Complier directives

#include<c8051\_SDCC.h>

#include <stdio.h>

#include<stdlib.h>

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 read\_driver(void)

void readcompass(void)

void readLED (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

Motor task or compass task or LED task

End main function

Ranger task

//we need to wait 80ms(different from compass) in the main function

after 80ms

call read ranger function

start a ping

reset the 80ms flag

print the range

compass task

wait 40ms

call read compass

start a ping

reset the 40ms flag

print the compass

LED task

read the ranger

start a ping

reset the 80ms flag

print the light

other important functions

unsigned int ReadRanger() {

unsigned char Data[1];

unsigned int light = 0;

unsigned char addr=0xE0; // the address of the ranger is 0xE0

i2c\_read\_data(addr, \_\_, Data, \_); // read one byte, starting at reg 1

light = Data[0] return light;

}