File needed

#include <c8051\_SDCC.h>

#include <stdlib.h>// needed for abs function

#include <stdio.h>

#include <i2c.h>

Function prototype

void Port\_Init(void);

void PCA\_Init(void);

void SMB\_Init(void);

void ADC\_Init(void);

void Interrupt\_Init(void);

void PCA\_ISR(void) \_\_interrupt 9;

int read\_compass(void);

void set\_servo\_PWM(void);

int read\_ranger(void); // new feature - read value, and then start a new ping void set\_drive\_PWM(void);

int pick\_heading(void); // function which allow operator to pick desired heading

//define global variables

unsigned int PW\_CENTER = \_\_\_\_\_;

unsigned int PW\_RIGHT = \_\_\_\_\_;

unsigned int PW\_LEFT = \_\_\_\_\_;

unsigned int SERVO\_PW = \_\_\_\_\_;

unsigned int SERVO\_MAX = \_\_\_\_\_;

unsigned int SERVO\_MIN = \_\_\_\_\_;

unsigned char new\_heading = 0; // flag for count of compass timing

unsigned char new\_range = 0; // flag for count of ranger timing

unsigned char print\_flag = 0; // flag for count of printing

unsigned int heading;

unsigned int range;

unsigned int light;

int compass\_adj = 0; // correction value from compass

int range\_adj = 0; // correction value from ranger

unsigned char r\_count; // overflow count for range

unsigned char h\_count; // overflow count for heading

unsigned char print\_count; // overflow count for printing

\_\_sbit \_\_at \_\_\_\_ RUN // a slide switch

Main function

Declare local variables

None

Funcion initialization

Do infinite while loop

If the run switch is stop position

Set the motor stop

Set the steer parallel to the car

Else

pick desired heading and range

If different heading

Read heading

Adjust servo PW

Flag to off

If different range

Read range (also the led data)

Flag off

If an object detected

Adjust steering PW

Adjust speed(motor PW)

End main function

Other function

(most of them use the LAB2 / LAB3 functions)