Complier directives

#include<c8051\_SDCC.h>

#include <stdio.h>

#include<stdlib.h>

#define PW\_MIN \_\_\_\_\_

#define PW\_MAX \_\_\_\_\_

#define PW\_NEUT \_\_\_\_\_

Function Prototypes

Void Port\_Init(void);

Void Timer\_Init(void);

Void Interrupt\_Init(void);

Void Timer0\_ISR(void) \_\_interrupt 1;

void PCA\_Init (void)

void XBR0\_Init(void)

void drive\_motar(void)

void steering servo(void)

void LEDblink(void)

Global variables

Sbit LED0 BUZZER SLDSW

unsigned int MOTOR\_PW = 0;

unsigned int steering-servo

unsigned int LED brightness

Main function

Declare local variables

(none)

Initialize function

Sys\_Init();

putchar(‘ ‘); //the quotes in this line may not format correctly

Port\_Init();

XBR0\_Init();

PCA\_Init();

Print some message to indicate start

Begin infinite loop

Drive motor

End main function

Void drive motor(void){

Initialize speed controller , need to leave it at that value for one second(use counter at PCA ISR)

USE BUZZER during initialization time

Get an input char from keyboard

If it is s change the pulsewidth signal and reduce the engine power

If it is f change the pulsewidth signal and increase the engine power

Remember The period must be 20ms and the pulsewidth must be initialized to be 1.5ms.

And use sysclk/12 and 16 bit counter

Pulsewidth should bigger than 1.1ms and smaller than 1.9ms

}

void steering servo(void){

Initialize steering servo

USE BUZZER during initialization time

Get and input char from keyboard

If it is r increase the steering pulsewidth by 10 (turn right more)

If it is l decrease the steering puslswidth by 10(turn left more)

Remember The period must be 20ms and the pulsewidth must be initialized to be 1.5ms.

use sysclk/12 and 16 bit counter

Initial estimates of the left and right limits are 0.9 [ms] and 2.1 [ms].

}

Void LED(void){

Initialization (almost same as the 2 function before)

USE BUZZER during initialization time

Get and input char from keyboard

If it is b turn the LED BRIGHTER

If it is d turn the LED DIMMER

//Remember The period must be 20ms and the pulsewidth must be initialized to be 1.5ms.

//use a Crossbar setting XBR0=0x27 with a pulsewidth signal on CEX3

//use sysclk/12 and 16 bit counter

//The pulsewidth should never be less than 1[ms] or greater than 19[ms].

}

Void Port\_init(void)

{

Set up port pin 1

}

Void Timer\_Init(void){

Set up Timer

}

Void Interrupt\_Init(void) \_\_interrupt 9{

Set up interrupt

}