Table of Contents Assignment for IoT Lab 1

➢ Task 1 (20 points):

1. Set up a WDT that reset without clearing it.
2. Set the Blue LED at the beginning of the program.
3. Clock (clock generator 2) frequency: 2048 Hz
4. Set the WDT period to 2 seconds.
5. Observe the behavior of the blue LED.
6. Do nothing in the main loop ().

➢ Task 2 (20 points):

- **Case 1**: Set up a loop that clears the WDT.

✓ Set the Blue LED at the beginning of the program.
✓ Clock (clock generator 2) frequency: 2048 Hz
✓ Set the loop period to 1 second, loop 10 times in the main loop () function such that it repeats.
✓ Clear(kick) the WDT in the loop
✓ Count down the number of loops and print a countdown message.

```
SerialUSB.print("Countdown");
SerialUSB.println(number);
```

Output:

Countdown 9

Countdown 8

Countdown 7

✓ Set the period using the delay(ms) function.
✓ Set the WDT period to 4 seconds.

- **Case 2**: Try it again without clearing WDT by commenting on the corresponding lines.

✓ Compare case 1 with clearing WDT and case 2 without clearing WDT and record the system behavior from the LED and the serial monitor message.

Question Task 2:

Discuss and explain the differences between the two cases.

➢ Task 3 (10 points):

✓ Write a function that generates a WDT period by arbitrary input.
✓ Clock (clock generator 2) frequency: 2048 Hz
✓ Function input: period (millisecond)
✓ Function:

- Calculate the register value based on the period

- Take the floor to the closest value for values that cannot be mapped to register value.

✓ Example function:

```
int setWatchdog(int period)
{
// your code
return register_value;
}
// Configure WDT
WDT->CONFIG.bit.PER = setWatchdog(period);
```

✓ Test it in the scenario of Task 1

Question Task 3:

1. How to get an accurate WDT period?
2. Is it necessary to use an accurate period?

➢ Task 4 (10 points):

For CSCE 838 Students" Mandatory" and CSCE 438 Students "Bonus"

- ✓ After a reset event is there a way for the MCU to figure out if the last reset was due to WDT? If yes,
- ✓ -Write code that detects if the last reset was due to WDT:
- ✓ If it was due to WDT, print a message in the console.
- ✓ What could be the importance of checking if the last reset was due to WDT? If no,
- ✓ Explain why it is not possible.

➢ **Documentation:**

✓ Record your development process.

1. Define Requirements
2. Write Code
3. Test
4. Result

➢ **- Final Deliverables:**

✓ - Report:

1. The requirements for each task
2. Development plan:

- The procedure for solving the problem

- The configurations used for each task

3. Test plan

4. Results:

- Answer the questions following each task

- Code snippets for each function

- Figures in the report:

- Screenshots that show you complete the required functions (serial message and Arduino IDE warning)

- Pictures that show you complete the required functions if necessary

- Test results

- For example, varying the WDT period to see how results change

✓ The entire program (Arduino sketch) in the appendix

➢ **Submission Instructions:**

✓ Submit your lab on Canvas on or before the deadline (Sep 2nd, 8:29 a.m.)
✓ Your submission should include one single .pdf explaining everything asked in the tasks and screenshots if any.
✓ Your submission should also include all the code you have worked on with proper documentation.
✓ Failing to follow the instructions will make you lose points.

➢ **References:**

✓ SparkFun SAMD21 Pro RF Hookup Guide: https://learn.sparkfun.com/tutorials/sparkfun-samd21-pro-rf-hookup-guide?_ga=2.127628877.1139230921.1561643965-144910588.1557512622#setting-up-arduino

✓ SparkFun Pro RF Documentation: https://www.sparkfun.com/products/14916