Lab1: Familiar with Arduino

1. Connecting to Arduino. Verify the board is working by using the example code

File -> Examples -> Strings -> StringCharacters

Select board-> Arduino MKR WiFi 1010 -> upload

Tools -> Serial Monitor

Results:String charAt() and setCharAt():

SensorReading: 456

Most significant digit of the sensor reading is: 4

SensorReading= 456

Lab1

- Sketch uses 13108 bytes (5%) of program storage space. Maximum is 262144 bytes.
- Global variables use 2988 bytes (9%) of dynamic memory, leaving 29780 bytes for local variables. Maximum is 32768 bytes.
- Atmel SMART device 0x10010005 found
- Device : ATSAMD21G18A
- Chip ID : 10010005
- Version : v2.0 [Arduino:XYZ] Mar 19 2018 09:45:14
- Address : 8192
- Pages : 3968
- Page Size : 64 bytes
- Total Size : 248KB
- Planes : 1
- Lock Regions : 16
- Locked : none
- Security : false
- Boot Flash : true
- BOD : true
- BOR : true
- Arduino : FAST_CHIP_ERASE
- Arduino : FAST_MULTI_PAGE_WRITE
- Arduino : CAN_CHECKSUM_MEMORY_BUFFER
- Erase flash
- done in 0.855 seconds

- Write 13108 bytes to flash (205 pages)

- [=========] 31% (64/205 pages)
- [==========] 62% (128/205 pages)
- [==========] 93% (192/205 pages)
- [==========] 100% (205/205 pages)
- done in 0.117 seconds

Lab 1

- Modify the example: stringCharacter
- Adding
- `int testString[16]={0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15};`

```
int total_start = testString[0];
```

What happens with Flash Storage Memory and SRAM memory

- Sketch uses 13124 bytes (5%) of program storage space. Maximum is 262144 bytes.
- Global variables use 2988 bytes (9%) of dynamic memory, leaving 29780 bytes for local variables. Maximum is 32768 bytes.
- Atmel SMART device 0x10010005 found
- Device : ATSAMD21G18A
- Chip ID : 10010005
- Version : v2.0 [Arduino:XYZ] Mar 19 2018 09:45:14
- Address : 8192
- Pages : 3968
- Page Size : 64 bytes
- Total Size : 248KB
- Planes : 1
- Lock Regions : 16
- Locked : none
- Security : false
- Boot Flash : true
- BOD : true
- BOR : true
- Arduino : FAST_CHIP_ERASE
- Arduino : FAST_MULTI_PAGE_WRITE
- Arduino : CAN_CHECKSUM_MEMORY_BUFFER
- Erase flash
- done in 0.852 seconds

- Write 13124 bytes to flash (206 pages)

- [=========] 31% (64/206 pages)
- [==================] 62% (128/206 pages)
- [==========================] 93% (192/206 pages)
- [==============================] 100% (206/206 pages)
- done in 0.124 seconds

- Verify 13124 bytes of flash with checksum.
- Verify successful
- done in 0.011 seconds
- CPU reset.

Lab 1:

- Assignment 1: Write a program to change LED color every second. If the green color blinks more than 10 times, the program should print out the value ”stop blinking” and turn off LED
- Tips: Sketch->include library -> manage libraries
- LIBRARY MANAGER -> search: ” WiFiNINA.h” and install WiFiNINA

Examples

```
#include <WiFiNINA.h>
#include <utility/wifi_drv.h>

void setup() {
  WiFiDrv::pinMode(25, OUTPUT); //define green pin
  WiFiDrv::pinMode(26, OUTPUT); //define red pin
  WiFiDrv::pinMode(27, OUTPUT); //define blue pin
}

void loop() {
  WiFiDrv::analogWrite(25, 255); //turn on green
  WiFiDrv::analogWrite(26, 0);   // turn off red
  WiFiDrv::analogWrite(27, 0);   // turn off blue


}
```

Progmem

- What is progmem?
- Progmem keyword allows to store data directly in the program memory (flash memory) rather than the SRAM (static RAM)
- Flash memory is non-volatile. It retains data even when the power is off. However, it's read-only during normal operation.
- Static RAM (SRAM): is where variables and data are stored during program execution. It’s volatile, meaning it loses data when the power is off, and its capacity is typically much smaller than flash memory.
- Example:
  - Storing string: const char myString[] PROGMEM = "Hello, PROGMEM!";
  - Storing Arrays: const uint8_t myArray[] PROGMEM = {1, 2, 3, 4, 5};

Read from Flash

```
const char myString[] PROGMEM = "Hello, PROGMEM!";
const uint8_t myArray[] PROGMEM = {10, 20, 30, 40, 50};

void setup() {
    Serial.begin(9600);

    // Read and print string from PROGMEM
    char buffer[20];
    strcpy_P(buffer, myString);
    Serial.println(buffer);

    // Read and print array values from PROGMEM
    for (int i = 0; i < 5; i++) {
        uint8_t value = pgm_read_byte(&myArray[i]);
        Serial.print("Array value ");
        Serial.print(i);
        Serial.print(": ");
        Serial.println(value);
    }
}

void loop() {
    // Empty loop
}
```

WiFi connection

```
#include <WiFiNINA.h>

char ssid[] = "your_SSID";       // your network SSID (name)
char pass[] = "your_PASSWORD";   // your network password
int status = WL_IDLE_STATUS;     // the WiFi radio's status

void setup() {
  Serial.begin(115200);

  // Check for the WiFi module:
  if (WiFi.status() == WL_NO_MODULE) {
    Serial.println("WiFi module not detected.");
    while (true);
  }

  // Attempt to connect to WiFi network:
  while (status != WL_CONNECTED) {
    Serial.print("Attempting to connect to Network named: ");
    Serial.println(ssid);
    status = WiFi.begin(ssid, pass);
    delay(10000); // wait 10 seconds for connection
  }

  Serial.println("Connected to wifi!");
}

void loop() {
  // Your main code here
}
```