

TI DSP, MCU, Xilinx Zynq FPGA 프로그래밍 전문가 과정

Ultrasonic Ranging Module HC - SR04

2018.08.22

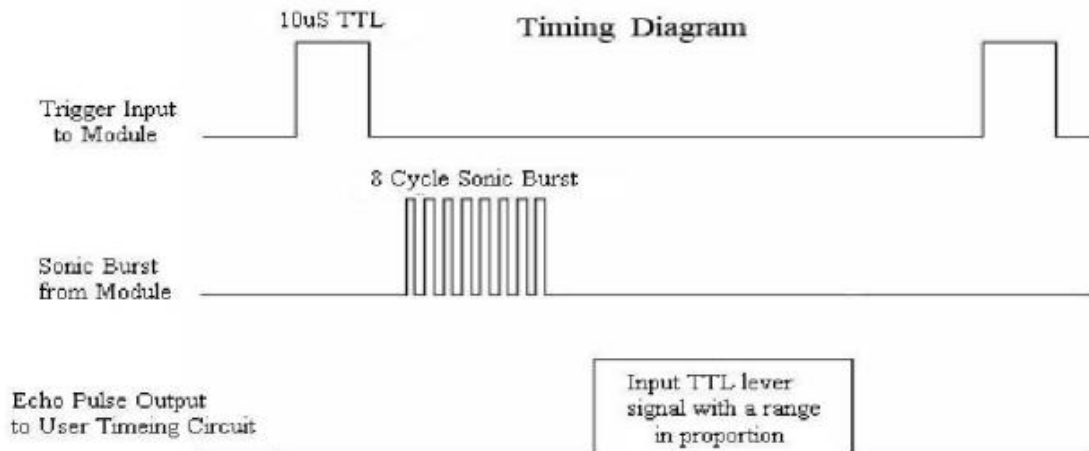
강사 – Innova Lee(이상훈)
gcccompil3r@gmail.com

학생 – 안상재
sangjae2015@naver.com

* 초음파 센서

Timing diagram

The Timing diagram is shown below. You only need to supply a short 10uS pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion .You can calculate the range through the time interval between sending trigger signal and receiving echo signal. Formula: $\mu\text{S} / 58 = \text{centimeters}$ or $\mu\text{S} / 148 = \text{inch}$; or: the range = high level time * velocity (340M/S) / 2; we suggest to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.



초음파 센서의 Trigger Input 에 10us 의 펄스를 주면, Echo 핀으로 펄스가 출력된다. Echo 핀의 펄스가 지속되는 시간을 통해 초음파 센서가 감지한 물체까지의 거리를 측정할 수 있다.

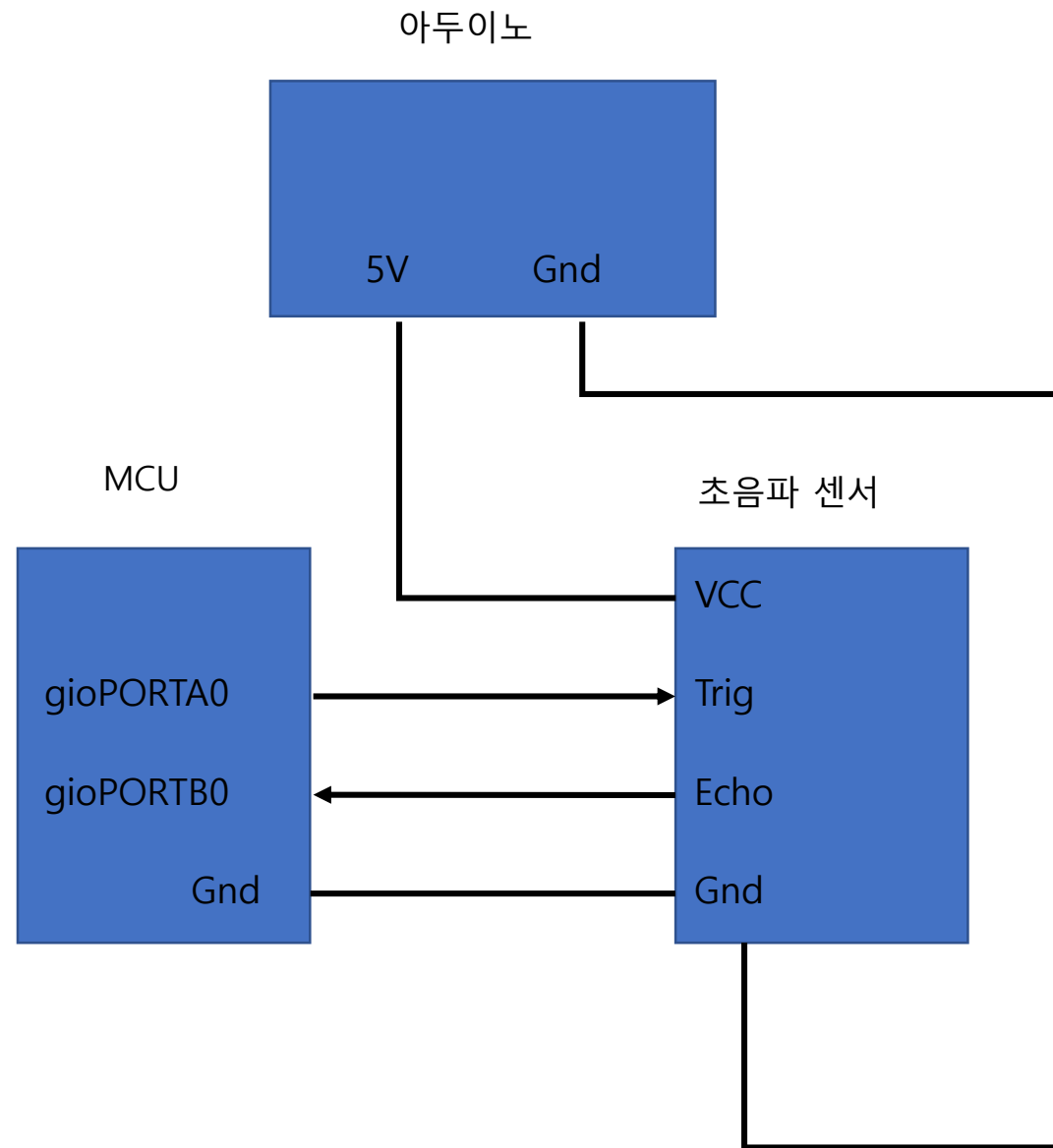
- 초음파 센서 입장에서

- 1) Trig 핀 입력
- 2) Echo 핀 출력

- 거리 공식

- 1) $\text{centimeters} = \mu\text{S} / 58$
- 2) $\text{inch} = \mu\text{S} / 148$

* 회로도



* 코드

- main 문

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

    _enable_IRQ_interrupt_(); // 전역 인터럽트 인에이블

    rtiEnableNotification(rtiREG1, 1); // rti 인터럽트 인에이블
    rtiStartCounter(rtiREG1, 0); // rti 시작

    while(1)
    {
        gpioSetBit(gioPORTA, 0, 0); // Trig 핀으로 펄스 인가
        delay(50); // 2us
        gpioSetBit(gioPORTA, 0, 1);
        delay(250); // 10us
        gpioSetBit(gioPORTA, 0, 0);

        count = 0;
        while(gioGetBit(gioPORTB, 0) == 0); // Echo 핀에서 펄스가 시작될 때까지 대기
        rtiStartCounter(rtiREG1, 0); // Echo 핀의 펄스가 시작되면 us 단위로 카운트 시작

        while(gioGetBit(gioPORTB, 0) == 1); // Echo 핀의 펄스가 끝날 때까지 대기
        rtiStopCounter(rtiREG1, 0); // Echo 핀의 펄스가 끝나면 카운트 종료

        Distance(count); // Echo 핀의 펄스 길이 동안의 us 시간을 거리로 계산함

        printf("%dcm\n", DistanceCalc);

        delay(25000000); // 1초 딜레이
    }
}
```

* 코드

- 헤더파일 및 전역변수, 함수 선언

```
#include "HL_sys_common.h"
```

```
#include "HL_gio.h"
#include "HL_sci.h"
#include "HL_rti.h"
#include "HL_sys_core.h"
#include <stdio.h>
```

```
long count = 0;
long DistanceCalc;
```

```
void delay(uint32 time);
void Distance(long time);
```

```
void delay(uint32 time) // time 50 : 2us
{
    uint32 i;

    for(i=0;i<time;i++);
}
```

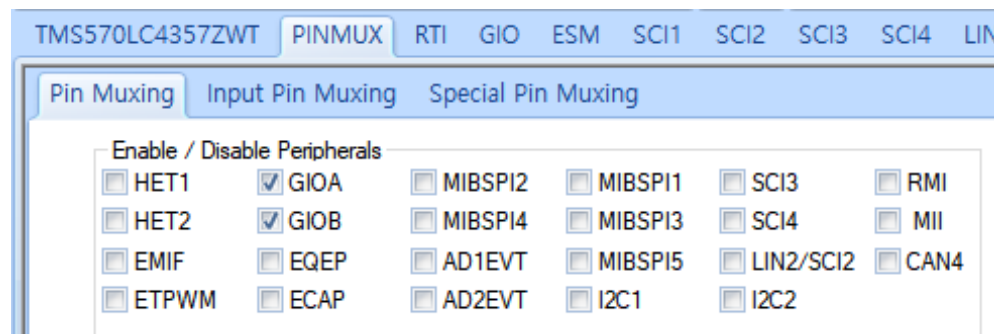
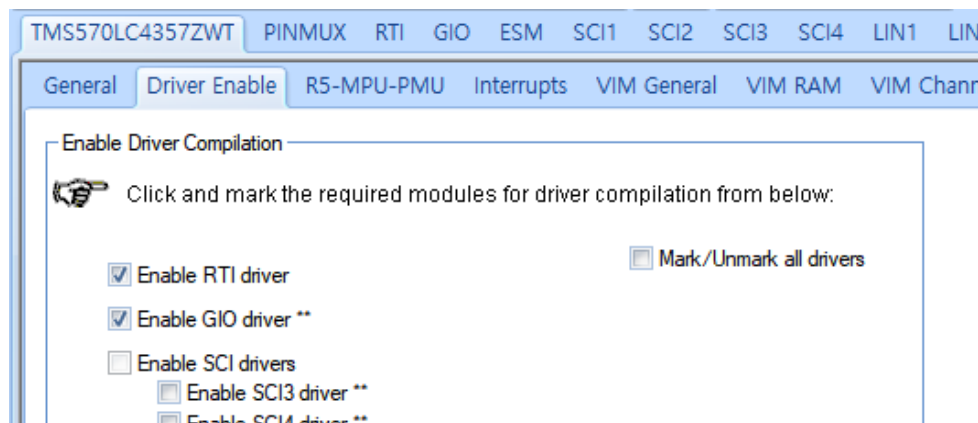
- 함수 정의

```
void Distance(long time) // us 시간 단위를 cm거리 단위로 변환하는 공식
{
    DistanceCalc = time / 58;
}
```

```
void rtiNotification(rtiBASE_t *rtiREG, uint32 notification) // 1us 마다 rti 인터럽트 발생해서 count 증가 시킴
{
    count++;
}
```

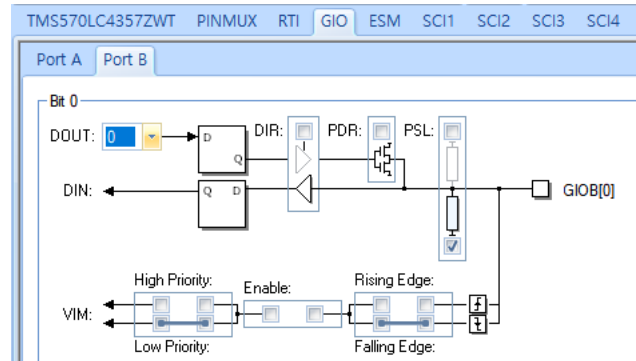
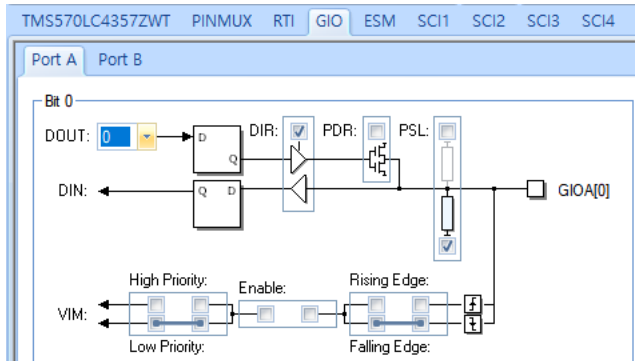
* Hal Code Generator 설정

- Driver 및 PINMUX 설정

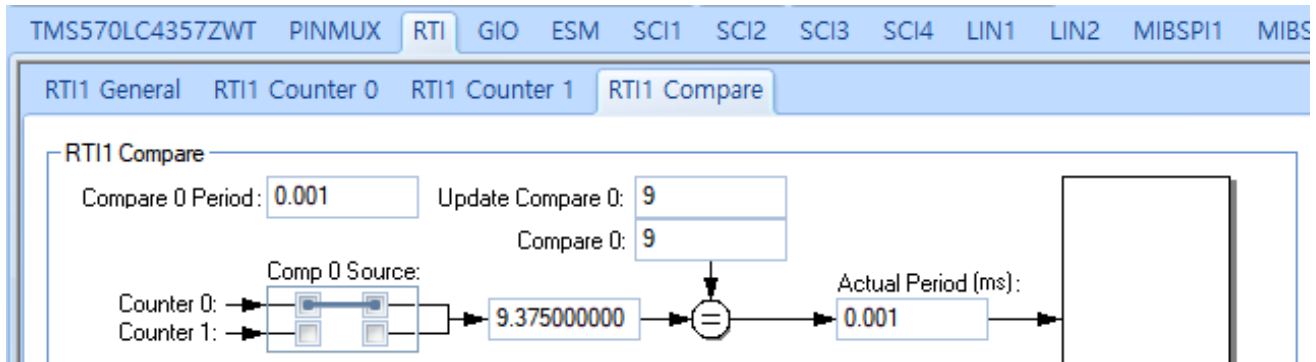


* Hal Code Generator 설정

- GIO 설정



- RTI 설정



* 결과

