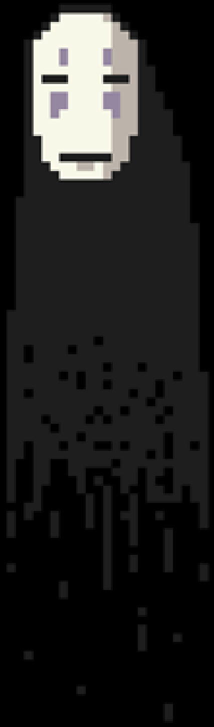
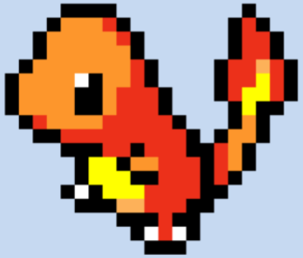
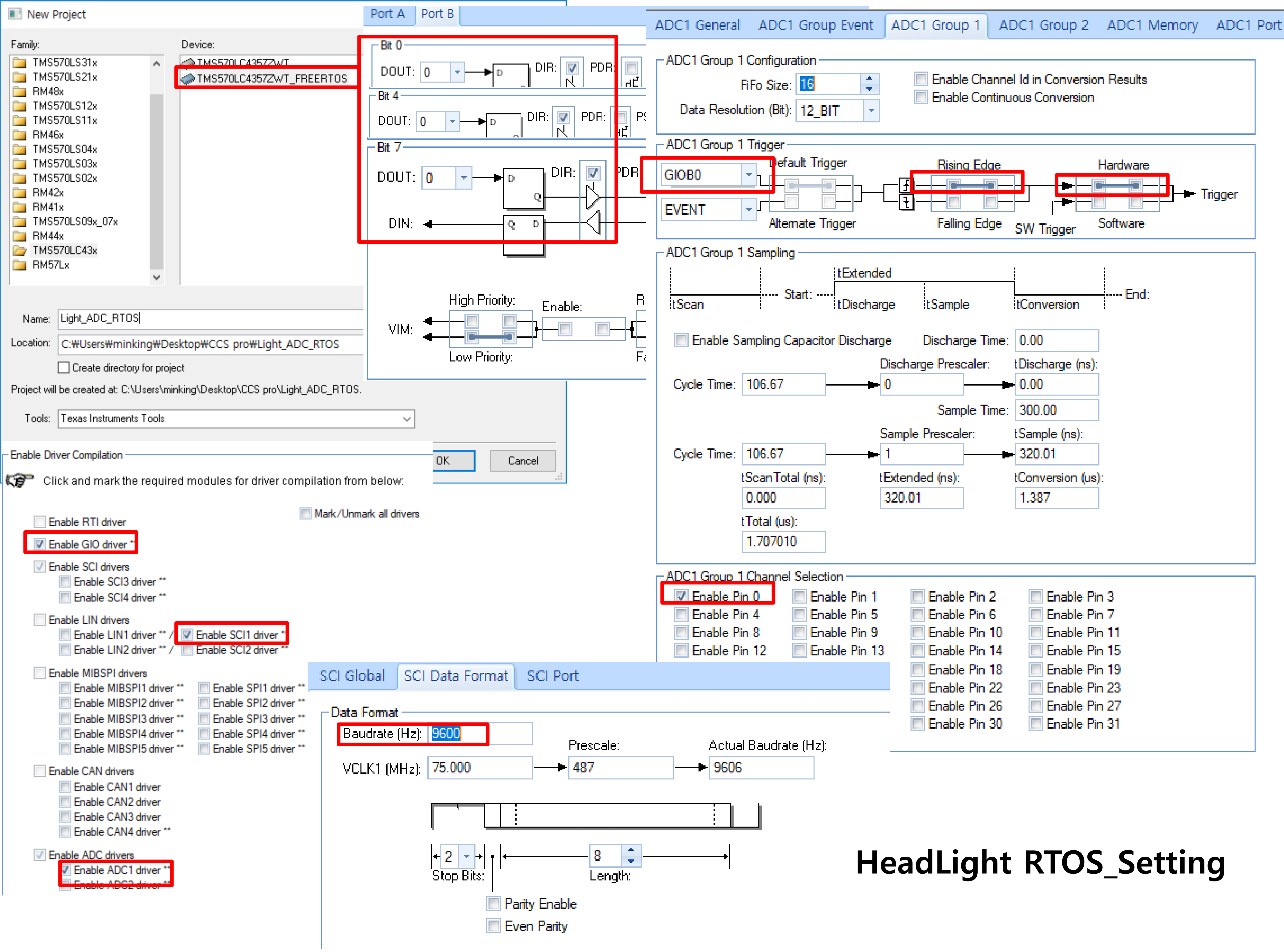
Xilinx
Zynq FPGA
TI DSP MCU 기반의
프로그래밍 및 회로 설계 전문가



강사 이상훈
gcccompil3r@gmail.com

학생 김민호
minking12@naver.com





New Project

Family:

- TMS570LS31x
- TMS570LS21x
- RM48x
- TMS570LS12x
- TMS570LS11x
- RM46x
- TMS570LS04x
- TMS570LS03x
- TMS570LS02x
- RM42x
- RM41x
- TMS570LS09x_07x
- RM44x
- TMS570LC43x
- RM57Lx

Device:

- TMS570LC4357ZWT
- TMS570LC4357ZWT_FREERTOS

Name: Light_ADC_RTOS

Location: C:₩Users₩minking₩Desktop₩CCS pro₩Light_ADC_RTOS

□ Create directory for project

Project will be created at: C:\Users\minking\Desktop\CCS pro\Light_ADC_RTOS.

Tools: Texas Instruments Tools

OK Cancel

Enable Driver Compilation

Click and mark the required modules for driver compilation from below:

□ Mark/Unmark all drivers

□ Enable RTI driver

☑ Enable GIO driver *

☑ Enable SCI drivers

□ Enable SCI3 driver **

□ Enable SCI4 driver **

□ Enable LIN drivers

□ Enable LIN1 driver ** / ☑ Enable SCI1 driver *

□ Enable LIN2 driver ** / □ Enable SCI2 driver **

□ Enable MIBSPI drivers

□ Enable MIBSPI1 driver ** □ Enable SPI1 driver **

□ Enable MIBSPI2 driver ** □ Enable SPI2 driver **

□ Enable MIBSPI3 driver ** □ Enable SPI3 driver **

□ Enable MIBSPI4 driver ** □ Enable SPI4 driver **

□ Enable MIBSPI5 driver ** □ Enable SPI5 driver **

□ Enable CAN drivers

□ Enable CAN1 driver

□ Enable CAN2 driver

□ Enable CAN3 driver

□ Enable CAN4 driver **

☑ Enable ADC drivers

☑ Enable ADC1 driver **

□ Enable ADC2 driver **

Port A Port B

Bit 0 — DOUT: 0 DIR: ☑ PDR: □

Bit 4 — DOUT: 0 DIR: ☑ PDR: □

Bit 7 — DOUT: 0 DIR: ☑ PDR

DIN:

VIM: High Priority: Enable: Low Priority:

ADC1 General | ADC1 Group Event | ADC1 Group 1 | ADC1 Group 2 | ADC1 Memory | ADC1 Port

ADC1 Group 1 Configuration

FiFo Size: 16

Data Resolution (Bit): 12_BIT

□ Enable Channel Id in Conversion Results

□ Enable Continuous Conversion

ADC1 Group 1 Trigger

Default Trigger: GIOB0

Alternate Trigger: EVENT

Rising Edge ☑ / Falling Edge □

Hardware ☑ / Software □ / SW Trigger

Trigger

ADC1 Group 1 Sampling

tScan Start: tExtended tDischarge tSample tConversion End:

□ Enable Sampling Capacitor Discharge

Discharge Time: 0.00

Cycle Time: 106.67 → Discharge Prescaler: 0 → tDischarge (ns): 0.00

Sample Time: 300.00

Cycle Time: 106.67 → Sample Prescaler: 1 → tSample (ns): 320.01

tScanTotal (ns): 0.000

tExtended (ns): 320.01

tConversion (us): 1.387

tTotal (us): 1.707010

ADC1 Group 1 Channel Selection

☑ Enable Pin 0 □ Enable Pin 1 □ Enable Pin 2 □ Enable Pin 3

□ Enable Pin 4 □ Enable Pin 5 □ Enable Pin 6 □ Enable Pin 7

□ Enable Pin 8 □ Enable Pin 9 □ Enable Pin 10 □ Enable Pin 11

□ Enable Pin 12 □ Enable Pin 13 □ Enable Pin 14 □ Enable Pin 15

□ Enable Pin 18 □ Enable Pin 19

□ Enable Pin 22 □ Enable Pin 23

□ Enable Pin 26 □ Enable Pin 27

□ Enable Pin 30 □ Enable Pin 31

SCI Global | SCI Data Format | SCI Port

Data Format

Baudrate (Hz): 9600

VCLK1 (MHz): 75.000 → Prescale: 487 → Actual Baudrate (Hz): 9606

Stop Bits: 2

Length: 8

□ Parity Enable

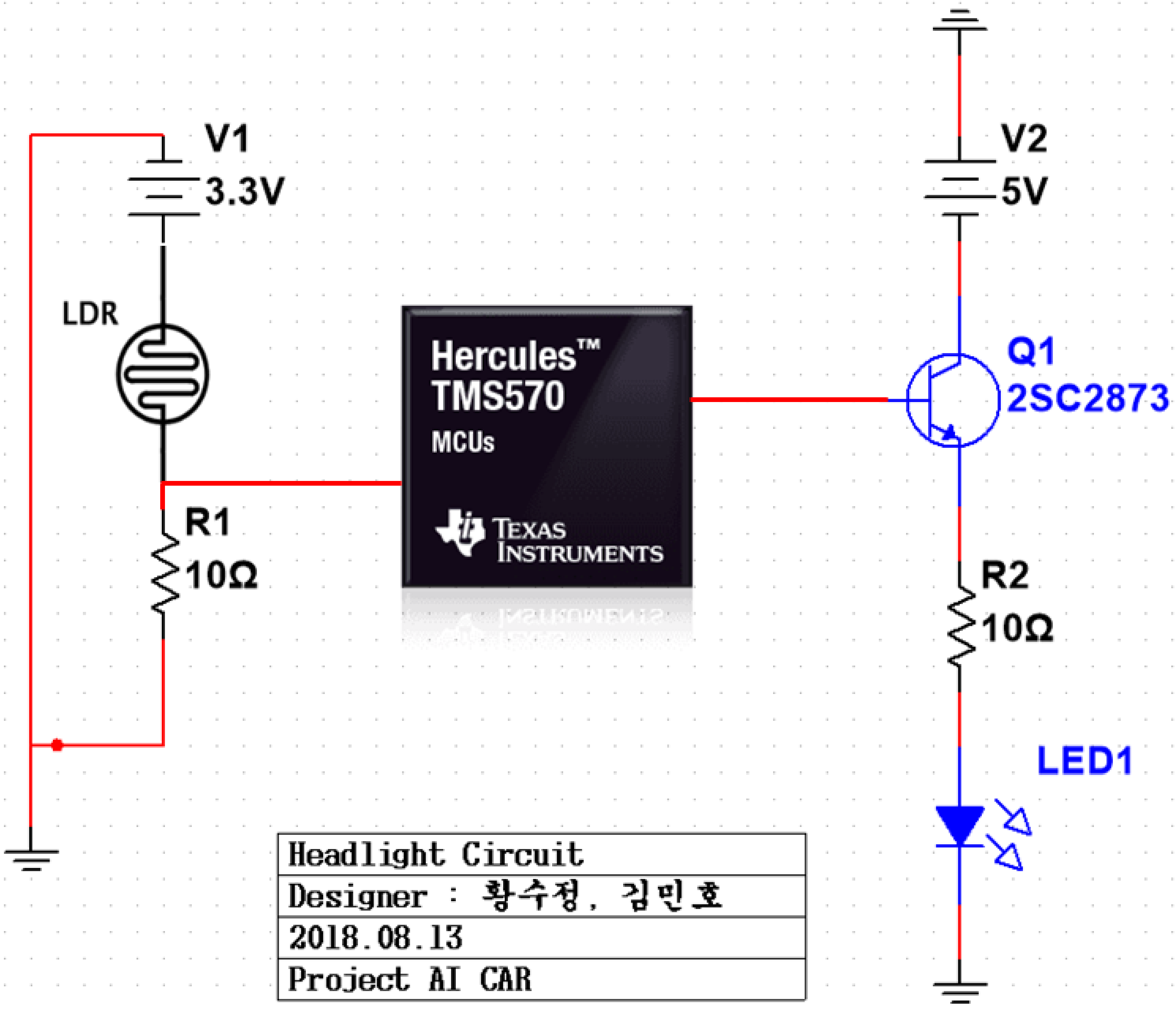
□ Even Parity

HeadLight RTOS_Setting

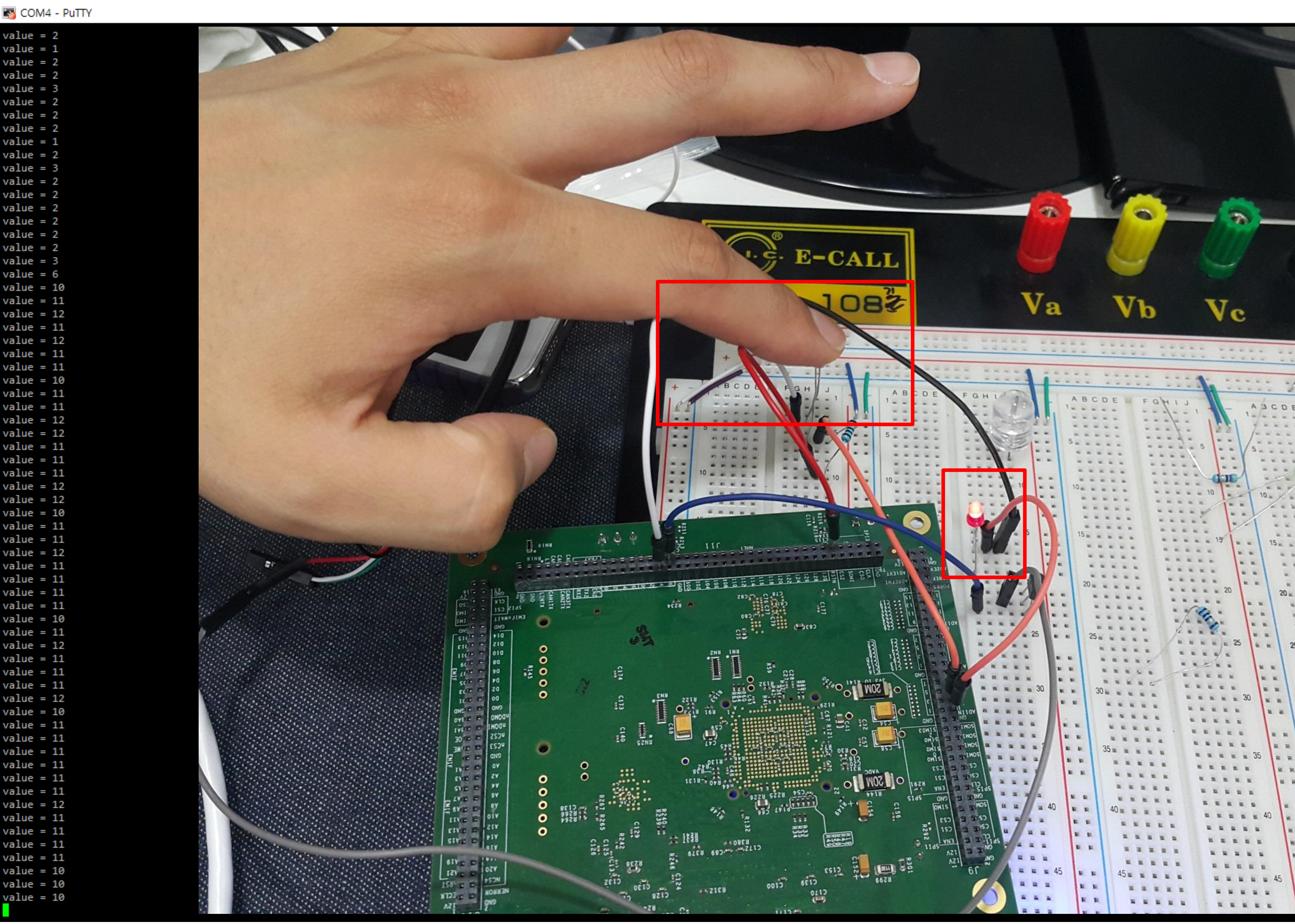
HeadLight RTOS_CODE

```
#include <HL_gio.h>
#include <HL_reg_gio.h>
#include <stdio.h>
#include <FreeRTOS.h>
#include <FreeRTOSConfig.h>
#include <HL_hal_stdtypes.h>
#include <os_mpu_wrappers.h>
#include <os_projdefs.h>
#include <os_semphr.h>
#include <os_task.h>
#include <string.h>
#include <HL_reg_sci.h>
#include <HL_sci.h>
#include <HL_adc.h>
#include <HL_reg_adc.h>
adcData_t counter;
uint8 msg[32] = { 0, };
uint32 value;
xTaskHandle xTask1Handle;
QueueHandle_t mutex;
void vTask1(void* pvParameters);
void send_data(sciBASE_t *sci, uint8 *msg, uint32 length)
{
    int i;
    for (i = 0; i < length; i++)
    {
        sciSendByte(sciREG1, msg[i]);
    }
}
void led(int bri)
{
    if (bri <= 5)
    {
        gioSetBit(gioPORTB, 7, 1);
    }
    else
    {
        gioSetBit(gioPORTB, 7, 0);
    }
}
```

```
int main()
{

    sciInit();
    gioInit();
    adcInit();
    adcStartConversion(adcREG1, adcGROUP1);
    gioSetBit(gioPORTB, 0, 0);

    if (xTaskCreate(vTask1, "Task1", configMINIMAL_STACK_SIZE * 8, NULL, 1,
&xTask1Handle) != pdTRUE)
    {
        while (1)
            ;
    }

    vTaskStartScheduler();
    while (1)
        ;
    return 0;

}

void vTask1(void *pbParameters)
{

    while (1)
    {
        gioSetBit(gioPORTB, 0, 1);
        gioSetBit(gioPORTB, 4, 1);
        while (adcIsConversionComplete(adcREG1, adcGROUP1) == 0)
            ;
        adcGetData(adcREG1, adcGROUP1, &counter);
        sprintf(msg, "value = %d₩r₩n", counter.value);
        send_data(sciREG1, msg, strlen(msg));
        led(counter.value);
        vTaskDelay(80);
        gioSetBit(gioPORTB, 0, 0);
    }

}
```



Headlight Circuit
Designer : 황수정, 김민호
2018.08.13
Project AI CAR



COM4 - PuTTY

```
value = 2
value = 1
value = 2
value = 2
value = 3
value = 2
value = 2
value = 2
value = 1
value = 2
value = 3
value = 2
value = 2
value = 2
value = 2
value = 2
value = 2
value = 3
value = 6
value = 10
value = 11
value = 12
value = 11
value = 12
value = 11
value = 11
value = 10
value = 11
value = 11
value = 12
value = 12
value = 11
value = 11
value = 11
value = 12
value = 12
value = 10
value = 11
value = 11
value = 12
value = 11
value = 11
value = 11
value = 11
value = 10
value = 11
value = 12
value = 11
value = 11
value = 11
value = 12
value = 10
value = 11
value = 11
value = 11
value = 11
value = 11
value = 11
value = 12
value = 11
value = 11
value = 11
value = 11
value = 10
value = 10
value = 10
```