

TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정

강사 - Innova Lee(이상훈)
gcccompil3r@gmail.com

학생 - 하성용
accept0108@naver.com

예제 : gpio 와 rti 이용해서 LED 하나당 개별카운터를 주어서 동작하게하기

< CCS >

```
#include "HL_sys_common.h"
#include "HL_gpio.h"
#include "HL_rti.h"
#include "HL_sys_core.h"

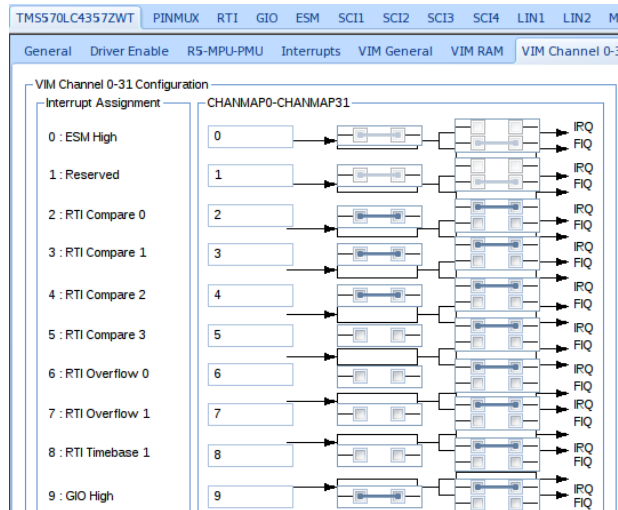
void main(void)
{
    gpioInit();
    rtiInit();

    gpioSetBit(gioPORTA, 0, 1); //gio 포트 A 에 0 번핀 1 값
    gpioSetBit(gioPORTA, 1, 1); //gio 포트 A 에 1 번핀 1 값
    gpioSetBit(gioPORTA, 2, 1); //gio 포트 A 에 2 번핀 1 값
    rtiStartCounter(rtiREG1, 0); //rti 카운터를 샘 0 부터
    _enable_interrupt_(); //인터럽트를 하기위해 필요
    rtiEnableNotification(rtiREG1, 1);
    rtiEnableNotification(rtiREG1, 2);
    rtiEnableNotification(rtiREG1, 4);
    while(1);
}

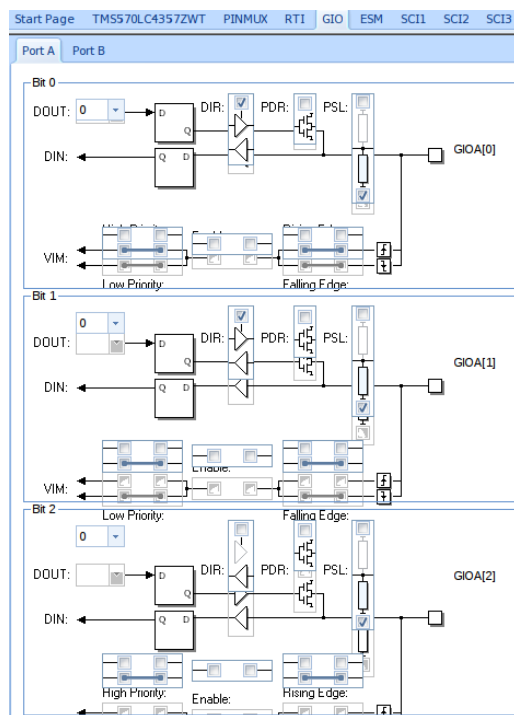
void rtiNotification(rtiBASE_t *rtiREG, uint32 notification)
{
    if(notification == 1) //0 번핀
    {
        gpioToggleBit(gioPORTA, 0);
    }
    else if(notification == 2) //1 번핀
    {
        gpioToggleBit(gioPORTA, 1);
    }
    else if(notification == 4) //2 번핀
    {
        gpioToggleBit(gioPORTA, 2);
    }
}
```

< HAL >

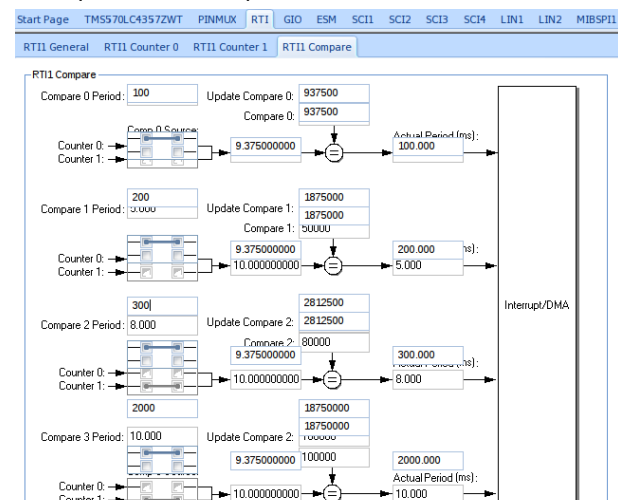
//빔채널



//GIO 출력비트 설정



//Compare 당 속도 값(period) 지정



//1 초마다 LED 밝기 6 단계 변화 예제

```
#include "HL_sys_common.h" //기본시스템 헤더 파일(타입 정의)
#include "HL_rti.h" //rti 헤더파일
#include "HL_sys_core.h" //인터럽트 헤더파일
#include "HL_etpwm.h" //pwm 헤더파일

//전역변수 -> 함수 어디에서든 사용할수있다
//배열 i, 25000 이 아닌 24999 인 이유는 0 부터 시작하기때문
int arr[6]={0, 5000, 10000, 15000, 20000, 24999};
int i = 0; //i 를 0 으로 초기화

void main(void)
{
    rtiInit(); //rtiInit 에 들어가서 필요한 함수들을 데려온다
    etpwmInit(); //etpwmInit 에 들어가서 필요한 함수들을 데려온다
    rtiStartCounter(rtiREG1,0); //카운터를 세겟다(REG 주소, 어디(몇)부터)
//etpwm 변조를 시작하겠다, 0 하고 1 값이 들어와야 PinMux 에서 허용
    etpwmStartTBCLK();
    _enable_interrupt_(); //인터럽트 쓰려면 필요
    rtiEnableNotification(rtiREG1,1); //rti 인터럽트를 쓰겠다
    while(1); //반복
}

void rtiNotification(rtiBASE_t *rtiREG,uint32 notification)
{
    i++; //i 를 1 씩 더함
//CMPA 란 듀티비에 해당하는값으로 예제의 경우 0 을 쓰면 꺼져있고 24999 쓰면 계속 켜져있다
//들어가보면 etpwm->CMPA = value; -> 값
    etpwmSetCmpA(etpwmREG1, arr[i%6]);
//CMPA 를 셋해주겠다 etpwmSetCmpA(etpwmBASE_t *etpwm, uint16 value)
    if(i==1000000)
        i=0;
}
```

저번주 시행 오차

// 위처럼 rtiNotification 함수는 i 를 0 으로 초기화하고 1 을 더해서 배열에 해당하는 값이 들어가고

다시 1 을더해서 배열에 해당하는값(밝기)이 들어가야하는데

int i=0 을 전역변수가 아닌 **void rtiNotification** 의 지역변수로 쓰게되면

i 를 0 으로 초기화하고 1 을 더하고 다시 0 으로 초기화하는 문제가 발생한다

< HAL >

TMS570LC4357ZWT PIN

General Driver Enable

Enable Driver Compilation

Click and mark t

☒ Enable RTI driver

☒ Enable ETPWM driver

//VIM Channel → 인터럽트를 쓰겠다 선언

General Driver Enable R5-MPU-PMU Interrupts VIM General VIM RAM VIM Channel 0-31

VIM Channel 0-31 Configuration

Interrupt Assignment

0 : ESM High

1 : Reserved

2 : RTI Compare 0

CHANMAP0-CHANMAP31

0 → IRQ FIQ

1 → IRQ FIQ

2 → IRQ FIQ

//GCM → 14 분주

General Driver Enable R5-MPU-PMU Interrupts VIM General VIM RAM VIM Channel 0-31 VIM Channel 32-63 VIM Channel 64-95 VIM Channel 96-127 RAM Flash GCM

GCM

Clk Srcs

Standard Src

Wakeup Src

PLL1

Power Down Src

PLL1

HCLK Divider

1

VCLK1 Divider

1

VCLK2 Divider

1

VCLK3 Divider

14

GCLK Config

HCLK Config

VCLK1 Config

VCLK2 Config

VCLK3 Config

Clk Domains

GCLK

300.000

HCLK

150.000

VCLK1

75.000

VCLK2

75.000

VCLK3

10.000

Notes

GCLK Max -- 330 MHz

HCLK Max -- 150 MHz

VCLK Max -- 110 MHz

VCLK2 Max -- 110 MHz

VCLK3 Max -- 110 MHz

VCLKA1 Max -- 110 MHz

VCLKA2 Max -- 110 MHz

//PINMUX : 위치는 GIOA 의 5 번핀

TMS570LC4357ZWT PINMUX RTI GIO ESM SCI1 SCI2 SCI3 SCI4 LIN1 LIN2 MIBSPI1 MIBSPI2 MIBSPI3 MIBSPI4 MIBSPI5 SPI1 SPI2

Pin Muxing

Input Pin Muxing

Special Pin Muxing

GIOA[5]

NONE

NONE

EXTCLKIN

NONE

eTPWM1A

B5

//RTI - RTI Comapare

TMS570LC4357ZWT PINMUX RTI GIO ESM SCI1 SCI2 SCI3 SCI4 LIN1 LIN2

RTI1 General RTI1 Counter 0 RTI1 Counter 1 RTI1 Compare

RTI1 Compare

Compare 0 Period: 1000

Update Compare 0: 9375000

Compare 0: 9375000

Counter 0: 9.375000000

Counter 1: 1000.000

Actual Period (ms): 1000.000

//ETWTM - ETPWM1 만 쓸거니까 1 만 켜주기

RTI SPI2 SPI3 SPI4 SPI5 CAN1 CAN2 CAN3 CAN4 ADC1 ADC2 HET1 HET2 I2C1 I2C2 EMAC DCC RTP DMH DMF POM CRC ETPWM

General ETPWM1 ETPWM2 ETPWM3 ETPWM4 ETPWM5 ETPWM6 ETPWM7

Enable ETPWM modules

☒ Enable ETPWM1

☐ Enable ETPWM2

☐ Enable ETPWM3

☐ Enable ETPWM4

☐ Enable ETPWM5

☐ Enable ETPWM6

☐ Enable ETPWM7

//ETPWM - ETPWM1 - CLKDIV → 7 기입

General ETPWM1 ETPWM2 ETPWM3 ETPWM4

Clock Configuration

TB Clock (MHz): 110.000

VCLK3 (MHz): 10.000

Clock Prescale

HSPCLKDIV: 0

CLKDIV: 7

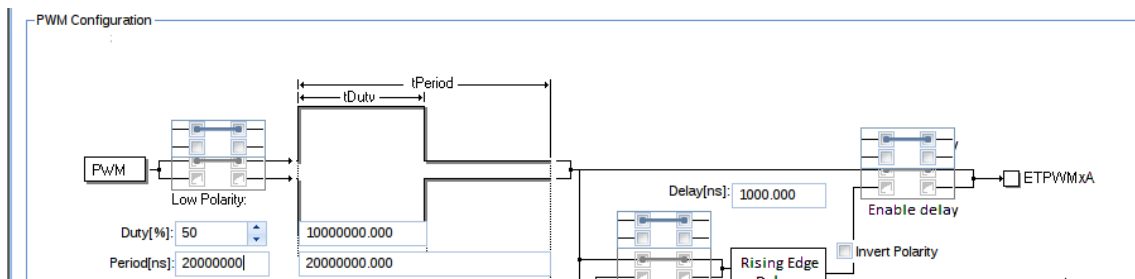
ms=millisecond 는 천분의 1 초 → 0.001 이 단위로

na=nanosecond 는 10 억분의 1 초 → 0.000,000,001 이 단위

20m/s 는 $0.001 * 20 = 0.02$ 로

na 에서 20m/s 로 바꾸려면 $0.000,000,001 * 20,000,000 = 0.02$

//Period → 20,000,000



감사합니다