UNC CHARLOTTE

IEEE SoutheastCon 2007

Hardware Competition

Richmond Virginia 27 March 2007

Santhosh Ramani

ECGR 6185 Advanced Embedded Systems

UNC CHARLOTTE

Competition Rules:

1. Robot should be able to follow a white line.
2. Shoot ping pong (table tennis) balls through a Hoop.
3. Get data serially through an IR receiver and display the required part of the code received.
4. The dimension for the Robot should not be greater than 12”x12”x12” at the start of the match.
5. The playing court was 4’ x 12’ (full size) and 2 teams would compete at a time.
6. The team that scored the most points was declared the winner.
7. The robot should be self powered and autonomous. No pneumatic or hydraulic parts can be used.

Different Zones of the Playing Field

1. There are 3 zones on each half of the playing field.

2. Start when the Red LED at the center of the Start zone glows.

3. Each team can get additional balls from the Rebound zone. (max no. of balls per match was 30)

4. Get code from the Ball Request zone and then display the 3 digit number.

5. Sound the alarm once you reach the Rebound zone, and if the number displayed is correct the team gets additional balls.

6. The number is transmitted every 1sec and transmission starts as soon as any part of robot enters the Ball Request zone and changes when it leaves.

Capabilities:

1. Ability to shoot 3 ping pong balls in the shortest time possible.

2. Navigate using the white-line (sensor to detect white line and software for deciding the path).

3. Have the circuitry to get the code serially with help of an IR receiver.

4. To extract the 3 digits from “ieee*xyz*ieee” received serially.

5. Display the numbers and sound the buzzer once in the Rebound zone.

6. It was an Autonomous vehicle and was within the size restriction mentioned.

Mechanical Aspect:

1. Robot was driven by 4x4 drive with 4 motors for each wheel.

2. The shooter was a spring system with gears powered by a DC motor (could have also been powered by Stepper motor).

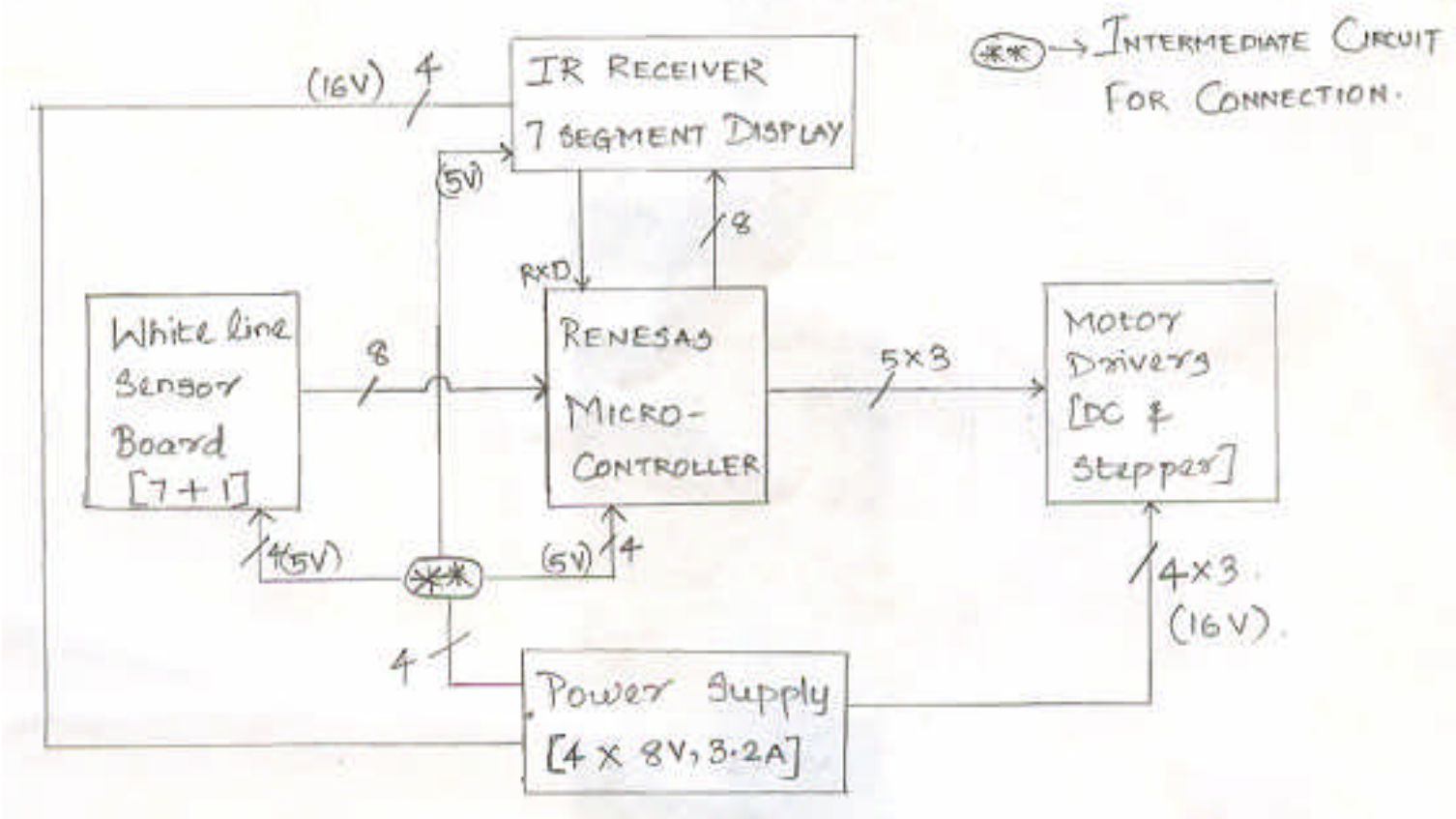
3. The mechanism was such that the ping pong balls could be shot continuously.

4. It used 2 gears, one was rotated by the motor which was coupled to the second gear connected to the spring.

5. It could shoot 3 ping pong balls in less than 10 seconds and the motion was continuous.

Electronics Aspect:

1. The Block Diagram of the Robots is as follows:



Electronics Aspect:

1. The Micro-controller used was Renesas on the SKP M16C/62P board.

2. The circuitry used could be divided into 3 categories:
   a. Motor driver circuits.
   b. Serial communication circuit (including display).
   c. White-line sensor board.

3. All the circuits were designed using Eagle 4.16r.

4. All the code was written using HEW 4.

5. The battery used was a NimH (8V 3200mA) and each circuit had its own voltage regulator to provide exact voltage.

Electronics Aspect:

1a. DC Motor Driver Circuit:

i. Used LMD18245 which is a 3A, 55V DMOS Full-Bridge Motor Driver

ii. The circuit also has a 7812 and 7805 voltage regulator for 12V and 5V supply.

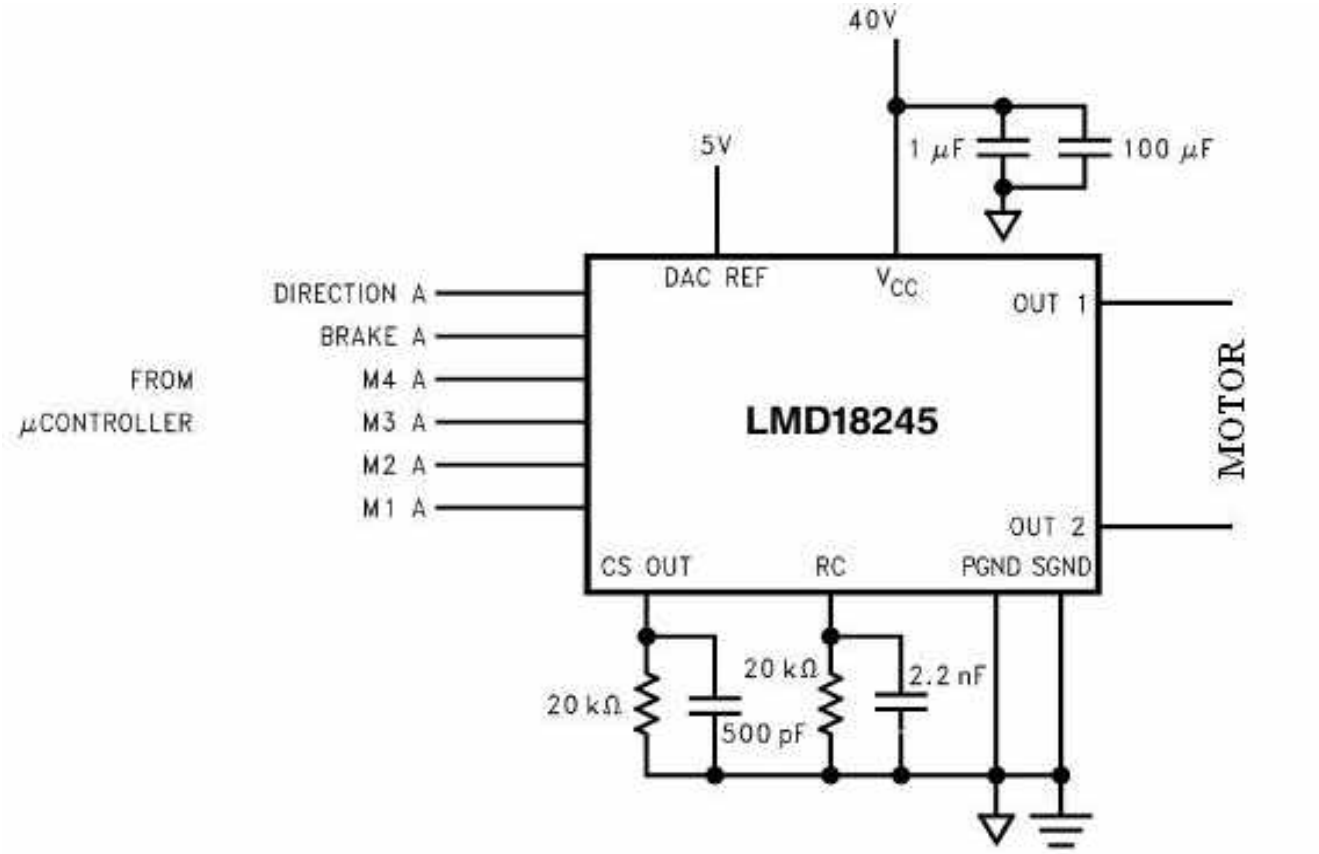
iii. The motors were run at 12V and the 5V signals were used for reference required by the chip and for testing purposes.

iv. Every LMD can be used to operate a single motor at 15 different speed levels and in both directions.

v. The speed is controlled by the chopping effect of the supply from the LMD to the motor.

Electronics Aspect:

Connections for LMD18245



Electronics Aspect:

1b. Stepper Motor Driver Circuit:

i. The circuit used 2 LMD18245 and can be used for driving 4,5 or 6 wire stepper motors.

ii. For 5,6 wire stepper motors the common supply required was supplied through TIP122 controlled by the micro-controller.

iii. The direction of motion and steps taken is controlled by the ON/OFF and direction signals given to both the LMDs in a synchronized fashion.

iv. The stepper can be rotated using Full Step, Half Step or Micro Step mode by changing the timing to each LMD.

v. The voltage can range from 0-40V and current rating is 3A so it can drive most Stepper motors which usually have a 2A current rating.

Electronics Aspect:

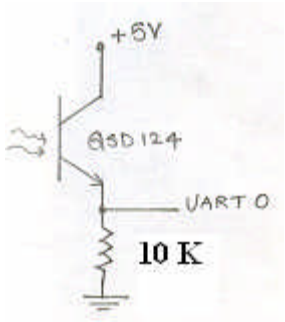
2a. Serial communication circuit

i. Used QSD 124 which is a Infrared phototransistor (QED123 and QSD124 are IR transmitter-receiver pair)

ii. The phototransistor was used at 5V and the output is given directly to the Micro-controller

iii. UART0 was used at a 2400 baud rate and with 8-N-1 protocol.

iv. Circuit Diagram:



Electronics Aspect:

2b. Display circuit

i. We used multiplexed 7 Segment LEDs (Common Anode).

ii. The 7 segement LEDs were driven using 74ls47 which is a 7-Segment Decoder/Driver with Open-Collector Outputs.

iii. The input from the micro-controller was in the BCD format.

iv. 3 more pins were used for multiplexing and each 7 segment LED display was switched on and off within 1/10th of a second.

v. This was achieved by using a timer interrupt switch executed a ISR which refreshed the display.

vi. Due to the current requirement the displays were powered through a TIP122 controlled by the micro-controller at 5V.

UNC CHARLOTTE

Electronics Aspect:

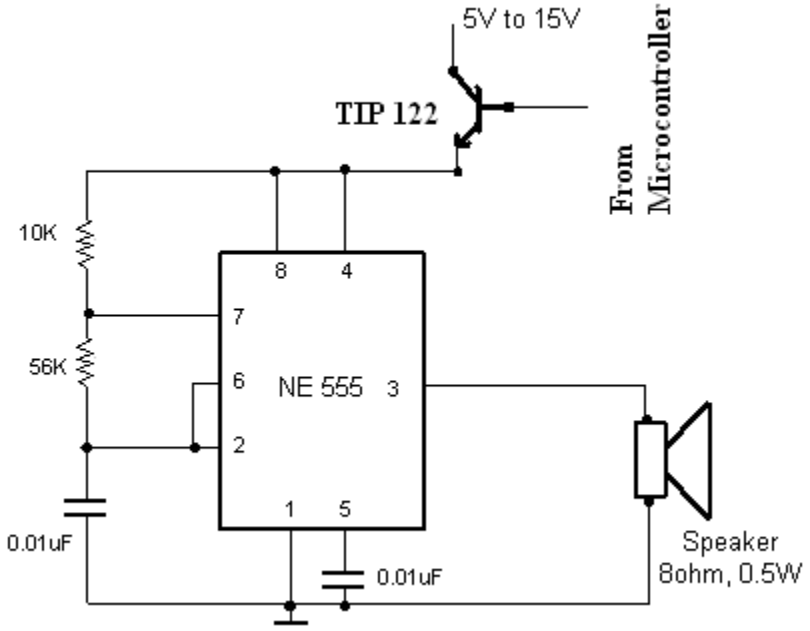
2c. Buzzer Circuit

i. The circuit used a 555 Timer in Astable multivibrator operating at a frequency of about 1kHz.

ii. TIP122 was used as a switch to start/stop the Buzzer.

iii. The frequency of operation could be changed by changing the 10K resistor.

iv. The 555 timer in this case was used at 5V.



Electronics Aspect:

3. White-line sensor board:

i. The sensor used was OPT101 from Texas instruments.

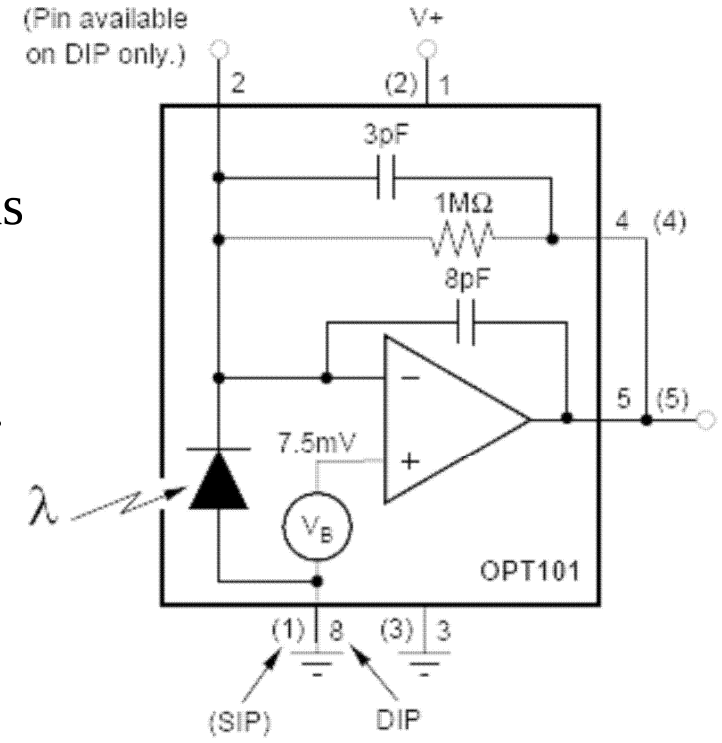
ii. It is a monolithic photodiode with on-chip transimpedance amplifier.

iii. It was used in conjunction with Pirana Leds and both were operated at 5V.

iv. The output for white surface was 1.5-4.0V.

v. There were 7 OPT101-LED pair at the front of the robot used for navigation.

vi. One pair was used at the center to start.



Electronics Aspect:

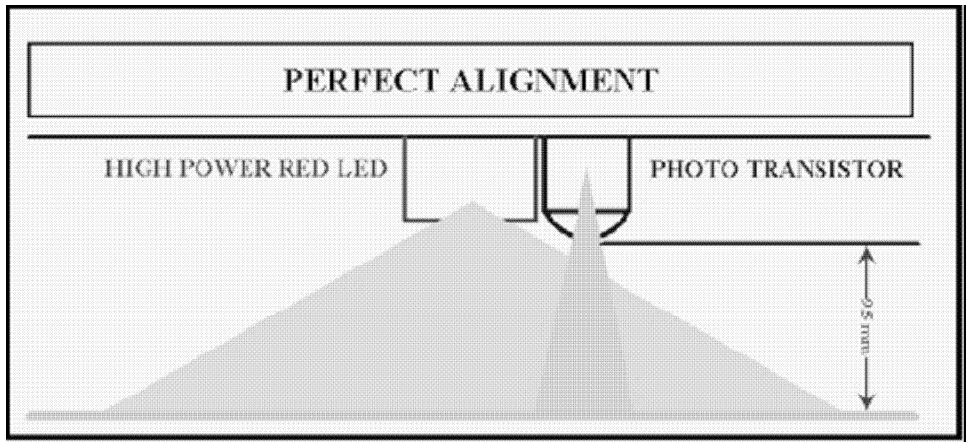
3. White-line sensor (Principle):

i. The principle of operation is based on difference in the intensity of light reflected from a surface.

ii. White surface→ more light reflected→ higher intensity→ higher output voltage.

iii. The reverse is true for the black surface and this relation is linear.

iv. The effect of ambient light is completely removed by the use of ADC allowing software threshold adjustment for different surfaces.



Voltage supply and Control signals:

1. Each circuit was directly supplied from the batteries and hence each circuit had its own voltage regulator and protection circuit.

2. This made all circuits independent. (useful for testing and debugging).

3. The ports used from the Renesas micro-controller were:
   a. Port 7→ Display

      (4 pins→ BCD data, 3 pins→ Selecting display, 1pin→ Buzzer on/off)

   b. Port 0→ Motor Drivers (front and back wheels)

      (4 pins for each circuit such that [1pin→ ON/OFF, 1pin→Direction]x2 was for each LMD)

   c. Port 2→ Used as ADC using all 8 pins (7 at front + 1 at center)

   d. Port 8→ Used for the Shooter motor (DC motor)

      (Just 2 pins used 1pin→ON/OFF, 1pin→Direction for single LMD)

Software Aspect:

1. Serial Communication

i. The serial communication was achieved using UART0 with reception being interrupt driven.

ii. Once all the characters were recieved the 3 numbers were extracted.

iii. All characters were received once more and checked with previous result.

iii. This incorporated Fault Tolerance (Time redundancy→Repeating Tasks)

iv. Once the received digits were confirmed they were displayed (converted to BCD format).

v. This was achieved with the help on an interrupt every 0.1 sec.

Software Aspect:

2. Motor Driver

i. The motors were tested and driven by using bitwise ouputs that were declared as (*# define)* constants.

ii. These were based on the pins used for each motor driver.

iii. Accurate delays using timers were used to get precise motion in a particular direction.

iii. The functions used provided millisecond and micro second delays.

iv. The shooter was also operated in a similar fashion using delays.

v. This was achieved with the help on an interrupt every 0.1 sec.

Software Aspect:

3. White line (Main)

i. The algorithm was based on moving in a particular direction unless there was a change in the input sensor reading.

ii. When the sensor on the right gave a higher output we turned right and vice versa.

iii. Nodes were reference points and node transition/count was very important.

iii. All functions were synchronized using node count.