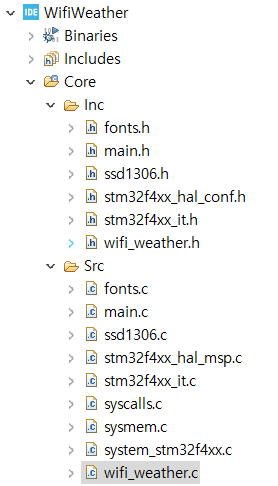
|  |  |
| --- | --- |
| 년도-학기 | 2021년 1학기 |
| 과목명 | 임베디드시스템설계 |

|  |  |
| --- | --- |
| **LAB번호** | **제목** |
| 11 | Cortex-M Oled Display |

|  |  |
| --- | --- |
| 실험 일자 | 2021년 5월 21일 |
| 제출자 이름 | 강\*\* |
| 제출자 학번 | 201803\*\*\*\* |
| 팀원 이름 |  |
|  |  |

**Chapter 1. 프로그램의 동작 방식 설명**

우선 oled를 사용하기 위해 강의자료 예제와 마찬가지로 fonts.h와 ssd1306.h 파일을 복사해 WifiWeather의 Core의 Inc에 붙여넣고, fonts.c와 ssd1306.c 파일을 복사해 Core의 Src에 붙여 넣었다.



기상청에서 받은 기상 정보 중에서 시간, 온도, 습도, 기상환경 정보를 확인한다. ret1 = (int)strstr((char \*)buffer, (char \*)"<temp>");

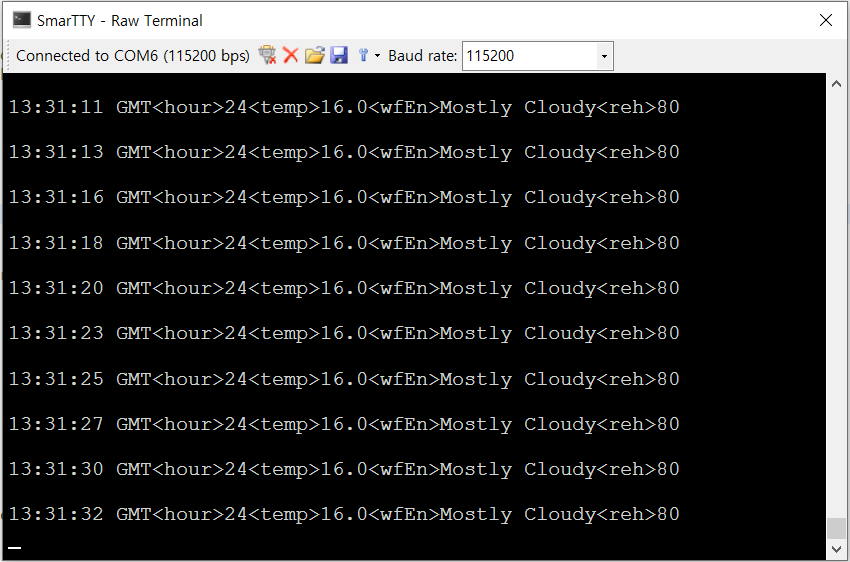
ret2 = (int)strstr((char \*)buffer, (char \*)"</temp>");와 같이 strstr을 통해 단어의 첫 주소를 확인해 그 사이의 기상 정보를 알아낸다.

현재 시간과 기상 정보의 숫자 값을 문자열로 저장하기위해 char timestr[50], char tempstr[50], char weatherstr[50], char humidstr[50] 변수를 설정하였다. for(int i = 0; i< ret2 -ret1 -6 ; i++) tempstr[i]=\*(uint8\_t \*)(ret1+i+6);

tempstr[ret2-ret1-6]=0; 와 같이 반복문을 통해 배열에 기상정보를 대입한다.

ssd1306\_WriteString((char\*)tempstr, Font\_16x26, White); ssd1306\_WriteString 함수를 이용해 문자열을 oled에 출력한다. 시간과 습도, 기상 상태도 같은 방법으로 출력할 수 있다. 단, 시간은 GMT 시간보다 9시간 빠르므로 이에 대한 계산이 필요하다. 기온의 ℃ 기호에서 작은 동그라미는 ssd1306\_DrawPixel을 통해 직접 그려 표현하였다.

**Chapter 2. 결과**



포트를 통해 확인한 결과로, 이 중 현재 GMT 시각, 기온, 기상 상태, 습도 값을 oled에 출력할 것이다.



현재시각, 기온, 기상상태, 습도를 Oled에 출력한 결과이다.

**Chapter 3. 결론 및 Discussion**

이전 wifi 통신 실습 강의와 Oled 실습 강의에서 해봤던 예제를 바탕으로 진행하니 reh 정보를 추가로 찾아내는 것과, oled에 출력하는 것 모두 어렵지 않게 해결할 수 있었다. 숫자 정보만을 배열에 문자열로 저장하는 것도 먼저 질문해준 학우 덕분에 교수님의 도움으로 바로 적용할 수 있었다.

어려웠던 점은 기온의 기호를 표현하기 위한 작은 동그라미였다. 비트맵을 구현하거나 인터넷에 찾아본 결과 도형을 그리는 것에 대한 설명이 있는 것 같았지만, 따라 해보기엔 시간이 부족하여 일단 픽셀을 하나하나 설정해 그려보았다. 조금 두껍게 그렸다면 조금 더 자연스러웠을 것 같다. 이후 여유가 있을 때 비트맵을 구현하거나 함수를 설정하여 선이나 도형들을 oled에 출력하는 것을 해보면 좋을 것 같다. 그리고, 기상 상태의 길이가 긴 경우 글씨 크기가 크면 뒷부분 일부가 출력되지 않는 경우가 생기는데 이를 두줄로 나눠서 표현하는 방법을 알고 싶었다. 문자열의 공백이 생기면 조건문을 통해 커서를 다음으로 옮기면 되지 않을까 생각해보았다.

**부록**

**#include** "stm32f4xx\_hal.h"

**#include** "string.h"

**#include** "stdio.h"

**#include** "ssd1306.h"

**#include** "fonts.h"

**extern** UART\_HandleTypeDef huart2;

**extern** UART\_HandleTypeDef huart3;

**void** **PrintString**(uint8\_t \* string);

**void** **my\_itoa**(uint32\_t val, uint8\_t\* str, uint8\_t len)

{

uint8\_t i;

**for** (i = 1; i <= len; i++)

{

str[len - i] = (uint8\_t)((val % 10UL) + '0');

val /= 10;

}

str[i - 1] = '\0';

}

**void** **WifiSetup**(**void**)

{

uint8\_t string[200];

uint8\_t buffer[200];

**int** length;

**for** (**int** i = 0; i < 200; i++)buffer[i] = ' ';

**strcpy**((**char** \*)string, "AT+CWMODE=3");

length = **strlen**((**char** \*)string);

string[length] = 0x0D;

string[length + 1] = 0x0A;

string[length + 2] = 0;

HAL\_UART\_Transmit(&huart3, (uint8\_t \*)string, length + 2, 0xFFFF);

HAL\_UART\_Receive(&huart3, (uint8\_t \*)buffer, 100, 1000);

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)buffer, 100, 0xFFFF);

string[0] = 0x0D;

string[1] = 0x0A;

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)string, 2, 0xFFFF);

**strcpy**((**char** \*)string, (**char** \*)"AT+CWJAP=\"iptime\_jk\",\"rkdwjdrud94\"");

length = **strlen**((**char** \*)string);

string[length] = 0x0D;

string[length + 1] = 0x0A;

string[length + 2] = 0;

HAL\_UART\_Transmit(&huart3, (uint8\_t \*)string, length + 2, 0xFFFF);

HAL\_UART\_Receive(&huart3, (uint8\_t \*)buffer, 100, 4000);

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)buffer, 100, 0xFFFF);

string[0] = 0x0D;

string[1] = 0x0A;

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)string, 2, 0xFFFF);

}

**void** **OpenWeather**(**void**)

{

uint8\_t string[200];

uint8\_t buffer[200];

**int** length;

**int** ret;

**do**

{

**for** (**int** i = 0; i < 200; i++)buffer[i] = ' ';

**strcpy**((**char** \*)string, (**char** \*)"AT+CIPSTART=\"TCP\",\"www.kma.go.kr\",80");

length = **strlen**((**char** \*)string);

string[length] = 0x0D;

string[length + 1] = 0x0A;

string[length + 2] = 0;

HAL\_UART\_Transmit(&huart3, (uint8\_t \*)string, length + 2, 0xFFFF);

HAL\_UART\_Receive(&huart3, (uint8\_t \*)buffer, 100, 4000);

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)buffer, 100, 0xFFFF);

string[0] = 0x0D;

string[1] = 0x0A;

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)string, 2, 0xFFFF);

ret = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"CONNECT");

} **while** (ret == 0);

}

/\*

void init\_oled(void)

{

ssd1306\_Init();

HAL\_Delay(1000);

ssd1306\_Fill(Black);

ssd1306\_UpdateScreen();

HAL\_Delay(1000);

}

void display\_oled(char str[])

{

ssd1306\_SetCursor(0, 0);

ssd1306\_WriteString("str", Font\_11x18, White);

}

\*/

**void** **WifiWeather**(**void**)

{

uint8\_t string[200];

uint8\_t buffer[2000];

uint8\_t query[500];

uint8\_t queryLength;

uint8\_t queryLengthString[5];

**int** length;

**char** timestr[50];

**char** tempstr[50];

**char** weatherstr[50];

**char** humidstr[50];

OpenWeather();

**for** (;;)

{

**sprintf**((**char** \*)query, "GET /wid/queryDFSRSS.jsp?zone=4127152500 HTTP/1.1\r\nHost: www.kma.go.kr\r\nConnection: keep-alive\r\n\r\n");

queryLength = **strlen**((**char** \*)query);

**for** (**int** i = 0; i < 50; i++)buffer[i] = ' ';

my\_itoa(queryLength, queryLengthString, 3);

**sprintf**((**char** \*)string, "AT+CIPSEND=%s", queryLengthString);

length = **strlen**((**char** \*)string);

string[length] = 0x0D;

string[length + 1] = 0x0A;

string[length + 2] = 0;

HAL\_UART\_Transmit(&huart3, (uint8\_t \*)string, length + 2, 0xFFFF);

HAL\_UART\_Receive(&huart3, (uint8\_t \*)buffer, 23, 4000);

**for** (**int** i = 0; i < 2000; i++)buffer[i] = ' ';

HAL\_Delay(1);

HAL\_UART\_Transmit(&huart3, (uint8\_t \*)(query), queryLength, 0xFFFF);

HAL\_UART\_Receive(&huart3, (uint8\_t \*)buffer, 2000, 4000);

//HAL\_UART\_Transmit(&huart2, (uint8\_t \*)buffer, 2000, 0xFFFF);

string[0] = 0x0D;

string[1] = 0x0A;

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)string, 2, 0xFFFF);

**int** ret1,ret2;

ret1 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"ERROR");

**if** (ret1 != 0) **break**;

ret1 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"GMT");

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)(ret1 -9), 12, 0xFFFF);

**for**(**int** i = 0; i< 5 ; i++) timestr[i]=\*(uint8\_t \*)(ret1+i-9);

timestr[5]=0;

timestr[0]=timestr[0]-48;

timestr[1]=timestr[1]-48;

timestr[1]=timestr[1]+9;

**if**(timestr[1]>10){

timestr[1]=timestr[1]-10;

timestr[0]=timestr[0]+1;

**if**( (10\*timestr[0]+timestr[1]) > 23 ){

timestr[0]=timestr[0]-2;

timestr[1]=timestr[1]-4;

}

}

timestr[0]=timestr[0]+48;

timestr[1]=timestr[1]+48;

ssd1306\_SetCursor(0, 0);

ssd1306\_WriteString((**char**\*)timestr, Font\_7x10, *White*);

ret1 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"<hour>");

ret2 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"</hour>");

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)(ret1), ret2 - ret1, 0xFFFF);

ret1 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"<temp>");

ret2 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"</temp>");

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)(ret1), ret2 - ret1, 0xFFFF);

//숫자만 분리하기 원래 출력되는건 <temp>13.0 이런식, 앞에 6개 빼버리고 문자열로 만들었음

**for**(**int** i = 0; i< ret2 -ret1 -6 ; i++) tempstr[i]=\*(uint8\_t \*)(ret1+i+6);

tempstr[ret2-ret1-6]=0;

ssd1306\_SetCursor(0, 37);

ssd1306\_WriteString((**char**\*)tempstr, Font\_16x26, *White*);

ssd1306\_DrawPixel(64, 35, *White*);

ssd1306\_DrawPixel(64, 36, *White*);

ssd1306\_DrawPixel(64, 37, *White*);

ssd1306\_DrawPixel(65, 34, *White*);

ssd1306\_DrawPixel(65, 38, *White*);

ssd1306\_DrawPixel(66, 33, *White*);

ssd1306\_DrawPixel(66, 39, *White*);

ssd1306\_DrawPixel(67, 33, *White*);

ssd1306\_DrawPixel(67, 39, *White*);

ssd1306\_DrawPixel(68, 33, *White*);

ssd1306\_DrawPixel(68, 39, *White*);

ssd1306\_DrawPixel(70, 35, *White*);

ssd1306\_DrawPixel(70, 36, *White*);

ssd1306\_DrawPixel(70, 37, *White*);

ssd1306\_DrawPixel(69, 34, *White*);

ssd1306\_DrawPixel(69, 38, *White*);

ssd1306\_SetCursor(71, 37);

ssd1306\_WriteString("C",Font\_16x26,*White*);

ret1 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"<wfEn>");

ret2 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"</wfEn>");

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)(ret1), ret2 - ret1, 0xFFFF);

**for**(**int** i = 0; i< ret2 -ret1 -6 ; i++) weatherstr[i]=\*(uint8\_t \*)(ret1+i+6);

weatherstr[ret2 -ret1 -6]=0;

ssd1306\_SetCursor(0, 15);

ssd1306\_WriteString((**char**\*)weatherstr, Font\_7x10, *White*);

ret1 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"<reh>");

ret2 = (**int**)**strstr**((**char** \*)buffer, (**char** \*)"</reh>");

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)(ret1), ret2 - ret1, 0xFFFF);

**for**(**int** i = 0; i< ret2 -ret1 -5 ; i++) humidstr[i]=\*(uint8\_t \*)(ret1+i+5);

humidstr[ret2 -ret1 -5]='%';

humidstr[ret2 -ret1 -4]= 0;

ssd1306\_SetCursor(93, 45);

ssd1306\_WriteString((**char**\*)humidstr, Font\_11x18, *White*);

ssd1306\_UpdateScreen();

string[0] = 0x0D;

string[1] = 0x0A;

HAL\_UART\_Transmit(&huart2, (uint8\_t \*)string, 2, 0xFFFF);

/\* wait for a while to slow down \*/

HAL\_Delay(2000);

}

PrintString((uint8\_t \*)"ERROR: Connection Closed. Restart the program.\r\n");

}