Accelerometer Controlled R/C Tank
Milestone I Report for CMPEN 352W Final Project

by

David Carley ______________________

Bryan Frantz ______________________

April 14, 2009

1 Detailed Architecture

1.1 Level-1

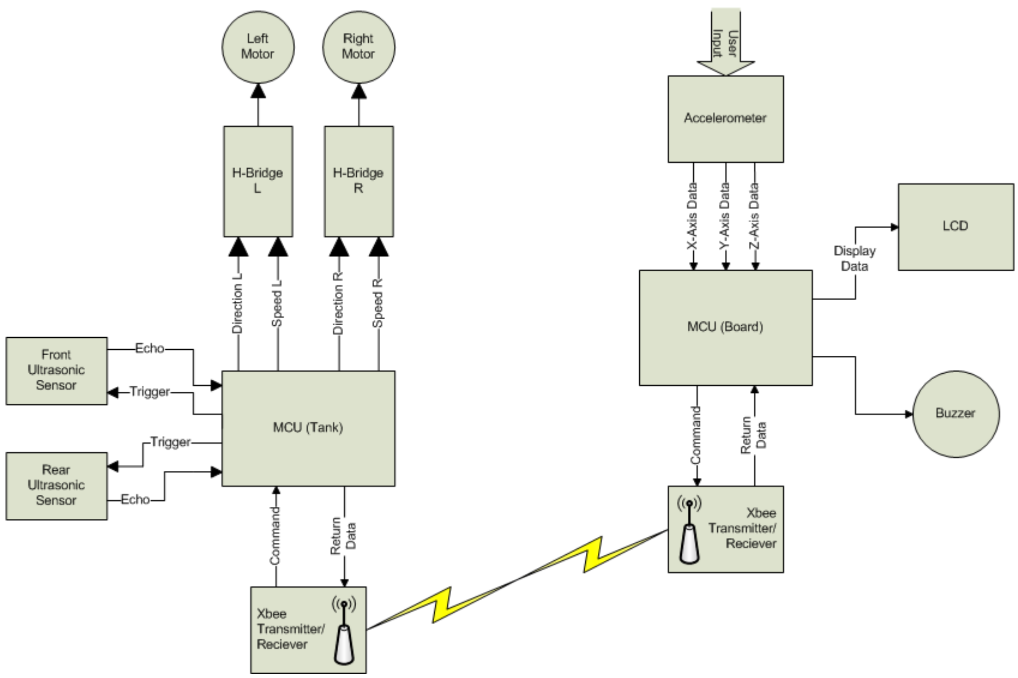

Figure 1: Level-1 Block Diagram

1.1.1 User side

Module	PIC 18F4520 MCU
Inputs	Accelerometer, XBee
Outputs	LCD, Piezo speaker, XBee
Behavior	`Finite state machine in Level-2` See Figure (2).

Table 1: Level-1 Description for PIC 18F4520 MCU

Module	Accelerometer
Inputs	User motion
Outputs	x, y, z vectors acceleration
Behavior	• Collect acceleration from user in x, y, z simultaneously • Output to PIC 18F4520 MCU

Table 2: Level-1 Description for Accelerometer

Module	LCD
Inputs	Data and control
Outputs	Viewable text to user
Behavior	• Receive data to display • Output on screen

Table 3: Level-1 Description for LCD

Module	Piezo alarm
Inputs	Input signal, Ground
Outputs	Sound
Behavior	• Receive signal from PIC 18F4520 MCU

Table 4: Level-1 Description for piezo alarm

Module	XBee
Inputs	Data, wireless
Outputs	Data, wireless
Behavior	• Receive mode – Receive data from wireless signal – Send data to PIC 18F4520 MCU • Transmit mode – Receive data from PIC 18F4520 MCU – Send data as wireless signal

Table 5: Level-1 Description for XBee

1.1.2 Tank side

Module	PIC 18F4520 MCU
Inputs	Ultrasonic Sensors, XBee
Outputs	Left Motor H-Bridge, Right Motor H-Bridge, Ultrasonic Sensor Trigger XBee

Behavior	`Round-Robin Interrupt Driven Controller`
	
	`Two Interrupts:`
	`   1. Update motor control signals (High Priority)`
	`   2. Run ultrasonic sensors to update distance (Low Priority)`
	
	`wait for a command`
	`parse command`
	`execute`
	`if(change motor control)`
	`   update PR1 and PR2`
	`   update direction`
	`if(sensor distance request)`
	`   transmit most recently updated senor values`
	
	`High Priority:`
	`   change PWM for motor 1 and motor 2`
	`   clear flag`
	
	`Low Priority:`
	`   trigger sensor to measure distance`
	`   wait for high-to-low transition`
	`   clear timer 3`
	`   wait for low-to-high transition`
	`   store timer 1 value in global variable`
	`   clear flag`
	`   repeat for second sensor`

Table 6: Level-1 Description for PIC 18F4520 MCU Tank

Module	Ultrasonic Sensor
Inputs	Trigger from PIC18F4520 MCU (Tank)
Outputs	Echo Pulse
Behavior	10 μs trigger pulse activates the sensor a echo pulse is returned proportional to the distance from closest object if(echo pulse $>$ 36ms) no object was detected

Table 7: Level-1 Description for Ultrasonic Sensor

Module	H-Bridge
Inputs	PWM and direction bits from PIC18F4520 MCU (Tank)
Outputs	Voltage levels to motor
Behavior	• The duty cycle of the PWM sets the rotation speed of the motor • 1(forward) or 0(reverse) determines direction

Table 8: Level-1 Description for LCD

Module	XBee
Inputs	Data, RX, TX
Outputs	Data, wireless
Behavior	• Receive mode – Receive data from wireless signal – Send data to PIC 18F4520 MCU (Transmitter) • Transmit mode – Receive data from PIC 18F4520 MCU (Tank) – Send data as wireless signal

Table 9: Level-1 Description for Tank XBee

1.2 Level-2

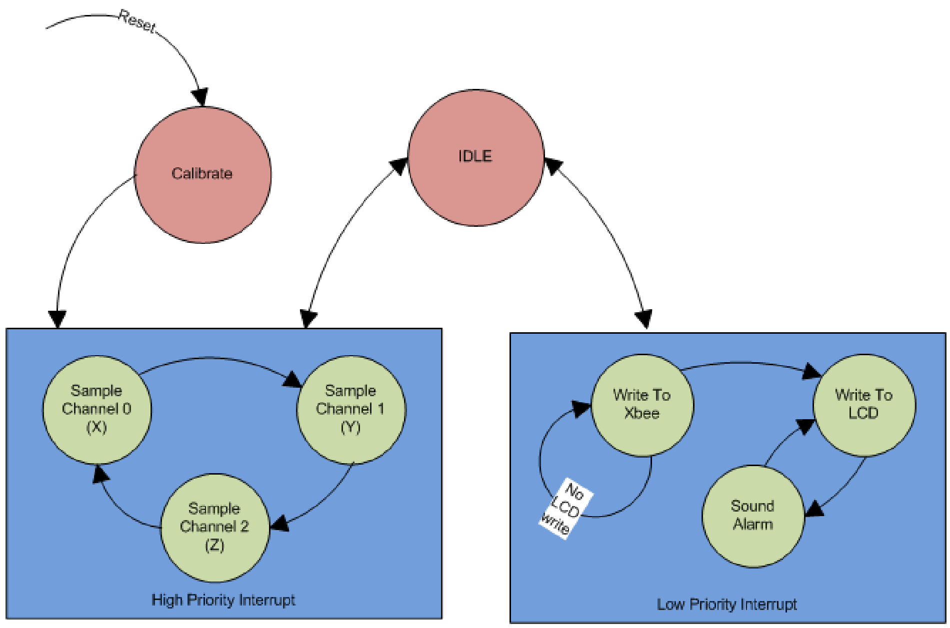

Figure 2: Level-2 FSM for User Transmitter

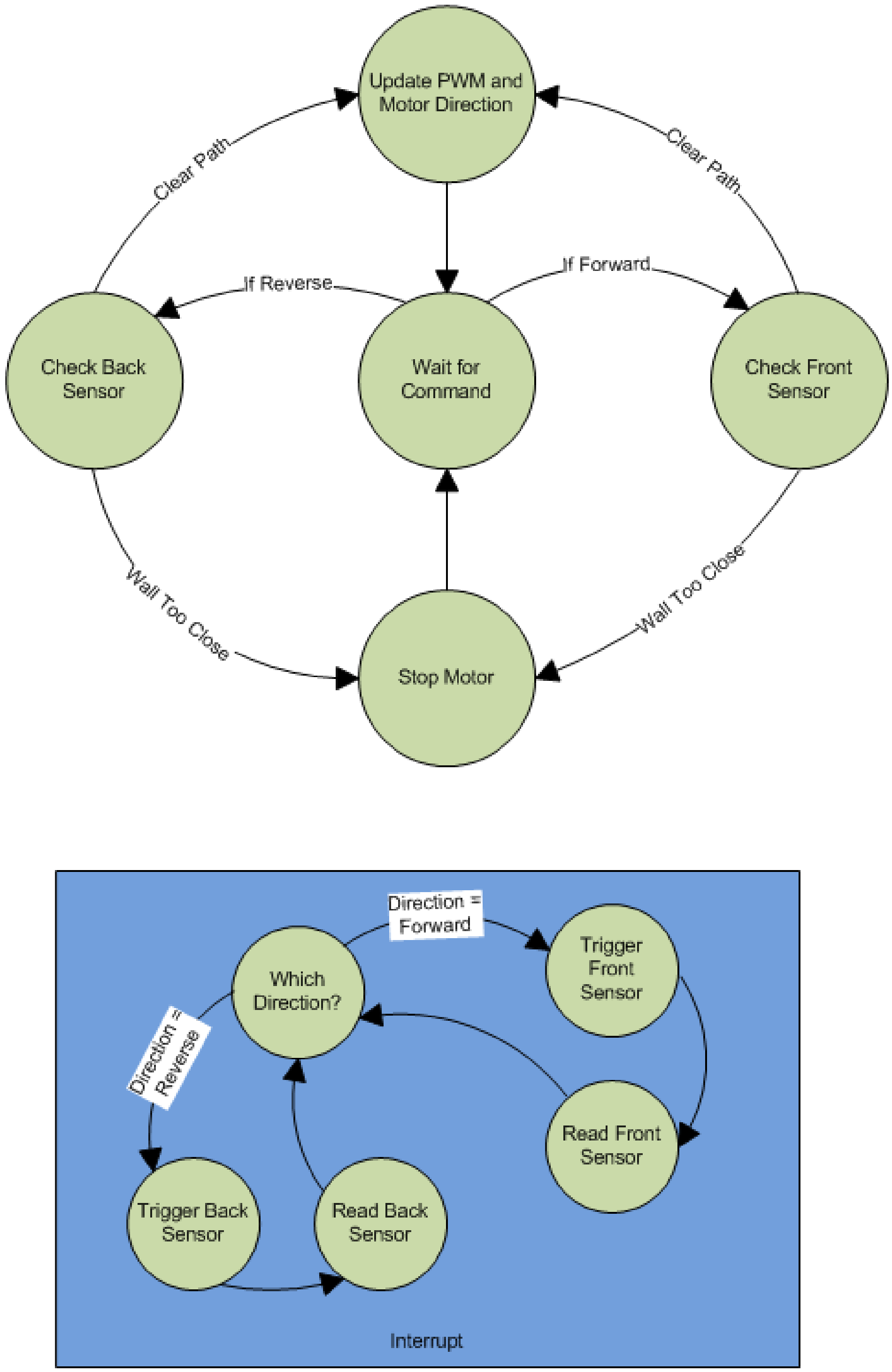

Figure 3: Level-2 FSM for R/C Tank receiver

1.2.1 Communication protocol

A simple communication protocol is established so that both the user interface and the tank may be developed as independent entities. To minimize the parsing, a deterministic context free grammar is chosen to represent instructions to and feedback from the tank. Our vocabulary of non-terminating characters is

$$V_{NT} = \{\mathrm{R}, \mathrm{L}, \mathrm{F}, \mathrm{B}, \mathrm{M}, \mathrm{S}, ?, \mathrm{f}, \mathrm{r}, n \in \mathbb{Z}_0^9\},$$

with the disjoint terminating character

$$V_T = \{;\}.$$

If multiple instructions are transmitted simultaneously, then the start condition immediately follows the terminating character. The start character $S \in V_S$, where

$$V_S = \{\mathrm{R}, \mathrm{L}, \mathrm{F}, \mathrm{B}, \mathrm{S}, ?\}.$$

Thus, our grammar G is

$$G = \{V_{NT}, V_T, V_S\}.$$

1.2.2 Naming conventions

The following conventions were adhered to during the development of the grammar so that maximal readability is established for the instructions at the development level.

- **Location:**
  - "R" for 'right'
  - "L" for 'left'
  - "F" for 'front'
  - "B" for 'back'
- **Component:**
  - "M" for 'motor'
  - "S" for 'sensor'
- **Request:**
  - "?" for 'request'
- **Adjective:**
  - "f" for 'forward'
  - "r" for 'reverse'

- **Decimal value:**
  - integers, $n \in \mathbb{Z}_0^9$, are allowed to represent decimal values.

1.2.3 Language subset

The language is restricted by the deterministic constraint placed on the context free grammar. These restrictions are governed by the following semantic rules

- Location must come first during instruction; unless it is a request, then both sensors are sent back.
- Component always follows the location.
- This may be followed by an adjective.
- And can be followed by a decimal value.
- Must end with semicolon;

2 Milestone I

The goal for milestone I is to get the major functional blocks working. We do not plan to have these blocks working together, only functioning independently.

2.1 XBee

1. Transmit Data ✔
2. Receive Data ✔
3. This will be assisted by eavesdropping using hyper-terminal ✔

Each XBee transmitter/reciever was setup with X-CTU[1] by configuring the XBee's in command mode as instructed by Mechatronics Wiki[2] With no program installed immediate communication is achieved over wireless USART using transparent wireless transmission.

2.2 Accelerometer

1. Generate usable values from the accelerometer by adjusting the sample rate ✔
2. Experimentally check for the range of values that can be read from each axis ✔
3. Allow the accelerometer to be calibrated to any initial position ✔

The accelerometer is tested by writing an interrupt service routine (ISR) that samples eight values from each of the x, y, and z vector directions. This is to ensure that the sampling frequency was great enough and was not being slowed down by other programming tasks. After the total of 24 values were recorded into the x, y, and z arrays, a function is called from the ISR that disables interrupts (so that new values do not alter this test) and prints out the collected results on Hyper Terminal. Once all of the values are printed, interrupts are enabled so that this process continues indefinitely.

The following test was performed on the calibrated output of the accelerometer. Initially the accelerometer was calibrated in the static state sitting on a table; increments based on a degree measurement. The axis's x and z are measured as shown in Figure 4, and y is measured as shown in Figure 5.

Results from this experiment are shown in Figure 6. It is noted in the test that the x and y values correspond for their respective angles. **NOTE: Enter a table environment, rather than a figure environment for the experimental results.**

[1]Software downloaded from manufacturers website: http://www.digi.com/support/productdetl.jsp?pid=3352&osvid=57&tp=4&s=316. Accessed 12 April 2009.

[2]XBee_radio_communication_between_PICs. Mechatronics Wiki. http://hades.mech.northwestern.edu/wiki/index.php/XBee_radio_communication_between_PICs. Accessed 12 April 2009.

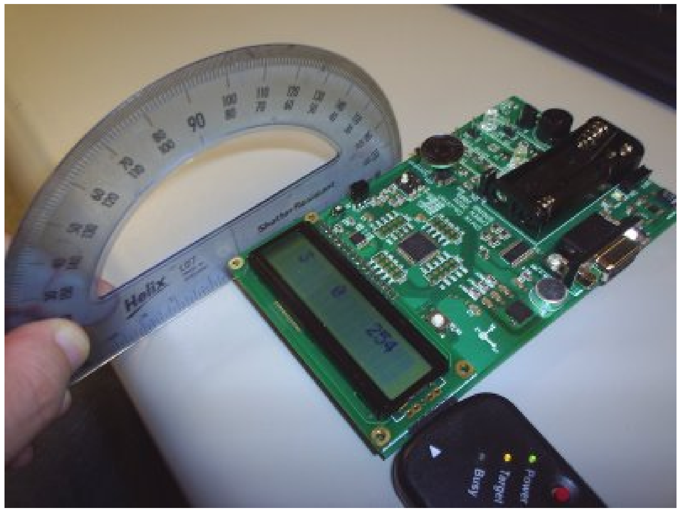

Figure 4: Setup for accelerometer experiment for the x and z axes

Figure 5: Setup for accelerometer experiment for the y axis

Degree	X_calibrated	Y_calibrated	Z_calibrated
0	0	0	255
15	238	235	
30	223	224	
45	210	213	
60	198	202	
75	194	196	57
90	192	193	
105	195	195	
120	203	203	
135	213	216	
150	223	224	
165	234	241	
180	253	1	124

Figure 6: Experimental results measured for calibrated accelerometer at various angles

2.3 H-Bridge Network

1. Generate individual motor speeds and direction ✔
2. Remove any heat dissipation problems ✔

The H-bridge network is tested by wiring the motors to match the wiring diagram in the datasheet for 754410 chip. The enable bits are fed the PWM from the PIC. The H-bridge reads the amount the pulse is high as an average voltage to determine the speed of the motor. If the pulse is high for 50% of the period, the equvalent voltage is 2.5 V. This is tested by assigning different pulse widths and looking for a change in speed. We found the H-bridge networks to be functioning properly.

2.4 Ultrasonic Sensors

1. Read pulse length using interrupt on pin change ✔
2. Develop formula for translating pulse length to distance (inch or centimeter) ✔

To test the ultrasonic sensors, an interrupt had to be written to trigger the sequnce of events needed to activate and then read the sensor. The first stage of the interrupt is to send a 10 μs pulse to the trigger pin of the sensor. This activates the interal circuitry allowing the sensor to return a pulse on the echo pin that is proportional to the distance of the object detected. To read the length of the pulse, Timer1 is cleared when the echo goes high and read when the echo returns low. The sampled Timer1 value is then diplayed on Hyper Terminal. The ultrasonic sensors functioned properly with very stable distance readings.