UNIVERSITY OF Waterloo

Department of Systems Designs Engineering

Biomedical Engineering

Digital Systems Laboratory
BME 393L Lab manual

Final Project

Systems Design Engineering
University of Waterloo
Waterloo, Ontario N2L 3G1, Canada

1. Intended Learning Outcomes

By the end of this lab, students will be able to:

- Identify design possibilities for a project offered by digital systems.
- Use course theory, guidelines, and best practices to implement a digital systems design.
- Demonstrate the results of their design to relevant stakeholders.

2. Background

Most groups choose to base their final project around a microcontroller (the Arduino Uno). Microcontrollers offer many important features for the digital designer, of these the final project focuses on Timers and Interrupts. Check Learn for a separate document that will cover these three concepts. Please note that using a microcontroller is not mandatory, but recommended as they are used extensively in industry and completing a project with one can provide valuable experience.

3. Lab Setup

This is a four-period laboratory. In the first two lab periods, each group will be expected to design and implement a project (see Section 4 for details). In the last period, the project must be demonstrated to the lab staff in a 3-minute informal presentation, followed by a short marking session. The grading session will be broken up over a two-week period.

Marks for this lab will be based on the following:

- 25% for the project's functionality.
- 37.5% for effective use of hardware (appropriate amount of functional course content, any additional hardware, polish or otherwise going above and beyond basic expectations)
- 37.5% for the group's comprehension of their design.

Note that simple projects have an upper grade limit, regardless of how effective they are at accomplishing the task.

Important: Each group member will be expected to understand the entire design, questions regarding any aspect of the project will be randomly asked to any group member.

3. Pre-Lab

Before the first lab session, each group must have had their project proposal approved, in person, by the lab instructor. To do this, each group should fill out Table 3.1 (below) and show it to the lab instructor who will then either approve the project or suggest modifications such that the project is appropriate. Fill out the table only after reading the entire lab manual.

Core project components should be on hand and ready to use before the start of the first lab session.

Table 3.1

Project Description	Which course concepts are used?	Key design functionality and constraints
NICU incubator - Heart rate - Respiration Rate - Temperature - Crying	Timers Interrupts Combinational/sequential circuitry	Inputs ● Baby's heart rate, read as normal (0) or abnormal (1). -> output: RED LED ○ Input determined by a switch ● Baby's respiration rate, read as normal (0) or abnormal (1). -> output: YELLOW LED for first 3 seconds, if not return to normal after 3 seconds output RED LED ○ Input determined by a switch ● Baby's temperature, read as normal (00), low (01), or high (10). -> output: BLUE LED ○ Input determined by a switch ● Baby is crying, read as not crying (0) or crying (1) -> output: WHITE LED ○ Input determined by a switch ● An input **button1** used by nursing staff to indicate that a required manual vitals check has been done -> return from YELLOW, BLUE or WHITE to GREEN ● An input **button2** to clear an emergency state (and put the system into the normal state) -> return from RED to GREEN

Inputs

Sensor Input	Sensor Meaning
Monitor	
Heart Rate (HR)	Normal
	Abnormal
Respiration Rate (RR)	Normal
	Abnormal (High or low)
Crying (C)	Not Crying
	Crying
Temperature T(1:0) -> 2 buttons	Normal
	Low
	High
Green Button (to resolve Crying)	
Red Button (to resolve Alarm state -> from abnormal HR or/and RR)	

Possible Cases

1. All is normal -> GREEN LED
2. Too hot (press yellow button) or too cold (press yellow and blue button), HR, RR normal + not crying -> BLUE LED
    a. If too hot -> print “Turning on AC.”
    b. If too cold -> print “Turning on Heat.”
    c. Timer
        i. Timer:
            1. print -> countdown from 3
            2. print -> “Temperature adjusted.”
            3. return to GREEN LED
3. Crying, T, HR + RR normal -> WHITE LED
    a. print “Baby is crying. Check on baby.”
    b. flash WHITE LED
    c. press **button1** to return to GREEN LED
4. RR is abnormal for 3 seconds, RR, T + not crying -> YELLOW LED + **timer**
    a. print “Alert. An abnormal respiration rate was first detected. Abnormal respiration rate protocol engage.”
    b. Timer:
        i. print -> countdown from 5
        ii. If RR abnormal after 5 seconds -> RED LED -> **interrupts**
            1. flash RED LED
            2. print “Warning. Abnormal respiration rate continues. Check on baby.”
            3. press **button2** to return to GREEN LED
        iii. If RR returns to normal after 5 seconds -> GREEN LED

1. print “Respiration has returned to normal.”

5. Too hot or too cold + Crying, HR, RR normal -> BLUE LED + WHITE LED
    a. If too hot -> print “Turning on AC.”
    b. If too cold -> print “Turning on Heat.”
    c. flash WHITE LED
    d. Timer
        i. Timer:
            1. print -> countdown from 3
            2. print -> “Temperature adjusted.”
    e. print “Baby is crying. Check on baby.”
    f. press **button1** to return to GREEN LED
6. Too hot or too cold + RR abnormal for 3 seconds -> BLUE LED + YELLOW LED + **timer**
    a. If too hot -> print “Turning on AC.”
    b. If too cold -> print “Turning on Heat.”
    c. print “Alert. An abnormal respiration rate was first detected. Abnormal respiration rate protocol engage.”
    d. Timer:
        i. print -> countdown from 5
        ii. If RR abnormal after 5 seconds -> RED LED -> **interrupts**
            1. flash RED LED
            2. print “Warning. Abnormal respiration rate continues. Check on baby.”
            3. press **button2** to return to GREEN LED
        iii. If RR returns to normal after 5 seconds -> remain on YELLOW LED
            1. print “Respiration has returned to normal.”
            2. return to GREEN LED
    e. Timer
        i. Timer:
            1. print -> countdown from 3
            2. print -> “Temperature adjusted.”
            3. BLUE LED OFF
7. Crying + RR abnormal -> WHITE LED + YELLOW LED + **timer**
    a. flash WHITE
    b. print “Baby is crying. Check on baby.”
    c. print “Alert. An abnormal respiration rate was first detected. Abnormal respiration rate protocol engage.”
    d. Timer:
        i. print -> countdown from 5
        ii. If RR abnormal after 5 seconds -> RED LED -> **interrupts**
            1. flash RED LED
            2. print “Warning. Abnormal respiration rate continues. Check on baby.”
            3. press **button2** to return to GREEN LED

iii. If RR returns to normal after 5 seconds -> remain on YELLOW LED + WHITE LED.
    1. flash WHITE
    2. print "Respiration has returned to normal. Baby is crying. Check on baby."
    3. press **button1** to return to GREEN LED

8. HR abnormal (don't care about any of the other states) for 3 seconds -> RED LED -> **interrupts**
    a. flash RED LED
    b. print "Warning. Heart rate is abnormal. Check on baby."
    c. press **button2** to return to GREEN LED

Include an error in case the wrong button is pressed

- per the specific case -> pressing the wrong button causes a print statement telling you to press the other button

Outputs

System Output	LED Condition
Monitor	Green LED On
Adjusting Temperature	Blue LED On
Baby is Crying	White LED On
Alert RR is abnormal	Yellow LED On
Alarm	Red LED On
Required to press a button to resolve	LED is flashing

More than one LED can be on at one time, except in the Alarm state. In the Alarm state only the red LED may be on.

4. In Lab Procedure

4.1 Project expectations

Each group is to select a project that interests them. The projects should be sufficiently complicated as to warrant approximately 12 hours of work (6 hours in lab and 6 hours out of the lab). See section 4.2 for a project ideas students may choose to design.

Projects must be hardware-based and typically include:

- microcontroller timers, not running at the Uno default speed
- timing of multiple concurrent events
- have multiple microcontroller interrupts

In addition, any combinational/sequential circuitry can be used. If a group wishes to do a project with only combinational/sequential circuitry, please connect early with the lab instructor.

The undergrad teaching lab will be open during the regularly scheduled lab times. A TA will be present during this time, and the lab instructor will also be available to answer questions. In addition to each group member's lab kits and the Falstad applet, each group may also use the DEN (den.uwaterloo.ca) and any additional hardware accessible to the group. No group will be allowed to take lab equipment or materials out of the lab.

For the presentation students should have their code available, clear schematics (hand drawn are acceptable), and a way to demonstrate timers used (i.e. serial.print time stamps, or an array with time stamps that are printed at an appropriate time).

4.2 Project Ideas

As a default project, students may create an embedded version of lab 4, a NICU incubator. The embedded system does <u>not require a dedicated FSM as lab 4 did</u>, instead the embedded system is based around a microcontroller, typically a group's Uno. This project's upper grade limit would be in the 85-90 range. For this project, the incubator has the following inputs:

- Baby's heart rate, read as nominal or abnormal.
- Baby's respiration rate, read as nominal or abnormal.
- Baby's temperature, read as nominal, low, or high.
- An input button used by nursing staff to indicate that a required manual vitals check has been done.
- An input button to clear an emergency state (and put the system into the nominal state)

The NICU incubator's Outputs:

- A status indicator denoting an 'all is good' state using a green LED.
- An alarm indicated by a red LED.
- An alert stated indicated by a yellow LED.
- A 'temperature is being adjusted' indicator, a blue LED.
- A slow flashing indicator to inform staff that it's time for a manual vitals check, colour determined by the group.

The system must behave in the following way:

- During normal operation (heart, respiration, and temperature are all nominal) a green LED must be on.
- Should the heart rate leave the nominal range for 3 seconds, the system will go into an alarm state.
- Should the respiration rate leave the nominal range for 3 seconds, the system goes into an alert state, indicated by a yellow LED. Should the respiration rate fail to return to a nominal rate within 9 seconds total (3 seconds before alert state plus 6 seconds in the alert state) the system will go to the alarm sate.

- When the temperature of the incubator is being adjusted, the heating or venting should be on for 5 seconds before the baby's temperature is rechecked. During temperature adjustments a blue LED should be on. The behaviour of the blue LED needs to be dynamic (not just on) and unique for heating and cooling. Each group will determine the heating and cooling encoding.
- If the system is in the nominal state for 10 seconds, it should flash the manual vital check LED. The only way to leave this state is for the nursing staff to press the vital check done button unless an alarm is triggered.
- The only way to leave the alarm state is for the nursing staff to press the reset button.

The system must use timers and interrupts in the following way:

- The two buttons must use the same interrupt (pin 2), after which the system will determine which button has been pushed.

Table 4.1: Default Project Input Table

Sensor Input	Possible values	Sensor Meaning
Heart Rate	GND	Normal
	5 V	Abnormal
Respiration Rate	GND	Normal
	5 V	Abnormal
Temperature	Determined per group	Normal
		Low
		High
Vital Check Done	5 V	Button not pressed
	GND	Button pressed
Clear Alarm State	5 V	Button not pressed
	GND	Button pressed

Table 4.2 Output Encoding Table

System Output	LED Condition
Monitor	Green LED On
Adjusting Temperature	Blue LED On and Flashing
Alert RR is abnormal	Yellow LED On
Sound Alarm	Red LED On

Time to do vitals check	Flashing (colour determined per group

Other project ideas:

Safe with unique unlocking requirements

Games (often inspired from boardgames, video games, etc.)

Light based Morse code system

Waveform generator

5. First Steps

I would recommend the following steps for your design:

1. Determine the inputs and outputs of your design.
2. Determine how time will be tracked in your system.
3. Determine how you will implement multiple interrupts from a single interrupt pin. Determine how additional inputs will be implemented.
4. Create a block diagram:
    a. Start with the input signals and end with the outputs.
    b. In between the inputs and outputs determine what series of blocks, or steps, you should take. A flow chart may also be helpful at this step.
    c. Between each block you should be clear what the input and output signals are (i.e. buses, single bits) and how those signals behave (i.e. normal high, low, bus encoding etc.)
    d. For digital systems it is normal for blocks not to interact with each other, or for only a subset of blocks to interact together.
5. Once you have determined your blocks, each one can be designed.
6. Implement each block and ensure it is functioning as intended.

6. Post Lab Questions

There are no post lab questions, or lab report due for these labs, it will be graded live during weeks three and four.

- How long the system has been running for
- How long between reset (clearing an alarm)
- Time between crying

No adding times together to track time (subtracting time)