Accelerometer Controlled R/C Tank
A final project plan for CMPEN 352W

by

David Carley ______________________

Bryan Frantz ______________________

April 8, 2009

1 Proposal

1.1 Need Statement

Current motion controlled input devices lack a sense of customizable control. Users of these devices, such as the iPod Touch and Wiimote, are forced to adapt to a strict range of motion.

Allowing a user to calibrate a more comfortable or intuitive starting position, their experience is improved by reducing the movement required to accomplish an action[1]. Designing a motion controlled input that can be calibrated to conform to the user will not only improve interaction, but allows for a platform to develop motion controlled input applications.

Such applications could be more intuitive interaction with video games [2], allowing a disabled person with limited motion the ability to control some external device, or improving the ergonomics of any motion controlled input device that has a corresponding kinetic action[3].

1.2 Marketing Requirements

- Control should be intuitive
- Calibration should be allowed for multiple control styles
- The tank should respond how the user expects it to respond

1.3 Level-0 Description

[1] Fitts,P.M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology, 47, 381-391

[2] The video on our development blog shows that calibration is required for the user to have an intuitive control with the machine "http://352w.blogspot.com/2009/04/accelerometer-calibration.html"

[3] What is ergonomics and why is ergonomics important? "http://www.safetynewsandreviews.co.uk/article.asp?c=21" Accessed 8 April 2009.

Module	User transmitter
Inputs	User motion, XBee wireless interface
Translator	MMA7260 Accelerometer
Outputs	XBee wireless interface, LCD

Behavior	`__only on startup`
	`    get accelerometer data`
	`    calibrate to user relative position`
	`__end startup`
	
	`Priority of RTOS`
	`  * (1) Accelerometer`
	`        - get values from accelerometer`
	`        - form instruction based on new [x,y,z]`
	`  * (2) Send data`
	`        - send motor instructions to tank receiver`
	`  * (3) Request data`
	`        - periodically send request for sensor data`
	`        - wait for interrupt of response`
	`        - parse response packet`
	`        - update global values for front and back sensor`
	`  * (4) Update LCD`
	`        - write the global values of front and back sensor to LCD`
	`  * (5) sound alarm`
	`        - when front or back sensor is within distance threshold`
	`             send alarm signal`

Table 1: Level-0 Description for Transmitter

Module	R/C tank receiver
Inputs	Front and back ultrasonic range sensors, XBee wireless interface
Outputs	Right and Left motor, XBee wireless interface

Behavior	`stay in a idle state until a command is received`
	
	`if(motor control)`
	`  update PR1 and PR2`
	`  update motor direction`
	`if(sensor request)`
	`  transmit sensor data`
	
	`wait for next instruction`
	
	`ISR's will handle PWM and sensor control`

Table 2: Level-0 Description for R/C Tank

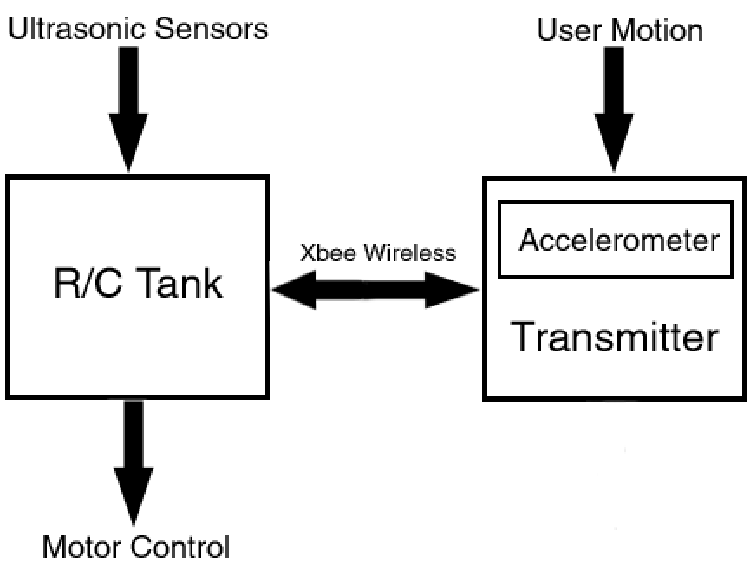

Figure 1: Level-0 Graphical Description

2 Detailed Architecture

2.1 Level-1

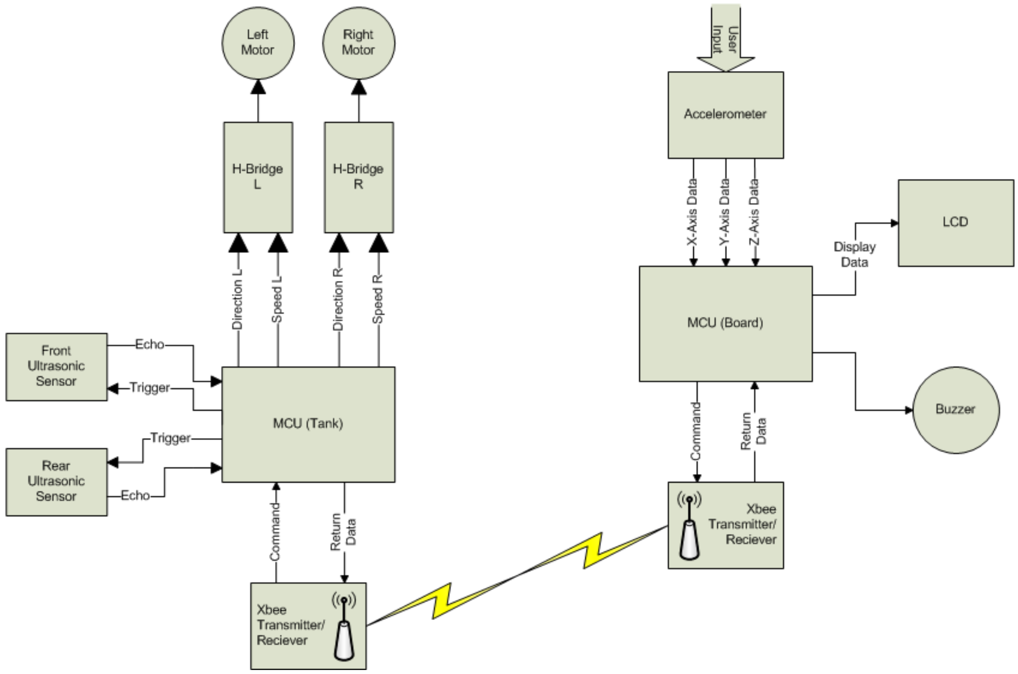

Figure 2: Level-1 Block Diagram

2.1.1 User side

Module	PIC 18F4520 MCU
Inputs	Accelerometer, XBee
Outputs	LCD, Piezo speaker, XBee

Behavior	`CONTROLLED BY RTOS`
	`Finite state machine in Level-2`
	
	See Figure (3).

Table 3: 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 4: 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 5: Level-1 Description for LCD

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

Table 6: 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 7: Level-1 Description for XBee

2.1.2 Tank side

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

Behavior	<pre>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</pre>

Table 8: Level-1 Description for PIC 18F4520 MCU Tank

Module	Ultrasonic Sensor
Inputs	Trigger from PIC18F4520 MCU (Tank)
Outputs	Echo Pulse
Behavior	10us 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 9: Level-1 Description for Ultrasonic Sensor

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

Table 10: 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 11: Level-1 Description for Tank XBee

2.2 Level-2

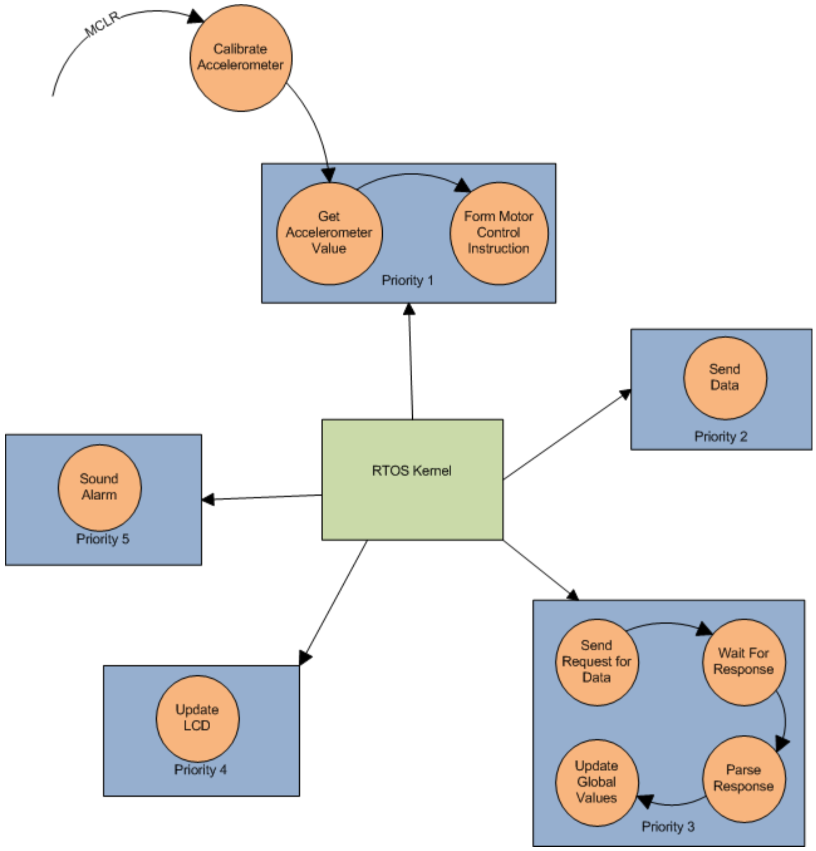

Figure 3: Level-2 FSM for User Transmitter

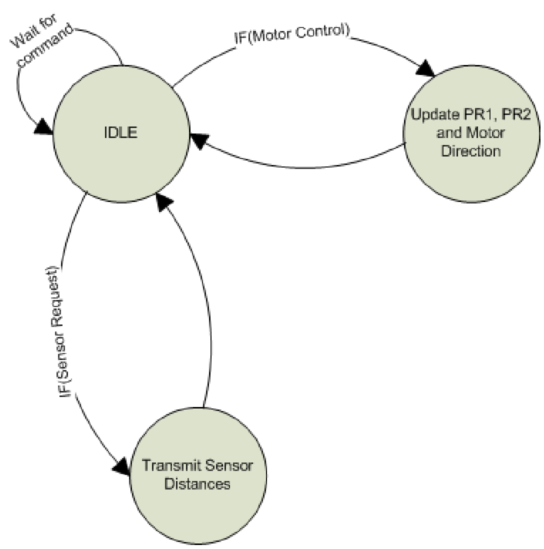

Figure 4: Level-2 FSM for R/C Tank receiver

2.2.1 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

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

2.2.2 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.

2.2.3 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;

3 Calculations

- Ultrasonic sensor distances into centemeter or inch
- More to come ...

4 Technical Requirements

- Layout of circuit in tank reciever must fit within the tank
- Power rails are limited to 5 v to reduce speed and power consumption, therefore increasing battery life
- Sample accelerometer at $\geq$ 11 kHz
- More to come ...

5 Bill of Materials

All materials have been purchased in the past or are being supplied by the school

- 2 PIC18F4520 Chips
- 1 MMA7260 Accelerometer
- 2 XBee Wireless Transmitter/Receivers
- 2 SRF04 Ultrasonic Sensors
- 1 Modified R/C Tank
- 1 CRYSTALFONTZ LCD
- 4 SN754410NE Quadruple Half-H Drivers
- 1 MAX232N Dual ELA-232 Driver (DIP Package)
- 1 MAX3232CSE ELA-232 Driver (Surface mount)

- 1 TPA511NORMAL Joy Stick
- 1 CD4069 Inverter Circuit
- 1 F/CM12P Buzzer
- 1 Null-Modem to AMP9806 Connector
- 4 LM7805AC 5v Regulars
- 2 XBee Breakout Boards
- 1 PicKit2

6 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.

- XBee
  - Transmit Data
  - Receive Data
  - This will be assisted by eavesdropping using hyper-terminal
- Accelerometer
  - Generate usable values from the accelerometer module with filters and by adjusting the sample rate
  - Experimentally check for the range of values that can be read from each axis
  - Allow the accelerometer to be calibrated to any initial position
- H-Bridge Network
  - Generate individual motor speeds and direction
  - Remove any heat dissipation problems
- Ultrasonic Sensors
  - Read pulse length using interrupt on pin change
  - Develop formula for translating pulse length to distance (inch or centimeter)

7 Milestone II

The goal of milestone II is to get the individually functioning blocks integrated as a functioning system.

- Send commands with XBee from spring board to R/C tank.
  - Use a deterministic context free language

  - Calculate motor speed and direction using values from the accelerometer
  - Request sensor data via interrupt
  - Use the R/C tank to interpret command, perform and respond if needed.
- Interpret XBee commands and perform them on the R/C tank
  - Each command needs to be parsed to fulfill the request
  - The request being performed assigns any registers or interrupts needed.
  - If the command is a request for data, the data packet will be transmitted over the XBee module
- Use Accelerometer for intuitive user control
  - Accelerometer is set to the users desired initial position
  - The values from the accelerometer are then generated using this offset
- Read distance from the ultrasonic sensors and return data
  - Use an ISR to initialize, capture and return data.
  - The captured value is updated on a regular interval
  - When the distance is requested, the most recently sampled distances are transmitted back with the XBee module