TI DSP, MCU 및 Xilinx Zynq FPGA
프로그래밍 전문가 과정

강사 – Innova Lee(이상훈)

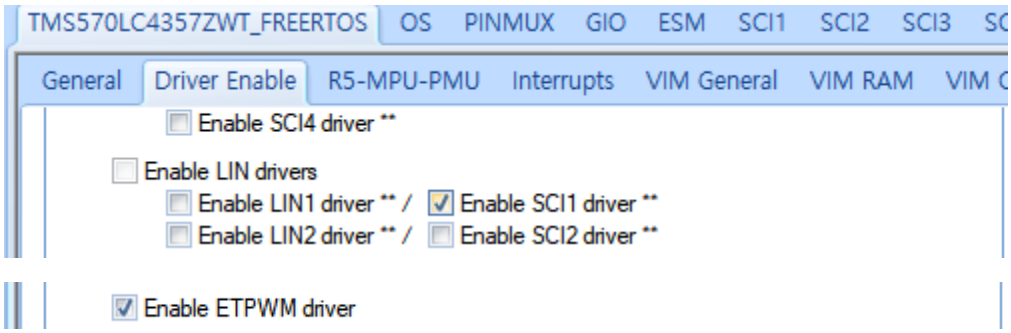
gcccompil3r@gmail.com

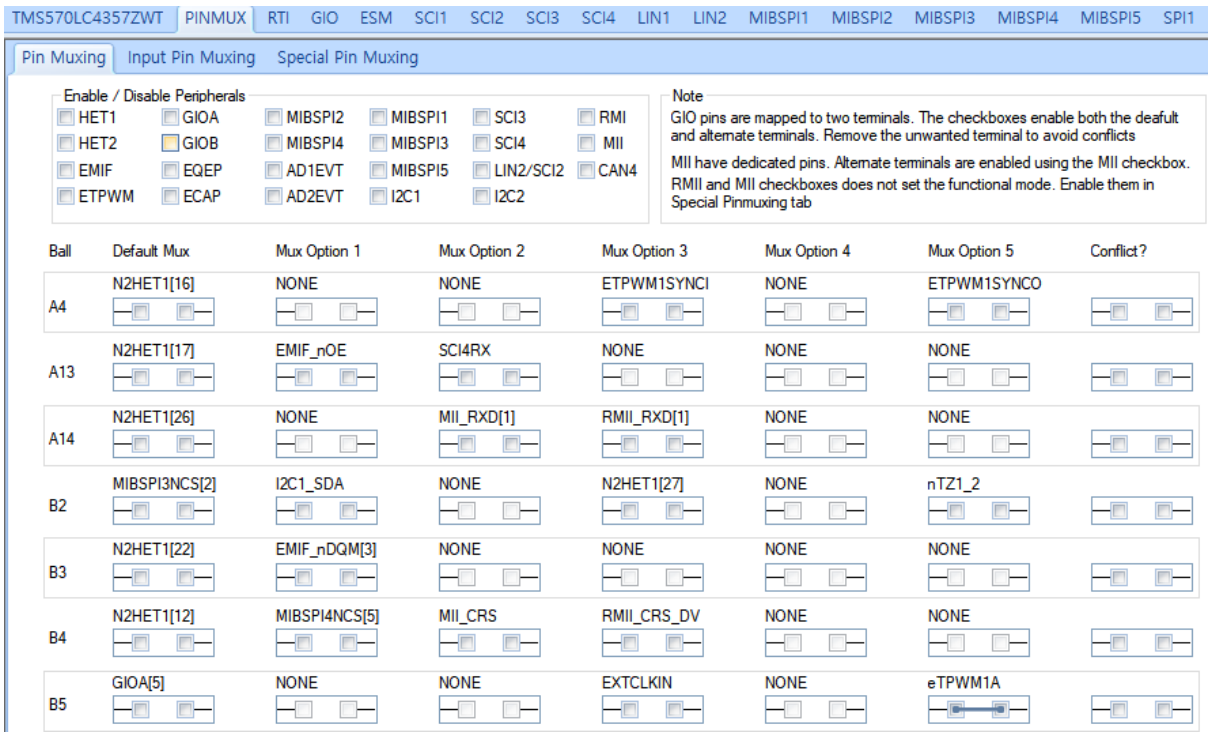
학생 – 문한나

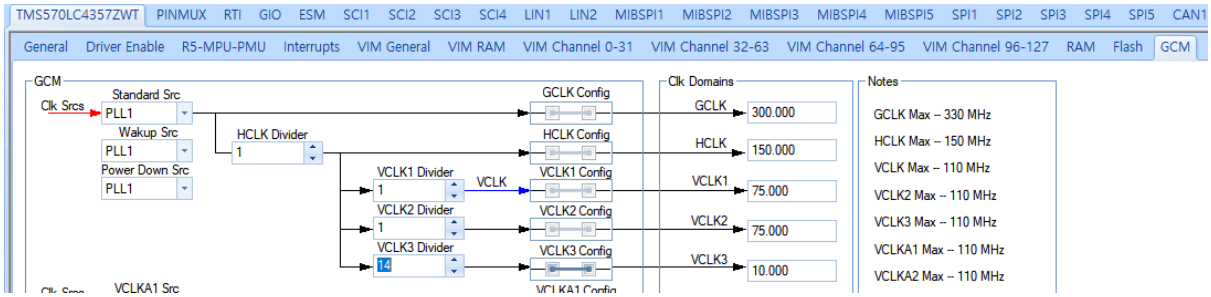
mhn97@naver.com

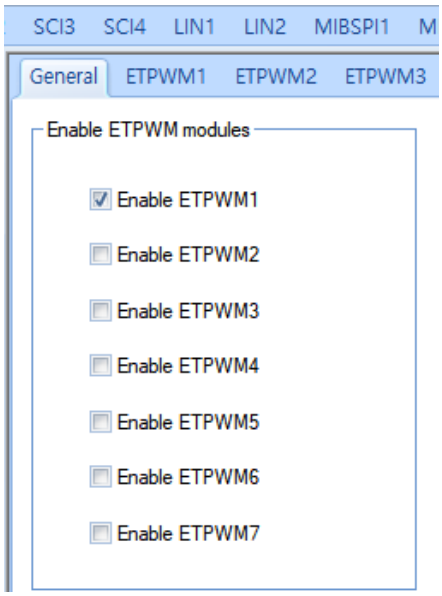
uart 통신을 이용한 모터제어

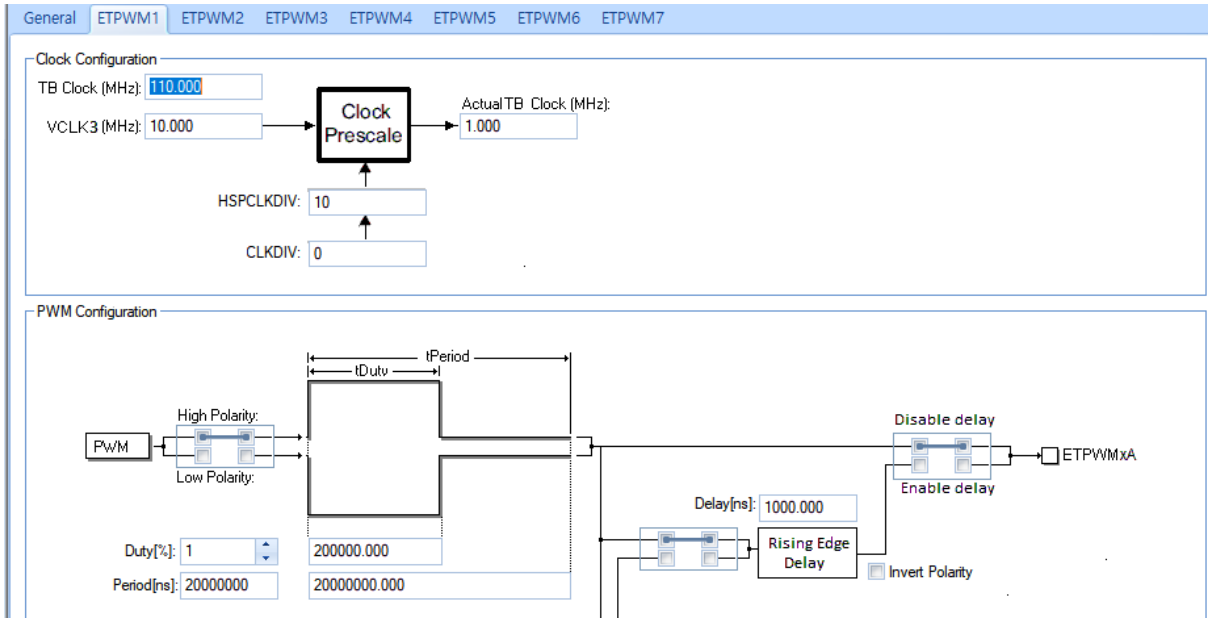
HCG 설정








SCI3
SCI4
LIN1
LIN2
MIBSPI1
M
General
ETPWM1
ETPWM2
ETPWM3
Enable ETPWM modules
Enable ETPWM1
Enable ETPWM2
Enable ETPWM3
Enable ETPWM4
Enable ETPWM5
Enable ETPWM6
Enable ETPWM7


General
ETPWM1
ETPWM2
ETPWM3
ETPWM4
ETPWM5
ETPWM6
ETPWM7
Clock Configuration
TB Clock (MHz):
110.000
VCLK3 (MHz):
10.000
Clock Prescale
Actual TB Clock (MHz):
1.000
HSPCLKDIV:
10
CLKDIV:
0
PWM Configuration
tPeriod
tDuty
High Polarity:
PWM
Low Polarity:
Disable delay
ETPWMxA
Enable delay
Delay[ns]:
1000.000
Rising Edge Delay
Invert Polarity
Duty[%]:
1
200000.000
Period[ns]:
20000000
20000000.000

CCS 코드

```
#include "HL_sys_common.h"
#include "HL_etpwm.h"
#include <HL_sci.h>
#include "string.h"
#include "stdio.h"
#include "stdlib.h"

uint8 input[5] = {0};

void send_data(sciBASE_t* sci, uint8* msg, int length)
{
    int i;
    for(i=0;i<length;i++)
        sciSendByte(sciREG1,msg[i]);
        sciSendByte(sciREG1,'\r');
        sciSendByte(sciREG1,'\n');
}

int main(void){

    int i;

    sciInit();
    etpwmInit();

    etpwmStartTBCLK();

    while(1){

        for(i=0; i<5; i++){
        input[i] = sciReceiveByte(sciREG1); //저장
        }

        send_data(sciREG1, input, strlen(input)); //출력

        etpwmREG1->CMPA = atoi(input); //제어

    }
    return 0;
}

/* etpwmREG1->TBPRD = 19999U;

   etpwmREG1->CMPA = 200U;

   duty는 5%(1000) ~ 10%(2000)이고, 7.5%(1500)이 기준이다.

   모터는 1500보다 작을수록 정방향으로 빨라지고 클수록 역방향으로 빨라진다. */
```