Basics: Arrays

If you need to store 1000 different numbers, you’ll need to declare 1000 variables. Start typing. Just kidding, you only need 1 array that has 1000 slots. Let’s see how arrays work.

Suggested Prerequisites

- Circuits Intro
- Programming Intro
- Conditional Statements
- Functions
- Loops

Setup and Preparation

List of Supplies

- TI MSP430G2 LaunchPad
- Breadboard
- M/F Jumper Wires (4)
- LEDs of any color (3)

Prepare the Circuit

Create the following circuit (Figure 1) with small resistor values (around 100 Ω). This is the same circuit that was used in the lab Loops.

Figure 1: Arrays Circuit
**Pin Numbering**

In this lab, it will be easier to refer to pins by their pin number, instead of their pin name. The pin map for the MSP430G2 is shown in Figure 2.

![LaunchPad with MSP430G2553](image)

**Figure 2: MSP430G2 Pin Map**
Program 1: Arrays

Create and upload the following program (Figure 3) to the LaunchPad.

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

    // Make an array with 3 slots
    int arr[3];

    arr[0] = 3;
    arr[1] = 7;
    arr[2] = 11;

    Serial.println();
    Serial.println(arr[0]);
    Serial.println(arr[1]);
    Serial.println(arr[2]);
    Serial.println();
}

void loop() {
}
```

Figure 3: Arrays

Things to consider

- You can observe the output of this program with the Serial Monitor.
  - You may need to press the RESET button if the Serial Monitor is garbled with text.
- Try to draw a picture of what you think an array looks like conceptually.
- You can try to assign numbers to “slots” that are outside the range given. See what happens.
- You can find more information about arrays at the following link:
  - http://energia.nu/Reference_Index.html
Program 2: Arrays and For Loops

Create and upload the following program (Figure 4) to the LaunchPad.

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

    // Make an array with 5 slots and fill them right away
    int arr[] = {1, 3, 5, 7, 9};
    int numSlots = 5;

    Serial.println();
    for (int i = 0; i < numSlots; i++) {
        Serial.println(arr[i]);
    }
    Serial.println();
}

void loop() {
}
```

Figure 4: Arrays with For Loops

Things to consider

- You can observe the output of this program using the *Serial Monitor*.
- You may need to push the RESET button on the LaunchPad if the text in the *Serial Monitor* is garbled.
Program 3: Light Show

Create and upload the following program (Figure 5) to the Launchpad.

```c
void setup() {
    pinMode(11, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(13, OUTPUT);

    // Declare array with 7 slots, and fill them right away
    int pins[] = {11, 11, 12, 13, 11, 13, 12};
    int numSlots = 7;

    for (int i = 0; i < numSlots; i++) {
        blinkLED(pins[i]);
    }
}

void loop() {
}

void blinkLED(int pinNumber) {
    digitalWrite(pinNumber, HIGH);
    delay(250);
    digitalWrite(pinNumber, LOW);
    delay(250);
}
```

Figure 5: Light Show

Things to consider

- You can replay the program by pressing the RESET button on the LaunchPad.
- You can try to change the pattern of the light show.
Program 4: Reading and storing data

Create the following program (Figure 6), and upload it to the LaunchPad.

```cpp
void setup() {
    Serial.begin(9600);
    pinMode(RED_LED, OUTPUT);
    pinMode(PUSH2, INPUT_PULLUP);

    int numDataPoints = 100;
    int data[numDataPoints];
    int increment_ms = 50; // equates to 20 readings per second

    // Reading data...
    digitalWrite(RED_LED, HIGH);
    for (int i = 0; i < numDataPoints; i++) {
        data[i] = !digitalRead(PUSH2);
        delay(increment_ms);
    }
    digitalWrite(RED_LED, LOW);

    // Printing data in CSV format...
    Serial.println("Time (s), Value");
    for (int i = 0; i < numDataPoints; i++) {
        Serial.print(i*increment_ms);
        Serial.print(",");
        Serial.println(data[i]);
    }
}

void loop() {
}
```

**Figure 6: Reading and storing data**

**Things to consider**

- The program runs for approximately 5 seconds (Indicated by the LED). Use the PUSH2 button to input a pattern of data. You can observe the output of the program by using the Serial Monitor.
  - You may need to use the RESET button after opening the Serial Monitor.
- You can plot the data from this program. (Refer to the tutorial Plotting Data.)
Review

After this lab, you should have a good understanding of the following topics. If you’re not sure about some of them, go back through the lab and try to find a good place to explore the topic.

You should be able to do the following

- Declare an array with a certain number of elements ("slots"), and assign numbers into those elements.
- Use elements of arrays like normal variables. (i.e. add the 1st element to the 2nd element)
- Use a for loop to access numbers in an array in strategic patterns.
- Use an array to store meaningful data, such as in the light show example.
- Collect data from a real-world sensor and store the numbers in an array.

Application

Try applying the skills you learned from this lab. You can come up with your own project idea or try one from the flowchart.