2 PID Terms and Their Effects

The proportional term, K_p, makes a change to the output that is proportional to the error between a set point and the output. If the proportional gain is too high, the system can become unstable and begin oscillating. If the proportional gain is too small, the system cannot follow the change of the control object and the response will be too slow.

The contribution from the integral term is proportional to both the magnitude of the error and the duration of the error. The integral term accelerates the movement of the process towards a set point and eliminates the residual steady-state error that occurs with a proportional only controller. However, the integral term can make the system overshoot the desired output because it accumulates past errors.

The derivative term adjusts the feedback according to the rate of error change and is useful to prevent overshooting the target. The derivative term slows the rate of change of the controller output.

Integrator windup occurs if the control error is very large. When the error is very large, the joint motor will become saturated, and the joint will become twisted. Normally the integrator term is a small number but at integrator windup, it can become very large. When the error is reduced, it can take a considerable amount of time for the integrator term to reduce to a normal value of less than 6 when the feedback control begins functioning.

One way to avoid integrator windup is to multiply a variable, Limit_Switch, to the integral updating. When the actuator is saturated, this variable is zero, and the integral is no longer updated. Another method is to feed the error between the actuator output and the controller output back to the integrator with a gain of 1/Ta.

3.4.3 PID Parameters Tuning Software

In order to tune the PID parameters to obtain more suitable values, LabVIEW software was developed. Using switches on the Graphical User Interface (GUI) screen, the integral and derivative terms may be switched either on or off and the parameters can be easily changed. The curves displayed in Figure 3-26 below are the screenshots from real time controlling. The software shows the different effects of each PID terms as described above. By using the software, suitable parameters for each term can be found.

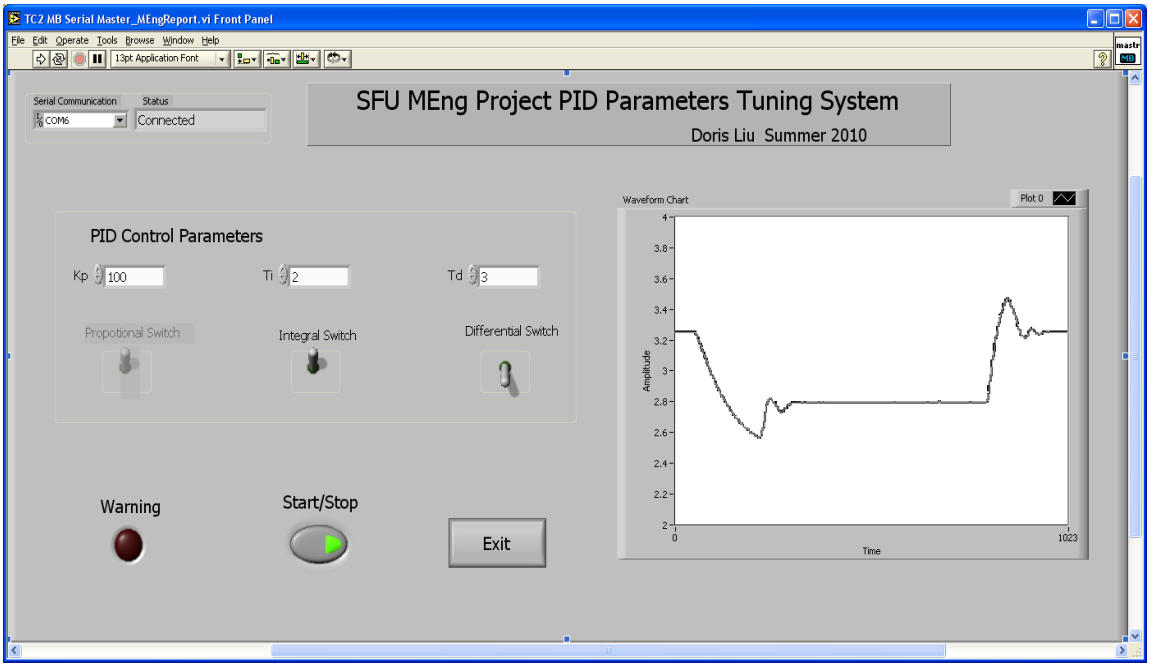


Figure 3-26: PID Parameters Tuning Software

A set of Block Diagrams was created for the LabVIEW interface to communicate with the robot microcontroller. Upon start-up, the communication port is initialized and it communicates with the microcontroller in order to obtain the current value of the PID control parameters. When one of the three PID coefficients is changed by user, an interrupt is generated and the new parameters are sent to the microcontroller on the control board.

3.5 Robot Control LabVIEW Graphical User Interface (GUI) Design

3.5.1 LabVIEW Interface

LabVIEW graphical programming allows the design of rapid and cost-effective interfaces with measurement and control hardware as well as the ability to quickly analyze data, share results, and distribute systems [25]. There are

many standard libraries available online and the serial port driver is one of them provided by National Instruments (NI).

A user friendly interface used to control the timing belt robot was developed using the integrated graphics, control logic and communication protocol. The interface is shown in Figure 3-27.

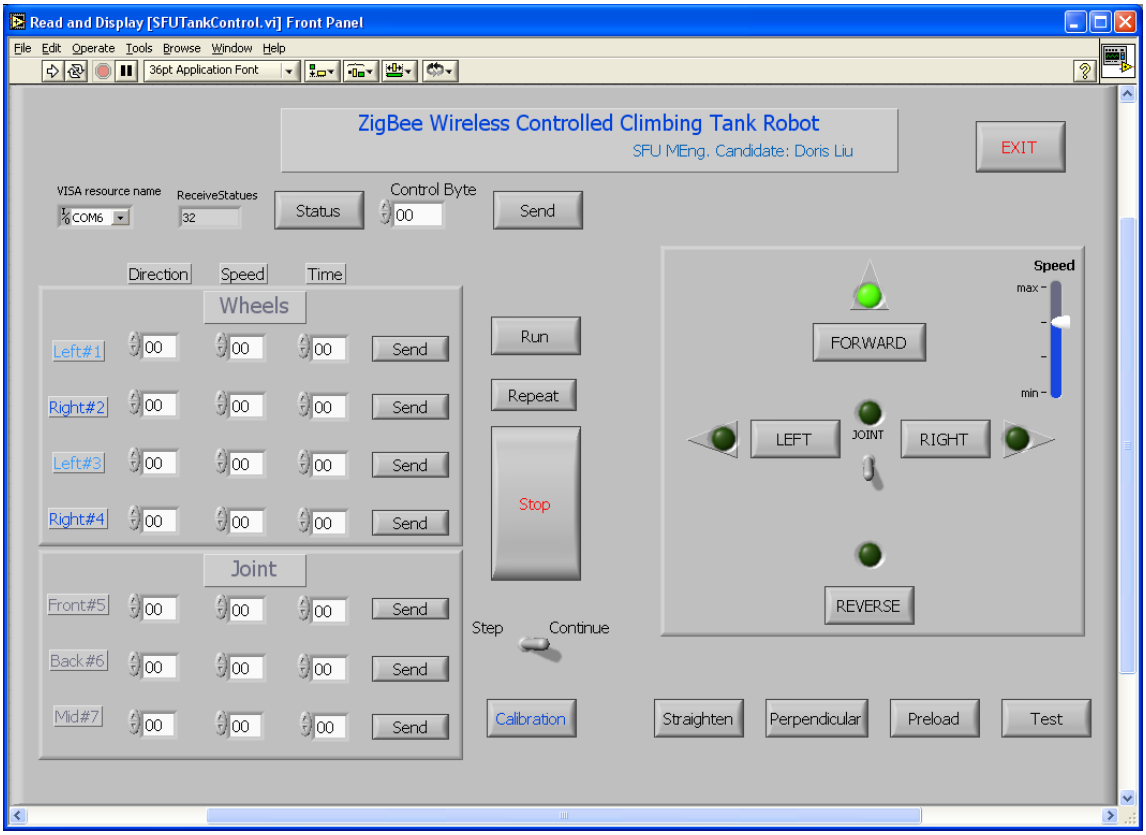


Figure 3-27: LabVIEW GUI Screenshot

In this Graphical User Interface (GUI), each of the seven motors may be controlled individually. Each motor can be controlled by three simple parameters which are direction, speed and time duration of rotation. After sending the command to run a motor for a single step, the motor turns in the specified direction for the specified speed and time duration. A direction command of '0'

instructs the specified timing belt to turn forwards or the specified joint to move upwards. A direction command of ‘1’ instructs the motor to turn in reverse or to move the joint rod downwards. The speed commands are from zero to three so the speed may be controlled at four different levels. A speed command of zero specifies the slowest speed while three specifies the fastest speed. The timer runs from zero to eight seconds, and is divided into seven levels. A command of zero means that the timer won’t run and the motor will not rotate. A command of six means timer will run continuously for eight seconds. After pressing the Send button, the command is sent to the TMS320F2808 MCU. When the Run button is pressed, the motor begins turning. The red Stop button has highest priority, and it is able to stop any motor at any time.

The timing belt robot can also be controlled in continuous mode. When the FORWARD button is pressed, the timing belt robot will move forwards. The speed can be controlled by using the slide bar beside the FORWARD button. This timing belt robot is currently designed for just one turn speed. When the FORWARD button is pressed, the robot runs continuously until it detects a wall. It then adjusts itself to be perpendicular to the wall, and warning buzzer will sound. There are several other buttons to control the robot, and they are LEFT, RIGHT, REVERSE, JOINT up/down, etc. The Calibration button is used to initialize a calibration of the sensors.

3.5.2 LabVIEW Control Logics

A set of control logic was created for the LabVIEW interface to communicate with the ZigBee Co-ordinator. Once the program starts, it initializes the serial communication port and variables, and then communicates with the ZigBee Co-ordinator via serial communication. After these initializations, the program remains in a while loop until the user presses a button. If one of the button states is changed, an interrupt is generated. The computer then goes to that interrupt service routine, and sends the new parameters or commands to the ZigBee module on the Co-ordinator board. The Co-ordinator then passes these commands on to the End Device through wireless communication. The control logic flowchart is shown in Figure 3-28.

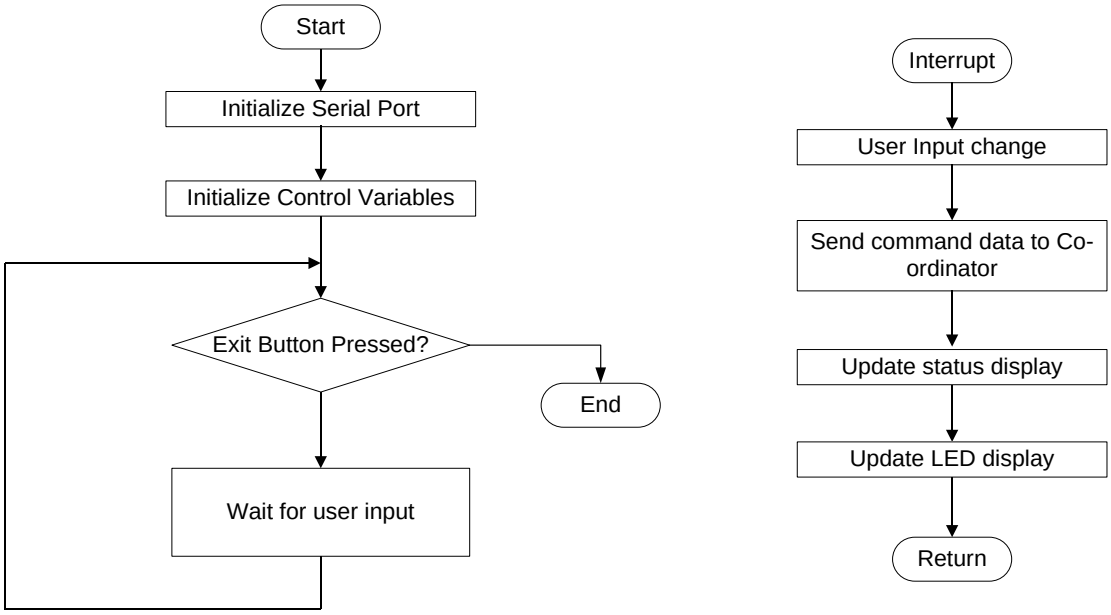


Figure 3-28: LabVIEW GUI Flowchart

Figure 3-29 is the LabVIEW block diagram for one interrupt service routine.

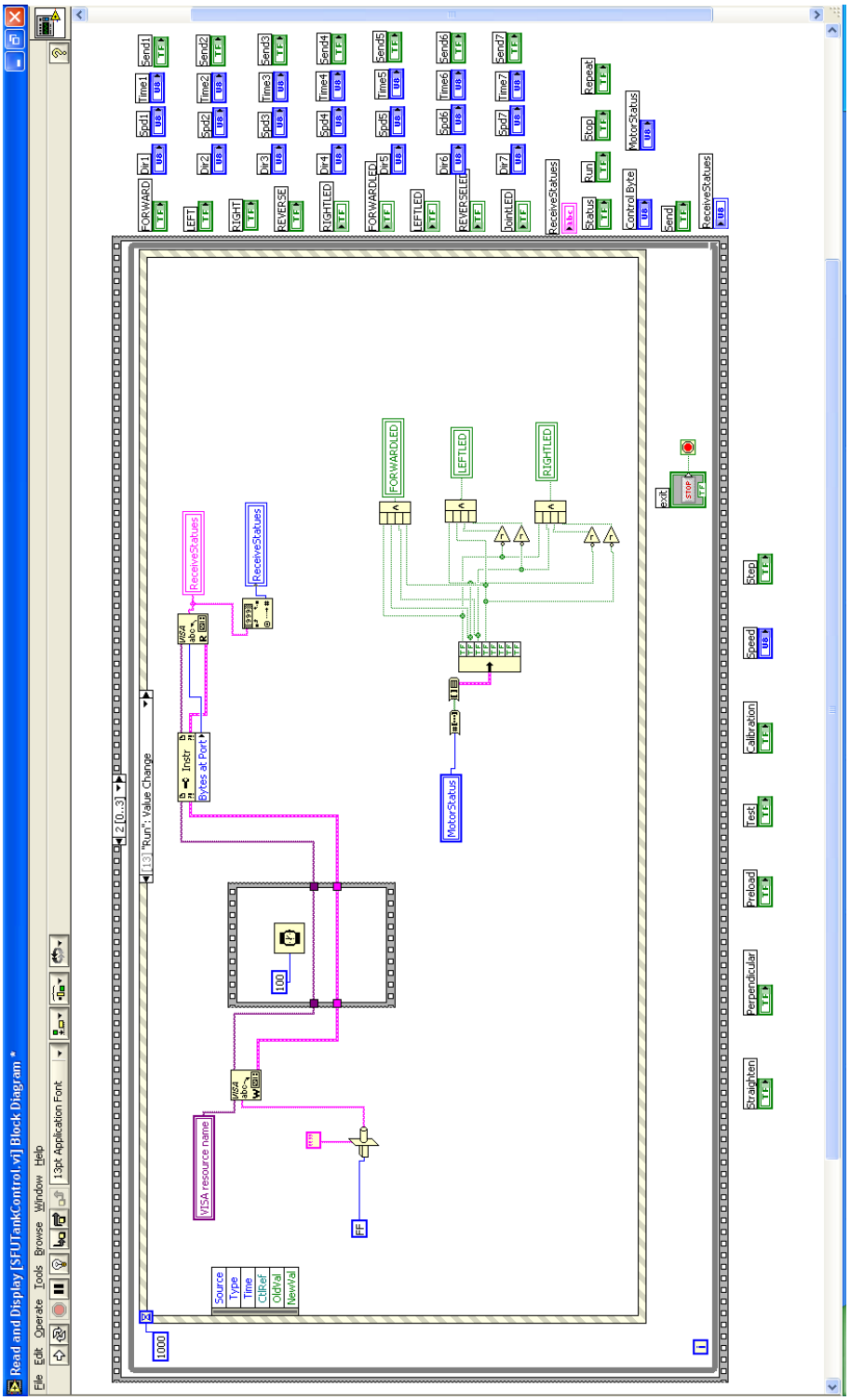


Figure 3-29: LabVIEW Block Diagram

4: SIGNAL STRENGTH MEASUREMENT FOR WIRELESS SYSTEM

The PC software is designed to read and display the signal strength from different End Devices. The software was developed in LabVIEW. Although LabVIEW can only display one signal with an historical graph, this problem was solved by writing measurement data to an Excel file. When a user clicks on the OK button, current measurement values for each device are written to an Excel file. The advantage of this is that it not only makes the measurements more convenient to save due to automatic recording, but it also allows the user to analyze the data and model attenuation in an Excel file. The measurement software PC User Interface is shown in Figure 4-1.

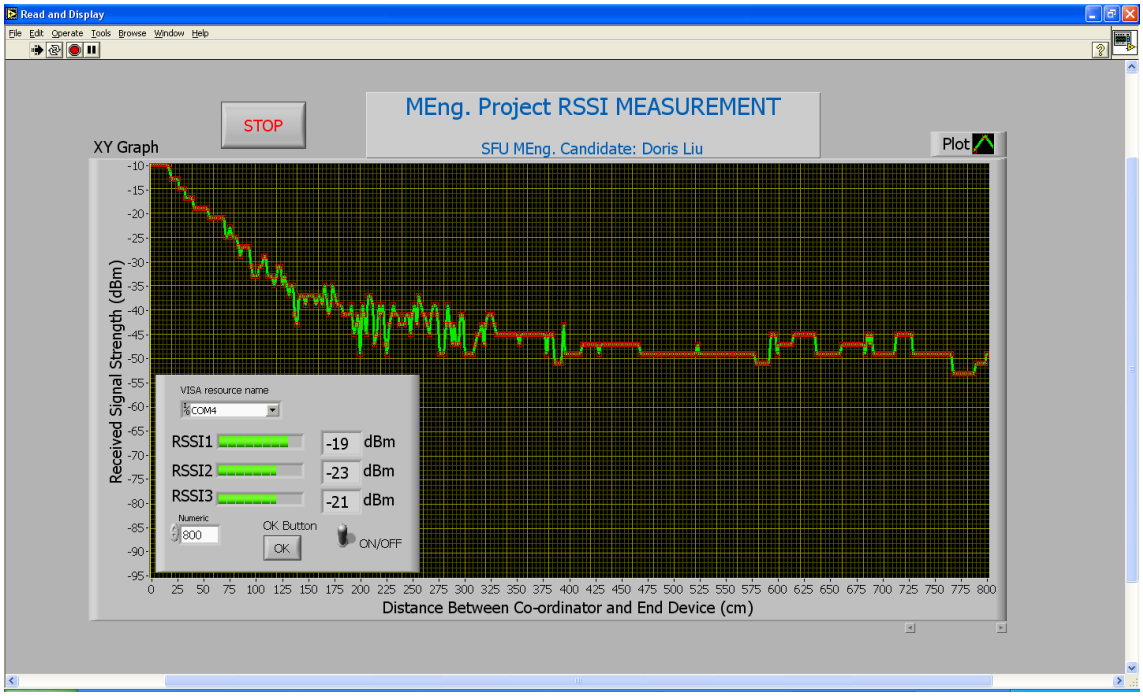


Figure 4-1: RSSI Measurement Software User Interface

By measuring the signal strength and modelling the path loss, we are able to estimate the mean signal strength between the transmitters and receivers based on the distance between them. While free space propagation dictates that a signal attenuates as a function of the square of the distance, measurements have shown that the signal strength is greatly dependent on the environment in practice.

The measurement was performed in the ASB building at Simon Fraser University. Every time the distance between the Co-ordinator and End Device board is increased by 2cm a single measurement of signal strength was made. For the setup seen in Figure 4-2, the three End Device antenna distances are the same. Two of theEnd Devices are SMA modules and the other End Device module is a SMD module which has a built-in ceramic antenna, and is suitable to be embedded into a small climbing timing belt robot. The measured waveforms are shown in Figure 4-3. Compared to a SMA antenna module, the measured signal strength of the SMD module was approximately 10 dBm less than that of the SMA antenna module. All three waveforms are shown in Figure 4-3 with the green waveform corresponding to the SMD antenna module.

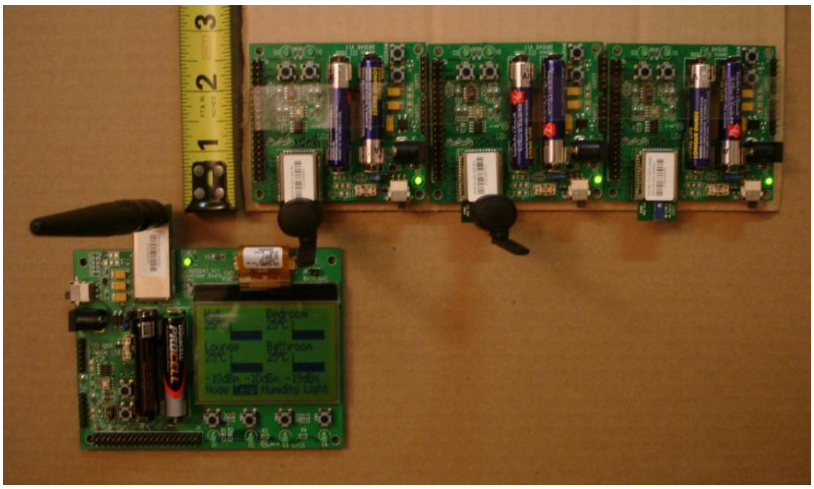


Figure 4-2: Measurement Setup for One SMD and Two SMA Modules

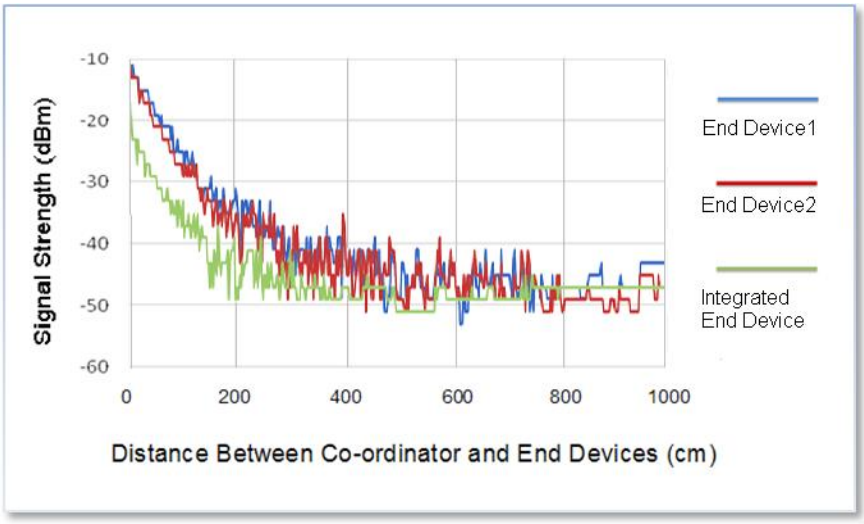


Figure 4-3: Signal Strength for One SMD and Two SMA Modules

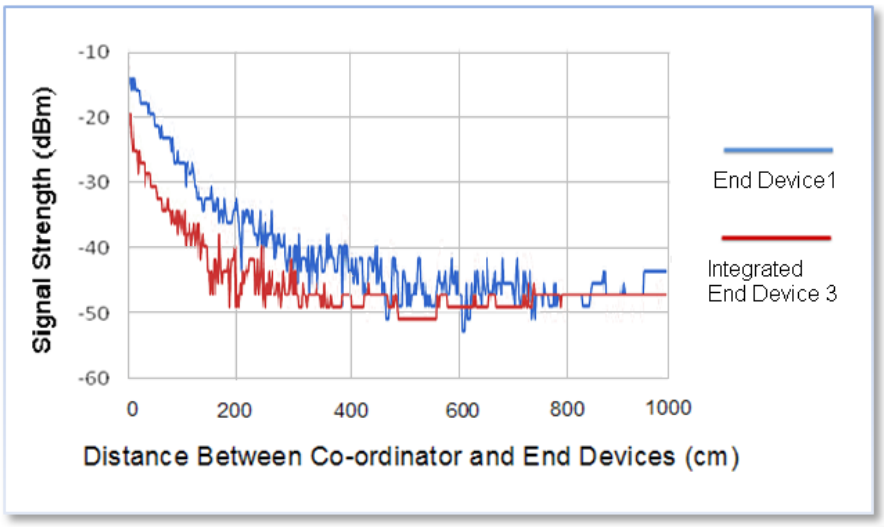


Figure 4-4: Signal Strength Comparison of SMD Module with SMA Module

In order to predict the received signal strength between the Co-ordinator and the End Devices, a signal strength attenuation model is used. In Figure 4-5, the received power is in dBm, and the distance is in meters. The path loss can be calculated using the following formula [13]:

$$L = 10\ n\ \log_{10}(d) + C \quad \text{.....(4-1)}$$

where *L* is the path loss in decibels, *n* is the path loss exponent, *d* is the distance between the transmitter and the receiver, usually measured in meters, and *C* is a constant which accounts for system losses [13]. The actual path loss equation found from this measurement is shown in equation 4-2,

$$y = -10 \times 2.3 \times \log(x) - 29 \quad \text{.....(4-2)}$$

where *y* is the path loss in dBm, and *x* is the distance between the transmitter and the receiver in meters. The path loss exponent found from the experiment was 2.3. This result demonstrates that mean signal strength can be

predicted using simple models. However, it is not possible to predict the variations from the multipath which are superimposed onto the mean path loss.

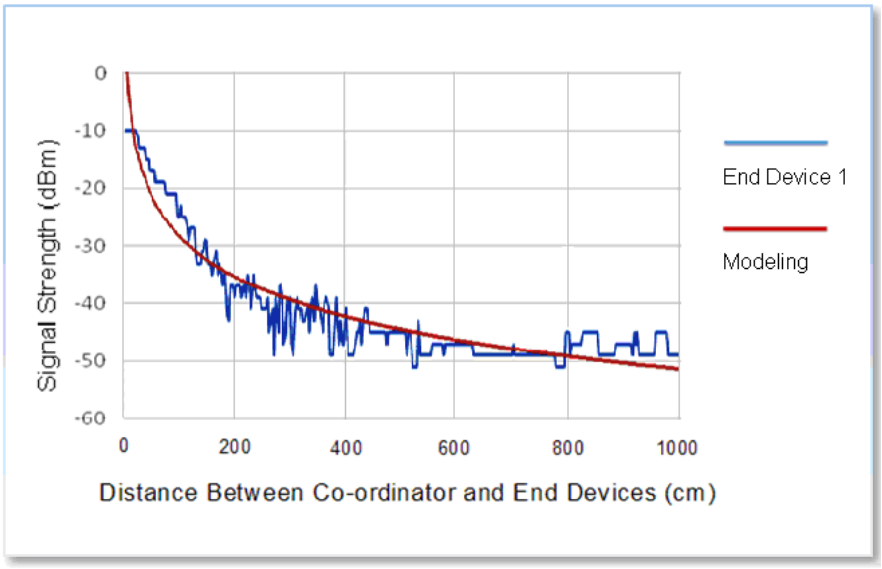


Figure 4-5: Modeling Attenuation for End Device

5: CONCLUSION

A wirelessly controlled climbing timing belt robot was designed and implemented in this M.Eng project. The robot is developed with the potential to transfer from one surface to another under different circumstances. The project began with a microcontroller and wireless communication system selection based on the requirements of the climbing timing belt robot, and progressed with hardware design, prototype implementation, software design and system testing.

A prototype board was built based on the hardware design. All the ICs on the prototype board were placed into sockets, and many connectors were used to provide the system with increased flexibility. The TI 32-bit microcontroller used in this robot control system is very powerful, and runs at 100MHz. This allows the system to have the ability to add more complex functionalities in the future.

This timing belt robot is controlled by a wireless communication system based on ZigBee, and it allows users to access information and control the robot remotely. It also has the capability to control several robots at the same time. For a wirelessly controlled climbing robot, wireless communication increases the system flexibility. ZigBee is used in the system due to the low cost of a ZigBee network system. With many Application Programming Interfaces (API) functions provided by the manufacture, and many more application notes online, it is possible to develop an application faster and achieve effective IEEE 802.15.4 applications rapidly.

This robot control system integrates many topics including microcontrollers, wireless communication systems, both digital and analog hardware, real time operation, control algorithms design, and programming languages for both the microcontrollers and the computer user interface. The robot was verified that it can follow commands received via the user input. Future work includes improving the performance of the climbing robot and the addition of dry adhesive timing belts to enable the robot to climb smooth surfaces.

6: FUTURE WORK

The hardware system has been verified to work, so a PCB can now be designed and surface mount components can be used. The final PCB can be designed to be very small in size and light in weight making it suitable for a climbing robot.

Several methods were attempted to enable the climbing of smooth vertical surfaces using dry adhesive timing belts. In the near future, this method of traction and locomotion could be realized.

A Star Topology was used in the ZigBee wireless communication system and because of this it is possible to use one computer to control several climbing robots at the same time while allowing the robots to communicate with each other.

An inclinometer is an instrument used to measure the angle of a slope, and an inclinometer may be added to the robot to detect the angle of the slope while climbing in order to enable the robot to transfer between surfaces that intercept at any slope.

REFERENCE

[1] NASA Mars Rover 2009, URL http://www.expo21xx.com/automation21xx/14901_st3_university/default.htm

[2] A. Sadeghi, “ENSC 494 Special Projects Laboratory - Direct Study, Mattoid2 – Tank Climber”, Figure 4, pp.7, 2009.

[3] Sharp GP2Y0A21YK, URL http://search.digikey.com/scripts/DkSearch/dksus.dll?Detail&name=425-2063-ND

[4] Avago HSDL-9100 datasheet, URL http://www.avagotech.com/docs/AV02-2259EN

[5] D. Liu, “Bluetooth, ZigBee, Wi-Fi and WiMax for Wireless Communications”, 2008.

[6] JN5139 Wireless Modules (IEEE802.15.4 and JenNet), URL http://www.jennic.com/products/modules/jn5139_modules

[7] Data Sheet – JN5139-xxx-Myy, IEEE802.15.4/ZigBee Module Family, URL http://www.jennic.com/files/support_files/JN-DS-JN5139MO-1v5.pdf

[8] JN5139-EK020 802.15.4/JenNet Starter Kit User Guide, URL http://www.jennic.com/files/support_files/JN-UG-3040-JN5139-EK020-User-Guide-1v1.pdf

[9] TMS320F2808 Digital Signal Processors Data Manual, URL http://focus.ti.com/lit/ds/symlink/tms320f2808.pdf

[10] TMS320F2808 Experimenter Kit - TMDSDOCK2808, URLhttp://focus.ti.com/graphics/tool/f2808_usb_dock.jpg

[11] IEEE 802.15.4 protocol stack.svg, URL http://en.wikipedia.org/wiki/File:IEEE_802.15.4_protocol_stack.svg

[12] Wheeler, Andy. “Embedded ZigBee Design Considerations”, URL http://www.wirelessdesignmag.com/ShowPR.aspx?PUBCODE=055&ACCT=0030563&ISSUE=0512&RELTYPE=PR&PRODCODE=00000&PRODLETT=A&CommonCount=0

[13] Path loss, URL http://en.wikipedia.org/wiki/Pathloss

[14] ZigBee - Do we still need wires?, URL http://allserv.kahosl.be/projecten/rabbit/studienamiddag/pres/rmoni.pdf, Page 19

[15] ISM band, URL http://en.wikipedia.org/wiki/ISM_band

[16] Zigbee, URL http://en.wikipedia.org/wiki/Zigbee

[17] RF power values, URL http://www.cisco.com/en/US/tech/tk722/tk809/technologies_tech_note09186a00800e90fe.shtml

[18] Link budget, URL http://en.wikipedia.org/wiki/Link_budget

[19] JN5139-EK010 ZigBee Evaluation Kit Getting Started, URL http://www.jennic.com/files/support_files/JN-UG-3030-JN5139-EK010-Getting-Started-1v2.pdf

[20] IEEE 802.15.4 Application Development Reference Manual, URL http://www.jennic.com/files/support_files/JN-RM-2024-IEEE802.15.4-App-Dev-2v0.pdf, Figure 1, Figure 2, Figure 4, Page 7

[21] L293D datasheet, URL http://www.datasheetcatalog.org/datasheet/texasinstruments/l293d.pdf

[22] Re: what is the difference between anolog ground and digital ground, URL http://www.epanorama.net/wwwboard/messages/1942.html

[23] Feedback control System, URL http://www.eolss.com/eolss/images/6.43.1.0Figure15.JPG

[24] PID controller, URL http://en.wikipedia.org/wiki/PID_controller

[25] Labview, URL http://www.ni.com/labview/

[26] Inclinometer, URL http://en.wikipedia.org/wiki/Inclinometer

[27] TMS320x280x, 2801x, 2804x Enhanced Pulse Width Modulator (ePWM) Module Reference Guide, URL http://focus.ti.com/general/docs/litabsmultiplefilelist.tsp?literatureNumber=spru791f

[28] TMS320x280x Analog to Digital Converter (ADC) Module Reference Guide, URL http://focus.ti.com/lit/ug/spru716d/spru716d.pdf

[29] TMS320F2808.pdf, URL http://focus.ti.com/lit/ds/symlink/tms320f2808.pdf

[30] W. Wolf, “Computers as Components: Principles of Embedded Computing System Design.” Second Edition. San Francisco, CA: Morgan Kaufmann, 2008.

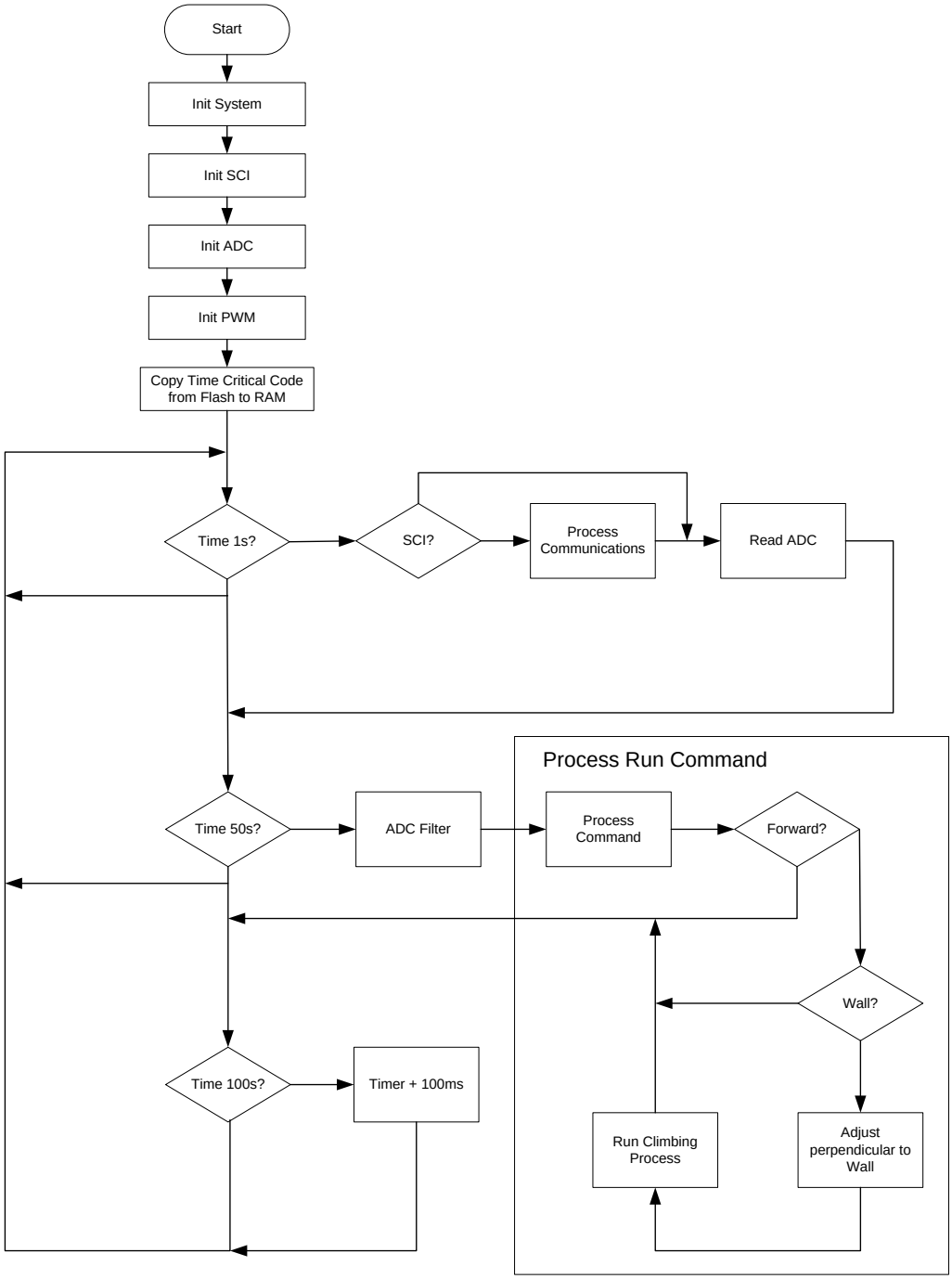
[31] Jenie API User Guide, URL http://www.jennic.com/support/user_guides/jn-ug-3042_jenie_api

[32] D. Liu, “ENSC 891 Directed Studies – Practical Short Range Wireless Communications Module Final Report”, 2008.

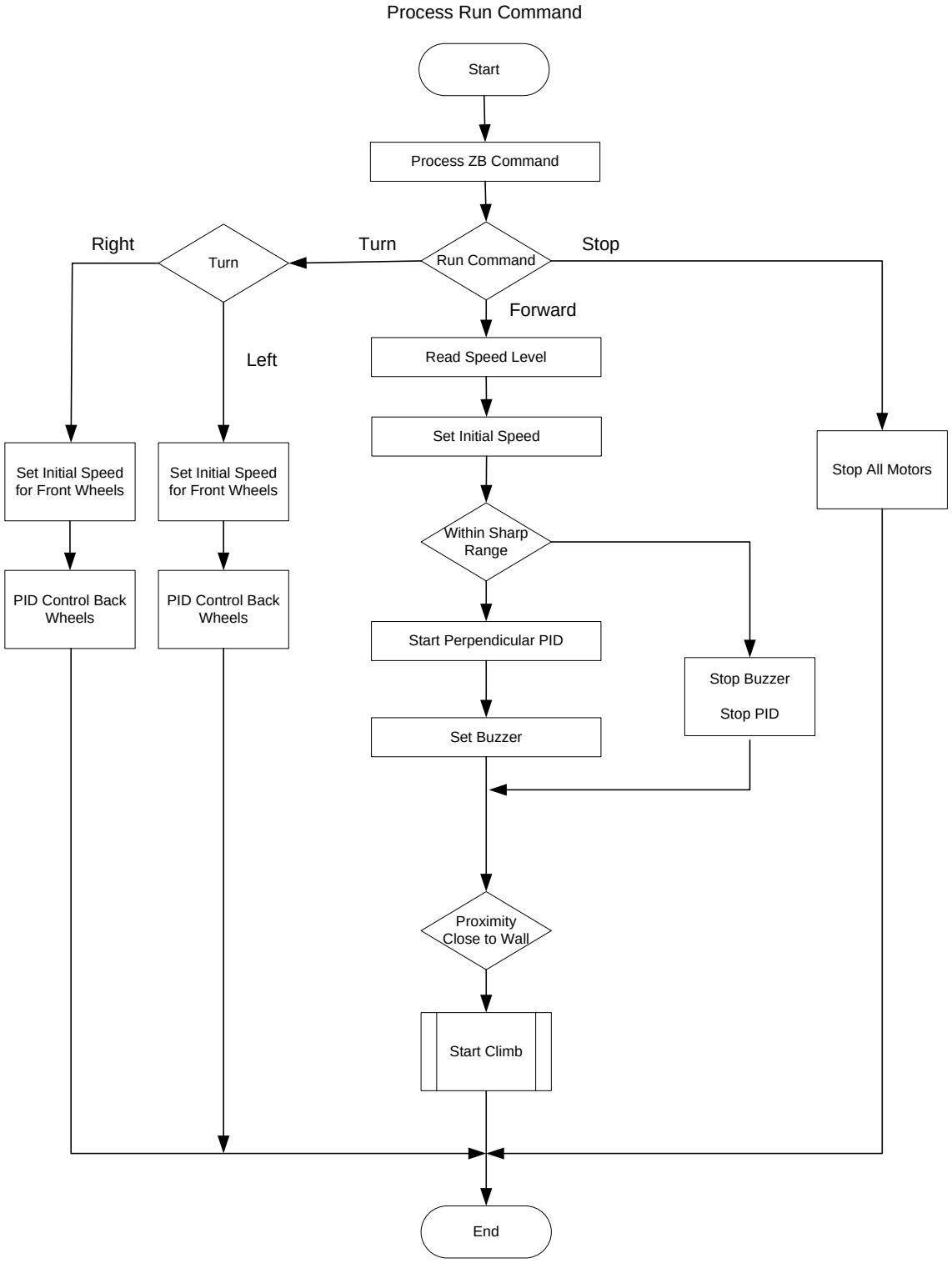
APPENDICES

Appendix A: Control Software Flowchart

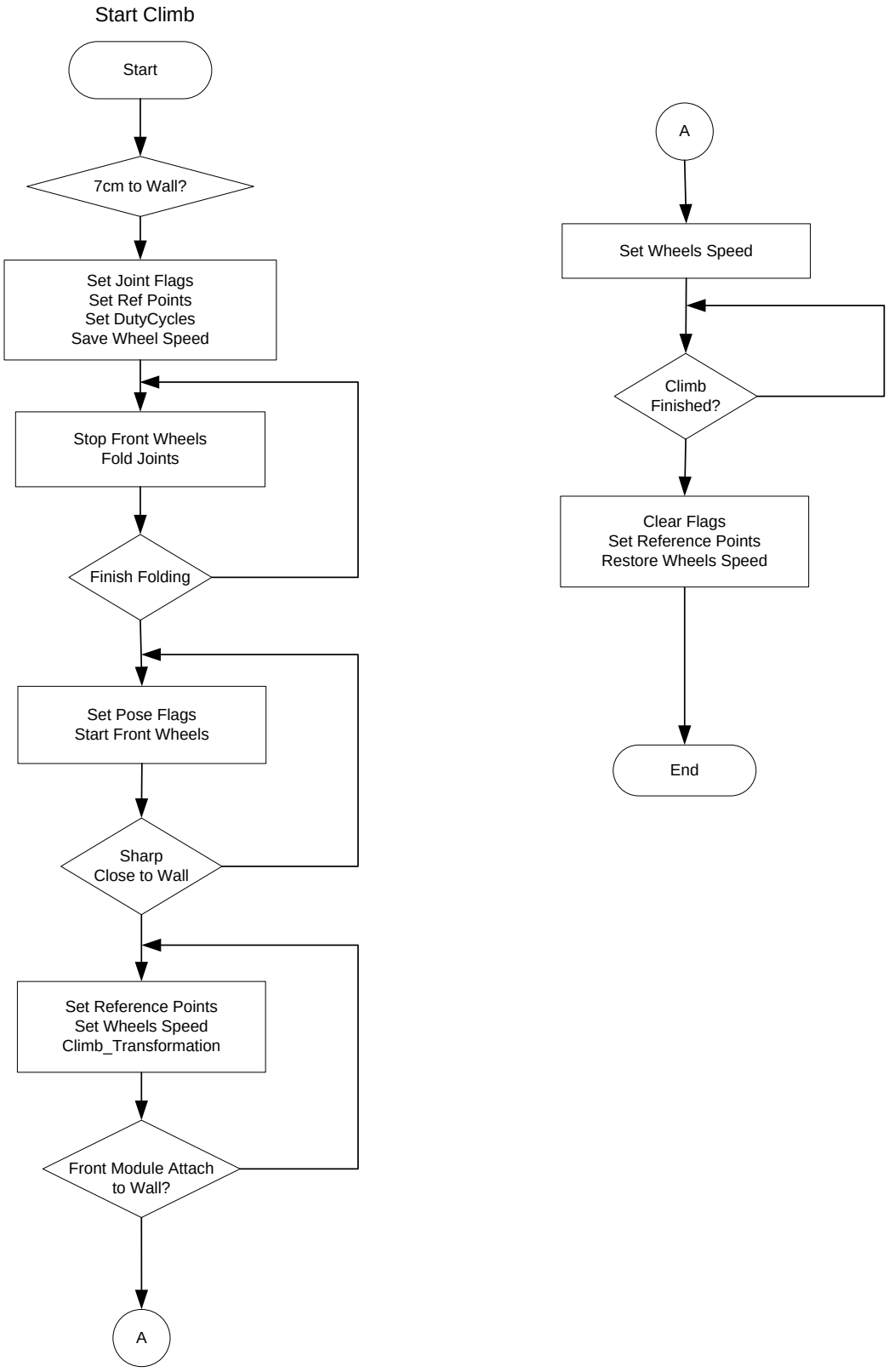
A.1 Control Logic Flowchart:



A.2 Background Running Flowchart:



A.3 Climbing Flowchart:



Appendix B: Jennic ZigBee System Design Information

The Jennic ZigBee wireless system enables users to implement systems using Jennic's Networking Stack with low cost as they do not require the development of base level functions. The modules use Jennic's JN5139 wireless microcontroller to provide a comprehensive solution with high radio performance and all RF components included. All that is required to develop and manufacture wireless control or sensing products is to connect a power supply and peripherals such as switches, actuators and sensors, considerably simplifying product development. Please see Figure B-1 for Module Block Diagram [7].

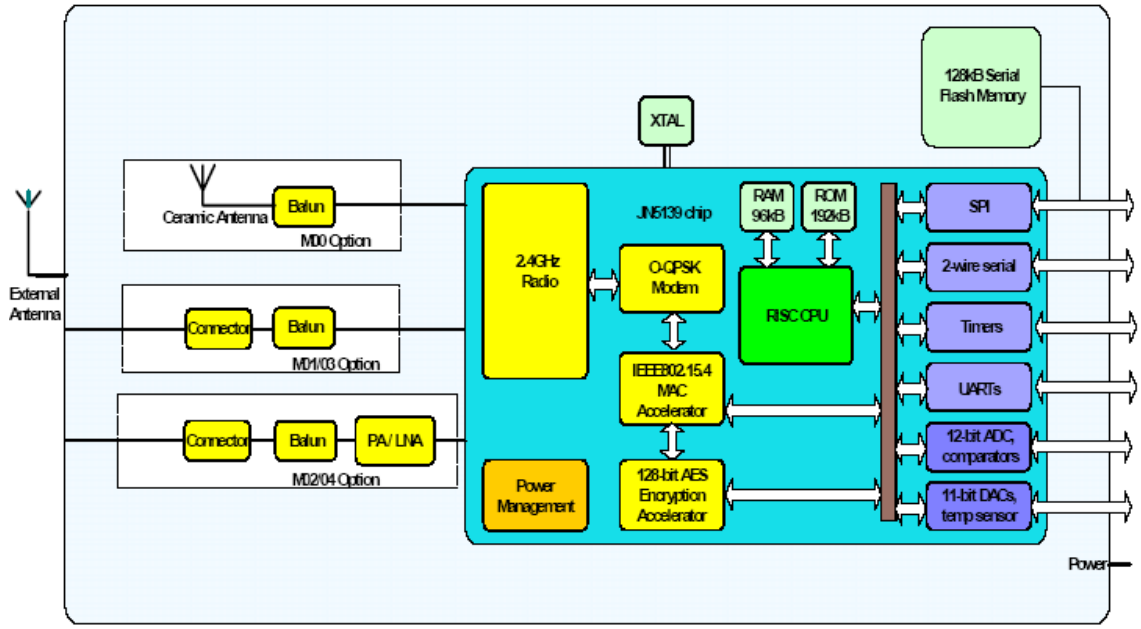


Figure B-1: Module Block Diagram [7]

The module's pin configuration is listed in Figure B-2 [7]. The size is 18mm by 30mm.

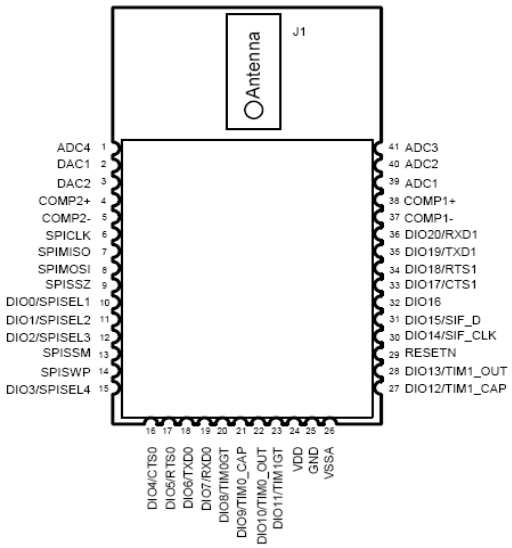


Figure B-2: Module Pin Configuration [7]

B.1 Node Types, Network Topologies, and Software Stack

The network essentials for using the Jennic ZigBee module are: node types, network topologies, and software stack. For a more complete introduction to wireless networks, please refer to the Jennic User Guides.

Node Types — In addition to running an application (e.g. turn on a LED), each node of a wireless network has a networking role also known as Node Type. A wireless network can contain three types of nodes, and they are described in Table B-1 below [8]:

Table B-1: Node Types in a Wireless Network [8]

Node Type	Description
Co-ordinator	Every wireless network must have a Co-ordinator. This node has a role in starting and forming the network, and can also have a routing role (passes messages from one node to another).
Router	This node passes messages from one node to another, although this routing functionality need not be used (in which case the node acts as an End Device). Messages can also originate and/or terminate at a Router.
End Device	This node is simply a place where messages can originate and/or terminate (the node does not have a routing role).

The wireless system used for climbing robots has only two implemented types of node: End Device and Co-ordinator.

Network Topologies — Different applications may require different configurations of the network. Examples of network topologies that may be achieved with the Jennic wireless system are illustrated in Figure B-3 below [8]**.**

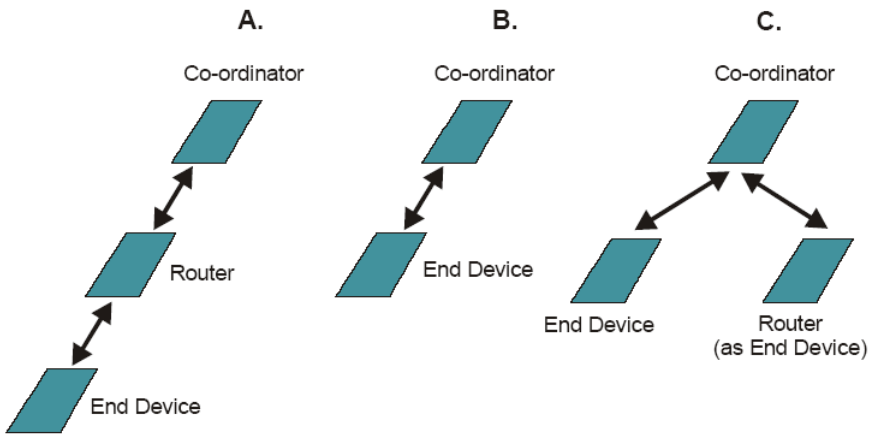


Figure B-3: Possible Network Topologies [8]

The wireless system used for the climbing robot utilizes topology shown in Figure B-4 because at the present stage, the required distance between End

Device and Co-ordinator is well within the ZigBee system's range which is approximately one kilometre. The required network has only one End Device and one Co-ordinator, so no router is needed. In my ENSC 891 Directed Studies course, a Star topology of four End Devices and one Co-ordinator was implemented and tested. The implemented Star topology is shown in Figure B-4 below.

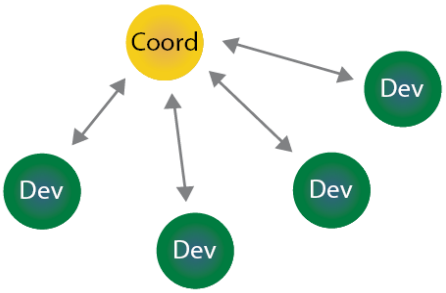


Figure B-4: Star Network

Software Stack — the software which runs on a wireless network node deals with both application specific tasks (e.g. turn on a LED or temperature measurement) and networking tasks (e.g. sending/receiving a message). The software is organized as a number of layers, forming a stack as illustrated in Figure B-5 [8].

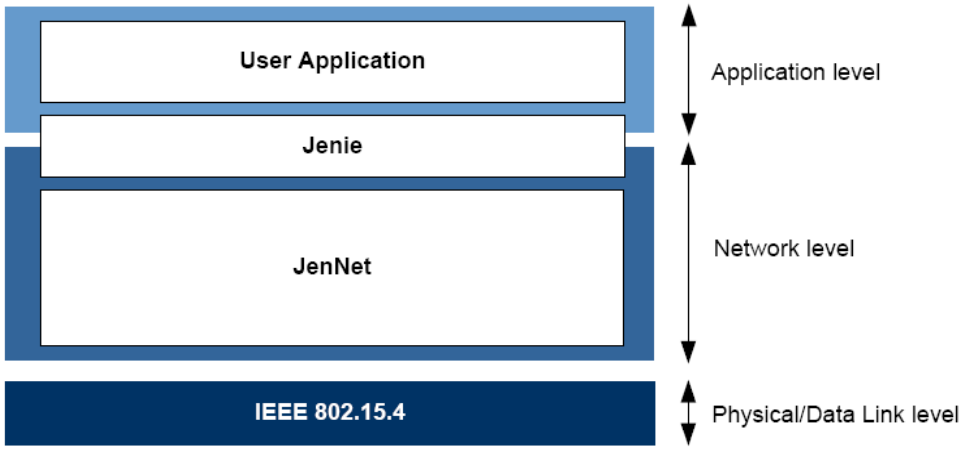


Figure B-5: Jenie/JenNet Stack Architecture [8]

In addition, Jennic provide an easy-to-use software interface, called Jenie, which allows the application to interact with the JenNet network protocol. The Jenie interface is available as an Application Programming Interface (API), consisting of a library of C functions. The Jenie functions can be used directly in the application code or via a serial command set known as AT-Jenie (refer to User Guides [8] Section 3.1 for more details).

B2. Jennic JN5139-EK010 ZigBee Evaluation Kit

The Jennic JN5139-EK010 ZigBee Evaluation Kit allows a small wireless network to be quickly assembled and used. The Kit contains one Co-ordinator board with a LCD display and four End Device boards, pre-installed with Jennic JN5139 wireless modules. The Kit also includes three SMA-connector antennas. The JN5139 wireless module can be chosen from either a SMA-connector module or integral ceramic antenna module.

The Maximum Transmit power is 2.5dBm [7] for the Jennic JN5139 ZigBee module. The 802.15.4 standard requires radios to have a minimum

output power of -3 dBm, or 0.5 mW. Radios on the market today have output power of between 0 and 3 dBm or 1 and 2 mW. Jennic provides very high transmit power among ZigBee vendors. See Figure B-6 for Jennic JN5139-EK010 ZigBee Evaluation Kit [19].

Figure B-6: Jennic JN5139-EK010 ZigBee Evaluation Kit [19]

The boards may be powered from batteries or, alternatively, from an external power source. In addition, each board has the following features [8]:

- LED indicators for user output (D1 and D2)
- 1 LED indicator for power status (D9)
- push-button switches
- switches for user input (SW1 and SW2)
- switches for board reset (RST) and programming (PRG)
- 1 slider switch for power on/off (SW6)

- UART interfaces (UART0 and UART1) for communication and program downloads
- Expansion port for connecting the JN5139 wireless microcontroller to external circuitry, such as sensors

B.3 Attenuation, Interference and Multi-path

In any network — wired or wireless — the reliability of communication between any pair of nodes is degraded by three phenomena: attenuation, interference and multi-path [12]. Especially in wireless networks, attenuation, interference and multi-path are important considerations at the system design stage.

Attenuation — Radio waves are attenuated with distance, similar to candlelight becoming dimmer as you move away from a candle [12]. Attenuation (path loss) may be due to many effects, such as free-space loss, refraction, diffraction, reflection, aperture-medium coupling loss, and absorption. Attenuation (path loss) is also influenced by terrain contours, environment (urban or rural, vegetation and foliage), propagation medium (dry or moist air), the distance between the transmitter and the receiver, and the height and location of antennas [13]. In free space, the signal attenuation is inversely proportional to the square of the distance between the transmitter and the receiver, so in free space the path loss exponent is equal to two. In the real world, the path loss exponent is normally in the range of two to four. Radio waves are further attenuated when

passing through solid materials such as doors or walls. Please see Table B-2 for ZigBee Signal Attenuation Caused by Various Types of Objects [14].

Table B-2: ZigBee Signal Attenuation Caused by Various Types of Objects [14]

Object in Signal Path	Signal Attenuation through Object
Plasterboard wall	3 dB
Glass wall with metal frame	6 dB
Cinder block wall	4 dB
Office window	3 dB
Metal door	6 dB
Metal door in brick wall	12 dB
Human body	3 dB

Interference — When electromagnetic sources generate signals in the same frequency band as the transmitted radio signal, the radio receiver experiences interference. Interference can come from intentional radiators, such as other radio transmitters, or from unintentional radiators, such as microwave ovens. Spread spectrum techniques, whether frequency-hopping or direct-sequence, offer some degree of immunity from interferers [12]. The multipath interference is dynamic and unpredictable.

Multipath — When a transmitted radio signal arrives at a receiver by means of two or more paths (e.g., due to reflections off nearby objects), the multiple signals combine at the receiver's antenna [12]. The wavelength for 2.4 GHz ISM band is about 0.125 meters (4.9 inches), so when direct and reflected paths differ by multiples of one half wavelength, the signals cancel one another.

B.4 ISM Band

The 2.4 GHz ISM band is used for ZigBee. The industrial, scientific and medical (ISM) radio bands were originally reserved internationally for the use of RF electromagnetic fields for industrial, scientific and medical purposes other than communications. In recent years these bands have also been shared with license-free error-tolerant communications applications such as wireless LANs and cordless phones in the 915 MHz, 2450 MHz, and 5800 MHz bands. Because unlicensed devices already are required to be tolerant of ISM emissions in these bands, unlicensed low power uses are generally able to operate in these bands without causing problems for ISM users. The 2.4 GHz ISM bands defined by the ITU-R is from 2.400–2.500 GHz (centre frequency 2.450 GHz) [15]. In the 2.4 GHz band there are 16 ZigBee channels, with each channel requiring 5 MHz of bandwidth. The center frequency for each channel can be calculated as, FC = (2350 + (5 * ch)) MHz, where ch = 11, 12, ..., 26 [16].

B.5 Wireless Link Budget

A wireless link budget analysis is an important step to take in order to determine the feasibility of a system, and it is also an excellent means for anyone to begin understanding the various factors which must be traded off to realize a given cost and level of reliability for a communications link. Receiver sensitivity is defined as the minimum signal power level (in dBm or mW) that is necessary for the receiver to accurately decode a given signal. [17]. Transmission power drives radio range — the higher the power, the longer its range for a desired signal strength. A link budget is the accounting of all of the gains and losses from the

transmitter, through the medium (free space, cable, waveguide, fiber, etc.) to the receiver in a telecommunication system. A simple link budget equation is shown below [18]:

Received Power (dBm) = Transmitted Power (dBm) + Gains (dB) − Losses (dB)

The link budget influences both the line-of-sight range and the robustness of non line-of-sight transmissions of a transmitter/receiver pair. For the Jennic JN5139 module, the receiver sensitivity is -96 dBm [7], and the transmit power is 2.5 dBm [7]. The link budget equals 98.5 dB.

B.6 Jennic ZigBee System Application Design

The ZigBee standard provides network, security and application support services operating on top of the IEEE 802.15.4 MAC and PHY wireless standard. The ZigBee application uses functions of the 802.15.4 Stack API to interact with the underlying hardware to access hardware registers through the IEEE 802.15.4 stack layers [20]. The ZibBee application also interacts with the Co-ordinator or End Device board hardware peripherals using the functions of the Board API. The application interacts with on-chip hardware peripherals using functions providing by the Integrated Peripherals API. The Board API then uses the Integrated Peripherals API to access the hardware. The hardware generates interrupts which are handled by an interrupt handler. The interrupts were routed to the IEEE 802.15.4 stack or peripheral hardware drivers. The software architecture is described below in Figure B-7 [20].

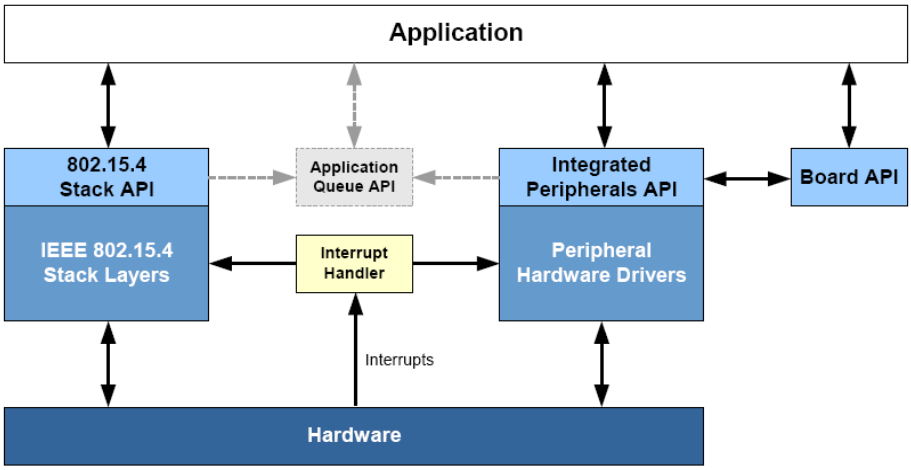


Figure B-7: Software Architecture [20]

Star topology was implemented and tested. The system is able to connect up to four End Devices using a single Co-ordinator.

The network is started by first completing the configuration of the device which will act as the Co-ordinator and then starting the device in Co-ordinator mode. The Co-ordinator is then opened for requests from the other End Devices to join the network.

An End Device wishing to join the network must first be initialized by calling the function vInitSystem to initialize the IEEE 802.15.4 stack on the device. In order to find the Co-ordinator, the device performs an Active Channel Scan in which it sends out requests across relevant frequency channels.

When the Co-ordinator acknowledges the request, it determines whether it has the resources to support the new End Device or reject it. Once the Co-ordinator accepts the device, the network is ready to exchange data.

The flowchart in Figure B-8 provides an overview of the steps in setting up a ZigBee based network [20], and Figure B-9 presents the Co-ordinator setup process.

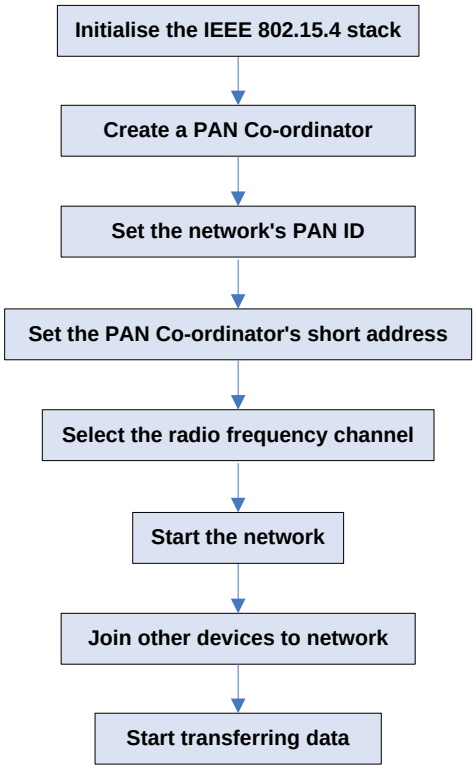


Figure B-8: Network Set-up Process [20]

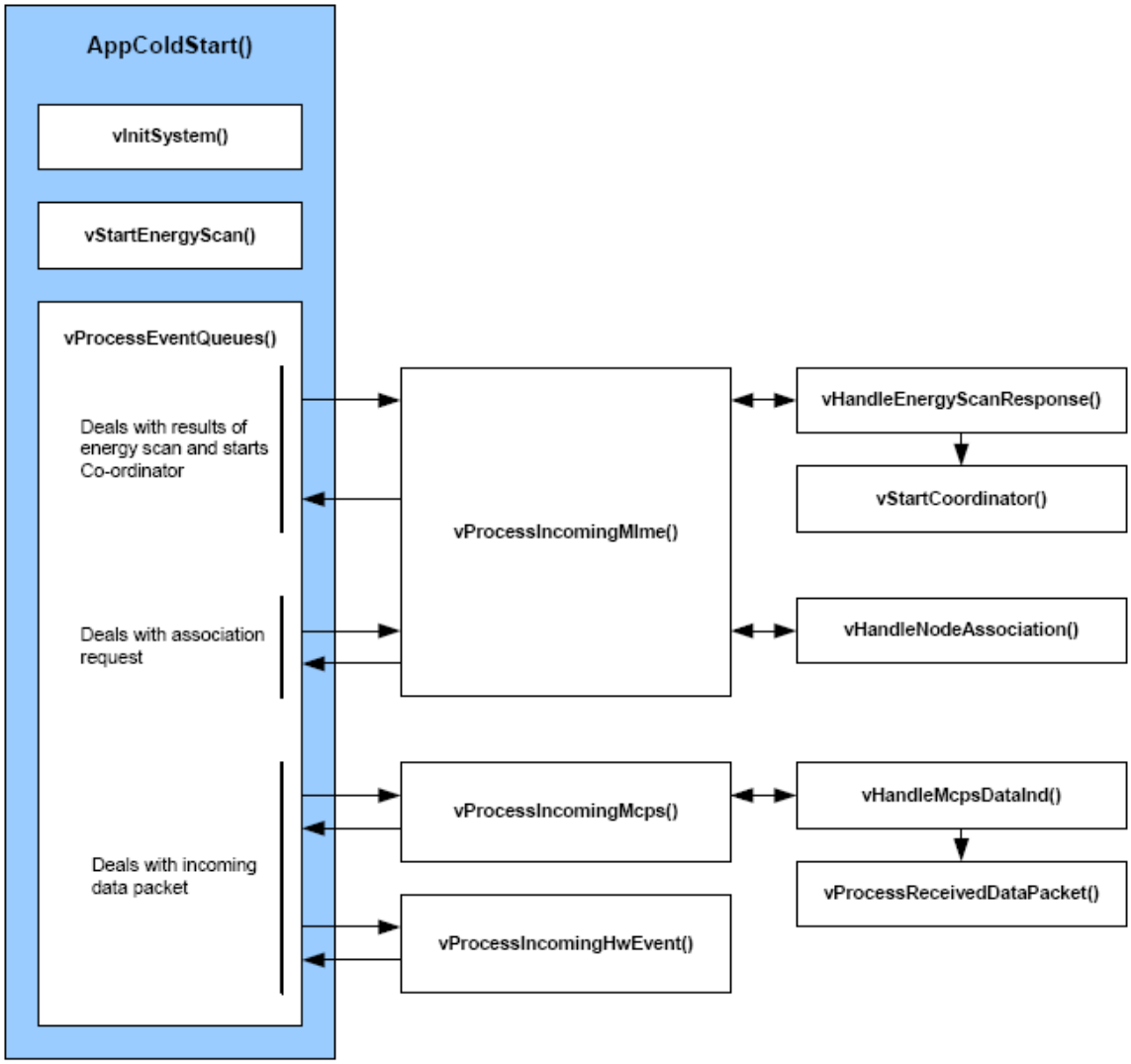


Figure B-9: Co-ordinator Setup Process [20]

Appendix C: Digital Signal Processor (DSP) TMS320F2808 Specifications

The hardware platform is based on the Texas Instruments (TI) Digital Signal Processor (DSP) TMS320F2808. It is a high-performance 32-Bit, low-power (1.8-V Core, 3.3-V I/O) designed CPU. The main features of TMS320F2808 hardware include [9]:

- High-Performance Static CMOS Technology – 100 MHz frequency
- 128K x 16 Flash
- 18K x 16 SARAM
- 12-Bit analog-to-digital converters (ADC), 16 Channels
- Two asynchronous serial communications interfaces (SCI)
- Four serial peripheral interfaces (SPI)
- Three 32-Bit CPU Timers
- 16 pulse-width modulator (PWM) Outputs
- Two Controller Area Network (CAN) Modules
- One Inter-Integrated-Circuit (I2C) Bus
- 35 Individually Programmable, Multiplexed General Purpose I/O (GPIO) Pins With Input Filtering

The 100-pin Plastic Quad Flat Pack (QFP) package of the TMS320F2808 is used to control the robot. The package information is listed in Figure C-1, and all linear dimensions are in millimeters.

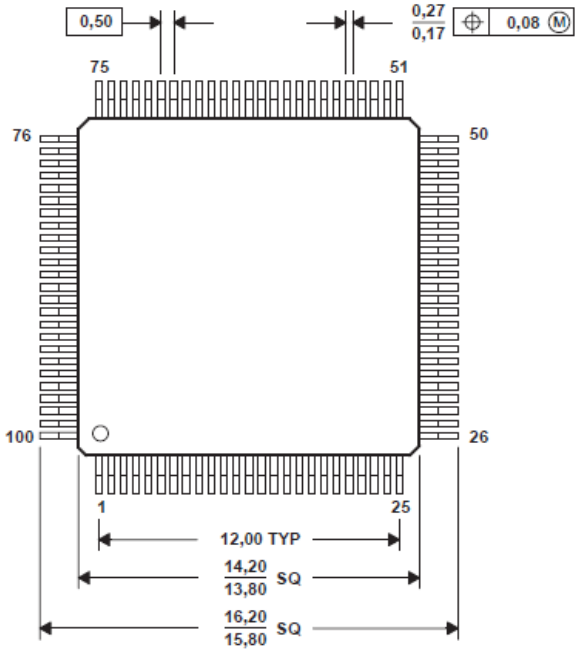


Figure C-1: 100-pin Plastic Quad Flat Pack Package [9]

The TMS320F2808 Experimenter Kit from Texas Instruments is an ideal product for initial device exploration, prototyping and testing. The following Figure C-2 illustrates the TMS320F2808 Experimenter Kit - TMDSDOCK2808. The docking station has a size of 90mm x 25mm.

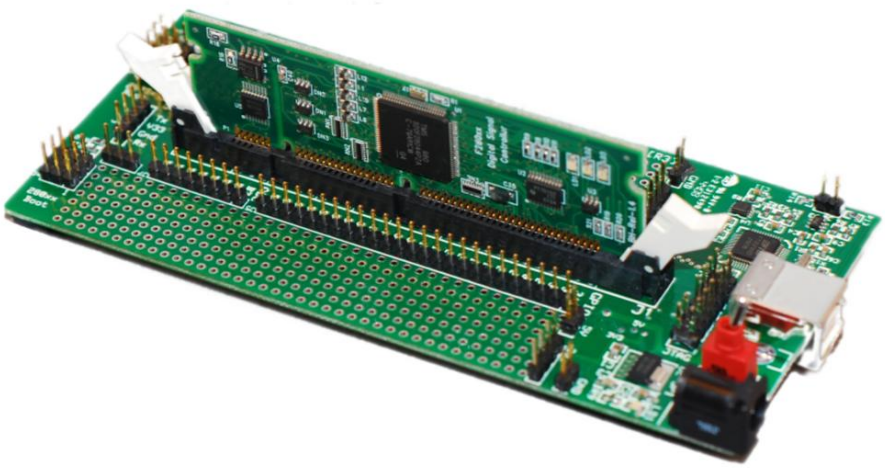


Figure C-2: TMS320F2808 Experimenter Kit - TMDSDOCK2808 [10]

The TMS320F2808 Experimenter Kit has a docking station with access to all control card signals, prototype board areas and RS-232, JTAG connectors, and features on board USB JTAG emulation. Each kit contains a F2808 control card. The control card is a complete board level module that utilizes the industry-standard to provide a single-board controller solution. No external JTAG emulator is required as the docking station has on board USB JTAG emulation. The climbing timing belt robot prototype can be built on the Experiment Kit - TMDSDOCK2808 board. After verifying that the entire system works, a PCB can be designed at miniature size so that Surface Mount Device (SMD) chips can be used in the future.

The TMS320F2808 Experimenter Kit - TMDSDOCK2808 contains the following features:

- TMS320F2808 MCU based control card
- Docking station with prototype area
- On board USB JTAG emulation
- Code Composer Studio™ IDE v3.3, C28x™ Free 32K Byte Version