**-<Project Name>: Team ?**

|  |  |  |
| --- | --- | --- |
| *Team Members (left-to-right on picture, above)* | *Class No.* | *Lab Div* |
|  |  |  |
|  |  |  |
| Iaman Alkhalaf | 5528-A | 6 |
| John Mahony | 8160-M | 4 |

|  |  |
| --- | --- |
| *Report/Functionality Grading Criteria* | *Points* |
| Originality, creativity, level of project difficulty | 20 |
| Technical content, succinctness of report | 10 |
| Writing style, professionalism, references/citations | 10 |
| Project functionality demonstration | 20 |
| Overall quality/integration of finished product | 10 |
| Effective utilization of microcontroller resources | 10 |
| Significance of individual contributions\* | 20 |
| *Bonus Credit Opportunities* | *Bonus* |
| Early completion | 0.5% |
| PCB for interface logic | 2% |
| Poster (required for Design Showcase participation) | 1% |
| Demo video (required for Design Showcase participation) | 1% |
| Design Showcase participation (attendance required)\* | 1% |

##### \**scores assigned to individual team members may vary*

|  |  |
| --- | --- |
| *Grading Rubric for all Criteria (Including Bonus)* | *Multiplier* |
| *Excellent* – among the very best projects/reports completed this semester | 1.0 - 1.1 |
| *Good* – all requirements were amply satisfied | 0.8 - 0.9 |
| *Average* – some areas for improvement, but all basic requirements were satisfied | 0.6 - 0.7 |
| *Below average* – some basic requirements were not satisfied | 0.4 - 0.5 |
| *Poor* – very few of the project requirements were satisfied | 0.1 - 0.3 |

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| 1.0 Introduction | 1 |
| 2.0 Interface Design | 2 |
| 3.0 Peripheral Utilization | 3 |
| 4.0 Software Narrative | 4 |
| 5.0 Packaging Design | 5 |
| 6.0 Summary and Conclusions | 6 |
| 7.0 References | 7 |
| Appendix A: Individual Contributions and Activity Logs | 8 |
| Appendix B: Interface Schematic and PCB Layout Design | 17 |
| Appendix C: Software Flowcharts | 19 |
| Appendix D: Packaging Design | 21 |

1. **Introduction**

*Provide a brief functional description of your project and describe the role each team member played,*

*Length should be about one page.*

1. **Interface Design**

## *Describe any external interfaces utilized (e.g., switches, LEDs, sensors). Include your Eagle or OrCAD schematic as Appendix B.*

*Length should be about* *one page.*

# **3.0 Microcontroller Resource Utilization**

In our code, we utilized five of the microcontroller’s peripherals as well as the the real time interrupt. The first peripheral we utilized was the pulse width modulation (PWM.) We used the PWM to to create a variable voltage that we used to create an sounds to confirm functionality on both the receiver side and the transfer side. We set MODR to 0x08 on the transmitter side in order to make make sure that the buzzer on PT3 would be used as the PWM output. MODR was set to 0x01 on the transmitter in order to enable PT0 as the output for the PWM on the receiver. The same values were set for PWME and PWMPOL corresponding to each the transmitter and the receiver in order to make sure the proper channels were being used.

We then used a potentiometer to change the duty cycle in order to control the volume of the buzzers by changing the DC voltage. In order to change the duty cycle, we connected the potentiometer to an analog-to-digital (ATD) input. In order to initialize the ATD sequence, we set ATDCTL2 to 0x80 in order to enable the ATD. We set ATDCTL3 to 0x08 in order to initiate 1 conversion per sequence. We did this in the timer interrupt service routine (TIM\_ISR.) We set TSCR1 to 0x80 in order to enable the power on bit. We set TSCR2 to 0x0C in order to reset TCNT when OC7 occurs and to set the prescale factor to 16.

We used the serial peripheral interface (SPI) to send data to the LCD screen similarly to how we have done in previous labs on the transceiver side. We used MOSI and SCK from PT6 and PT5 to provide a clock and data in to a 8-bit clock shift register to power the LCD on the screen on the receiver side. We used the same pins on the transmitter side on the transmitter side to control the sequence of our LED lights using a 10-bit clock shift register.

In order to get our transmitter and receiver to communicate with each other, we had to communicate using the serial communications interface (SCI.) The transmitter sent the data for the receiver from the TX pin on the SCI which the reciever acquired on the the RX pin of the SCI. In order to do this, we set SCIBH to 0x07 and SCIBDL to 0x50 on both the transmitter and the reciever in order to set the baud rate to 2000 bit per second in order to stay within the accepted baud region on our RF module. In the SCI interrupt service routines on the transmitter and reciever we created a buffer that was used to store the characters being sent and recieved between the microcontrollers. Finally, the transmitter utilized the real time interrupt (RTI) service routine in order to sample the pushbuttons. We did this by setting the RTICTL register to 0x27.

**4.0 Software Narrative**

*Describe what the software does and how it is organized/structured (i.e., event-driven, state machine, etc.). Submit your complete software listing on-line separately. Include a flowchart to document program structure in Appendix C.*

*Length should be about one page.*

The software architecture for this project can be broken down into two main parts: The transmitter, and the receiver. Each part is being managed by a single microcontroller.

* The Transmitter:
  + The main task for the transmitter side is to capture the customer’s feedback and send that to the receiver side. Customers evaluate the service based on three criteria: Food, Service, and overall satisfaction. Food and Service can be rated on a scale of 3, and overall satisfaction can be rated on a scale of 5. Receiving those ratings is obtained in an event-driven fashion through the use of pushbuttons. Each criteria has its own set of pushbuttons that span the whole scale of ratings. Once the customer enters all ratings, a submit pushbutton needs to be held in order to successfully send date to the receiver side.
  + Sampling the 12 pushbuttons is being in RTI. Sampling is accomplished through the use of seven PTAD ports. Three of those ports are programmed as outputs and they alternate so that only one port is low every time an RTI interrupt occurs. The other four ports are programmed as inputs. Each pushbutton is represented by a combination of an output port being low, and by an input port switching state from high to low, leading to a total of 12 possible combinations.
  + Each pushbutton has a corresponding LED. LEDs are being clocked manually by an output port pin and a data pin. The clock and data pin goes into a 12-bit shift register. Each time a pushbutton is pressed, an LED sequence is determined and then clocked through the use of a dedicated clocking function that receives the required LED sequence and clocks appropriately.
  + After the customer holds the submit button, a ‘ready’ flag set to indicate the the transmitter is done gathering data. After that, a fixed start byte and three other bytes based upon ratings are sent to a buffered character output routine BCO. BCO determines if the buffer is full or empty, places the character in the buffer, and enables the transmitter interrupt to see if data are ready to be sent to the receiving microcontroller.
* The Receiver:
  + The receiver is responsible for capturing ratings entered by the customer from the transmitter side and displaying a summary of those ratings on an LCD screen.
  + A buzzer is activated based upon a successful date receive. The volume of the sound is adjustable through adjusting the PWM duty cycle using a potentiometer connected to ATD input 0.
  + The receiver reads the four bytes received through using a buffered character input routine in a manner analogous to the transmitter side. The receiver maps the four received bytes (including the start byte) to the appropriate ratings. This is done through detecting which received byte of the four is a start byte and that constitute what the remaining three bytes refer to.
  + Through trial and error, the software team noticed that for each character sent through the RF link transmitter, there are three possible values that can be read by the receiver. This is most likely caused by noise and not having a MAX chip that regulates the voltage going into the Rx pin.
  + The Receiver then calls a display function that use the SPI module to send data to the LCD.  **Packaging Design**

*Describe the packaging design for your project; include drawings/photos in Appendix D.*

*Length should be about one page.*

1. **5.0 Summary and Conclusions**

*Describe what you learned from completing the project and what you might do to improve your design if you had more time.*

*Length should be about one page.*

Here is a summary of the most significant challenges we had and the lessons we learned through overcoming them:

* Challenge 1: interfacing 12 pushbuttons and LEDs
  + Lesson learned: “interface people, write something here”
* Challenge 2: Using SCI to communicate between two microcontrollers
  + Lesson learned: when working with multiple microcontrollers, using tera term and simple inchar/outchar messages is more efficient than stepping through the code or observing values as they update in the debugger.
* Challenge 3: High costs of reliable communication methods (i.e Xbee)
  + Solution: replace the with RF transmitters, much cheaper but requires additional software adaptation
  + Lesson learned: Under certain circumstances, cost can be minimized if there is a will to spend more time adjusting hardware/software to cheaper parts.
* Challenge 4: Having a dysfunctional voltage regulator chip
  + Solution: Use ‘pattern recognition’ to see how differently the receiver gets the data. It was noted that each character is received as one of three possible values. None of which corresponds to the original value sent. The software was edited to adopt to this observation and the receiver was able to successfully encode the received bytes.
* Challenge 5:
* Challenge 6:

**References**

*List any references (e.g., data sheets, application notes, web sites) used in formulating your solutions.* ***Be sure to cite these references in your report.***

*NOTE: Use IEEE format.*

**Appendix A:**

**Individual Contributions**

**and**

**Activity Logs**

**Activity Log for:** <name-1> **Role:** <role on team>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Activity*** | ***Date*** | ***Start Time*** | ***End Time*** | ***Time Spent*** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Written Summary of Technical Contributions:** <name-1>

*Provide a concise but sufficiently detailed description of your technical contributions to the project.*

*Length should be about one page.***Activity Log for:** <name-2> **Role:** <role on team>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Activity*** | ***Date*** | ***Start Time*** | ***End Time*** | ***Time Spent*** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Written Summary of Technical Contributions:** <name-2>

*Provide a concise but sufficiently detailed description of your technical contributions to the project.*

*Length should be about one page.***Activity Log for:** Iaman Alkhalaf **Role:** Software Leader

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Activity*** | ***Date*** | ***Start Time*** | ***End Time*** | ***Time Spent*** |
| Initial team meeting |  |  |  | 1 hour |
| Software Considerations Discussion |  |  |  | 3 hours |
| Writing the receiver side software, debugging breadboard circuit | Dec 5th | 12:00 pm | 10:00 pm | 8 hours |
| Collaborating with Peripheral leader to configure transmitter/receiver communication | Dec 6th | 12:00 pm | 10:00 pm | 8 hours |
| Consulting TAs | Dec 7th | 7:00 pm | 10:00 pm | 2 hours |
| Collaborating with Peripheral leader to transmitter/receiver communication | Dec 8th | 12:00 pm | 10:00 pm | 8 hours |
| Exploring the RF module, software changes | Dec 9th | 4:40 pm | 6:30 pm | 2 hours |
| Verifying software after installing PCBs | Dec 10th | 12:30 pm | 2:30 pm | 2 hours |
| Writing Appendix C and 4.0 of the report | Miscellaneous | | | 3 hours |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Written Summary of Technical Contributions:** Iaman Alkhalaf

*Provide a concise but sufficiently detailed description of your technical contributions to the project.*

*Length should be about one page.*

Mainly responsible for writing the software for the receiver side of the project. Spent a significant amount of time configuring and debugging the transmitter/receiver interaction through SCI. Verified project’s functionality using wired communication. Analyzed circuit’s behaviour after installing the RF module and adjusted the software accordingly. **Activity Log for:** <name-4> **Role:** <role on team>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Activity*** | ***Date*** | ***Start Time*** | ***End Time*** | ***Time Spent*** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Written Summary of Technical Contributions:** <name-4>

*Provide a concise but sufficiently detailed description of your technical contributions to the project.*

*Length should be about one page.*

**Appendix B:**

**Interface Schematic**

**and**

**PCB Layout Design**

*Paste a copy of your Eagle or OrCAD interface schematic here and (optionally) PCB layout.*

*Be sure to clearly identify the team member(s) responsible for producing this documentation.*

**Appendix C:**

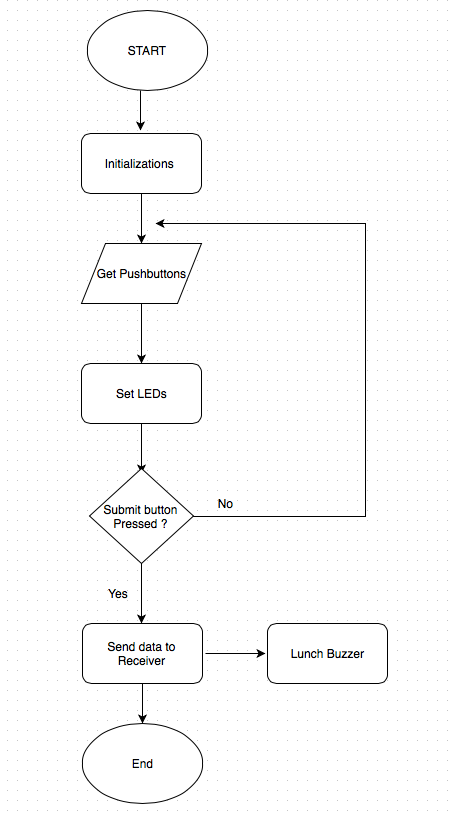
**Software Flowcharts**

*Include software flow diagrams and/or pseudo code here.*

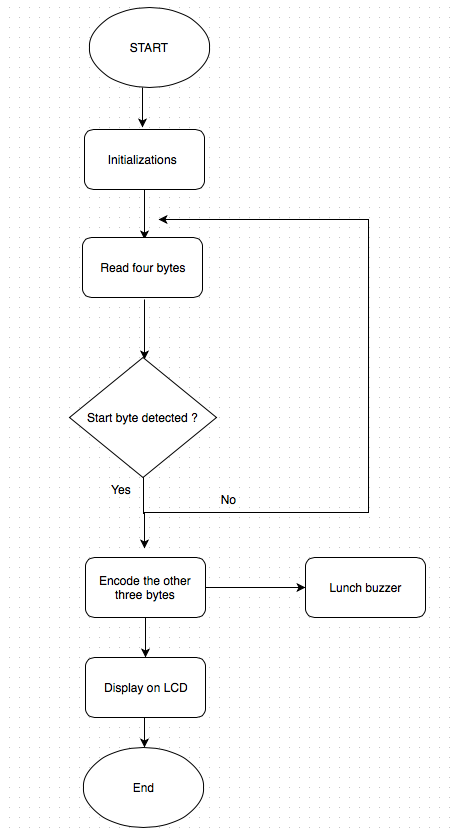
*Be sure to clearly identify the team member(s) responsible for producing this documentation.*

*NOTE: Software source listing file must be submitted on-line and should NOT be included here.*

**C - 1**: The Transmitter: flowchart made by Iaman Alkhalaf

**

**C - 2**: The Receiver: flowchart made by Iaman Alkhalaf

****

**Appendix D:**

**Packaging Design**

*Paste illustrations/pictures of your project packaging here.*

*Be sure to clearly identify the team member(s) responsible for producing this documentation.*