

Project Final Report: Simple RFID Spoofer

David Marchevsky

Project Abstract

I have implemented an RFID spoofer system using the ZNEO development kit, a Sparkfun ID-12 RFID reader module, and a handmade 125khz antenna. The system will read the data and unique identifier from a RFID card, decode and store the data in memory, and allow the user to spoof the card by transmitting an equivalent signal (with the card's unique hardware ID) using the antenna. The antenna transmission component of the project will be similar to <http://www.instructables.com/id/Stupid-Simple-Arduino-LF-RFID-Tag-Spoofers/>, which is an RFID spoofer antenna without a reader component. Only EM4000 and EM4001 specification spoofing is supported, but similar concepts can be applied to spoofing HID-based cards. Reading and spoofing cards using more recent specifications will likely require a different reader and antenna setup, as well as different encoding, decoding, and transmission functions.

Status

My project's status is about 80% where I would have liked it to be. In short: the reader module works perfectly, the antenna spoofer module doesn't work very well. The reader module reads the ID from my sample Sparkfun card accurately and without fail; the only thing lacking from the reader module is read range, but this is not surprising considering the cost of the ID-12 and the small internal antenna it possesses. The spoofer module is unable to recreate a signal that the reader module can read as a valid EM4000 transmission. I don't have a separate RFID reader to confirm this, but it is unlikely that it is correctly transmitting. When the reader module and the spoofer module are placed onto separate ZNEO boards, the reader module will occasionally pick up some data from the spoofer, but the antenna must be touching or very close to touching the ID-12, and the data received always fails checksum tests and is of improper length to be a valid EM4000 id. Regardless of this failure, the ID-12 will occasionally pick up a few stray bits from the handmade antenna, which is surprising considering my lack of experience in antenna design.

I was planning on purchasing a premade 125khz antenna, but all the websites that sold the antennas I needed required bulk purchases of about 1000 units, which would have been an unreasonable purchase as I only needed one antenna. Creating my own antenna certainly added to the amount of error, and although winding a 125khz antenna isn't considered to be hard by any means by those with experience in such matters, it was a challenge for me. The similar project which I was using as a baseline was only able to spoof a signal over several centimeters with a handmade antenna, so this project would have been mostly a proof-of-concept even if I had gotten the antenna to work.

Specification & Construction

I used the following hardware: Zilog ZNEO contest kit, Sparkfun ID-12 RFID reader, Sparkfun RFID card, 30 gauge radio shack coated antenna wire, a NPN transistor

(model 2N3904), a 10k Ohm resistor, a 10 nF capacitor from Radio Shack, and a toilet paper roll cardboard inside section. Note that it is not necessary to purchase exactly the same wire, as the antenna calculations can be modified for various thicknesses of wire as well as various diameter toilet paper rolls.

The capabilities of the ZNEO that were useful to me were basic timer functionality and the ability to send pin input to a UART and treat it as serial input, with the decoding done for free.

Hardware Setup 1)Reader module: ID-12 Connections

The connections for the ID-12 component are fairly simple. I found a guide online (<http://www.markfickett.com/stuff/artPage.php?id=373#id12>) featuring a similar project which contains a great deal of information relevant to getting the ID-12 working. Specifically, consider this table listing the functions of the ID-12 pins when ASCII output over serial is desired:

1	Ground	Connect to ground.
2	Reset	Connect this to +5v for normal operation. The unit does not function with it connected to ground.
3	Antenna	Unused
4	Antenna	Unused
5	CP	Unused
6	NC	Unused
7	Format Select	Connect this to ground for ASCII output over serial.
8	D1	RX; Unused.
9	D0	TX; Sends data at 9600 baud.
10	Buzzer	Turns on briefly (and oscillates) when an RFID tag is detected. Connect this to the base of a transistor to control an LED, or to a speaker for a beep.
11	+5V	Connect to +5V.

As the table implies, we only need to connect 6 of the 11 pins: 1,2,7,9,10,11 (10 isn't strictly necessary, but is a good idea to confirm desired operation). All pins should be connected as the table indicates, specifically: pins 2 and 11 should be connected to VCC, pins 1 and 7 should be connected to GND, pin 9 should be connected to PD4 (so we can read the RS232 simply using UART1, and pin 10 should be connected to the rightmost input of the J3 section of the modem so that the TX light will light up when a card is successfully read.

Hardware Setup 2)Antenna/Spoofers Module

Setting up the spoofer is slightly more complicated. A close look at the following links was helpful for me:

<http://hamwaves.com/antennas/inductance.html> (antenna winding calculator)

<http://ww1.microchip.com/downloads/en/AppNotes/00831b.pdf> (Antenna winding theory)

<http://www.corerfid.com/coreDocs/CoreIDSpecs.pdf> (ID-2 specs; the ID-2 is an ID-12 without an internal antenna, page 5 of this sheet describes how to wind an internal antenna, which was a good baseline for me to determine which of my calculations were sane although I obviously did not attempt to wind this antenna). IMPORTANT: If the wire is coated or enameled, this may affect conductivity, so scrape off the coating on the contacts.

Software Algorithm 1: Completed Program

The completed program should function, at a high level, as follows:

- 1) Wait for the ID-12 to report a card read
- 2) Read and decode the card ID sent through UART1 (possibly sending through UART0 for debug purposes)
- 3) Encode the card ID into a valid EM4001 spec signal.
- 4) Send the signal continuously though the antenna, with some constant break between transmissions

This program may be separated into two distinct components, each of which is also a functioning program: the RFID reader (1 and 2) and the RFID transmitter (3 and 4).

Specific implementation details for 1) Wait for the ID-12 to report a card read and 2) Read and decode the card ID sent through UART1:

This part is simple; if the ID-12 is connected correctly as per the above diagram, make sure to set_port() to UART1 and wait for a transmission. Because the transmission is in ASCII and the card's ID is a large number, it is necessary to do some fancy decoding to get an integer representation (note that it won't fit in a native int).

Note that the ID-12 takes care of error correction and, in my experience, consistently recognizes the same code for a given card.

Specific implementation details for 3) Encode the card ID into a valid EM4001 spec signal:

It is necessary to use Manchester encoding (http://en.wikipedia.org/wiki/Manchester_code) to encode the signal. Also note that the antenna is driven high when no signal is being sent, although since there is no signal change.

Testing Procedure:

First, test that the RFID reader component of the lab works properly. It is not necessary to have the spoofer code written at this point. Connect the reader to the board and store the ID that it reads from the card in memory somewhere (preferably not as an ascii string, as we'll have to convert at some point anyways). Useful things to check for are: is the ID read from one card consistent? What distance does the ID-12 pick up reads from a card? From what angles does the ID-12 pick up reads?

The testing procedure for the spoofer is slightly more complicated. Because I didn't have two RFID reader chips or a PC-based one, it was necessary to attempt to read my own spoofed signal with my program. Since my program is either waiting for a read or continuously sending out a spoofed signal (and not both), I used two ZNEO boards to test this phase: one with the reader component sending debug output to a terminal and the second with the writer component continuously spoofing a fixed signal. When both programs are running, try the following: instead of attempting to read the signal from a distance (at least initially), place the antenna right next to the ID-12 (hair's width away, or even touching). If any reads are successful, slowly repeat this test with the antenna moved slightly further away to determine what distances the antenna works at. Useful things to note at this stage are: How often does a read succeed? At what distance do the reads stop succeeding? How does this compare to the results in the previous tests (with the actual RFID card)?

Because the antenna is very unlikely to work the first time, the following methods of debugging/altering the antenna may be helpful: use the volt meter to ensure that current is flowing to the antenna and specifically into the antenna contacts; if touching the spoofer and the reader does not result in a read, try adding a turn of the coil to the antenna or removing a turn.

Retrospective

This was a challenging project and I am disappointed that I did not accomplish my goal of spoofing a simple RFID tag. There was a ruckus on the internet a few years ago with regards to how easy EM4000 and EM4001 tags are to spoof, followed by numerous proofs of concept, including the spoofer guide which I unsuccessfully followed. That being said, this project taught me a lot about RFID and the weaknesses of the specs I attempted to exploit; I am confident that with a lot more time, some recalculations, and some more education in antenna construction, I would be able to successfully spoof these simple tags. My respect for the whole hardware design/prototyping cycle has also increased immensely; as programmers we have numerous ways to debug our creations if they break while debugging an antenna and gaining useful information to apply to a new iteration of the hardware is basically impossible for beginners. Every failure resulted in me having to go back to the drawing board for the antenna construction with little new information gleaned from my previous failures.

If I was able to start over with the information I have now, I would attempt to design a better testing procedure for my antenna, as well as read up on basic electronics concepts and techniques and formulas related to antenna construction. It would be useful to obtain a working antenna and attempt to replicate it, but given my previous difficulties in purchasing only one unit this is unlikely to be possible. I would also have purchased more than one RFID card from SparkFun. I was under the impression that at least one of the RFID cards in my wallet would also be EM4000 or EM4001, but it seems that everyone has switched over to HID's protocols, as the basic ones are too insecure (they still kept me out, though!).

The software design for this project was very simple, and as such there weren't any design decisions that could make or break the project. My decision to eschew LED output and button input in favor of a simple debugging interface via serial was due to the fact that the spoofer is intended to be concealed, and therefore during normal use neither of those features would have been useful, and serial io is much more useful for normal debugging purposes. If my device were to be manufactured as actual physical hardware it would necessarily need to be cheap and small, so LEDs and buttons wouldn't be included in the design anyways.