Radio Frequency Identification Device Inventory Control System

Design Document Version 2

March 20, 2014

Brock Burdyl brockburdyl2014@u.northwestern.edu

Srinivas Balusu srinivas7642@gmail.com

Wenyi Zou wenyizou2013@u.northwestern.edu

Content

Executive Summary

Mission Statement

The mission of the Radio Frequency Identification Device Inventory Control System (RICe) is to provide a business’ patrons the most convenient, quick, and simple method of checkout as possible.

Introduction

RICe will allow a business or other entity to provide automatic checkout to users with normal vision. RICe improves upon current barcode systems by allowing users to checkout items themselves without having to inconvenience users by removing items from a carrying source. This is achieved by tagging each item with a RFID Tag which will be read by a RFID receiver at a designated self-checkout station. Once a user has checked out, an item will not set off an alarm as the user leaves the building.

Definitions

Staff Member/Worker - A user who has the ability to supervise patrons, typically hired by a business. May or may not have access to create additional staff members or workers in application and/or database.

Supervisor - A user who has the ability to add and subtract staff member and worker users from the application and/or database. A supervisor can track the hours worked by a staff member or worker user.

Master - A user who has full access to all parts of the software and hardware of our system.

Patron - A user who can only checkout and return books unassisted. In non-normal use cases that do not resolve in normal operation, a patron will be assisted by a staff member or higher.

Radio Frequency IDentification Tag(RFID tag)- A passive ISO 15693 compliant device capable of reception at 13.56MHz. Texas Instrument product number RI-I03-114A-S1.

RFID chip- An active ISO 15693 reader and writer which can read and write RFID Tags at a maximum range of 80 centimeters with external antenna. Texas Instrument product number RTF7960A.

Microcontroller - Microcontroller controls the RFID transmission and reception control the RICe system. Texas Instrument product number MSP430G2553 control the RICe system.

Item- an object the business is concerned about tracking in the inventory control system

Design Concept

Physical Layout

The system will consist of an entrance, exit, checkout station, and data base. The figure below depicts a possible layout of the RFID system in practice.

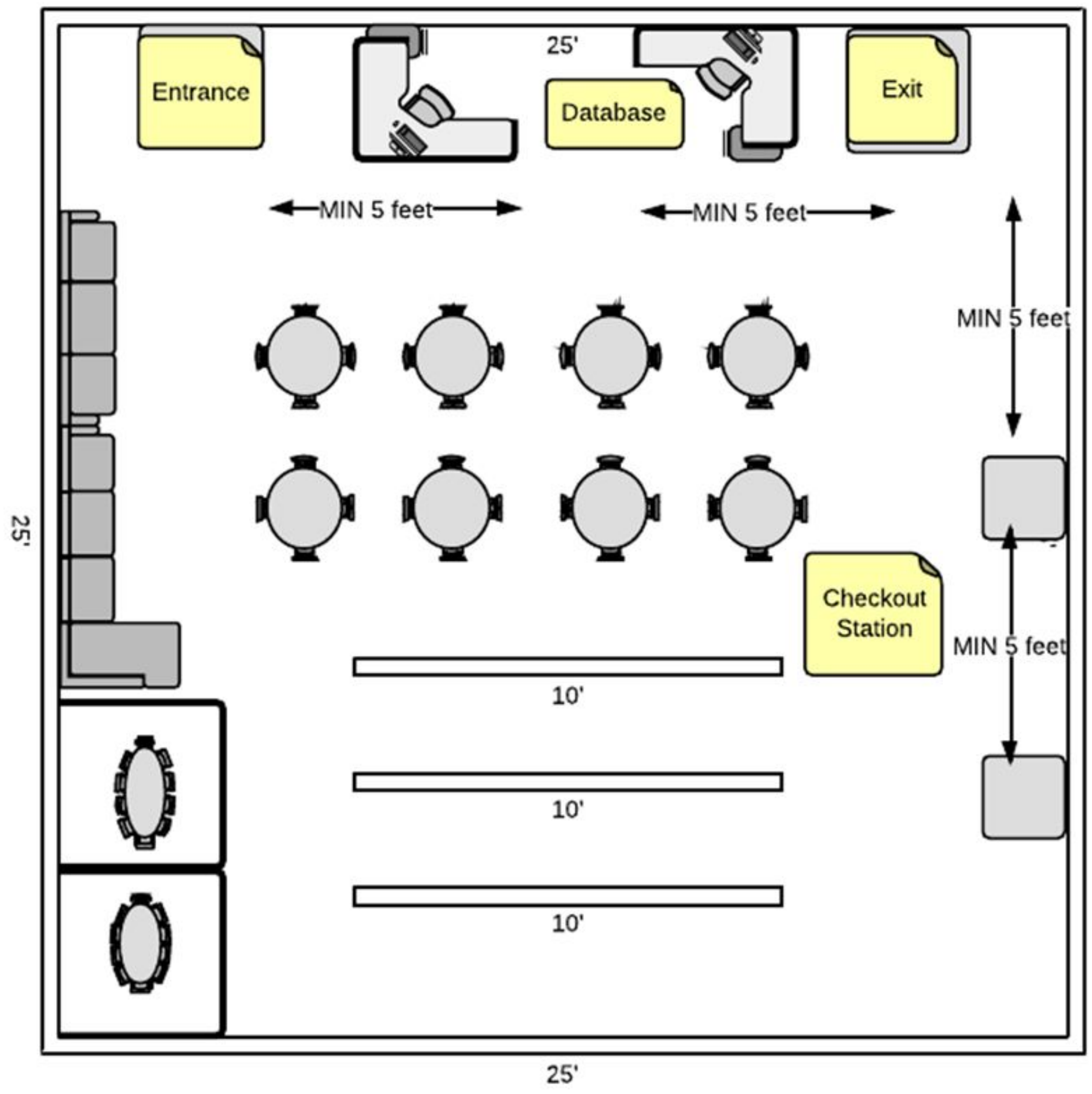


Figure 1: physical layout

Hardware Design

Antenna Design

The antenna design is a modified Full Wave Antenna using multiple loops. We use multiple turns in order to make our antenna a reasonable size and then calculate then treat the antenna essentially as an inductor for the rest of our circuit design process. The Joe Carr formula is then used to approximate the inductance of the antenna from which we can calculate the necessary capacitance for a target resonant frequency.

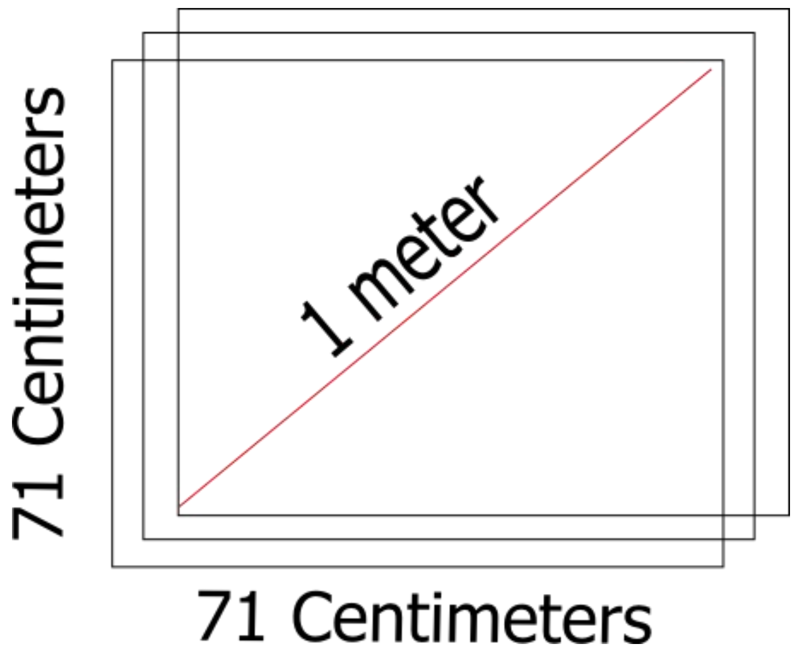


Figure 2: antenna dimensions

$$W_0 = 2\Pi 13.56 \ \ MHz$$

$$W_0 = 8.519 * 10^7$$

$$C_{distributive(pF))} = 60s \qquad s = side\ length\ in\ meter\,s$$

$$\sqrt{base^2 + height^2} = hyptonenus^2$$

$$\sqrt{\frac{1m}{\sqrt{2}}^2 + \frac{1m}{\sqrt{2}}^2} = 1^2 m = 1m$$

$$C_{distributive(pF)} = 60(\frac{1m}{\sqrt{2}})$$

$$C_{distributive(pF)} = 42.426\,pF$$

$$L(uH) = 0.008N^2s[ln(\frac{1.4142sN}{(N+1)l}) + 0.37942 + \frac{0.3333(N+1)l}{sN}]$$

$$W_0 = \frac{1}{\sqrt{L_{loop}C_{total}}}$$

$$W_0 = \frac{1}{\sqrt{L_{loop}(C_{matching} - C_{distributive})}}$$

N(number of turns)	l(coil length in centimeters)	s(side length in centimeters)	L(uH)	Cmatching (pF)
1	284	71	0.74575	~224
2	568	71	5.09082	69. 4863
2	662	71	6.24673	64.476
3	852	71	16.799	50.6264
4	1136	71	39.9531	45. 8743

Chart 1: antenna parameters

Based on these values, an antenna with 3 turns has a large flexibility of Cmatching. This allows for a wide deviation of calculated results depending on factors outside of the design criteria, such as line selection, impedance matching, average frequency shift caused by an adhesive, and bandwidth selection depending on RFID standard selection. The flexibility of Cmatching indirectly effects the resonant frequency in the event that a medium, such as a book cover, is found to significantly shift the frequency response of passive tags.

This flexibility is important because we need to build a power amplification circuit to increase the power supplied to the antenna in order to have enough power at 0.8 meters to power the passive RFID tags according to ISO 15693 cards.

The minimum unmodulated operating field shall be Hmin and has a value of 150 mA/m (rms).
The maximum unmodulated operating field shall be Hmax and has a value of 5 A/m (rms).

A tag shall operate as intended continuously between Hmin and Hmax.

Originally, our front-end was designed around a range of 10 cm driving a printed wire antenna at 200 mV.

The application note states that The design of an external power amplifier requires

detailed RF knowledge. There are also readily designed and certified high-power HF reader modules on the market.

Therefore a robust antenna and matching circuit must be designed to accommodate in matching the circuit at 50 Ω. The antenna design will include two parallel capacitors. The first capacitor will be a 36 pF capacitor in order to provide a base for a variable-capacitor (trimmer) of 5-60 pF. The trimmer will allow a total range of 36 pF to 96 pF providing an inductive tolerance of 237 uH.

$$W_0 = \frac{1}{\sqrt{L_{loop}(C_{matching} - C_{distributive})}}$$

$$8.52 * 10^7 = \frac{1}{\sqrt{(L_{loop} * 10^-6)[(43 - 42.426) * 10^-12]}}$$

$$L_{loop+} = 240\,uH$$

$$8.52 * 10^7 = \frac{1}{\sqrt{(L_{loop} * 10^-6)[(96 - 42.426) * 10^-12]}}$$

$$L_{loop-} = 2.57\,uH$$

$$L_{tolerance} = 237.43\,uH$$

The significance of this large inductor tolerance allows the antenna to resonate at antenna lengths + or – 2 turns different from the antenna built. This design allows for an antenna to be built wrong, but still usable as an antenna at 13.56MHz.

If the goal is to match the antenna and matching circuit together at 50 ohms, by standard design practice, then the power of our antenna can be increased if needed to operate at 80 cm by increasing the rms current while retaining a resistance of 50 ohms.

Activating the TRF7960A for external power amplification provides a modulated signal at the transmitter and configures the receiver inputs for an external demodulated subcarrier input.

While the external power amplification provides extra power to the antenna in order to power the passive ISO 15693 tags resulting in increased range of detection, it causes the receiver to receive voltages which can damage the device. Therefore the devices needs an attenuation circuit for the receiver inputs. A simple resistor network will effect the Q factor. An alternative is to use a voltage buffer to prevent distortion from overload. The current design is a generalized FET voltage buffer created by the common-drain configuration to attenuate and tune the received signals in order to protect the TRF7960A (front-end). As a result, investment in device components with a voltage tolerance of 200V may be beneficial based on the results of Kirschenbaum and Wooly who saw voltage swings of 180V with an antenna of half the size as our design plans to build.

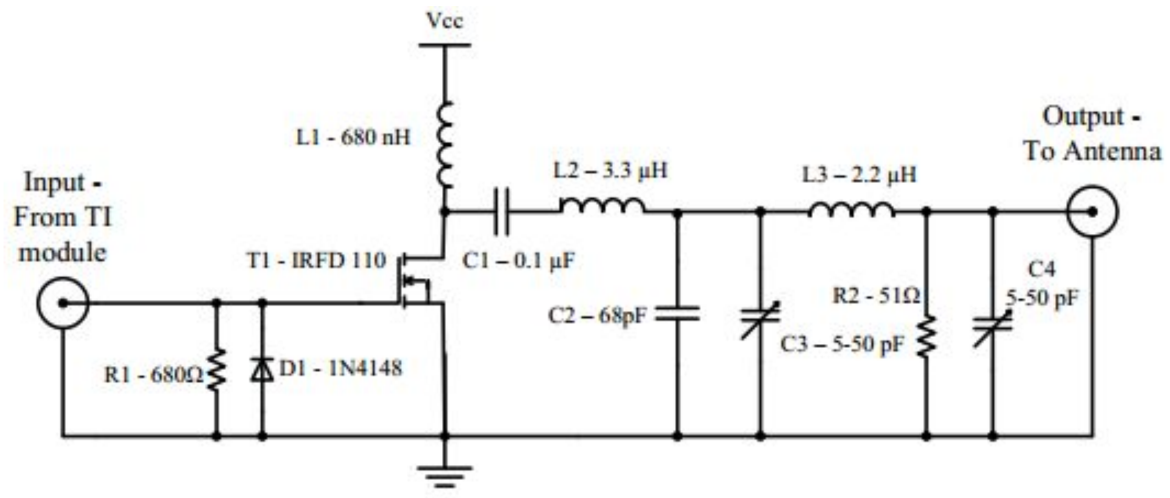


Figure 3: power amplifier

RFID transceiver circuit

RFID system circuit physically accomplish the task of information transmission from microcontroller to RFID chip and antenna. The whole circuit can divided into three parts.

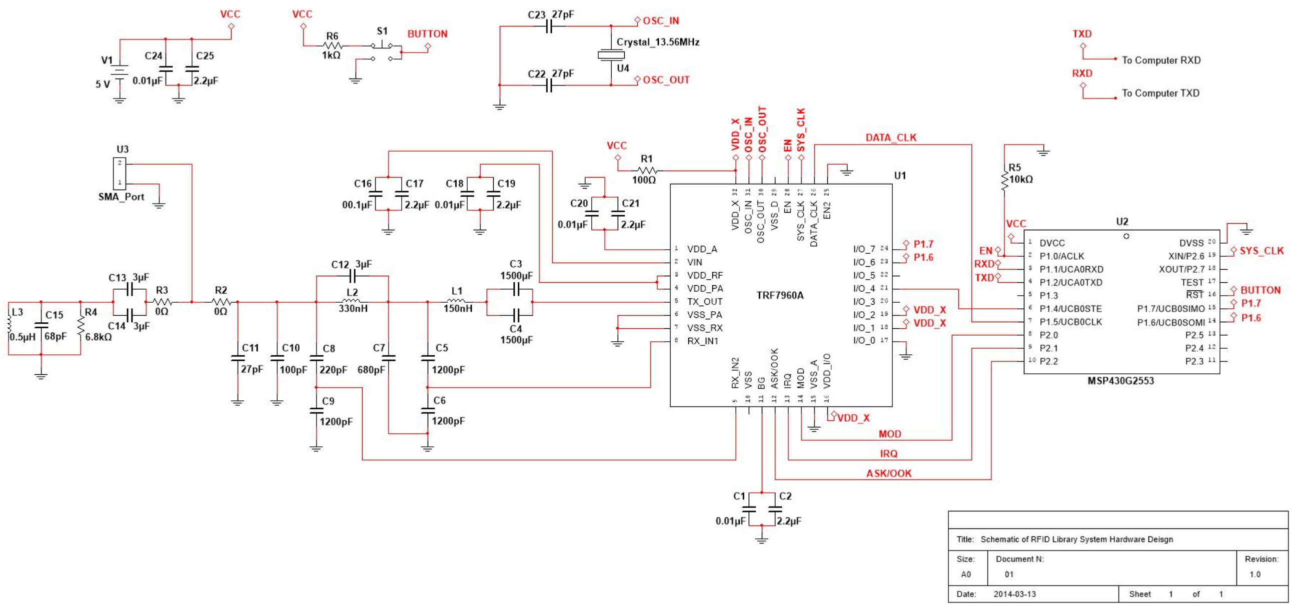


Figure 4: RFID transceiver circuit(Big figure in Appendix C)

First part is the connection between MSP430G2553 with TRF7960A(The right part of figure 4). The pin connection between this two chip and corresponding functions are listed below.

- DVCC (Pin1) connect to the VCC (5V power supply after filter circuit).
- P1.0 (Pin 2) connect to the EN on TRF7960A
- P1.1 (Pin 3) connect to the UART TXD on computer serial port.
- P1.2 (Pin 4) connect to the UART RXD on computer serial port.
- P1.3 (Pin 5) reserved.
- P1.4 (Pin 6) connect to the I/O_4 on TRF7960A. Used for SPI enable control pin.
- P1.5 (Pin 7) connect to the DATA_CLK on TRF7960A. Used for SPI clock signal pin.
- P2.0 (Pin 8) connect to the MOD on TRF7960A.
- P2.1 (Pin 9) connect to the IRQ on TRF7960A.
- P2.2 (Pin 10) connect to the ASK/OOK on TRF7960A.
- P2.3 - P2.5 (Pin 11 to Pin 13) reserved.
- P1.6 (Pin 14) connect to the I/O_6 on TRF7960A. Used for SPI SOMI in MSP430G2553 and SIMO on TRF7960A.
- P1.7 (Pin 15) connect to the I/O_7 on TRF7960A. Used for SPI SIMO in MSP430G2553 and SOMI on TRF7960A.
- RST (Pin 16) connect to the button for reset manipulation.
- TEST and P2.7 (Pin 17 and Pin 18) reserved.
- P2.6 (Pin 19) connect to the SYS_CLK on TRF7960A.

Second part shown in figure 5 is the filtering and matching circuit between RFID chip to antenna. This part of circuit realize two function. First function is filtering and picking of 13.56Mhz RF signal. The other is matching impedance to 50 ohms.

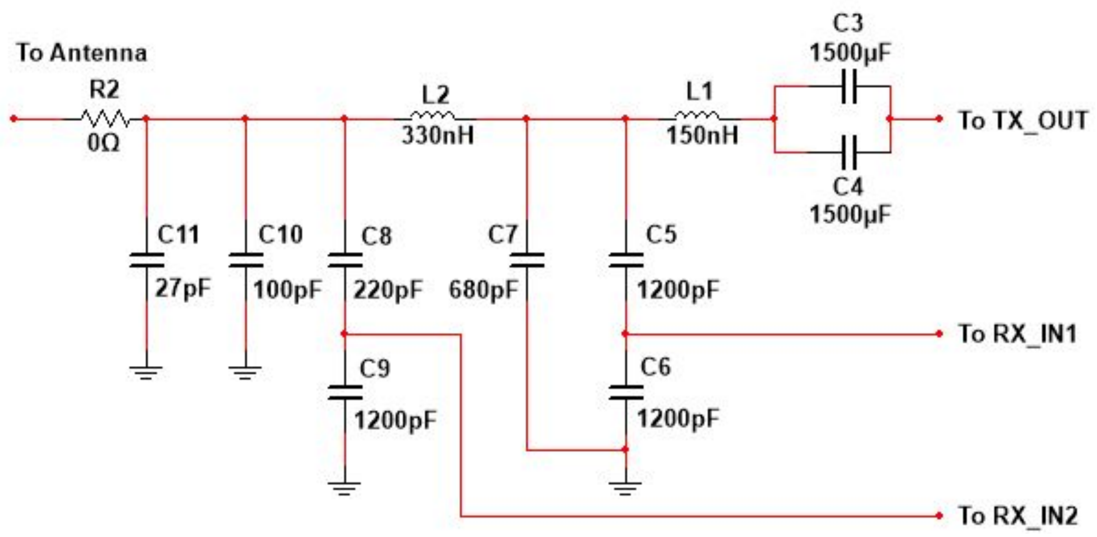


Figure 5: filtering and matching circuit

The third part shown in figure 6 is antenna part. The circuit on the bottom are sample of small distance antenna model. In our design, we use the antenna designed in Antenna Design chapter and hook up to the filtering and matching circuit.

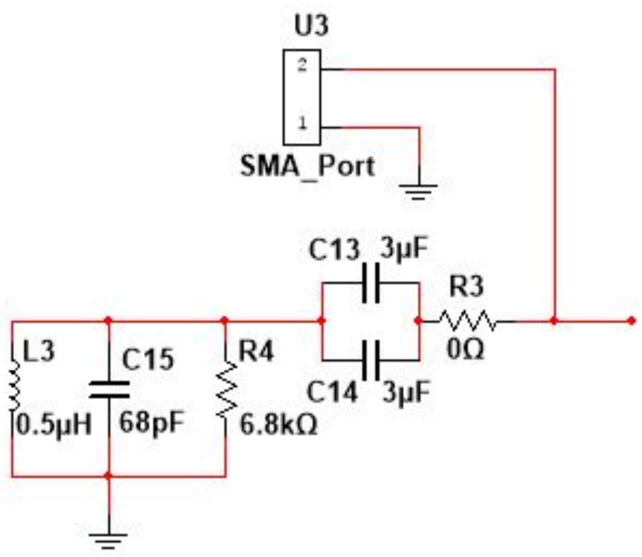


Figure 6: antenna model and connection port

Software Design

System Control and Communication

The control and communication of RICe System consists of two part: the microcontroller to the RFID chip and RFID tag, and the microcontroller to the computer and database.

The communication between MSP430G2553 and TRF7960A is by SPI protocol. The information transmit between them are command and data. The microcontroller send command to the RFID chip to be executed. If the command is for reading or writing data of RFID tag, then 16 bytes data will be transmitted between microcontroller and RFID chip each time after reading or writing command executed.

During this communication, the microcontroller is the master and the RFID chip the slave. The speed of SPI can be user defined.

The communication between MSP430G2553 and computer is by UART protocol. The computer sends a command to the microcontroller to be executed. If the command is for reading or writing data of RFID tag, then 16 bytes data will be transmitted between microcontroller and computer each time after reading or writing command executed.

During this communication, the microcontroller behaves as a slave, and the computer is the master. The speed of UART can be user defined. 9600 Baudrate is recommended.

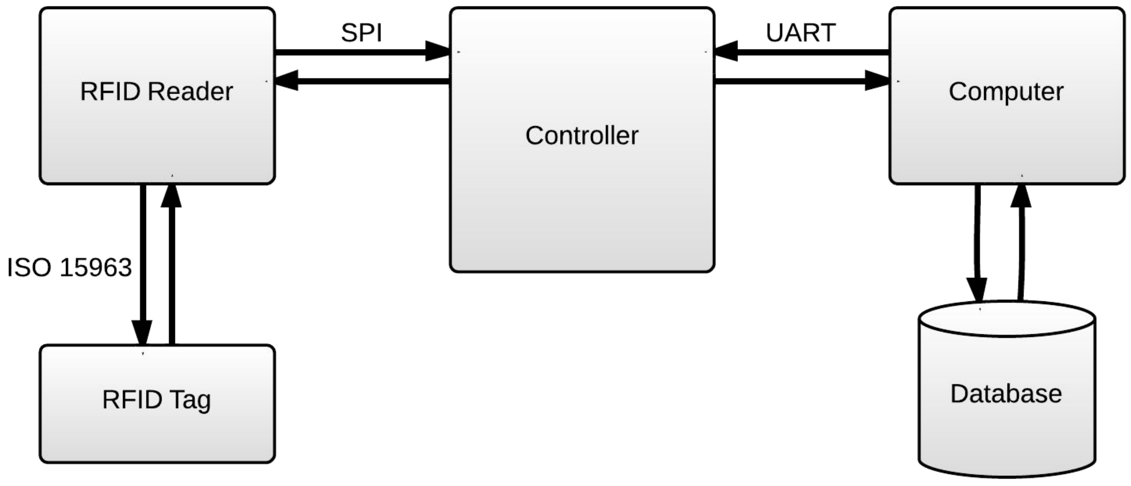


Figure 7: RICe System Control and Communication

Application Design

Our application needs to interface with the microcontroller and the central database and allow users to easily select which books to checkout. Here is the flow of the application as a patron goes through the checkout process.

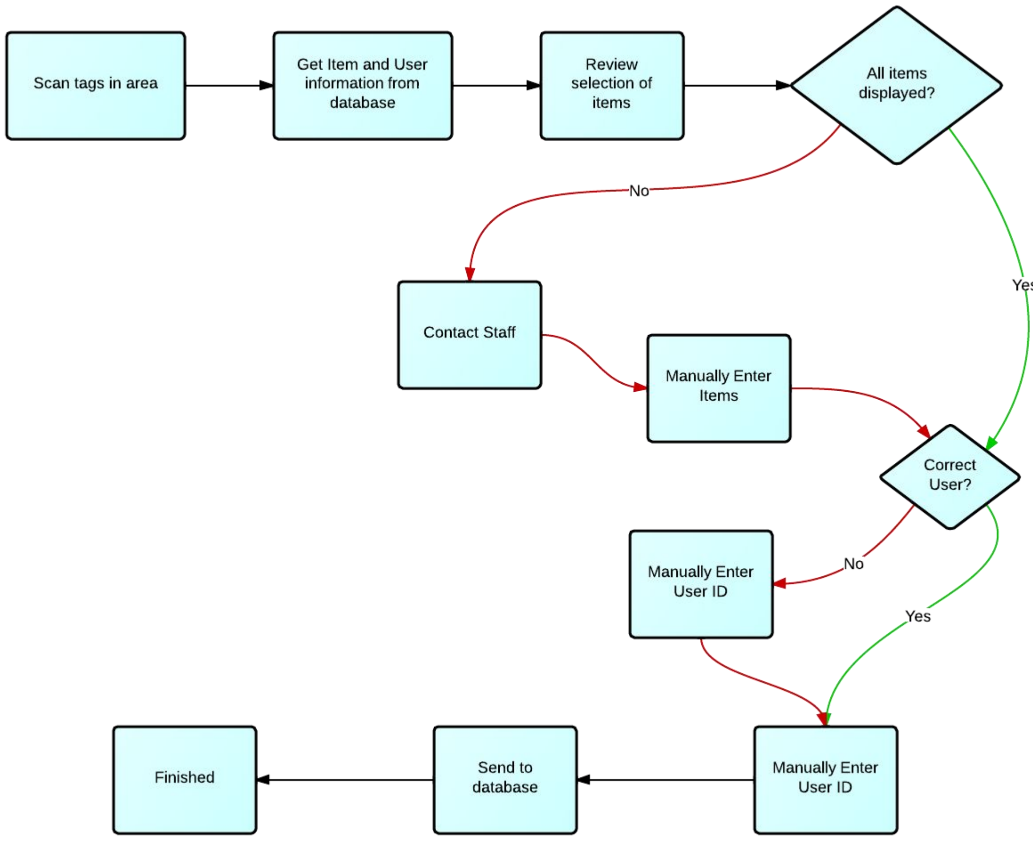


Figure 8: Application flow for a checkout

In addition, the staff should be able to easily add a new item to our system by just setting the item id associated with a tag. The application makes this just as easy as a new user checking out an item:

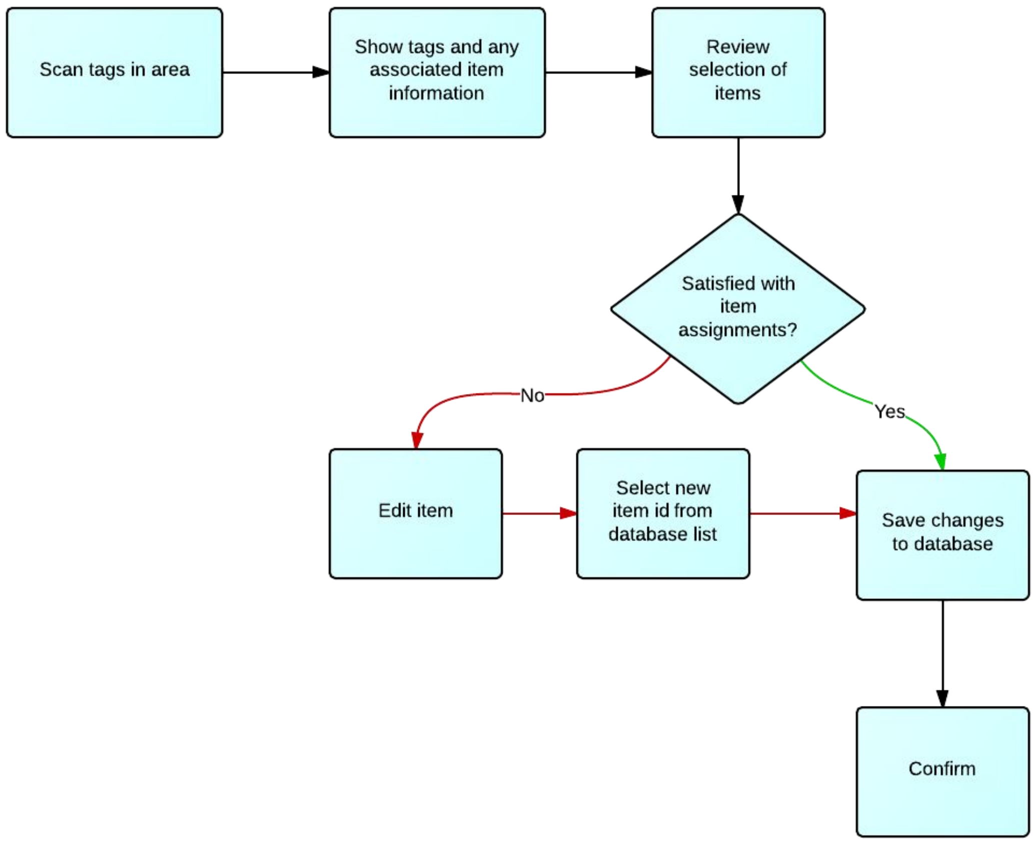

Scan tags in area
Show tags and any associated item information
Review selection of items
Satisfied with item assignments?
No
Yes
Edit item
Select new item id from database list
Save changes to database
Confirm

Figure 9: Application flow for an edit

We want to be able to plug our device into any modern computer without having to worry about the operating system. Java would be a good choice but in the interest of faster development and iteration, we have chosen to go with a Chrome app. Our application follows the common Model-View-Controller (MVC) design paradigm. The models, views, and controllers are listed below.

Models

Item

- Item Id: A string that uniquely identifies an item in stock.
- Item Name: A name for the item that can be displayed when a user is selecting the item.
- Item data: Different items may have more data. In our prototype with a library, an extra field would be the author.

User

- User Id: An id number that uniquely identifies each user and their checkout card.
- User Name: A string of the user’s name.
- User kind: A number that specifies the kind of the user, Ie. staff member, supervisor, master, patron?
- User data: Any other data that is needed by the application. Could be many more fields depending on the existing database structure of the client. For example, data of birth, address, etc.

Settings

- Database location: ip address and port of the database.
- Database user id: Most databases have login authentication.
- Database password: Password login for the database.
- Future settings: There is a lot of potential for customizability in this application so we are handling settings in a way that it scales easily as we add more options.

Controllers

RFID Reader Controller

Handles all communication with the microctronller and the buffer. Is a wrapper for that protocol handling so other parts of the application can use a simple interface.

- readTags gets an array of tags in the area within range of the reader.
- writeTag takes a tag ID, a key, and the data to be written.

Database Controller

Handles all interfacing with the database. Encapsulates the addressing and syntax needed by the database software as specified in the settings and provides the rest of the application with a simple interface.

- getObject takes a table name, field name, and field value to query the database and return any objects that match the criteria.
- updateObject sends an update command to the database and any fields set in the updated object will replace the existing object.

Settings Controller

Handles storage of the settings object, as well as saving , exporting, and loading of the settings so that it new setups can be configured quickly.

- setConfiguration takes a field string and a configuration value and also validates that the value is of a correct format for the specific field.
- getConfiguration is used to retrieve a specific setting as needed by the other parts of the application.
- exportConfiguration saves the setting as a json string to a file.
- loadConfiguration loads the settings from a json file.

Menu Controller

Handles switching between the other views as well as login to allow access to other parts of the application.

- loadView switches to a different view while halting any processes in the current view.
- login prompts the user with a login screen and checks the input with the users database.

Checkout Controller

Uses the reader to scan for tags and displays them in the checkout view. Uses the database controller to check the current status of the items and user. For example, if the books are already checked out, or the user is at their limit, this controller is responsible for checking and displaying the appropriate notification in the view. Then processes input in the view as needed, and finally uses the database controller to complete checkouts.

- checkoutItems takes an array of item IDs and a user ID and sends them to the database to update them as checked out, as well as update the amount of items currently checked out by the user.
- getItems uses the database controller to get the item information from the item IDs in the scanned tags. Now the useful information can be displayed to the user instead of an ID number

Edit Controller

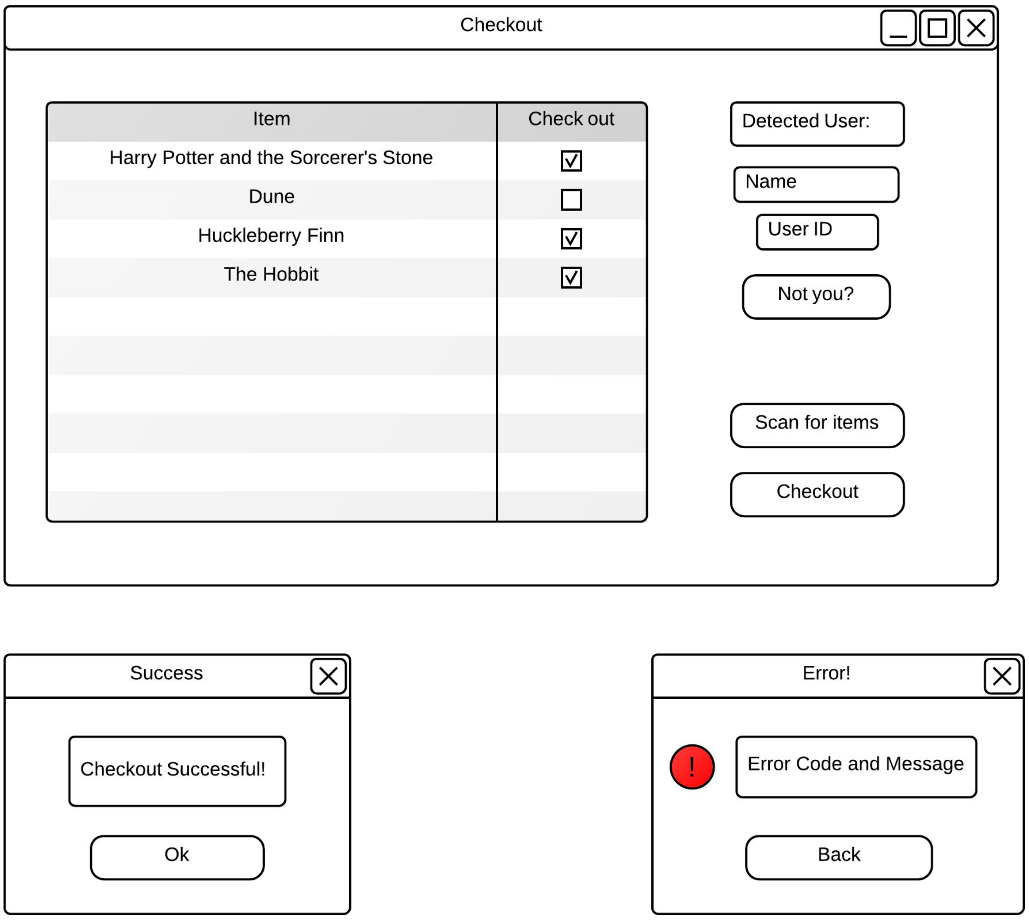
Uses the reader to scan for tags and displays them in the edit view. Uses the database controller to retrieve any item associated with the tag for display by the edit view. Also processes input on the edit view when the associated item is changed.

- editItems takes an array of item IDs and tag IDs that are sent to the database to update the tags database and change what item each tag is paired with

Views

Views are the user interface; what the user actually sees and interacts with. Our goal in the UI design was to make it simple and consistent with similar existing UIs so any user would intuitively know how to navigate and use the application.

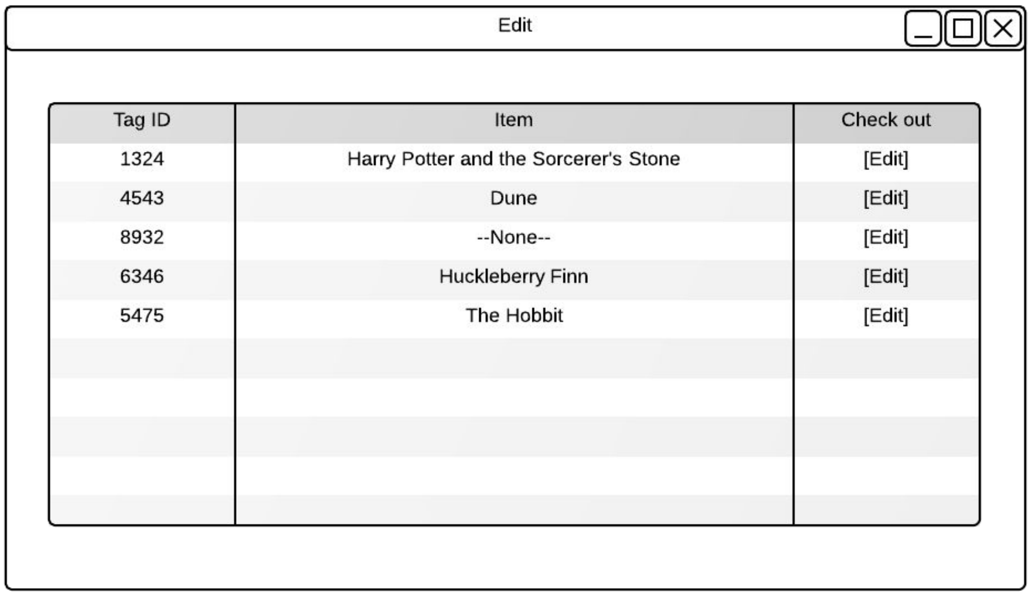
Checkout View



Item	Check out
Harry Potter and the Sorcerer's Stone	☑
Dune	☐
Huckleberry Finn	☑
The Hobbit	☑

Figure 10: checkout view

Edit View



Tag ID	Item	Check out
1324	Harry Potter and the Sorcerer's Stone	[Edit]
4543	Dune	[Edit]
8932	--None--	[Edit]
6346	Huckleberry Finn	[Edit]
5475	The Hobbit	[Edit]

Figure 11: edit view

Menus and login

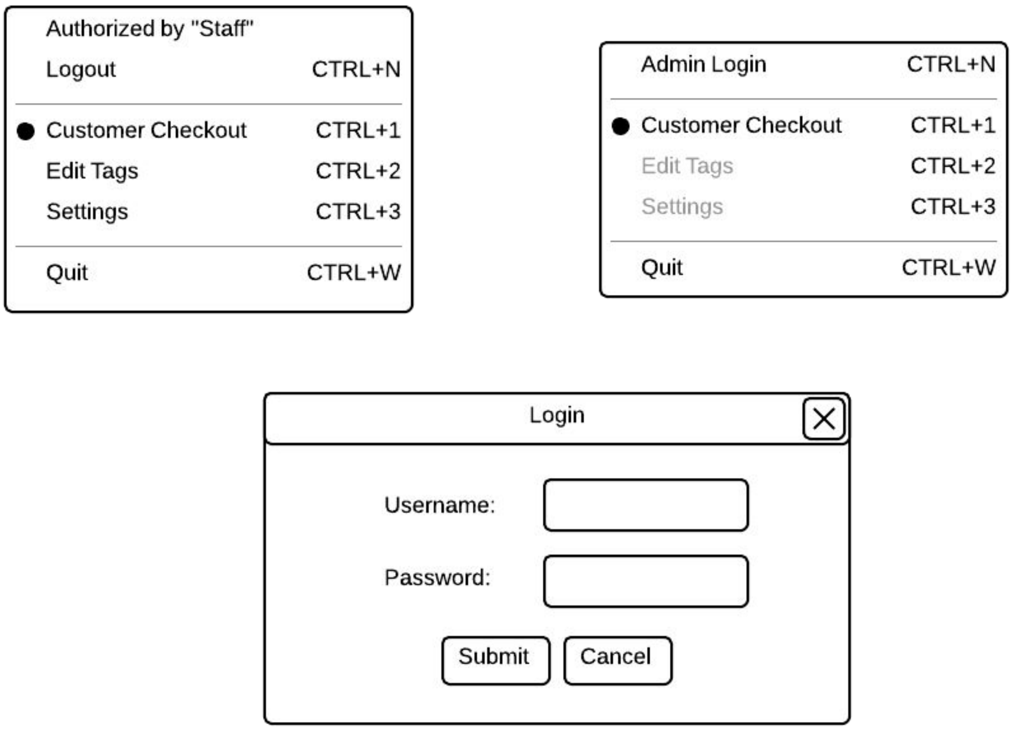


Figure 12: menus and login view

Database

Our application interfaces with the database through a REST API so any database software that supplies the needed commands can work with our application. The required commands are as follows:

- Get(tablename, field, value)
    - Given a string for the table name, a string for the field name, and a string for the field value, the database should return either a single object or an array if there are multiple matches, in JSON format.
- Update(tablename, object, upsert)
    - Given a string for the table name, a json string of the object or an array of objects, the table should update the objects with the same object IDs. The upsert parameter is a Boolean value that, when set to true, tells the database to insert the object(s) if no object with the ID was found.
- Insert(tablename, object)
    - Given a string for the table name, a json string of the object or an array of objects, the table should insert the objects into the table but return an error if and ID already exists
- Delete(tablename, field, value, return)
    - Given a string for the table name, a string for the field name, and a string for the field value, the database should delete all objects that match, but also return them if the return flag is set to true. For consistency, this search should be carried out the same way as the Get query

For our prototype, a sample database and interface will be created in order to show the full system at work.

Implementation Schedule

Task	Timeline
Antenna and matching circuit	Weeks 1-3
Controller and rfid reader communications	Week 1-3
Computer and Controller communications	Week 4
Application development	Weeks 1-4
Database communication	Week 5
Testing and completing design	Weeks 5-7
Additional settings and configuration options	Weeks 8-10

Chart 2: implementation Schedule

Parts List and Budget

All the prices and costs are in USD.

Cost of real product can be slightly different from cost of prototype.

Component	Price	Quantity	Cost	comment
Microcontroller	10	3	30	Use TI product MSP430G2553 Launchpad
RFID chip	0	3	0	Use TI product TRF7960A, free samples
32-pin QVN to DIP board	13.5	3	40.5	Used for TRF7960A pin expansion, shipping fee included
RFID tags	0.608	10	6.08	Use TI product RI-I03-114A-S1
Circuit board	0	3	0	Get from Northwestern University
Electrical components	0	many	0	Get from Northwestern University, include resister, inductor, capacitor, wire etc.
Testing board	49	1	49	Use T I product TRF7960ATB, for software testing
Total Cost : 125.58				

Chart 3: Parts list and Budget

Testing

Before the final prototype accomplished, testing procedure is necessary because it’s vital for a system to be reliable.

The testing procedure can be divided into two parts

Software testing

We use TRF7960A target board to do the software testing. This testing board include RFID chip, matching circuit and small detecting range antenna(3-5cm).

A successful software testing should realize the information transmission from computer to microcontroller, then to the testing board and generate RF signal, finally received by RFID tag. The tag should then transmit information back to the antenna, testing board, microcontroller, and computer. If this occurs, then our test was successful. This will allow us to continue building a dedicated software application which a working prototype to show at a final testing. At this point, we can begin to accurately test our antenna and matching circuit.

Hardware testing

Hardware testing is the second part after the completion of software testing. We use an external matching circuit that includes the power amplifier and external antenna to do the hardware testing.

A successful hardware test should realize the RF signal amplify from RFID chip to external antenna. The RFID tag should receive the RF signal at the maximum range of 80 centimeters

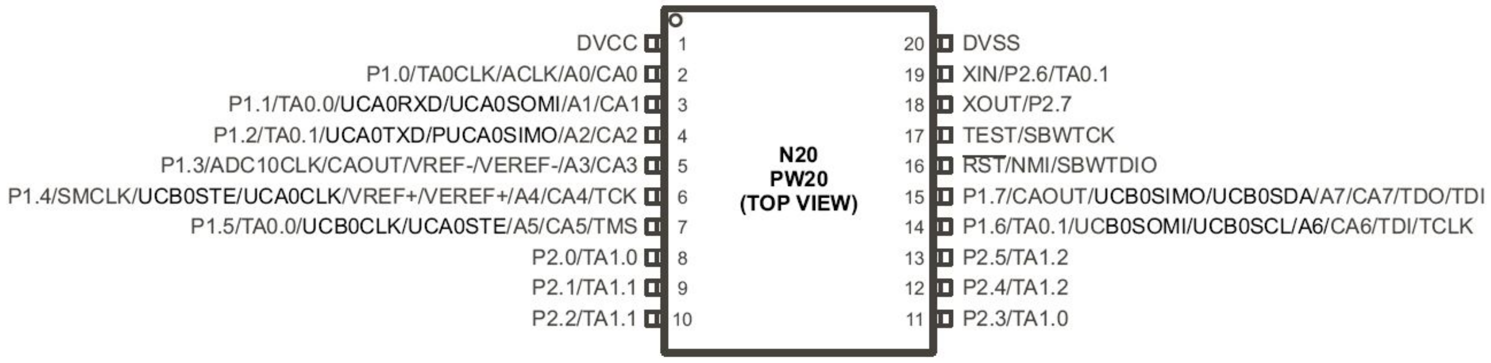
Conclusion

The RICe project will continue into the Spring of 2014 with the construction of a prototype. Construction and testing of the separate parts will continue in parallel as seen in the timeline. We have allowed ample time for fine tuning the prototype and it will meet the specifications outlined in the specifications sheet and realize most aspects of the design laid out here

Appendix

Appendix A

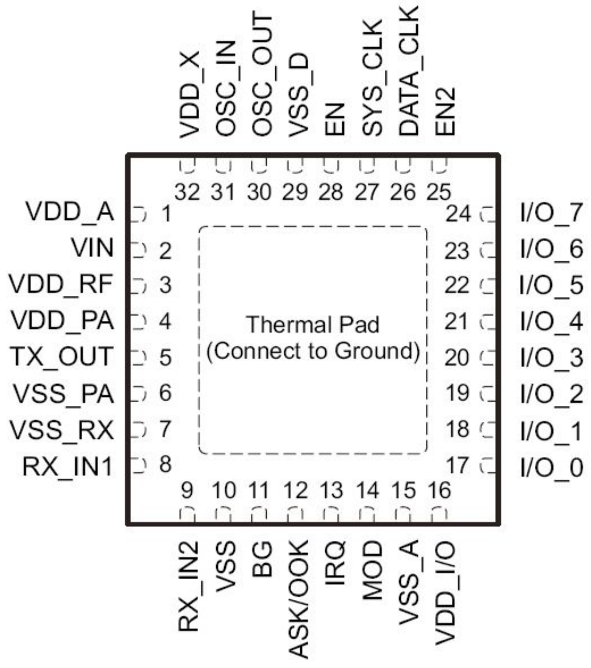
Pin configuration of MSP430G2553



DVCC 1
P1.0/TA0CLK/ACLK/A0/CA0 2
P1.1/TA0.0/UCA0RXD/UCA0SOMI/A1/CA1 3
P1.2/TA0.1/UCA0TXD/PUCA0SIMO/A2/CA2 4
P1.3/ADC10CLK/CAOUT/VREF-/VEREF-/A3/CA3 5
P1.4/SMCLK/UCB0STE/UCA0CLK/VREF+/VEREF+/A4/CA4/TCK 6
P1.5/TA0.0/UCB0CLK/UCA0STE/A5/CA5/TMS 7
P2.0/TA1.0 8
P2.1/TA1.1 9
P2.2/TA1.1 10
N20
PW20
(TOP VIEW)
20 DVSS
19 XIN/P2.6/TA0.1
18 XOUT/P2.7
17 TEST/SBWTCK
16 RST/NMI/SBWTDIO
15 P1.7/CAOUT/UCB0SIMO/UCB0SDA/A7/CA7/TDO/TDI
14 P1.6/TA0.1/UCB0SOMI/UCB0SCL/A6/CA6/TDI/TCLK
13 P2.5/TA1.2
12 P2.4/TA1.2
11 P2.3/TA1.0

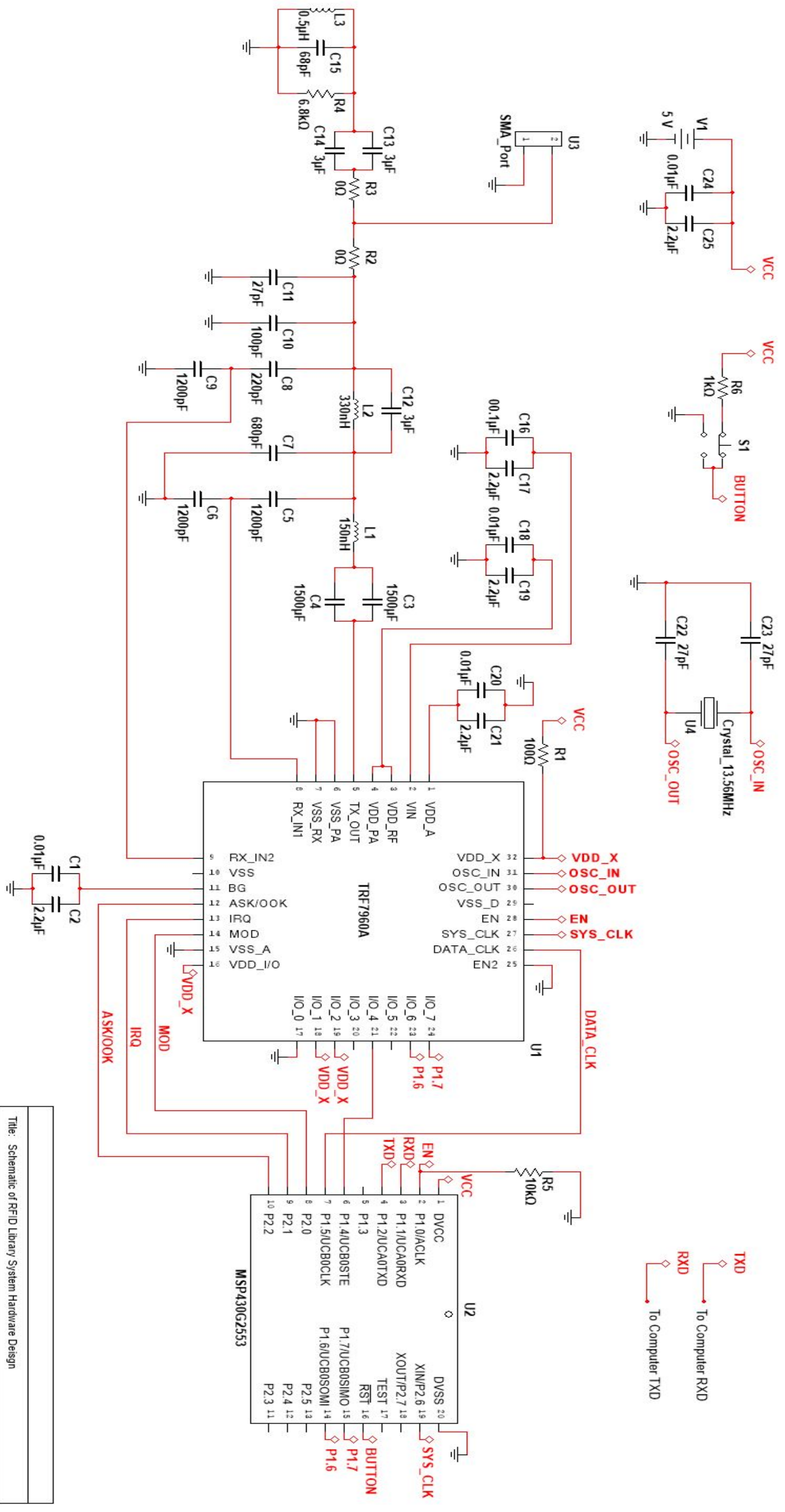
Appendix B

Pin configuration of TRF7960A



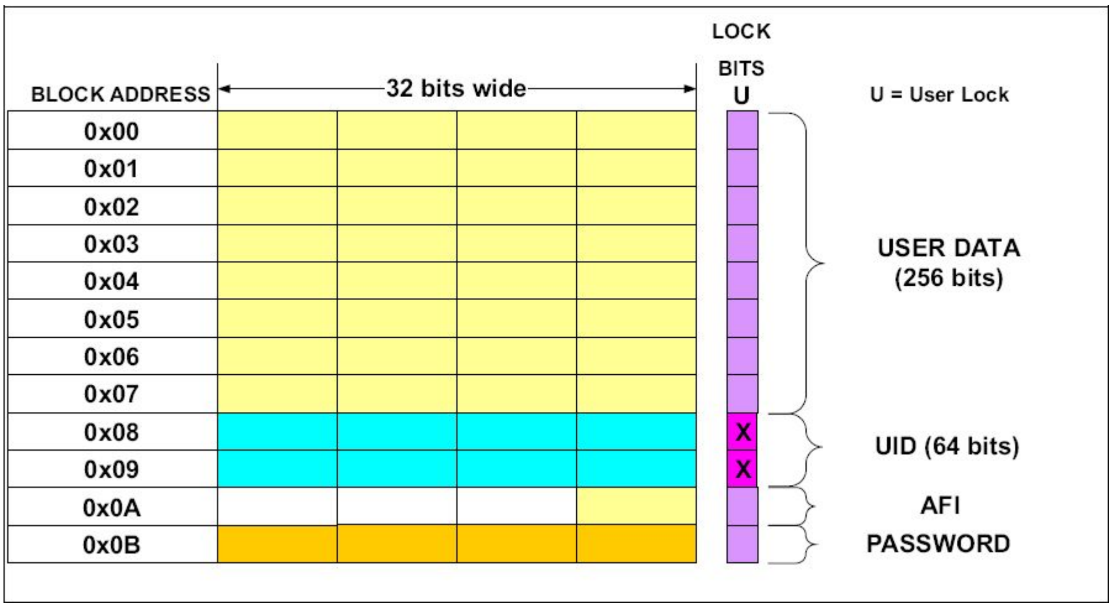
Appendix C

Big figure of RFID transceiver circuit.



Appendix D

RFID tag memory map


LOCK
BITS
U
BLOCK ADDRESS
32 bits wide
U = User Lock
0x00
0x01
0x02
0x03
0x04
0x05
0x06
0x07
0x08
0x09
0x0A
0x0B
X
X
USER DATA (256 bits)
UID (64 bits)
AFI
PASSWORD