Power - 2
Sensors -2
USer Interface -2
Motor Control -2
Chassis -2
Processor -2
Schematic -5

Power -4
Software H-bridge Control -4
Half H-Bridge -4
Full H-Bridge -4
IR Detectors / ADC -4
Software IR control -4

The Ram Roaster

"Where there's a Ram, we'll Roast it!"

100

Team 8

Team Members Will Flores Michael Yenik Alexander Youngs	Originator: Will Flores			
	Checked:	Released:		
	Filename: Team_08			
	Title: **The Ram Roaster "Where there's a Ram, we'll Roast it!"**			
This document contains information that is **PRIVILEGED** and **CONFIDENTIAL**; you are hereby notified that any dissemination of this information is strictly prohibited.	Date: **4/1/2012**	Document Number: **1-9-3-122012-1**	Rev: **2.5**	Sheet: **1 of 16**

Revision	Description
1.0	Originated document.
1.1	Modified document name to Team_08.doc
1.2	Modified metadata.
1.3	Changed the tagline.
1.4	Changed the product name.
1.5	Edited a team member's name.
1.6	Added system overview blocks.
1.7	Added abbreviations for readers.
1.8	Added system overview descriptions
1.9	Scope added.
2.0	Sections 4 and 6 stubbed.
2.1	Section 3 is fixed
2.2	Tests for the control board were stubbed
2.3	Timer tests, power system info and USB debugger info added to sections 6 and 4
2.4	Onboard User Interface and Navigation System added to section 4
2.5	ADC, switch interrupt, and line detection tests added to section 6

Table of Contents

Table of Figures

1. Scope

Have you ever needed groceries but didn't want to go out? What if your child is at soccer practice and you didn't feel like going to get them? Well, fret no more! With THE RAM ROASTER you no longer have to leave the comfort of your home. This state-of-the-art vehicle will follow a line of black electrical tape to the destination of your choice. All you have to do is unroll a few miles of black electrical tape and this car will do all those boring tasks for you! THE RAM ROASTER also uses the latest in Global Positioning System technology to navigate to common establishments in order to take care of chores as directed by the user. With two modes of navigation and a top speed of a quarter of a mile per hour, this vehicle will have all of your daily tasks completed… eventually.

2. Abbreviations

CPU – Central Processing Unit: the component of the product that performs all of the number crunching for the car.

GPS – Global Positioning System: the system that allows a device to determine its current location on Earth.

LED – Light Emitting Diode: a circuit component that emits light, which is used for providing a user interface.

LCD – Liquid Crystal Display: a component that outputs useful information to the user.

QSK – Quick Start Kit: the development environment that our product is built with.

USB – Universal Serial Bus: the protocol and hardware for interfacing with the vehicle's onboard computer.

PCB - The control board we commonly refer to that powers our car as well as provides inputs to the QSK.

3. System Overview

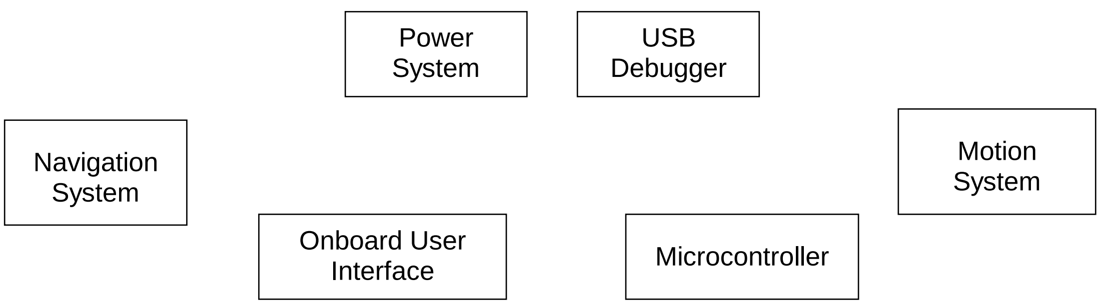

Figure 3.1: System Overview

3.1. Onboard User Interface

The user can interact with the vehicle through the buttons and onboard dial. This allows the user to adjust the vehicle's settings without the use of a computer. The user then, receives feedback from the LCD screen and the use of LEDs.

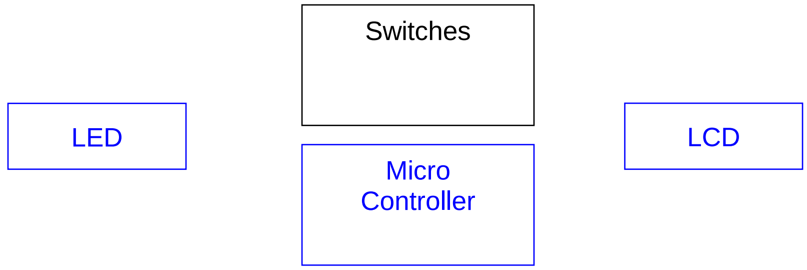

Figure 3.2: Onboard User Interface

3.2. USB Debugger

The USB debugger allows the vehicle to be connected to a computer so that new information can be transferred to the onboard computer. It also allows an advanced user to troubleshoot the current code in the onboard computer.

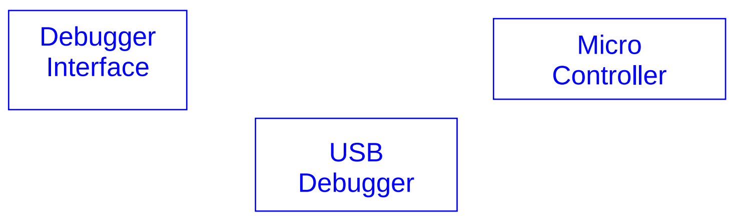

Figure 3.3: USB Debugger Overview

3.3. Microcontroller

The Microcontroller is the heart of the system. It processes inputs from: the users, sensors, and the GPS unit. It uses these inputs to cause the vehicle to follow lines or navigate through GPS waypoints.

Figure 3.4: Microcontroller Overview

3.4. Navigation System

The navigation system provides the vehicle with information for travelling. The navigation system comprises a GPS Unit and optical sensors.

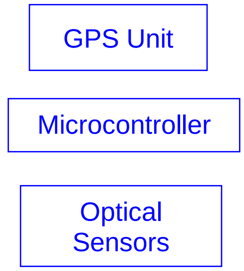

Figure 3.5: Navigation System Overview

The navigation system provides the vehicle with information for travelling. The navigation system comprises a GPS Unit and optical sensors.

3.4.1. GPS Unit

The GPS unit provides the vehicle with its current location on Earth, which allows it to navigate to a desired location.

3.4.2. Optical Sensors

The optical sensors are used by the vehicle to detect black lines, which it then follows.

3.5. Motion System

The motion system is comprised of the motors and their controllers. This is the heart of what moves the vehicle from place to place.

Figure 3.6: Navigation System Overview

3.5.1. Motors

The motors are connected to the vehicles wheels which are controlled by the microcontroller and the motor controller system.

3.5.2. Motor Controller System

The motor controller system contains the circuitry use to activate the motors in either direction at any number of speeds.

3.6. Power System

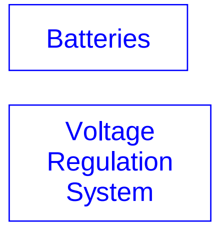

Figure 3.7: Power System Overview

3.6.1. Batteries

The vehicle runs uses 4 AA batteries, which provide the requisite raw power to the microcontroller, motors, and other devices connected to the vehicle.

3.6.2. Voltage Regulation System

The voltage regulation system takes the raw batteries input and controls it to a level which the microcontroller and other circuitry can comfortably use.

4. Hardware

This section goes into detail of describing the hardware used on our vehicle.

4.1. Onboard User Interface

4.1.1. Buttons

When pressed the buttons complete a circuit and transmit a high signal. Otherwise there is no signal transmitted. In this way the user can send their pulses or input to determine what the car should do. This can be set to control the motors or LCD screen or anything else that the user can provide input.

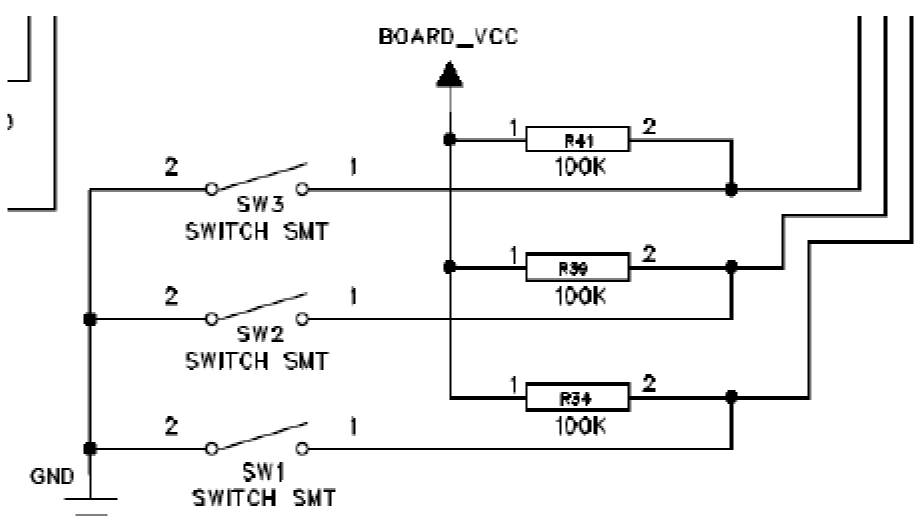

Figure 4.1: Switch Circuit

4.1.2. LCD

The LCD onboard the QSK62P is the HD44780U and uses a KS0066 controller IC. The LCD screen is a 2 row by 8 character display that can be written to for debugging/operation purposes. Our configuration hooks up 5VR to pin 2 on the LCD pinout on the QSK Board. 5VR is on pin 24 of our PCB and 5VR is created as depicted in figure 4.2. We suggest using the tutorial code to operate the LCD as they have configured functions to interface with the LCD such as BNSPrintf(). The tutorial is available on the CD that comes with the QSK62P board.

Note: In order for the QSK's LCD to be powered from the control board, a modification to the female connector and the LCD circuit needs to be made. A jumper must be soldered to these circuits as shown below:

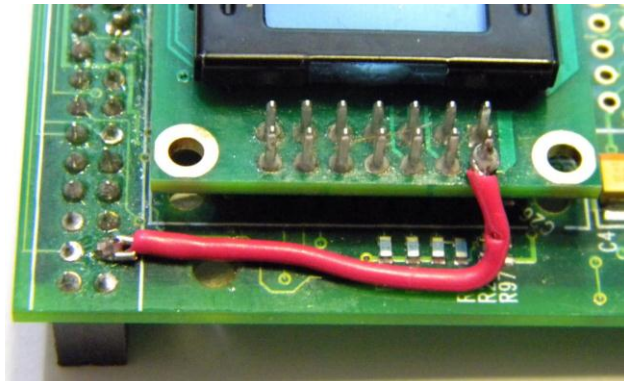

Figure 4.2: LCD circuit connected to the Control Board

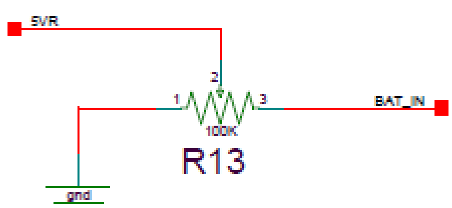

Figure 4.3: Power to LCD

4.1.3. LEDs

The LED circuit is shown in figure 4.3. LED0 is connected to Port 8 Pin 0, LED1 to Port 7 Pin 4, and LED2 to Port 7 Pin 2. The LEDs can easily be used as user interface. A common way to use them is in conjunction with the buttons. Since each LED is located above a corresponding button, it is common LEDs with their corresponding button. However, the LEDs can be used to display calibration modes, black line detection, motor control, etc. To turn on an LED you provide a low signal to its corresponding Port and Pin and a potential is created across the resistor and the LED enough to turn the LED on. To turn off the LED and remove the potential, provide a high signal to the corresponding Port and Pin.

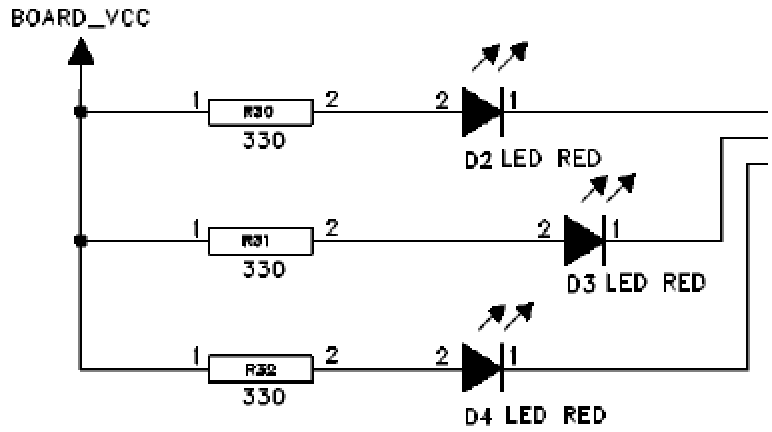

Figure 4.4: LED Circuit

4.2. Motion System

4.2.1. Motor Controller System

The motion system is comprised of the motors and the control board that is used to drive the motors with the requisite voltages. The main structure that controls the voltages applied to the motors is called a full H-Bridge. Each motor is controlled by its own H-Bridge. The control board implementation consists of two ZXMP4A16G P-channel Metal Oxide Semiconductor Field Effect Transistors (MOSFETs) and four ZXMN6A09G N-channel MOSFETs. These MOSFETs are current limited by 1206 package resistors as seen in the figure below:

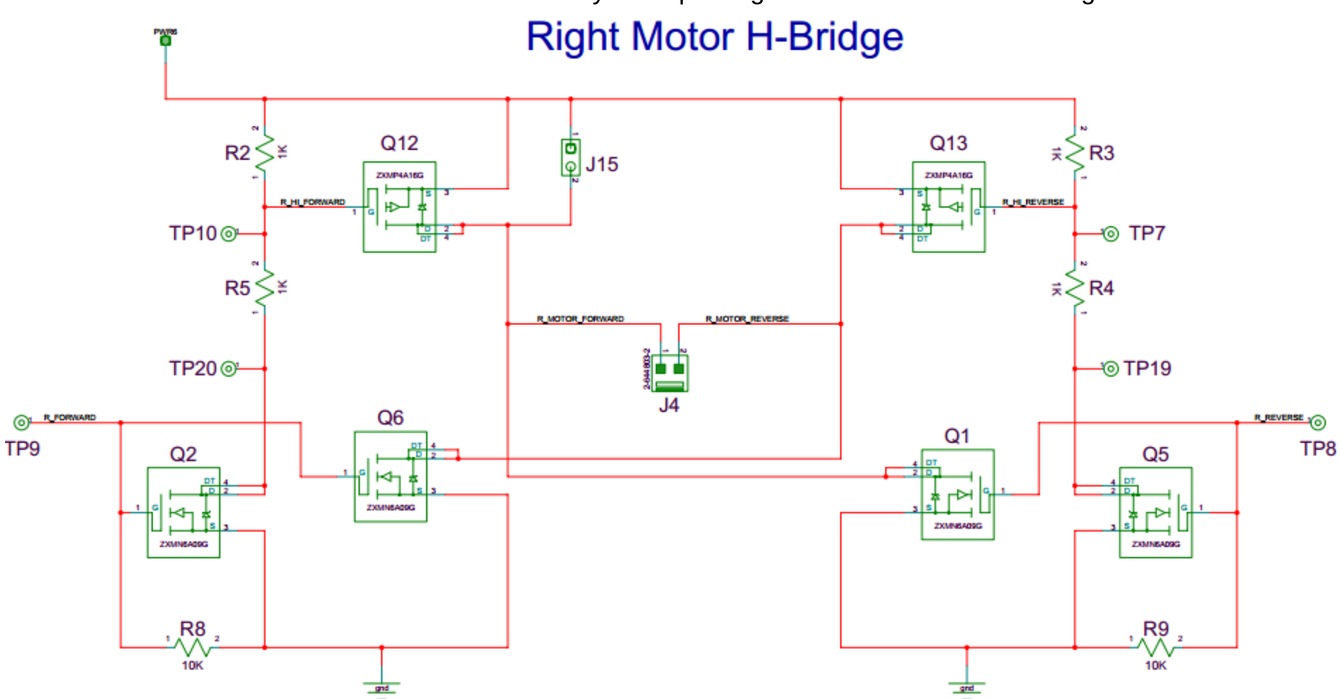

Figure 4.5: Right Motor Full H Bridge

Assuming that both R_FORWARD and R_REVERSE are configured to be low when the control board and QSK are connected together the motor in the middle will not be activated. When the R_FORWARD signal is activated from the QSK board's port 3 pin 0, Q12 and Q6 allow current to flow through the motor creating a voltage drop of positive raw battery voltage between J4. When the R_FORWARD signal is deactivated and the R_REVERSE signal is active from the QSK board's port 3 pin 1, Q13 and Q1 allow current to flow through the motor creating a voltage drop of negative raw battery voltage between J4. This is how each motor is able to travel forward and reverse.

Let us suppose that the R_FORWARD signal was still active when the R_REVERSE signal is applied. This would create a condition where the motor would be bypassed and the current would "shoot through" the transistors. This would likely damage, if not destroy, the H Bridge due to the amount of current that would run through the MOSFETs. Therefore much care should be exercised when programming the application of signals to the motors in order to avoid this occurring.

Below we have a figure of the Left motor full H bridge for completeness, which operates very similarly to the right motor H bridge.

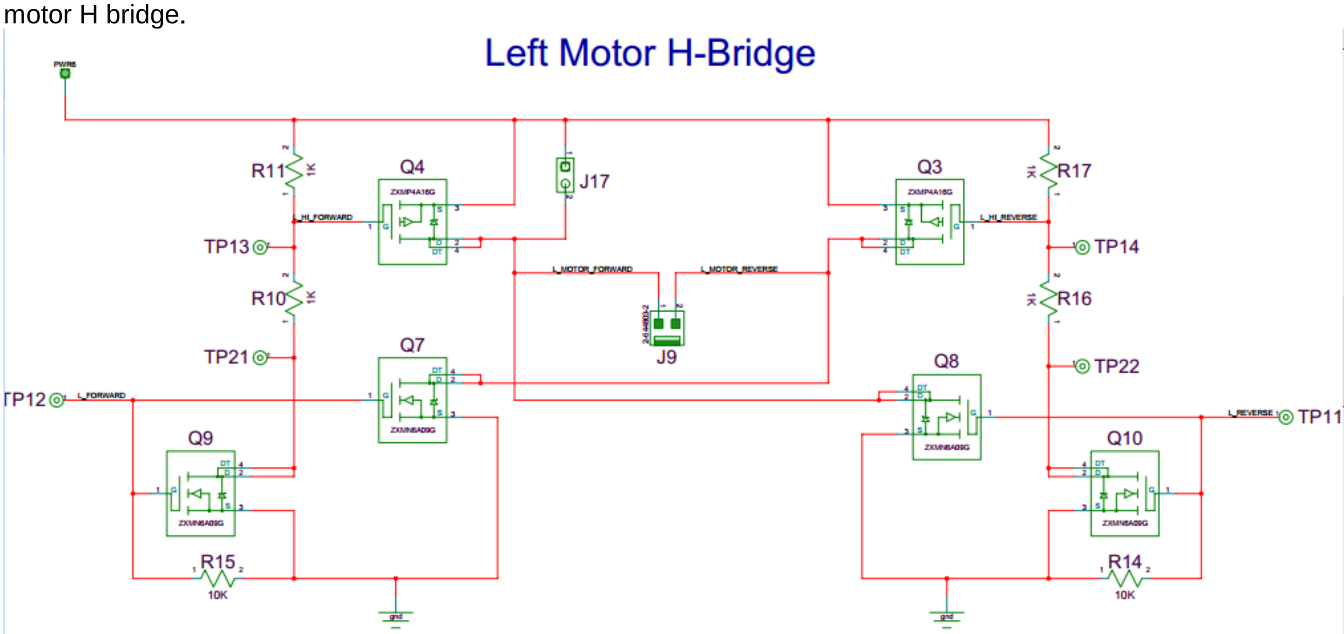

Figure 4.6: Left Motor Full H Bridge

The main difference between the Right Motor Full H Bridge and the Left Motor Full H Bridge is where the FORWARD and REVERSE signals are derived from on the QSK. The L_FORWARD signal comes from port 3 pin 2 and L_REVERSE comes from port 3 pin 3.

4.2.2. Motors

The vehicle consists of two 120:1 DC brushed motors that are connected to the control board by means of removable connectors. These connectors are crimped on to insulated stranded wire, which is soldered on to the connectors of the motors. The connectors on the motors are very fragile; therefore care must be exercised in order to not break this connection, once the connectors are attached to the motor.

Figure 4.7: Opened Motor for Vehicle

4.3. Power System

The power system uses two switching regulators to step the battery voltage down to the needed levels. The regulators use buck-boost ICs from Linear Technology, the LTC3531 and the LT1761ES5. The first, along with some external components, provides 3.3V for the majority of the logic. The latter provides 1.8V for the GPS and the level shifter used to communicate with the GPS. The LCD module needs a minimum of around 4.5V which is provided by R13, a potentiometer connected to the total battery voltage. It must be adjusted as the batteries drain from fully charged to discharged. The schematics for the 3.3V and 1.8V power systems are shown below:

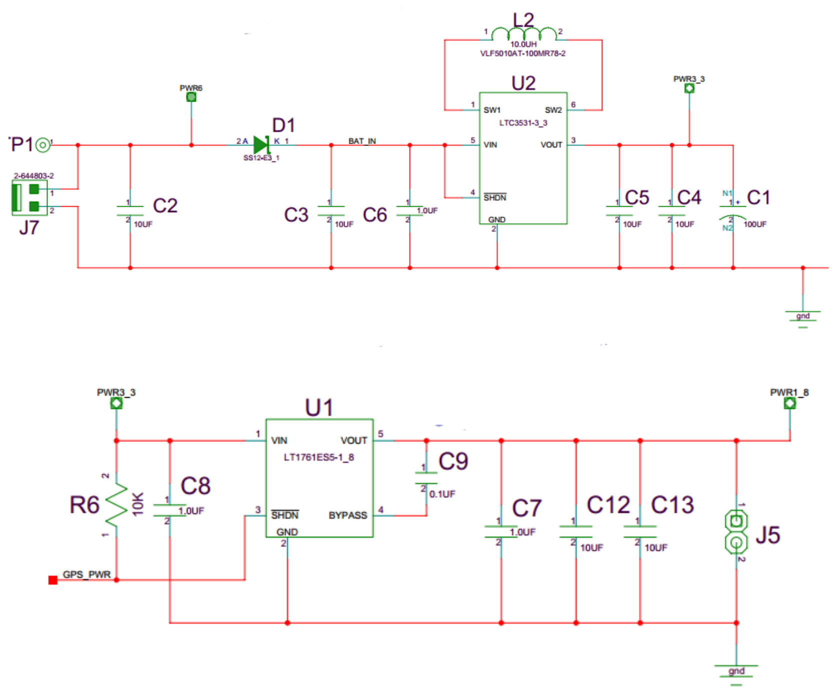

Figure 4.8: Power Regulation Circuit

4.4. Navigation System

4.4.1. Optical Sensors

There are two optical sensors and therefore two optical sensor circuits. We will refer to them as right and left side detectors. A circuit diagram is shown below of each side detector. We used two QSD123s for the detectors. These devices are infrared phototransistors and affect the amount of current running through them. The right side detector reading is given to the QSK on Port 10 Pin 3 and the left side is given on Port 10 Pin 2. These readings can be cycled through using Repeat Sweep mode on the ADC.

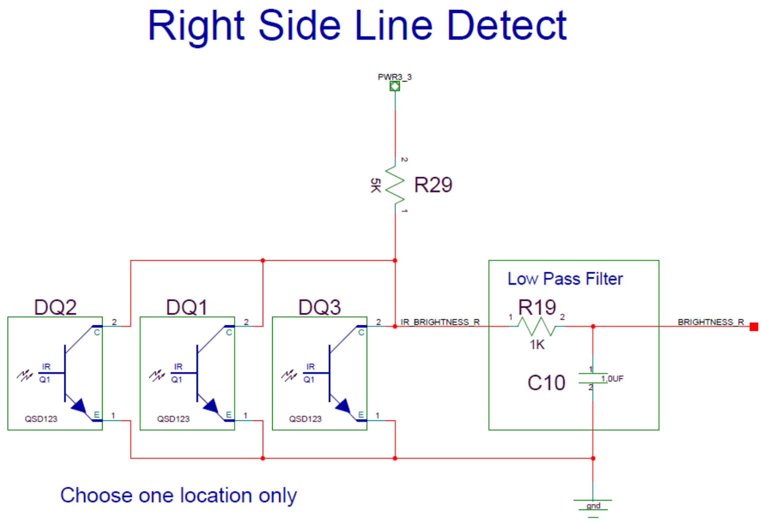

Figure 4.9: Line Detection Circuit (Right)

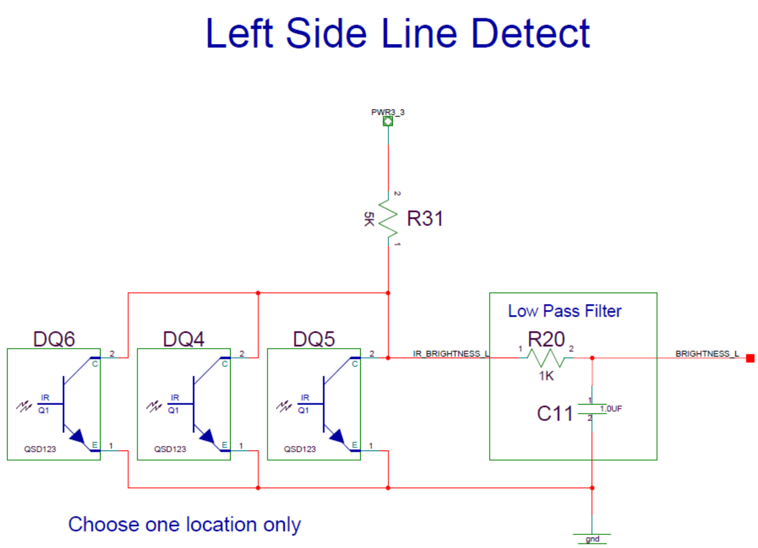

Figure 4.10: Line Detection Circuit (Left)

4.4.2. Emitter

The emitter circuit is centered about a QED123 infrared LED. This circuit is constructed as seen in the figure below. The IR LED is turned on when given a high signal from IR_LED. IR_LED is connected to Port 3 Pin 7 on the QSK board. The purpose of the emitter is to provide IR light onto the surface below the car so that the optical sensors described above can read the amount of IR light reflected back.

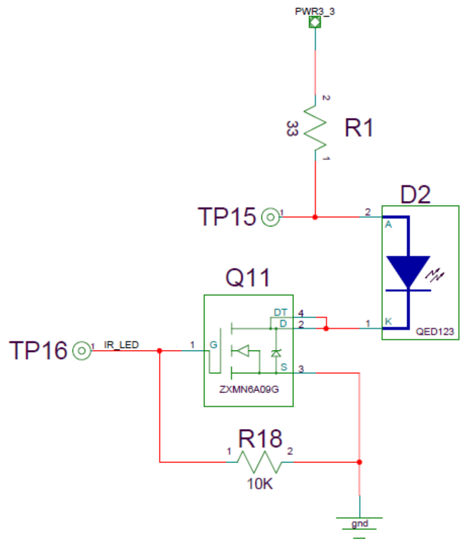

Figure 4.11: Emitter Circuit

4.5. USB Debugger

The USB debugger is a fairly complicated debugging facility exposed through the High-performance Embedded Workshop (HEW). The M16C architecture has support for debugging in hardware. When the user selects the option “erase flash and connect” in the HEW, a microkernel is loaded on to the M16C along with the user application. The microkernel handles interrupts generated by the BRK instruction as well as single step interrupts. Single step interrupts occur after execution of each instruction, but only when the D (debug) flag is set in the flag register (FLG). During each of these interrupts, the microkernel communicates with the onboard M16C E8 emulator, a tandem M16C microcontroller that communicates with the host PC. The E8 emulator communicates with the microkernel over USART 1. When the user requests information about the microcontroller such as register values, sets a breakpoint, or halts execution, the E8 emulator will appropriately request memory contents from the microkernel and send data to the microkernel to facilitate debugging of the user’s application. The E8 emulator is also responsible for uploading new programs to nonvolatile flash memory via the factory bootloader, which also communicates over USART 1, and resides in read-only “system” memory and is loaded into RAM to perform a flash re-write.

4.6. Microcontroller

The microcontroller that this vehicle is equipped with is the M30620FC from BNS solutions. This microcontroller is hosted on the Renesas QSK62P Quick Start Kit shown below:

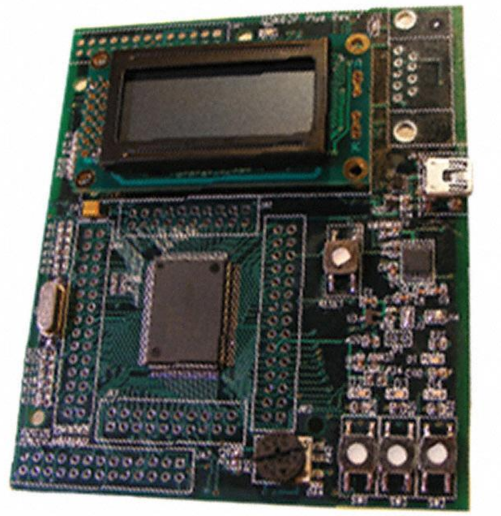

Figure 4.12: Renesas QSK62P Quick Start Kit

In order to properly connect the control board to the QSK board, a female connector must be soldered to the QSK Board as shown below:

Figure 4.13: Connection to Control Board

5. Power Analysis

Provide a description of the power consumption of each part. How long will it last running off of a battery(s).

6. Test Process

6.1. Control Board Tests

Before assembly, one should verify that J7 and J8 are not shorted by using a multi-meter and measuring a resistance in the range of mega-ohms.

After installing C2, C3, C4, C10, C11, and C12, verify J7 and J8 are not shorted by using a multi-meter and measuring a resistance in the range of mega-ohms. This test could also be done after installing each component to be thorough thus looking for a resistance in the range of mega-ohms.

After installing Q6 and Q7, test the motor connectors by probing them with either an oscilloscope or a multi-meter. Connect the battery to the control board. Use a jumper lead in order to connect J6 pin 1 to J6 pin 15 to confirm the raw battery voltage is output between the Left motor connector. Use a jumper lead in order to connect J6 pin 1 to J6 pin 17 to confirm the raw battery voltage is output between the Right motor connector.

After visually inspecting the control board for potentially shorted components, probe J8 with a multi-meter to confirm a resistance in the range of mega-ohms. Connect the battery pack and QSK board to the control board and download the tutorial program to confirm operation of the board on battery power. It may be necessary to adjust the installed R13 to control the function of the LCD screen.

After removing the short circuit between J15 and J17, use a multi-meter to test for a resistance in the order of mega-ohms between the two junctions.

After installing Q1 and Q8, test the motor connectors by probing them with either an oscilloscope or a multi-meter. Connect the battery to the control board. Use a jumper lead in order to connect J6 pin 1 to J6 pin 16 to confirm the raw battery voltage is output at J15. Use a jumper lead in order to connect J6 pin 1 to J6 pin 18 to confirm the raw battery voltage is output at J17.

Connect the battery to the control board and probe TP19, TP20, TP21, and TP22 with a multi-meter. These test points should all read the raw battery voltage.

After installing C2 and C4, connect the battery pack to the control board and probe J8 with a multi-meter in order to confirm an output voltage of approximately 3.3 volts. With the battery pack connected, probe J5 with a multi-meter in order to confirm a reading of approximately 1.8 volts.

After installing Q2, Q5, Q9, and Q10, connect the battery pack to the control board and probe TP10, TP7, TP13, and TP14 with a multi-meter. Connect a jumper lead to J6 pin 1 to J6 pin 15 to confirm that TP10 outputs half of the raw battery voltage. Connect a jumper lead to J6 pin 1 to J6 pin 16 to confirm that TP7 outputs half of the raw battery voltage. Connect a jumper lead to J6 pin 1 to J6 pin 17 to confirm that TP13 outputs half of the raw battery voltage. Connect a jumper lead to J6 pin 1 to J6 pin 18 to confirm that TP14 outputs half of the raw battery voltage.

After installing Q3, Q4, Q12, and Q13, connect the battery pack to the control board and connect the motors to the control board. Connect a jumper lead to J6 pin 1 to J6 pin 15 to confirm that the right motor is moving in the forward direction. Connect a jumper lead to J6 pin 1 to J6 pin 16 to confirm that the right motor is moving in the reverse direction. Connect a jumper lead to J6 pin 1 to J6 pin 17 to confirm that the left motor is moving in the forward direction. Connect a jumper lead to J6 pin 1 to J6 pin 18 to confirm that the left motor is moving in the reverse direction.

Confirm with an oscilloscope that JA6 pins 16 and 18 are not active while JA6 pins 15 and 17 are active.

After installing R1, R18, and Q11, using an oscilloscope probe D2 and confirm its function with software.

Visually check for shorts after all of the LEDs have been installed.

6.2. Timer Tests

The first and probably simplest timer test was to XOR an LED pin on the QSK board with a one on every timer interrupt. This means the LED is on for 50% of the time. Another LED was turned on permanently, and the light level difference was easily visible, indicating that the timer was at least running and the ISR was being run.

Once the timer was running, the voltage across the LED being controlled by the timer ISR was measured on an oscilloscope. The oscilloscope was used to measure the duration that the LED was on or off, and it was verified that the measured time corresponded to the timer interval defined in the software.

6.3. Power System Tests

After soldering on the 3.3V buck-boost circuitry, which also powers the 1.8V buck-boost circuitry, a multimeter was used to ensure the 3.3V output was the proper voltage within a small amount of tolerance. This was measured across the tantalum output capacitor (C1).

An oscilloscope was then used to examine the voltage across the output capacitor, to ensure a smooth voltage out with minimal output ripple. This was also measured at two of the power pads that connect to JA1 on the QSK board to make sure that clean power was getting to the QSK board.

After soldering on the 1.8V buck-boost circuitry, J5 on the extension board was probed with a multimeter to verify that the output is close to 1.8V. The output on J5 was then examined with an oscilloscope to verify that there was little output ripple.

6.4. Switch Interrupt

After configuring the switch interrupts for our car we had to ensure they worked. We set the interrupt service routine to “exclusive or” with the LED above the corresponding switch. This toggles the LED on/off every time the interrupt occurs. We tested the switches to make sure every time we pushed a button, the corresponding LED switched state.

After the switch interrupt was created and tested, we had to check to see if we needed to de-bounce the circuit. If the switch bounced, the interrupt may occur several times every time the button is pushed. We set a global variable called switch count to increment once per interrupt and then we set a conditional statement to flicker the LEDs after switch count was incremented to 20. Then we tested by pushing the buttons 20 times and making sure the flickering did not occur until we had in fact pushed the buttons 20 times. This turned out true in every case and we did not have to worry about de-bouncing our switches.

6.5. Line Detection

After soldering on the right side detector circuit, the left side detector circuit, and the emitter circuit to the PCB we tested to see if the emitter-detection system worked. Using an oscilloscope we connected Pin 11 (Left side detector reading) and Pin 12 (Right side detector reading) of the 26 pin male connector on the PCB. After making sure the emitter was on (Pin 22 of the connector), we made sure that the detectors picked up different readings based on what they were nearby. We noticed high readings when no IR was detected and low readings when IR was detected. This test ensured the proper values were being delivered to the ADC.

6.6. ADC

To test if we configured the ADC in software correctly, we set the ADC up in repeat sweep mode and set it to sweep from AN0 to AN3. AN2 and AN3 are the left and right detector readings, respectively. AN0 is the potentiometer resistance reading and AN1 is the voltage across the thermistor on the QSK. We were primarily concerned with AN2 and AN3 in this test, so we displayed each on the LCD Screen. We moved the car over black lines and over white areas and completely away from objects. The LCD screen displayed the measured values correctly as we moved over different areas confirming we configured the ADC properly.