Basics: Loops

Loops allow us to repeat sections of code, so that we don’t have to retype them.

Suggested Prerequisites

- Programming Intro
- Conditional Statements

Setup and Preparation

List of Materials

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

Prepare the Circuit

Create the following circuit (Figure 1).

Figure 1: Loops Circuit
Pin Numbering
Sometimes, it is easier to refer to pins on the LaunchPad by their pin numbers, as opposed to their printed pin name. The full pin map guide for the MSP430G2553 is given in Figure 2.

Figure 2: Pin Map for MSP430G2553
Program 1: While Loop

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

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

    int loopNumber = 0;
    int loopMax = 3;
    while (loopNumber < loopMax) {
        digitalWrite(11, HIGH);
        delay(500);
        digitalWrite(11, LOW);
        delay(500);
        loopNumber += 1;
    }
}

void loop() {
}
```

Figure 3: While loop program

**Things to consider**

- You can replay the program by pressing the RESET button on the LaunchPad.
- Consider the similarities between an *if statement* and a *while loop*.
- *Many things are happening in this program.* You should consider trying to explain this *while loop* using a block diagram.
- A description of the *while loop* can be found at the following link:
  - [http://energia.nu/reference/while/](http://energia.nu/reference/while/)
**Program 2: For loop**

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

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

  int loopMax = 3;
  for (int loopNumber = 0; loopNumber < loopMax; loopNumber += 1) {
    digitalWrite(11, HIGH);
    delay(500);
    digitalWrite(11, LOW);
    delay(500);
  }

  void loop() {
  }
}
```

**Figure 4: For loop program**

**Things to consider**

- You can replay the program by pressing the RESET button on the LaunchPad.
- Consider the similarities between a while loop and a for loop.
- You can try to explain this for loop using a block diagram.
- It is very common to replace loopNumber (also known as the “loop variable”) with a single letter variable, such as i, j, or k for the sake of simplicity.
- A description of the for loop can be found at the following link:
**Program 3: Iterating over a sequence**

Create the following program (Figure 5), and upload it to the LaunchPad.

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

  int loopMax = 3;
  for (int loopNumber = 0; loopNumber < loopMax; loopNumber += 1) {
    digitalWrite(11 + loopNumber, HIGH);
    delay(500);
    digitalWrite(11 + loopNumber, LOW);
    delay(500);
  }
}

void loop() {
}
```

**Figure 5: Iterating over LEDs program**

**Things to consider**

- You can replay the program by pressing the RESET button on the LaunchPad.
- This program is nearly identical to *Program 2*, but uses a subtle and clever modification.
- You can try to generalize when *for loops* are most useful.
Program 4: Waiting for a condition

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

```cpp
void setup() {
    pinMode(RED_LED, OUTPUT);
    pinMode(PUSH2, INPUT_PULLUP);

    boolean buttonIsPressed = 0;
    while (!buttonIsPressed) {
        digitalWrite(RED_LED, HIGH);
        delay(50);
        digitalWrite(RED_LED, LOW);
        delay(50);
        buttonIsPressed = !digitalRead(PUSH2);
    }
}

void loop() {
}
```

Figure 6: Waiting for condition program

Things to consider

- Use the PUSH2 button on the LaunchPad to characterize the behavior of the program.
- You can replay the program using the RESET button on the LaunchPad.
- You can try to generalize when `while loops` are most useful.
Program 5: Infinite loop

Create the following program (Figure 7), and upload it to the LaunchPad.

```c
void setup() {
  pinMode(RED_LED, OUTPUT);

  while (1) {
    digitalWrite(RED_LED, HIGH);
    delay(1000);
    digitalWrite(RED_LED, LOW);
    delay(1000);
  }
}

void loop() {
}
```

Figure 7: Infinite loop program

Things to consider

- This program behaves identically to the basic blink program, but does not utilize the native `loop()` function.
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

• Explain the logical flow of *while loops* and *for loops* with block diagrams.
• Use the “loop variable” of *for loop* to perform clever iterative operations, such as lighting up LEDs in a sequence.
• Use a *while loop* to execute code until a condition is met.

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.