SimpleNFC

Introduction

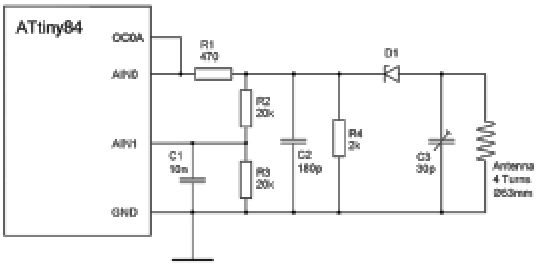
This article describes a simple approach for emulating NFC tags (to be accurate ISO 14443-3 compatible Type 2 tags). Nothing more than 4 resistors, 3 capacitors and 1 diode are required to connect the RFID antenna to a microcontroller.

On applications where range and high transfer rates aren’ t required, NFC can replace WIFI and Bluetooth to exchange data between smartphones and other devices. Compared to other RF interfaces supported by nowadays smartphones, NFC works with 13.56 MHz on a much lower frequency. Combined with the simple hardware layer and the lightweight protocol, it’ s ideal for low power and low cost as well as DIY applications.

The inspiration for that project comes from Micahs "software only" 125 kHz RFID tag. Because of the On-Off Keying for ISO 14443-3 tags my design requires a bit more Hardware, including a power supply (not shown in the folowing schematic).

Hardware

The LC-circuit consisting of C3 and antenna coil is tuned to 13.56 MHz resonance frequency. The received On-Off Keyed signal is rectified by the schottky diode D1. The diode combined with C2 and R4 works as envelope detector to recover the transmitted data. This signal is feed into the positive comparator input AIN0. The comparator reference AIN1 is generated by a voltage divider with an additional low pass filter consisting of R2, R3 and C2.



For sending data from the tag to the reader load modulation is intended. This is realized by internally connecting the counter 0 to its output OC0A. When OC0A is pulled low, more

power is drawn from the LC-circuit compared to the condition when OC0A is high.

Probably the design doesn’ t match the NFC specifications in a number of aspects. But anyway it worked very well during testing. My test device was a Lumia 620 which uses the very common NFC controller PN544 from NXP.

To be able to generate the subcarrier of approximately 847.5 kHz, the clock frequency of the microcontroller must be divisible by the subcarrier with a small or without remainder. The best choice would be a 13.56 MHz crystal as clock source. I used a 13.5925 MHz crystal which worked fine as well. Frequencies below 13 MHz will be critical because of insufficient computing time.

Deviation in the sub percent region seems to be not critical for the subcarrier frequency. For the average transmitting bit-rate even small deviation has to be considered. That can be done in software (the code is already prepared for 13.56 and 13.5925 MHz).

Code

The code is open source and published on GitHub: github.com/Nonannet/simple-nfc.

Further reading

Timo Kasper gives a short introduction on the physical RF-layer. He presents his own RFID emulator design consisting of significant more but only standard components.

A nice and short NFC introduction has been given by Charlie Miller in Exploring the NFC Attack Surface.

All Details on ISO14443-3 tags can be obtained from the ISO/IEC 14443-1, 14443-2 and 14443-3.

The format in which data is stored on the tag is called NDEF. Everything in details can be obtained from the official specification NFC Data Exchange Format (NDEF) from the NFC-Forum.

Niall Quirke wrote in his thesis about SimpleNFC: Passive NFC Environmental Sensor

Hackaday article about this implementation: hackaday.com/2013/...