Lab 3 (part 3): Steering and Speed Control

Preparation

Reading

Lab Manual

Chapter 4 - The Silicon Labs C8051F020 and the EVB (sections on the *C8051 timer functions, interrupts, and the programmable counter array*)

Chapter 5 - Circuitry Basics and Components (The buffer)

Chapter 6 - Motor Control (DC Motors)

Objectives

Use the sensor readings to control the hardware. The electronic compass pair will use the compass readings to control the steering. The ultrasonic ranger pair will use the ranger readings to control the car speed.

Motivation

In the previous lab, pulse width modulation was introduced to change speed of drive-motor and steering of servomotor. In this lab, the compass and ranger readings will be used to control the steering and the speed of the car respectively.

Lab Description & Activities - Ultrasonic Ranger Pair

1. Use the ranger distance reading to control the drive motor speed in a continuous loop. Inside the loop, poll a run/stop switch connected to Pin 6 of Port 3. If the switch is ON, the drive motor speed should be running. If the switch is OFF, the motor should stop running. The blimp will also have switches on the pins. Create a simple control program where

   – The motor has full power forward if an object is 10 cm or less above the car.
   – The motor is neutral when the object is 40-50 cm above the car.
   – The motor has full power reverse if the closest object is more than 90 cm from the car.
   – The pulse width varies linearly for distances from 10 cm to40 cm, between max forward and neutral pulsewidths.
   – The pulse width varies linearly for distances from 50 cm to 90 cm, between neutral and maximum reverse.

2. Have a TA verify the speed performance of your car.
   – Print the distance and the motor pulse width as you move any object near the ranger.

Lab Description & Activities - Electronic Compass Pair

1. Use the compass reading (heading) to control the steering servo in a continuous loop. Inside the loop, poll a run/stop switch connected to Pin 7 of Port 3. If the switch is ON, the steering servo should turn the car towards desired heading. If it is OFF, the wheels should be parallel to the car. The blimp will also have a switch on P3.7.
   – Install a slide switch on the protoboard and connect it to P3.7.
   – Assume a desired heading, make this a variable so it can be changed later, but fix the value for now. For example:

```
unsigned int desired_heading = 900;
```

   This example sets the heading to 90 degrees, assuming that the units used are 1/10 of a degree.
   – Read the actual heading
   – Use the difference between the actual and the desired heading to set the servo pulse-width using proportional control.

   Notes: consider the following control algorithm:

```
error = desired_heading - actual_heading;
temp_servo_pw = k*(error) + center_pw;
```

   – In the above equation, use a value for `k` that yields a maxim change about center_pw of 750.
   – This algorithm fails if the desired heading is in the range of 270 to 360 degrees and the actual is in the range of 0 to 90, and vice versa. Develop an algorithm that handles these cases.
   – Center_pw is the value obtained using your Lab3 Part 1 code.
   – The variable temp_servo_pw must be checked and adjusted to be within the range determined using your Lab 3 Part 1 code.
   – The CCMn (Capture/Compare Module n) value can then be set by a line of code such as:

```
temp_servo_pw = SERVO_PW;
cex0_lo_to_hi = 0xFFFF - SERVO_PW;//Implement current PWM
```

   – There is not an effective way to determine the best value of k at this point. Adjust the value of k so that steering will respond as the car is turned.

2. Have a TA verify the speed performance of your car.
   – Print the desired heading, the actual heading and the steering pulse width as you rotate to the desired heading.

Lab Check-Off: Demonstration and Verification

1. Complete the entries in your lab notebook as described below and present it to your TA.
2. Your TA may ask you to explain how sections of the C code or circuitry you developed for this exercise work. To do this, you will need to understand the entire system.
3. Be able to explain how the PCA is used to generate the PWM signals.
4. Capture the Hyper-terminal screen showing the following output for at least 5 different values:

   Ultrasonic ranger pair: Print the distance and the motor pulse width.

   Electric compass pair: Print the desired heading, the actual heading and the steering pulse width.

Writing Assignment - Lab Notebook

Your Lab Notebook must be kept up to date in recording the work you and your partner do in the lab. This should include pseudo-code for your system, data graphs, and a copy of your final C code for this lab. Further information on keeping a good Lab Notebook can be found in *LITEC Lab Notebook Requirements - Notebook Requirements*.

Grading-Preparation and Checkoff

Prior to the starting the laboratory you must complete

1) No Worksheet for Laboratory 3-3.
2) The Pin-out form (Port, Interrupt, XBR0, PCA, SMB initializations)
3) The Pseudocode (Revision when finished)

When you are ready to be checked off, the TAs will be looking at the following items

4) That your project performs all the indicated requirements (defined above in ***Lab Description & Activities***)
5) Appropriately formatted and commented source code
6) Clean and neat hardware, with appropriate use of colors for source and ground connections

Additionally, you will be asked a number of questions. The questions will cover topics such as

7) Understanding algorithms in the code, identifying locations in the software that perform specific actions, understanding the hardware components, understanding the test equipment

The final item that will be included in the Laboratory grade is your

8) Notebook.

The above 8 items each have an individual contribution to your Laboratory grade.