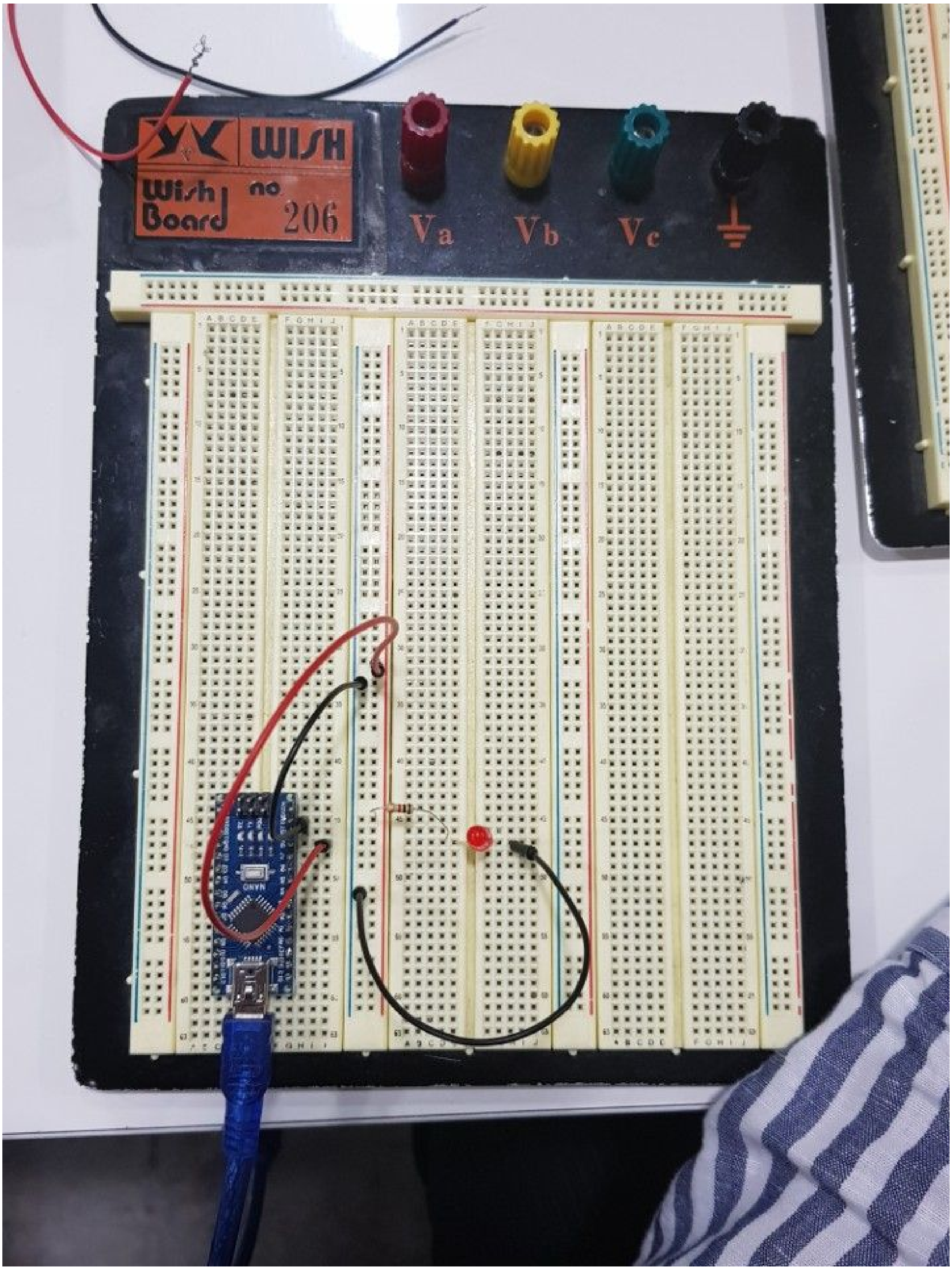
Xilinx Zynq FPGA, TI DSP, MCU기반의 프로그래밍 및 회로 설계 전문가 과정

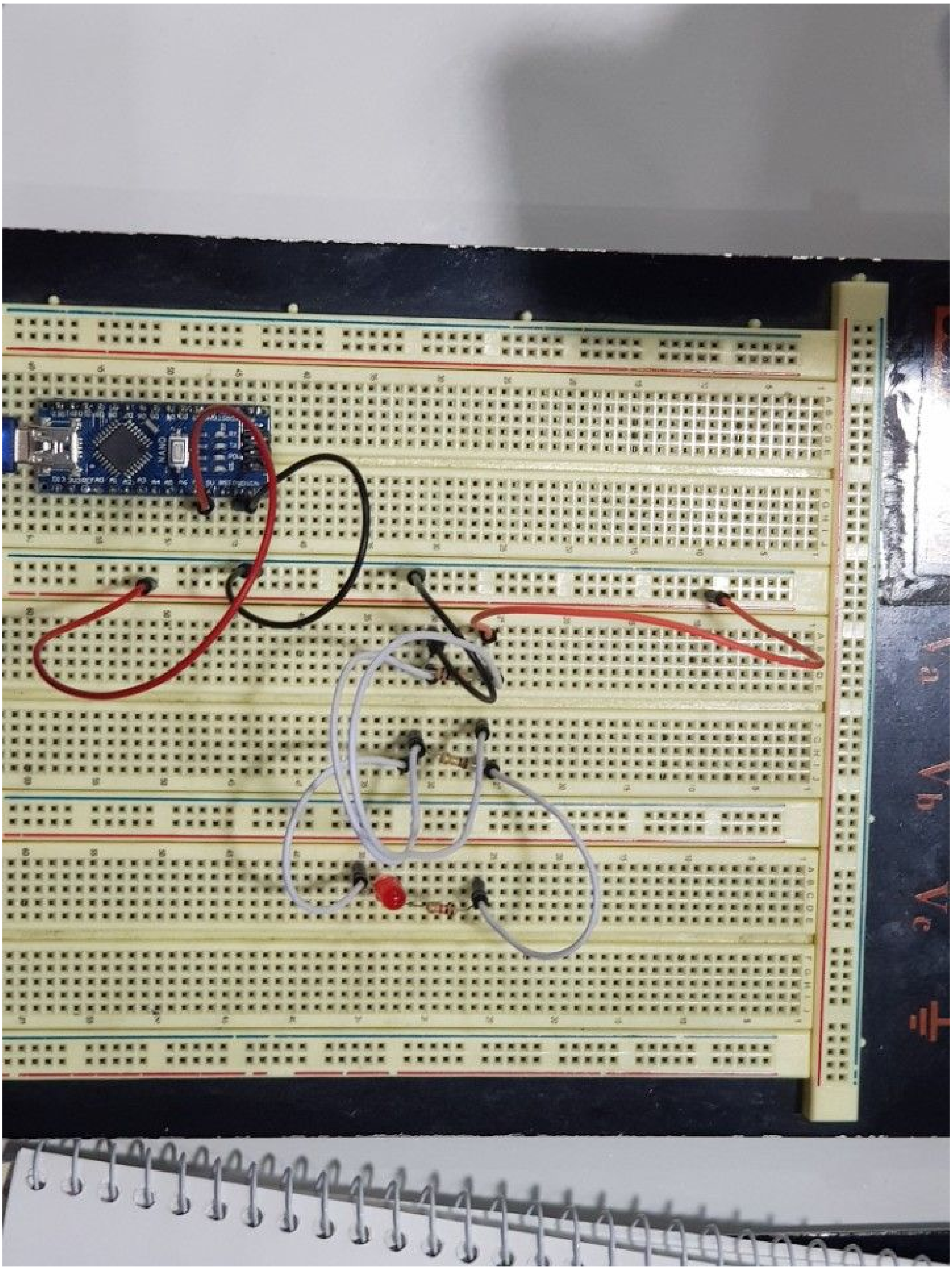
강사 - Innov (이상훈)
gcccompil3r@gmail.com
학생 - 이유성
dbtjd1102@naver.com

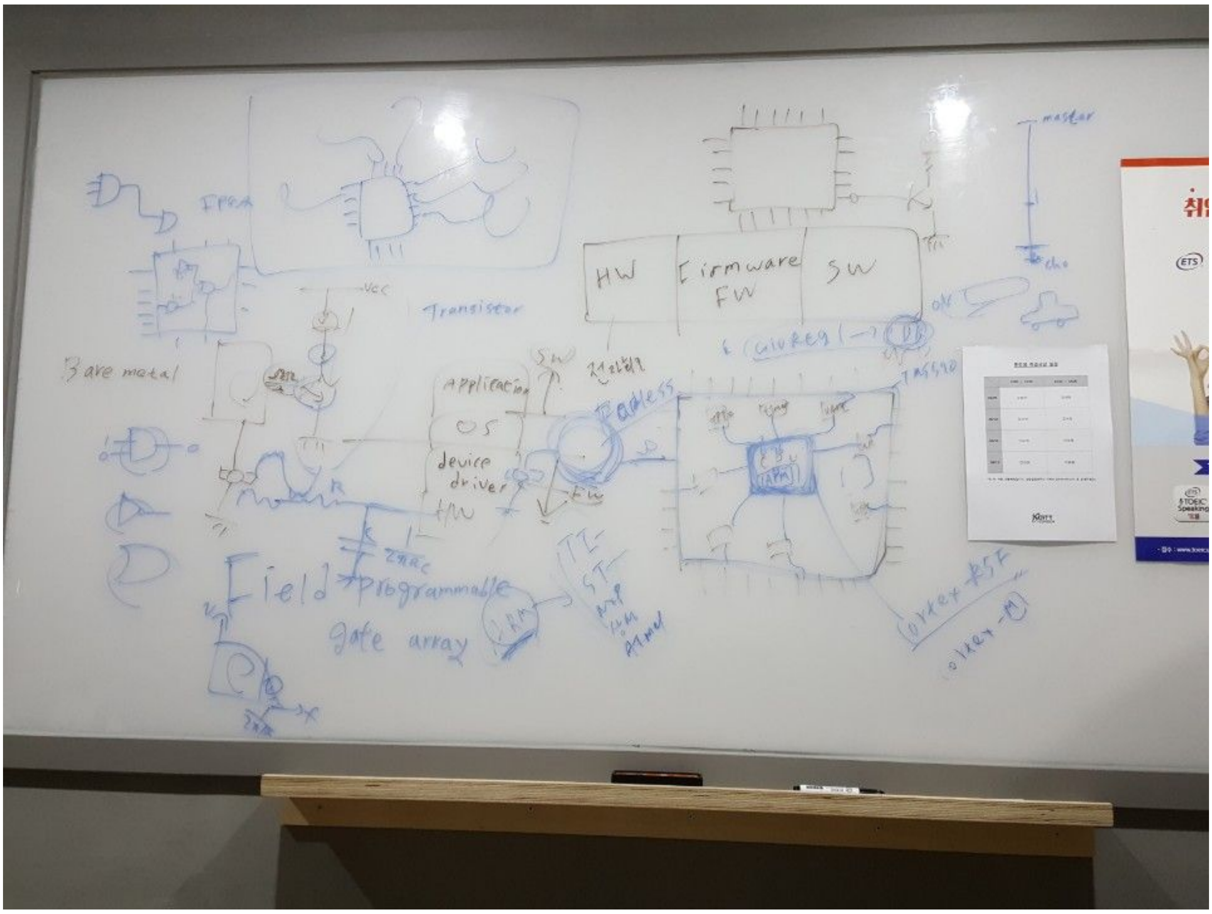
ADC_UART

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

#include "HL_esm.h"
#include "HL_adc.h"
#include "HL_sci.h"
#include "HL_gio.h"

#define TSIZE1 12
uint8 TEXT1[TSIZE1] = {'\r', '\n', 'i', '\t', 'c', 'M', '.', 'I', 'O', '-', '0', 'x'};
#define TSIZE2 9
uint8 TEXT2[TSIZE2] = {'\t', 'v', 'A', 'L', 'U', 'E', '=', '0', 'x'};

adcData_t adc_data[2];
void sciDisplayText(sciBASE_t *sci, uint8 *text, uint32 length);
void sciDisplayData(sciBASE_t *sci, uint8 *text, uint32 length);
void wait(uint32 time);

void main(void)
{
   uint32 ch_count = 0;
   uint32 id = 0;
   uint32 value = 0;

   gioInit();
   gioSetDirection(gioPORTB, 1);

   sciInit();

   adcInit();
   adcStartConversion(adcREG1, adcGROUP1);

   for(;;)
   {
      gioSetBit(gioPORTB, 0, 1);

      while((adcIsConversionComplete(adcREG1, adcGROUP1)) == 0);
      ch_count = adcGetData(adcREG1, adcGROUP1, &adc_data[0]);

      id = adc_data[0].id;
      value = adc_data[0].value;

      gioSetBit(gioPORTB, 0, 0);

      sciDisplayText(sciREG1, &TEXT1[0], TSIZE1);
```

```
        sciDisplayData(sciREG1, (uint8 *)&id, 4);
        sciDisplayData(sciREG1, &TEXT2[0], TSIZE2);
        sciDisplayData(sciREG1, (uint8 *)&value, 4);

        id = adc_data[1].id;
        value = adc_data[1].value;

        sciDisplayData(sciREG1, &TEXT1[0], TSIZE1);
        sciDisplayData(sciREG1, (uint8 *)&id, 4);
        sciDisplayData(sciREG1, &TEXT2[0], TSIZE2);
        sciDisplayData(sciREG1, (uint8 *)&value, 4);
    }
}

void sciDisplayData(sciBASE_t *sci, uint8 *text, uint32 length)
{
    uint8 txt = 0;
    uint8 txt1 = 0;

#if ((__little_endian__ == 1) || (__LITTLE_ENDIAN__ == 1))
    text = text + (length - 1);
#endif

    while(length--)
    {
#if ((__little_endian__ == 1) || (__LITTLE_ENDIAN__ == 1))
        txt = *text--;
#else
        txt = *text++;
#endif
        txt1 = txt;

        txt &= ~(0xF0);
        txt1 &= ~(0x0F);
        txt1 = txt1>>4;

        if(txt<=0x9)
        {
            txt += 0x30;
        }
        else if(txt > 0x9 && txt < 0xF)
        {
            txt += 0x37;
        }
        else
        {
            txt = 0x30;
        }

        if(txt1 <= 0x9)
        {
            txt1 += 0x30;
        }
        else if((txt1 > 0x9) && (txt1 <= 0xF))
        {
            txt1 += 0x37;
        }
        else
        {
            txt1 = 0x30;
        }

        while((sciREG1->FLR & 0x4) == 4);

        sciSendByte(sciREG1, txt1);

        while((sciREG1->FLR & 0x4) == 4);
        sciSendByte(sciREG1, txt);
    }
}
void sciDisplayText(sciBASE_t *sci, uint8 *text, uint32 length)
{
    while(length--)
    {
        while((sciREG1->FLR & 0x4) == 4);

        sciSendByte(sciREG1, *text++);
    }
}

void wait(uint32 time)
{
    while(time)
    {
        time--;
    }
}
```


WISH
Wish Board
no 206
Va
Vb
Vc
NANO




FPGA
Vcc
Transistor
Bare metal
application
OS
device driver
HW
SW
FW
2πRC
Field programmable gate array
ARM
TI
ST
Atmel
HW
Firmware FW
SW
전원회로
Fabless
GIVREG
TASSYO
APM
master
slave
ON
Cortex-R5F
Cortex-M
ETS
TOEIC Speaking