ECE 445

SENIOR DESIGN LABORATORY

DESIGN DOCUMENT

Clickers for ZJUI Undergraduate

BOWEN LI (Bowenli5@illinois.edu)
YUE QIU (yueq4@illinois.edu)
MU XIE (muxie2@illinois.edu)
QISHEN ZHOU (qishenz2@illinois.edu)

Sponsor : Timothy Lee

March 24, 2023

Contents

1 Introduction

1.1 Problem

Improving student engagement in class remains a challenge for instructors, not only for online classes but also for in-person classes. Engaging students through active learning pedagogies that rely on attendance is important for improving the quality of teaching. In fact, a 2010 meta-analysis found that attendance positively affects both course grades and GPA, and is the single strongest predictor of college grades. [1]. It is important to engage with them through active learning pedagogues that rely on student attendance. Thus, the i-clicker system can help stat the attendance rate effectively, and improve the students' concentration by encouraging them solve the in-class problems.

The I-clicker system used in UIUC for undergraduates was designed for solving these problems. It requires students to participate in class and answer questions to earn credit. Nowadays there are mainly two kinds of i-clickers: One is based on an embedded system and the other is based on a web application. Compared with the web applications running on laptops and cell phones, those wireless i-clickers have the following advantages: they are easy to carry, they can accurately match and identify user tags, they are difficult to cheat, and would not distract students. Unfortunately, most of the i-clickers used on campus have the following problems: inconsistency, high response delay, poor signal, and manual matching.

1.2 Solution

The signal transmission chip in some clicker in UIUC is time out. So we can deal with the problem mentioned above by using a newer signal transmission chip.
To deal with the problem of high delay and poor signal, we decide to replace the origin chip with nrf24l01, which has higher transmission efficiency and wider channel. It can match up to 256 receiving ports by modifying the communication protocol. More details with this chip will be covered on tolerance part. As it can support up to 256 receivers, it can meet the size of user group. But the cost of each clicker will increase by 0.3$.

1.3 Visual Aid

Here is a comparison between nRF24l01 chip and CC1101 chip.
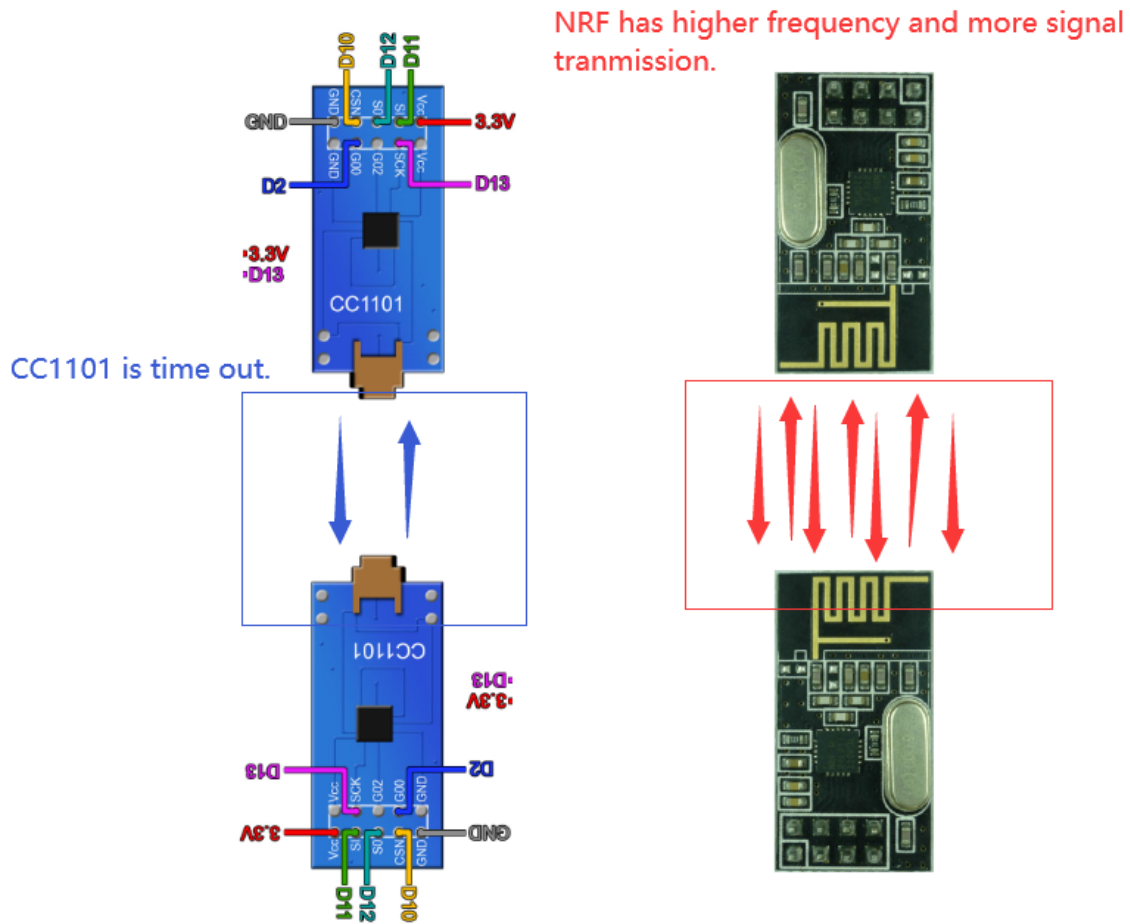
Figure 1: Visual Aid

1.4 High-level Requirement Lists

1. The total power of the portable i-clicker system should be between 2W to 5W to effectively prolong the battery life for long-time operation.

2. The transceiver's effective working radius should be at least 120 meters and support up to 125 multichannel connections to cover most classroom structures. The base station should also have a Low Noise Amplifier module to improve the quality of signal reception and eliminate noise signals.

3. The data transmission between i-clicker and transceiver should be completed within 500 milliseconds to ensure timeliness of the polling results.

2 Design

2.1 Block Diagram

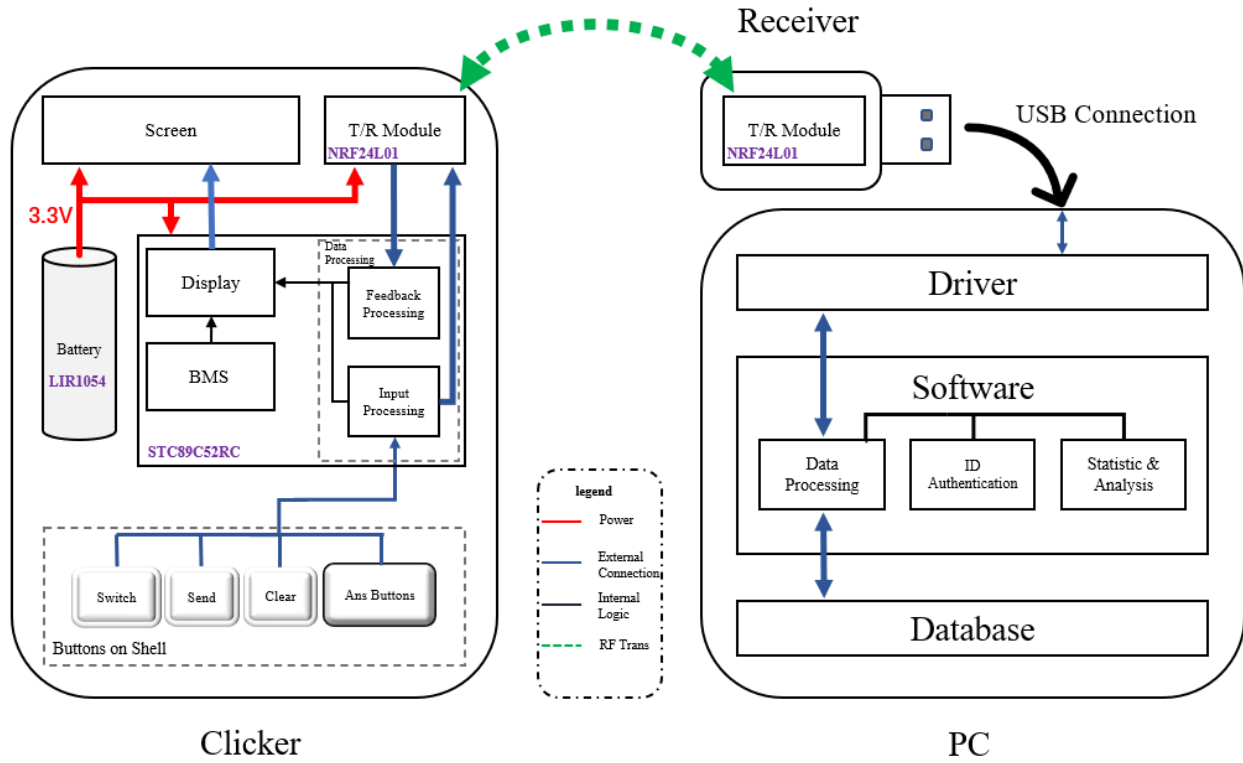Figure 2: Module Blocks

2.2 Circuit Design
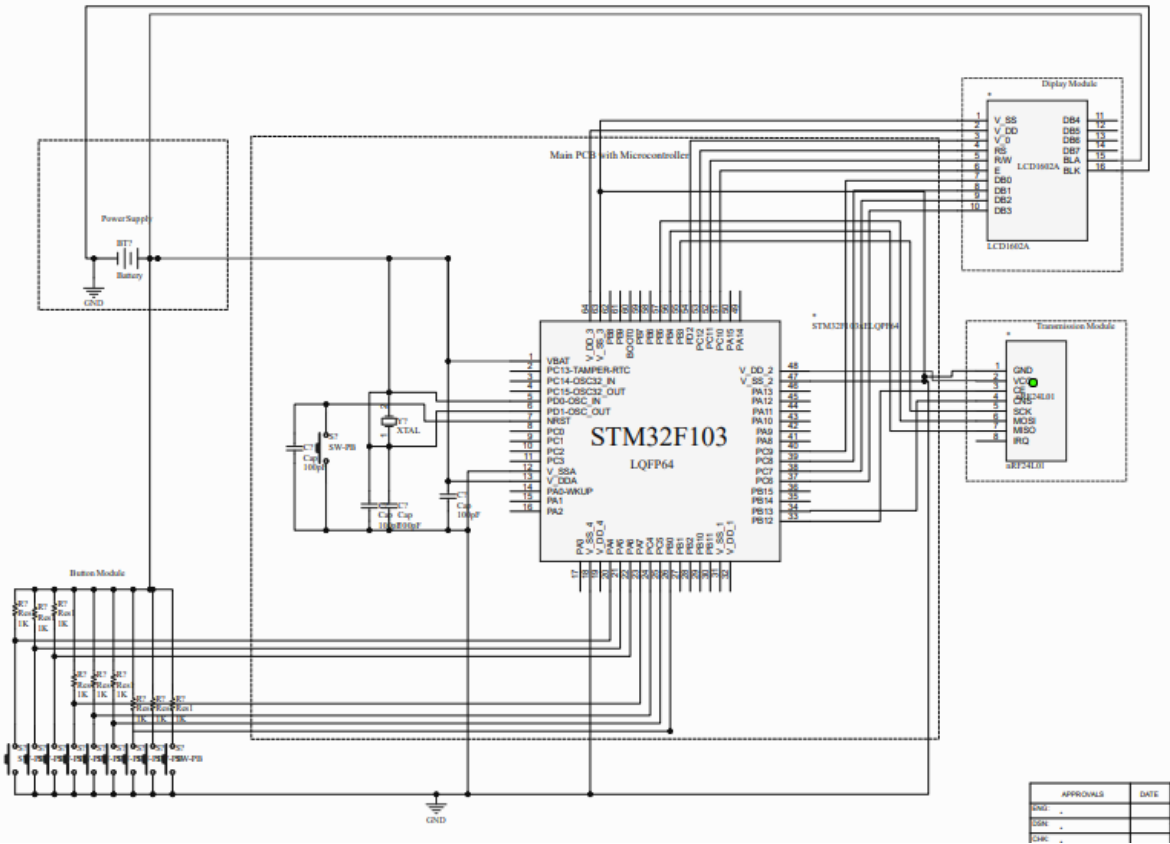
Figure 3: Circuit Design

2.3 Subsystem Overview and Requirement

2.3.1 DC Power Subsystem

We need a direct power supply to power our SCM and LCD display. Considering product size and LCD brightness, voltage should be 3.3V-5V and current should be less than 5 mA. Too high voltage can easily cause eye damage to the screen and too high a current is likely to cause security risks. And it must also be packaged into the clicker product. So we will going to use button battery LIR1054(fluctuation voltage from 3.6v to 4.2v) for power.

2.3.2 Display Module

Clicker needs a screen to provide feedback to users. We choose LCD1602A to display to users. LCD1602A will receive data stream from SCM and 3.3V - 5V voltage from DC power. Users can receive feedback on their display screen via driver, nRF24l01 Hardware side and STC89C52RC.

4

2.3.3 SCM Hardware Logic Subsystem

We use a SCM to compile the display module, send and receive message from wireless transceiver module and receive hardware signal from keystroke. It also needs a voltage supply. STC89C52RC is a common and popular SCM at present, and its supply voltage is between 3.4V to 5.5V. This is exactly what we need. This module should run all the time after power is on and send data stream to the wireless transceiver module upon receipt of a keystroke command. During power-on, SCM should always keep the screen refreshed, that is, update the contents of the screen every unit of time.

2.3.4 Wireless 2.4GHZ Signal Transceiver Module

This module is divided into two parts, one is connected to the software side of the computer, the other is connected to the hardware side of the SCM. A DC power is also needed. We choose nRF24l01 chips as our signal transmission part. One chip on the hardware side should be connected to STC89C52RC and DC power, the other chip on the software side should be connected to AS06 bottom plate and then connected to USB port on the computer. Once one side receives the data stream from the procedure, it sends it to the other one over 2.4GHZ wireless signal.

2.3.5 Hardware circuit connecting module

We will use eight buttons to form a three-bit keystroke input signal. The only function is to connect to the SCM, by giving different signals as different commands, and then transfer the SCM into different states. We choose a four-corner button switch with dimensions of 6mm x 6mm x 7mm for the specification.

2.3.6 Clicker Overview

After all the subsystems are connected, we need a plastic shell to encapsulate it. STC89C52RC, 8-buttons keystroke, DC power LIR1054, LCD1602 screen and one nRF24l01 wireless transceiver chip should be wrapped in it.

2.3.7 Signal Conversion Subsystem

A driver is required on the USB interface. After receiving data stream from hardware, computer need a driver to Convert hardware signals into software signals. Similarly, when the software wants to send a feedback signal to the hardware, it should also use this driver to convert the software signal into the hardware signal.
Functions of driver:

1. Reception

    (a) Read response messages from USB port.

    (b) Filter out messages with incorrect format.

    (c) Extract ID and answer information from messages.

2. Sending

    (a) Add wrapper to feedback messages from Data Processing Unit.

    (b) Write Feedback message to USB port.

We choose module pySerial which encapsulates the access for the serial port to implement the driver. It provides back ends for Python running on Windows, OSX, Linux, BSD (possibly any POSIX compliant system) and IronPython. The module named "serial" automatically selects the appropriate back end.

The driver we design should be able to detect the port the Receiver is plugged into. To receive data from USB port, serial port should be initialized with Baud Rate = 9600, which refers to the rate of the effective data signal modulation carrier, that is, the number of times the carrier modulation state changes per unit time.

Communication Protocol

The communication protocol is embedded into the driver. In the primary communication protocol, we set the format of response to be *"ID{4}+ANS{1}"*, totally 5 digits. Similarly, the format of feedback to be *"ID{4}+Correctness{1}"*. For "Correctness" field, 1 is for correct, and 0 is for wrong. The driver should filter out messages with incorrect format by a helper function.

Requirement	Verification
90% of responses from Clickers should be received correctly	1. Test the driver with multiple receivers send concurrently. 2. Check whether the database has collected more than 90% of the responses and all response are in the correct format.
Almost all of feedback to Clickers should be sent correctly with low latency	1. After sending a response, clickers will wait for corresponding feedback in a short period. 2. Check whether almost all Clickers could get feedback expected within a shord period after "Answer Section".

2.3.8 Data Processing Subsystem

Software ought to be able to process the data from driver accurately, analyse the data and generate feedback to each Clicker.

Functions of Data Processing Unit:

1. Attendance

2. Score

    (a) Mark each answer.

    (b) Record the answer or score in database.

3. Enquiry

(a) Query the class attendance history.

(b) Query students' answer or score history.

4. Generate Feedback

Teacher could keep track of the students' answers and display it visually through statistical charts. After each "Answer Section", data processing unit will count the number of each option and draw a histogram correspondingly, where the bar of correct answer is green and the one of wrong answer is red, Figure4. Data processing unit could correct students' answers according to the correct answer input by teacher previously. The data processing unit will score each student's final answer and record the score and option of each student in database. Meanwhile, the unit will generate feedback according to the correctness to students at the end of each "Answer Section". It will deliver feedback message with format "*ID{4}+Correctness{1}*" to the deliver.

At the beginning of each course, teacher starts a "Course Section". Once a student answer a question within the "Course Section", the unit will record his/her attendance. After class, teacher could check the detailed attendance on student list or overall attendance with a sector diagram, Figure5.

Interacting with database, Data Processing Unit is able to provide historical data to teacher. Hence, teacher could have a better grasp of students' learning and make adjustment to optimize teaching content.
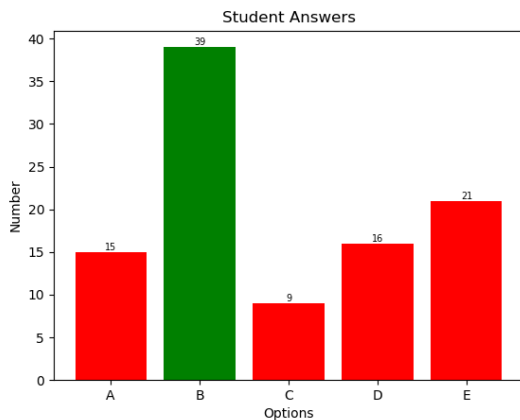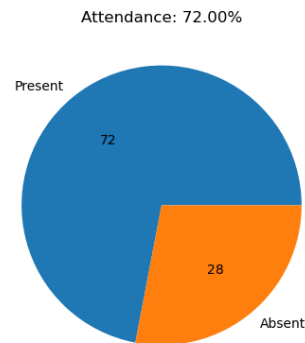

Figure 4: Student Answers

Figure 5: Attendance

Requirement	Verification
Attendance should be recorded accurately	1. Start a short "Course Section" with several "Answer Sections", and use a number of specific Clickers to answer one or more questions. 2. Check if all students' attendance is recorded if and only if they once answer a question and their responses are also received by driver successfully. And inspect attendance information in student list is matched correctly.
All scores and options should match the responses from Clickers	After an "Answer Section", compare the message delivered by driver with final result to inspect whether the options are the same as the one in message as well as whether only the correct answers score. This process could be done by a subroutine.
All enquiry on historical data should be correctly and completely	Maintain a log of statistic data, compare the log with the historical data provided by Data Processing Unit after a spectrum of periods. This process could be done by a subroutine.
All feedback should be generated and send correctly within low latency	Examine the feedback messages delivered to the driver after "Answer Section". Each feedback ought to be matched with student's ID and answer.

2.3.9 ID Authentication

Clickers is identified by the unique identifier on SCM chip. Correspondingly, users could also bind their identities to Clicker identifier at the beginning of a semester or the time they get the Clicker. The method of registration is yet to be decided. Current scheme is as follows.

Identity Binding Methods:

- Manual Binding. I-Clicker system in the UIUC use this method, which requires teaching professor to collect students' ID together with I-Clickers' identifier on a notebook, and then, bind them together manually. But. obviously, it is inefficient.

- Signal Binding. A type of clicker apply this method, which requires teaching professor to open up an ID collecting section for student to input and send their id to terminal equipment (Teacher's computer). This method is quite efficient, but could be unfriendly for students who missed the section and number keys should be implemented on the design.

- Internet Binding. An other potential scheme is to build up a website for registration where students can bind/unbind the machine on their own no matter when. But, there will be extra effort to build such a website.

Requirement	Verification
All ID Binding should be Correct	1. After a series of answering, check whether the score/status on Data processing Unit is matched to testers' operations. 2. Inspect whether the identity binding is one-to-one mapping.
Efficient to Operate	The method of registration should be handy to operate.

2.3.10 Database Subsystem

We need a local MySQL database to store the historical record on the computer. Before the data processing subsystem returns feedback, it will firstly store the results and scores of the data processing into the database. The database stores information in Key-Value Pairs and has interfaces for processing subsystem calls scores or raw data and for ID Authentication to record, modify, query or delete the ID binding information.
A valid entry in database should contains student's ID as primary key, Clicker's ID, total score & attendance, and detailed score & attendance for each Course Section.

Requirement	Verification
Stable database interface	Any error of database operations should be detected and fixed by program immediately during Course Sections.
Permanent and correct record	All entries in database should be stored correctly after a long-term.

2.4 Tolerance Analysis

Based on our hardware design, the RF24 wireless system are basically work with half-duplex network system, thus we should consider the Media Access Control to avoid the signal collision. Due to the performance limitation, we can not synchronize among all the i-Clickers, then we would apply unslotted CSMA/CA algorithm to control the communication between base and clickers. In order to reduce delay and ensure the reliability and stability, the performance of our network need to be deeply studied.

Now we assume that there are **n** clickers are associated to the central base server in a class, and each device only have one packet to transmit per communication window. It is possible that students may want to resubmit their answers, thus we assume that each device would generate packets according to Poisson process with the rate λ. Furthermore,

we assume that the payload packet transmission time would be T_{Tx}, and the ACK packet transmission time would be T_{ack}, and the expected server response time would be $E[S]$, and the average turnaround time would be T_{turn}, According to the algorithm, the clickers must wait for a whole Clear Channel Assessment(CCA) time before counting down their back-offs or start transmitting, thus we assume that the CCA time would be T_{cca}. We also define that the max trial times is M. After M trial, the packet would be discarded.

By referring to the article posted by Tae Ok Kim etc. [2], we could evaluate the whole model in M/G/1 queuing model. According to the model, we could know that during one busy cycle, the average successful transmission number would be $(1 - P_{Loss}) * E[\Gamma]$, where the P_{Loss} would be the packet loss rate, and the $E[\Gamma] = \frac{1}{1-\rho}$ is the number of packets served in that busy period, with the traffic intensity ρ:

$$\rho = \lambda * (E[D_{hol}] + T_{Tx} + 2 * T_{turn} + T_{ack}) \tag{1}$$

With the $E[D_{hol}]$ denotes that delay between the packet generated and the period that just before the packet being transmitted or discarded in our scenario. The $E[D_{hol}]$ can be calculated as below:

$$E[D_{hol}] = \sum_{v=0}^{m} \alpha^v (1 - \alpha) \sum_{i} = 0 \frac{W_i - 1}{2} * \delta + (v + 1) * T_{cca}$$

$$+ \alpha^{M+1} \sum_{i=0}^{M} \frac{W_i - 1}{2} * \delta + (M + 1) * T_{cca}$$

With the α denotes the possibility that the channel is busy when the node's back-off is zero, and the article also gives the equations to resolve the value:

$$\alpha = \frac{(n - 1)(1 - P_{Loss}) E[\Gamma] (T_{cca} + T_{Tx} + 2 * T_{turn} + T_{ack})}{\frac{1}{\lambda} + E[\Gamma] * E[D_{hol}]} \tag{2}$$

$$= \frac{(n - 1)(1 - P_{Loss})(T_{cca} + T_{Tx} + 2 * T_{turn} + T_{ack})}{\frac{1}{\lambda} - (T_{Tx} + 2 * T_{turn} + T_{ack})} \tag{3}$$

With the equations above we can calculate the expected total delay $E[D]$ of the system according to the queuing model given by [3]:

$$E[D = \frac{\lambda * E[s * *2]}{2(1 - \rho)} + E[D_{hol}] \tag{4}$$

In our model, according to our design, the nRF24 would use a payload packet with 64bytes and a Ack packet with 48 bytes. Hense, we would apply transmission rate at 256Kbps, and the average serving time would be 100ms for each packet. Based on the classroom structures, we would set our typical transmission time T_{turn} at 0.4ms, and the T_{cca} at 0.8 ms accordingly. By inserting in corresponding parameters obtained from datasheet, and consider the general conditions above, we could acquire the expect delay over the numbers of students with different i-clicker answering rate λ: We can find that the
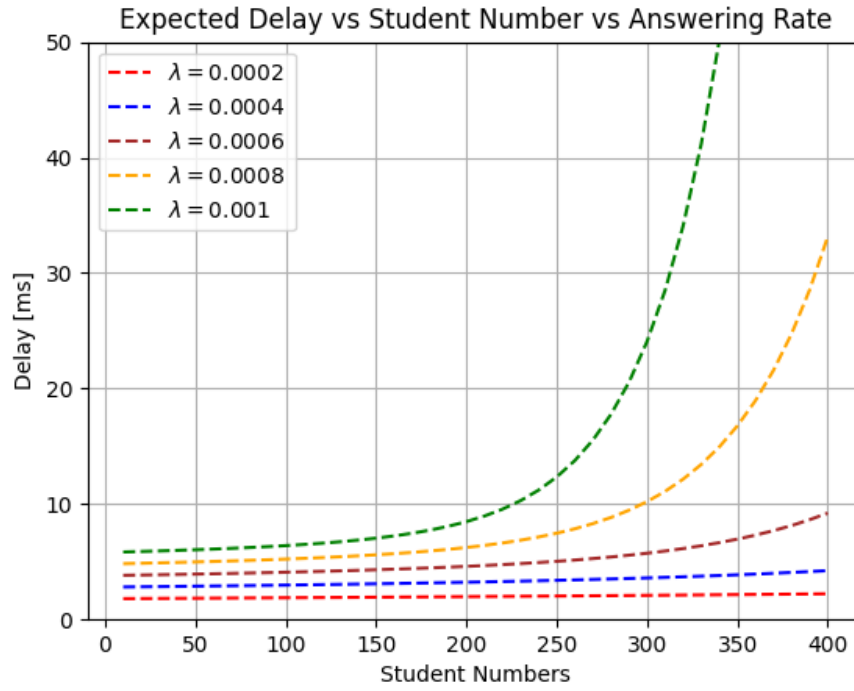
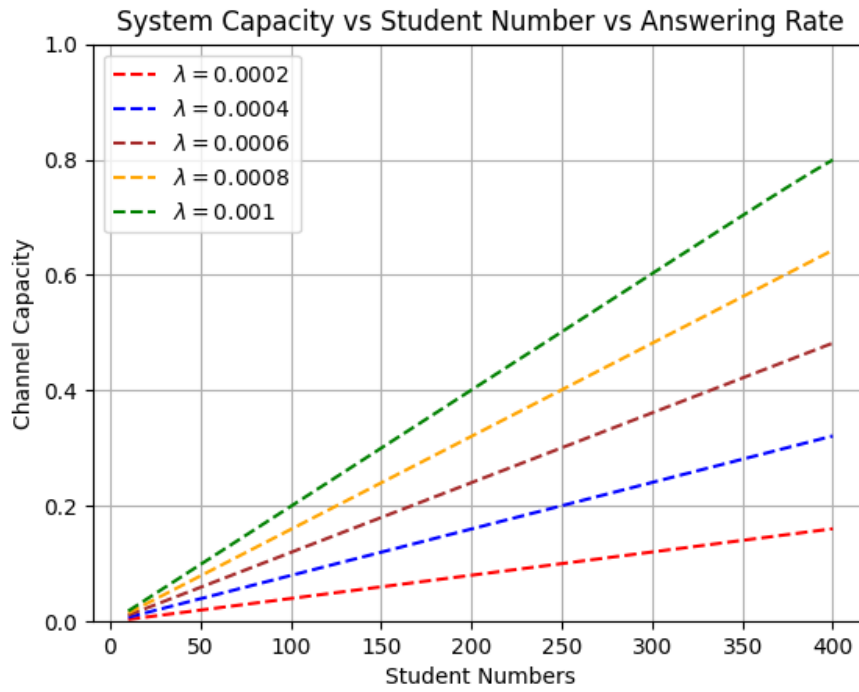Figure 6: Expected Delay vs Student Number vs Answering Rate

Figure 7: Expected Capacity vs Student Number vs Answering Rate

latency increased sharply when λ larger than 0.001, which means if all the students start to transmit answers at average rate of 1 times/sec, there is a high probability that the system will be clogged. The system capacity figures below would also shows this: Hence, we should set the built-in send intervals in our transmitting logic, in order to avoid the potential crash possibilities. The build-in send intervals would set around 2 seconds, to keep the overall flow of communication.

3 Cost and Schedule

3.1 Cost

3.1.1 Labor Cost

Our hourly rate is $10 per hour for each member and 20 hours per week. We need 12 weeks to finish our project. The Labor cost is approximately

$$4 * 10 * 20 * 12 * 2.5 = 24000\$$$

3.1.2 Parts Cost

We bought 3 arduino Uno for debugging, and will make 8 i-clickers to test many to one transmission.

Part	Quantity	Unit Price	Total Cost
Arduino Uno	3	128 CNY	384 CNY
3D Printing Consumables	2.5kg	50 CNY	125 CNY
Nrf24l01 chips	8	5 CNY	40 CNY
STM32F103ZE	8	2.5	20 CNY
STC89C52RC	8	3 CNY	24 CNY
PCB print	3	20 CNY	60 CNY
switch	150	0.15 CNY	22.5 CNY
LIR1054(button battery)	8	3 CNY	24 CNY
Total			689.5 CNY

Total cost for product prototype is 689.5 CNY. Each additional i-clicker costs 16 CNY more.

3.2 Schedule

	Bowen Li	Yue Qiu	Qishen Zhou	Mu Xie
2.20 - 2.26	Learn the knowledge on Arduino and SCM. Buy the related hardware components.	Learn how to build a communication protocol. Buy wireless transmission chips and Start to write proposal.	Review on how to build a database. Start to write the code on software side.	Learn the detailed knowledge on PCB design and start to write proposal and design document.
2.27 - 3.5	Design switch logic circuit and liquid crystal display. Connect it to arduino.	Realize wireless signal between computer and nrf24l01 module.	Write the communication protocol with C or C++.	Connect nrf24l01 module to Arduino. Finish relative program code on hardware.
3.6 - 3.12	Finish the whole code on hardware with Mu Xie. Finish proposal with others.	Write basic data processing program on software part and realize data exchange with database.	Build up a database with MySQL locally. Write proposal.	Write the hardware code in Arduino with C++. Ensure signal transmission on all the components.
3.13 - 3.19	Learn how to transfer the language on arduino to SCM. Write design document.	Finish design document with others. Resolve multiple channel conflicts and limitation on many to one signal transmission issues.	Finish design document. Debug for the whole program code on hardware and software until now.	Consult different types of SCM workbooks and choose the most suitable SCM by burning program to it one by one.
3.20 - 3.26	Build our I-clicker prototype version 1.0 with Arduino and purchase typical SCM for choice.	Realize sending a signal on the computer side and receive it on the arduino terminal.	Help with Yue Qiu in this week. Realize sending a signal on arduino terminal and receive it on the computer side.	Test. Ensure all the signal transmission on hardware side is as expected and same with version 1.0.

14

3.27 - 4.2	Burn the programs on Arduino to SCM and transfer all the connection to breadboard. This is our prototype version 2.0.	Ensure all the signal transmission in prototype version 2.0 is same with previous version.	Finish data process with Yue Qiu. Realize machine code identify detection.	Integrate hardware part and weld. Work together with Bowen Li.
4.3 - 4.9	Continue with prototype version 2.0. Deal with the code if language in Arduino and SCM have several difference.	Realize signal transmission many-to-one and one-to-many. Add machine code for each nrf24l01 chip.	Complete a mature data process program. Make and refine the front end with Yue Qiu	Refine the hardware design on prototype version 2.0 for the next PCB design.
4.10 - 4.16	Replace power supply with button battery. Test the prototype with software part.	Structure computer side applications to evoke the whole system with Qishen Zhou. Structure front end.	Research on how to realize 256-to-one wireless signal transmission.	Design the shell of the I-clicker with software modeling tools.
4.17 - 4.23	Simplify the prototype version 2.0. Create a PCB design sample and print it.	Modify communication channels and protocols with Qishen Zhou. Try to induce interference.	Modify the communication chanel and frequency band.Try to connect host to as many i-clickers as possible.	Debug for the final I-clickers prototype. Refine the Wireless receive part on USB port on computer.
4.24 - 4.30	Finish Version 3.0. Test all the i-clicker system. Handle signal problems in different scenarios.	Encrypt all the signal transmission.	Decryption the data on database. Try to give different feedback to different results.	Connect integrated prototype with PCB board and the plastic Shell with 3D print. Weld. That is final version.

5.1 - 5.7	Prepare for the final demonstration and report.	Prepare for the final demonstration and report.	prepare for the final demonstration and report.	prepare for the final demonstration and report.
5.8 - 5.14	Report and demo	Report and demo	Report and demo	Report and demo

4 Ethics & Safety

4.1 Ethics

During the development and use of our project, we must commit ourselves to the highest ethical standards and avoid any breaches. One important consideration is that our devices use 2.4GHz Wi-Fi as a data transmission medium, which may interfere with the normal operation of some vital devices, particularly medical devices. This violates the first principle of the IEEE Code of Ethics, which is to hold paramount the safety, health, and welfare of the public [4]. To address this, we must propose limits on the wireless frequency range of equipment used and specify the scenarios in which they can be used, as well as use warnings to relevant groups.

We should also take responsibility for any problems that arise from the use of our products. According to the sixth principle of the IEEE Code of Ethics, we must maintain and improve our technical competence and only undertake technological tasks for others if we are qualified by training or experience or after full disclosure of pertinent limitations [4]. Therefore, during the development process, we should ensure that our designs and codes are easy to maintain and upgrade for additional functions, user-friendly, and reliable since they will be used by various students for a long time.

More importantly, we should keep the users' privacy information safety. According to the Ethics 1.1 [4], we should protect others' privacy. Thus, we clearly declare that our i-clicker system would never collect other personal information from the users. Our system would only collect users' name for check-in and answering statistic, meanwhile, we would never collect the location information and the contact information. The users' name would be only collected and managed by the teachers and professors, and the central data management system and the communication among the devices would be encrypted to prevent unauthorized information leak.

4.2 Safety

In terms of safety concerns, our final product will contain widgets that may fall off and be swallowed by a child. Therefore, we should include safety-related labels on the device to keep it away from children. Additionally, our devices contain rechargeable Li-ion batteries, which means they can only be used within a specified room temperature range of 0 to 45 degrees Celsius and should be recycled through a formal process. All electronic devices used in the experiment are below the safe voltage for human bodies, but it is still important to follow safety rules to avoid electrocution.

References

[1]   M. Crede, S. G. Roch, and U. M. Kieszczynka, "Class attendance in college a meta-analytic review of the relationship of class attendance with grades and student characteristics," *Review of Educational Research*, vol. 80, no. 2, pp. 272–295, 2010.

[2]   T. O. Kim, J. S. Park, H. J. Chong, K. J. Kim, and B. D. Choi, "Performance analysis of ieee 802.15.4 non-beacon mode with the unslotted csma/ca," *IEEE Communications Letters*, vol. 12, no. 4, pp. 238–240, 2008. DOI: 10.1109/LCOMM.2008.071870.

[3]   H. Takagi, *Queueing analysis. A foundation of performance evaluation. Volume 1: Vacation and priority systems, part 1*, English. Amsterdam etc.: North-Holland, 1991, ISBN: 0-444-88910-8.

[4]   IEEE. ""IEEE Code of Ethics"." (2016), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 02/08/2020).