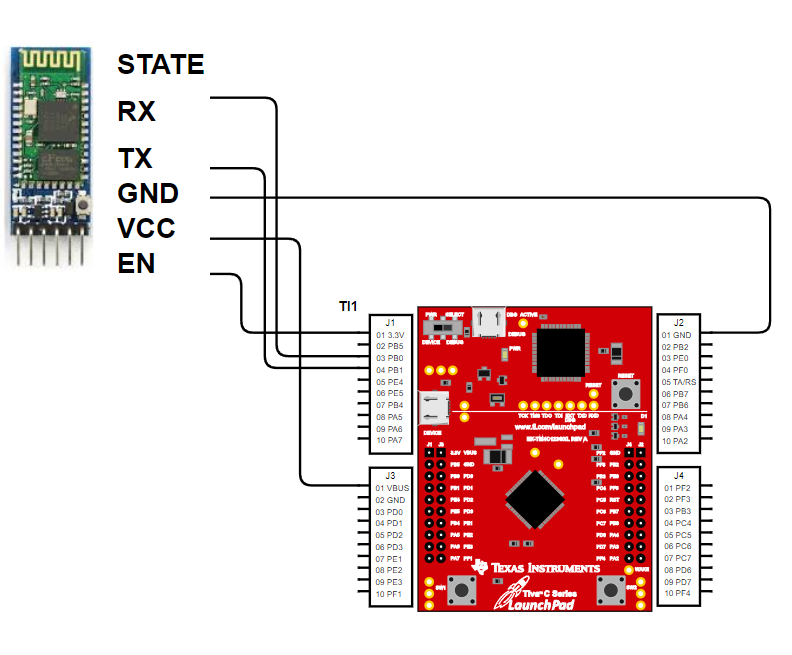
CECS447 Project 3 Bluetooth module

Introduction: In this project I will use the HC05 bluetooth module to make a bluetooth communication system between the TM4C123 MCU and PC.

Circuit diagram：



I used three GPIO Port in this project:

Port A: UART0, the communication between the PC and MCU.

Port B: UART1, the communication between the Bluetooth and MCU.

Port F: on board LED.

Instructions：

1.download the Bluetooth initializer with the EN pin connect with the MCU and Bluetooth module

2.run the program and then turn off the MCU and disconnect the EN pin of the HC05 Bluetooth module.

3.turn the MCU back on, use the PC to search the HC05 Bluetooth

4.after PC connected to the HC05, down the control program to the MCU

5.use the Bluetooth serial terminal to send the instructions to the Bluetooth module.

Code:

setupBLT.c

#include "tm4c123gh6pm.h"

#include "UART.h"

#include "string.h"

void Delay(void);

// main function for programming BT device with no UI

int main(void) {

char String[30];

UART\_Init();

// setup the HC-05 bluetooth module

UART0\_OutString("AT+NAME=KaiwenLiu\r\n"); // Name = Kaiwen Liu

UART1\_OutString("AT+NAME=KaiwenLiu\r\n"); // Name = Kaiwen Liu

while ((UART1\_FR\_R&UART\_FR\_BUSY) != 0){};

BLT\_InString(String);

UART0\_OutString(String);

UART0\_OutString("AT+UART=57600,0,1\r\n"); // baud rate = 57600, 1 stop bit, odd parity

UART1\_OutString("AT+UART=57600,0,1\r\n"); // baud rate = 57600, 1 stop bit, odd parity

while ((UART1\_FR\_R&UART\_FR\_BUSY) != 0){};

BLT\_InString(String);

UART0\_OutString(String);

UART0\_OutString("AT+PSWD=0825\r\n"); // Pass = 0825

UART1\_OutString("AT+PSWD=0825\r\n"); // Pass = 0825

while ((UART1\_FR\_R&UART\_FR\_BUSY) != 0){};

BLT\_InString(String);

UART0\_OutString(String);

UART0\_OutString("AT+ROLE=0\r\n"); // Mode = Slave

UART1\_OutString("AT+ROLE=0\r\n"); // Mode = Slave

while ((UART1\_FR\_R&UART\_FR\_BUSY) != 0){};

BLT\_InString(String);

UART0\_OutString(String);

// query the HC-05 bluetooth module

UART0\_OutString("AT+UART?\r\n"); // baud rate = 57600, 1 stop bit, odd parity

UART1\_OutString("AT+UART?\r\n"); // baud rate = 57600, 1 stop bit, odd parity

while ((UART1\_FR\_R&UART\_FR\_BUSY) != 0){};

BLT\_InString(String);

UART0\_OutString(String);

UART0\_OutString("AT+PSWD?\r\n"); // Pass = 0825

UART1\_OutString("AT+PSWD?\r\n"); // Pass = 0825

while ((UART1\_FR\_R&UART\_FR\_BUSY) != 0){};

BLT\_InString(String);

UART0\_OutString(String);

UART0\_OutString("AT+ROLE?\r\n"); // Mode = Slave

UART1\_OutString("AT+ROLE?\r\n"); // Mode = Slave

while ((UART1\_FR\_R&UART\_FR\_BUSY) != 0){};

BLT\_InString(String);

UART0\_OutString(String);

UART0\_OutString("AT+NAME?\r\n"); // Name = Kaiwen Liu

UART1\_OutString("AT+NAME?\r\n"); // Name = Kaiwen Liu

while ((UART1\_FR\_R&UART\_FR\_BUSY) != 0){};

BLT\_InString(String);

UART0\_OutString(String);

while (1) {}

}

#include "PLL.h"

#include "UART.h"

#include "tm4c123gh6pm.h"

#define GPIO\_PORTF\_DATA\_R (\*((volatile unsigned long \*)0x400253FC))

#define GPIO\_PORTF\_DIR\_R (\*((volatile unsigned long \*)0x40025400))

#define GPIO\_PORTF\_AFSEL\_R (\*((volatile unsigned long \*)0x40025420))

#define GPIO\_PORTF\_PUR\_R (\*((volatile unsigned long \*)0x40025510))

#define GPIO\_PORTF\_DEN\_R (\*((volatile unsigned long \*)0x4002551C))

#define GPIO\_PORTF\_LOCK\_R (\*((volatile unsigned long \*)0x40025520))

#define GPIO\_PORTF\_CR\_R (\*((volatile unsigned long \*)0x40025524))

#define GPIO\_PORTF\_AMSEL\_R (\*((volatile unsigned long \*)0x40025528))

#define GPIO\_PORTF\_PCTL\_R (\*((volatile unsigned long \*)0x4002552C))

#define SYSCTL\_RCGC2\_R (\*((volatile unsigned long \*)0x400FE108))

char lettersend;

//---------------------OutCRLF---------------------

// Output a CR,LF to UART to go to a new line

// Input: none

// Output: none

void OutCRLF(void){

UART\_OutChar(CR);

UART\_OutChar(LF);

}

//debug code

void UART1\_Init(void){

SYSCTL\_RCGCUART\_R |= 0x0002;//activate UART1

SYSCTL\_RCGCGPIO\_R |= 0x0002;//activate port B

UART1\_CTL\_R &= ~0x0001; //disable UART

UART1\_IBRD\_R = 17; // IBRD = int(16,000,000 / (16 \* 57600)) = int(17.3611111)

UART1\_FBRD\_R = 23; // FBRD = round(3611111 \* 64) = 27

UART1\_LCRH\_R = 0x0070;//8bit length,enable FIFO;

UART1\_CTL\_R = 0x0301;//enable RXE,TXE and UART

GPIO\_PORTB\_AFSEL\_R |= 0x03; // alt function on PB1-0

GPIO\_PORTB\_DEN\_R |= 0x03;

GPIO\_PORTB\_PCTL\_R = (GPIO\_PORTB\_PCTL\_R&0xFFFFFF00)+0x00000011;

//UART1: PB0 -> U1Rx,PB1 -> U1tx

GPIO\_PORTB\_AMSEL\_R &= ~0x03; // disable analog functionality on PB 0-1

}

void PortF\_Init(void){ volatile unsigned long delay;

SYSCTL\_RCGC2\_R |= 0x00000020; // 1) activate clock for Port F

delay = SYSCTL\_RCGC2\_R; // allow time for clock to start

GPIO\_PORTF\_LOCK\_R = 0x4C4F434B; // 2) unlock GPIO Port F

GPIO\_PORTF\_CR\_R = 0x1F; // allow changes to PF4-0

// only PF0 needs to be unlocked, other bits can't be locked

GPIO\_PORTF\_AMSEL\_R = 0x00; // 3) disable analog on PF

GPIO\_PORTF\_PCTL\_R = 0x00000000; // 4) PCTL GPIO on PF4-0

GPIO\_PORTF\_DIR\_R = 0x0E; // 5) PF4,PF0 in, PF3-1 out

GPIO\_PORTF\_AFSEL\_R = 0x00; // 6) disable alt funct on PF7-0

GPIO\_PORTF\_PUR\_R = 0x11; // enable pull-up on PF0 and PF4

GPIO\_PORTF\_DEN\_R = 0x1F; // 7) enable digital I/O on PF4-0

GPIO\_PORTF\_IS\_R &= ~0x11; // (d) PF4,PF0 is edge-sensitive

GPIO\_PORTF\_IBE\_R &= ~0x11; // PF4,PF0 is not both edges

GPIO\_PORTF\_IEV\_R &= ~0x11; // PF4,PF0 falling edge event

GPIO\_PORTF\_ICR\_R = 0x11; // (e) clear flags 4,0

GPIO\_PORTF\_IM\_R |= 0x11; // (f) arm interrupt on PF4,PF0

NVIC\_PRI7\_R = (NVIC\_PRI7\_R&0xFF1FFFFF)|0x00400000; // (g) bits:23-21 for PORTF, set priority to 2

NVIC\_EN0\_R = 0x40000000; // (h) enable interrupt 30 in NVIC

}

void carcontrol(char instructions){

if(instructions == 'w'){

GPIO\_PORTF\_DATA\_R = 0x08; //'w : forward,green LED on'

}

else if(instructions == 's'){

GPIO\_PORTF\_DATA\_R = 0x04; //'s : reverse, blue LED on'

}

else if(instructions == 'a'){

GPIO\_PORTF\_DATA\_R = 0x0A; //'a : left turn, yellow LED on'

}

else if(instructions == 'd'){

GPIO\_PORTF\_DATA\_R = 0x06; //'d : left turn, purple LED on'

}

else if(instructions == 't'){

GPIO\_PORTF\_DATA\_R = 0x00; //'t : stop, all LED off'

}

else if(instructions == 'u'){

GPIO\_PORTF\_DATA\_R =GPIO\_PORTF\_DATA\_R;//'u, keep the same LED on'

}

else if(instructions == 'l'){

GPIO\_PORTF\_DATA\_R =GPIO\_PORTF\_DATA\_R;//'l, keep the same LED on'

}

}

int main(void){

unsigned char letterget;

char string[20];

PortF\_Init();

UART\_Init(); // initialize UART

UART1\_Init();

while(1){

letterget = UART1\_InChar();

UART\_OutChar(letterget);

carcontrol(letterget);

}

}