NEBRASKA

Unlocking the Secrets of ID Cards: Exploring Wireless Magic of Near Field Communication with Software-Defined Radios

Dickson Akuoko Sarpong and Hongzhi Guo

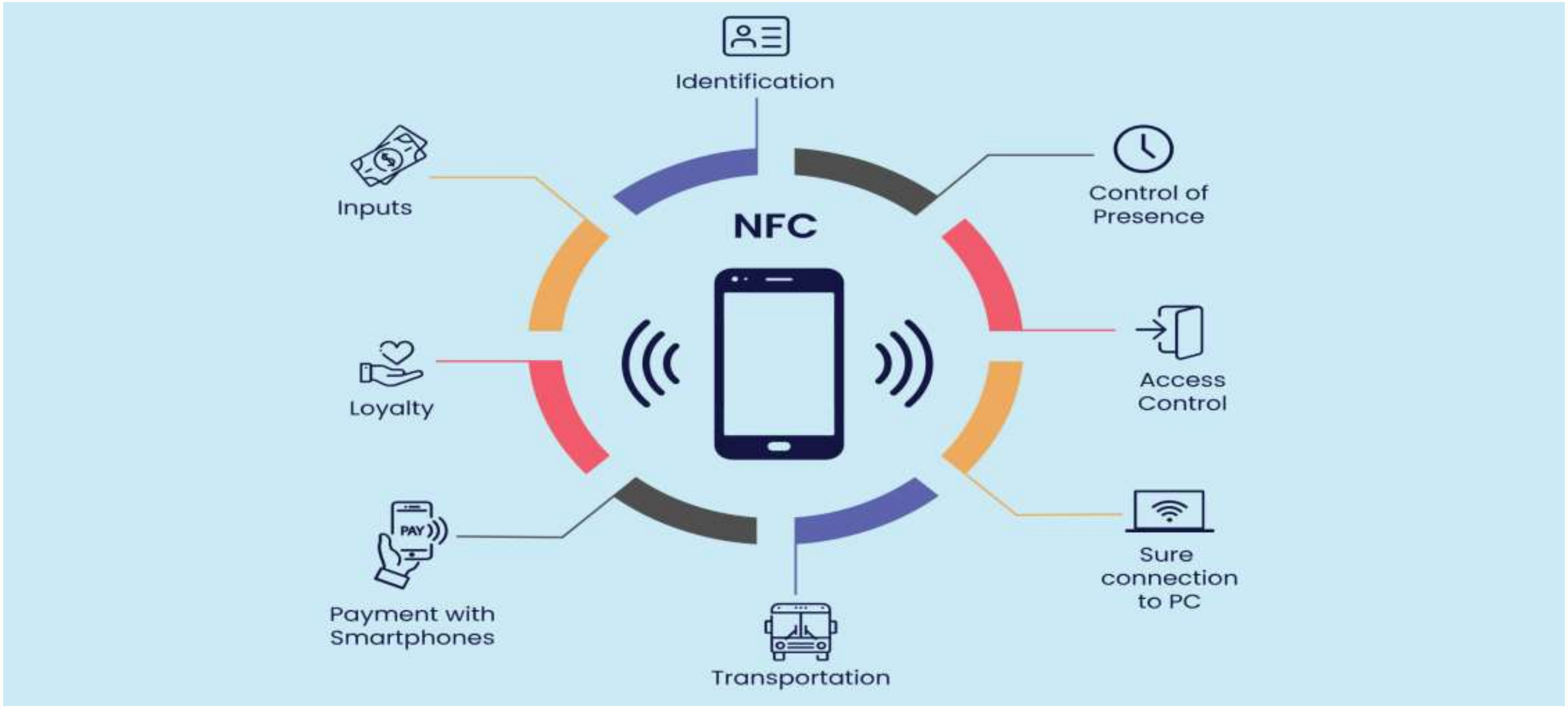
Outline

What is NFC?

VISA
Tap to pay
VISA

EUROPESE UNIE
KONINKRIJK DER NEDERLANDEN
PASPOORT
READID

Applications of NFC



https://www.rfid-smart.com/blog/rfid-journal/nfc-facts-and-applications-that-will-broaden-your-mind.html

NFC Basics

Overview:

- Near Field Communication (NFC) is a short-range wireless communication technology that enables the exchange of data between devices when they are brought close together, typically within a few centimeters operates at a frequency of 13.56 MHz.

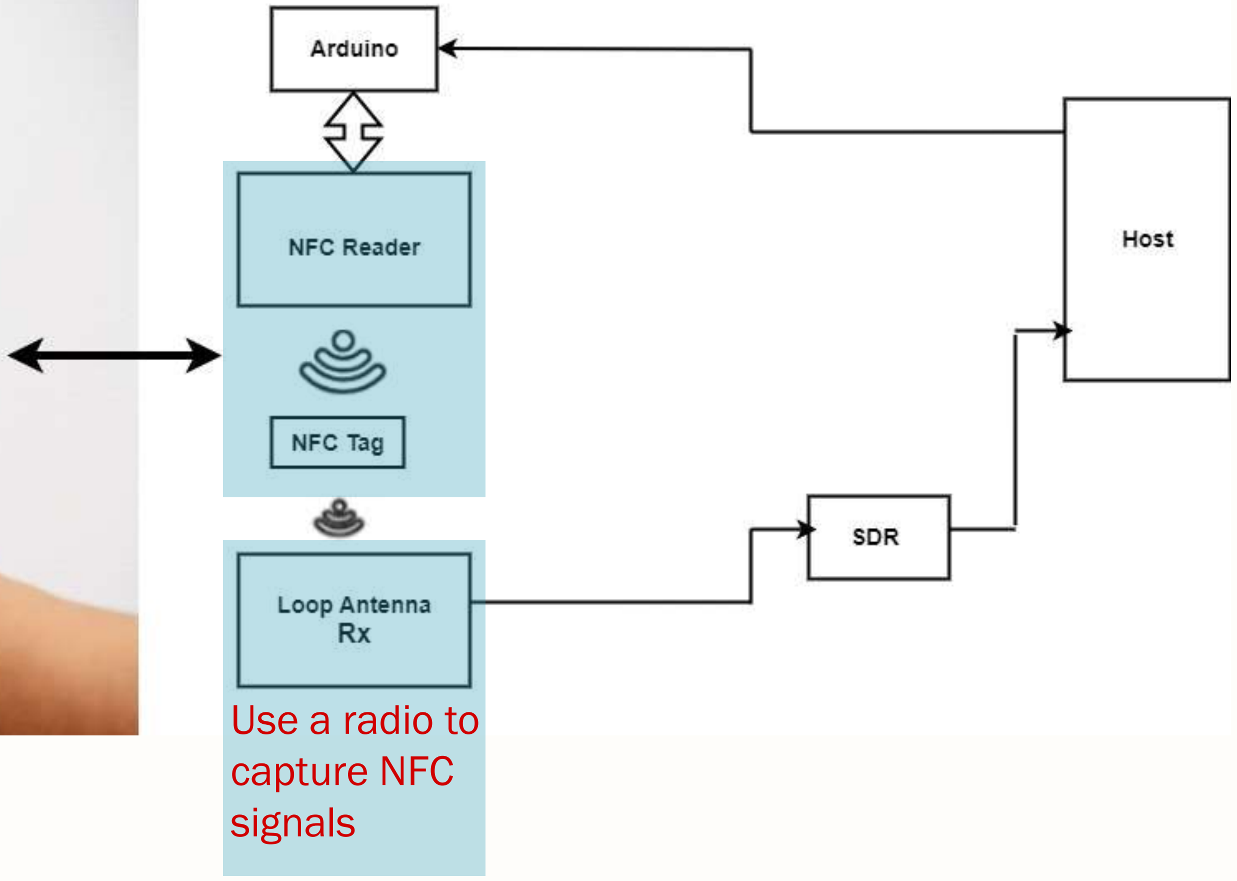
Functionality:

- Enables data exchange over distances of just a few centimeters.
- Utilizes inductive coupling between loop antennas to facilitate secure, contactless transactions.

Advantages:

- Ease of Use: Simple to implement and operate.
- Security Features: Provides a secure method for data transfer, ideal for a wide range of applications.

What are we going to do?

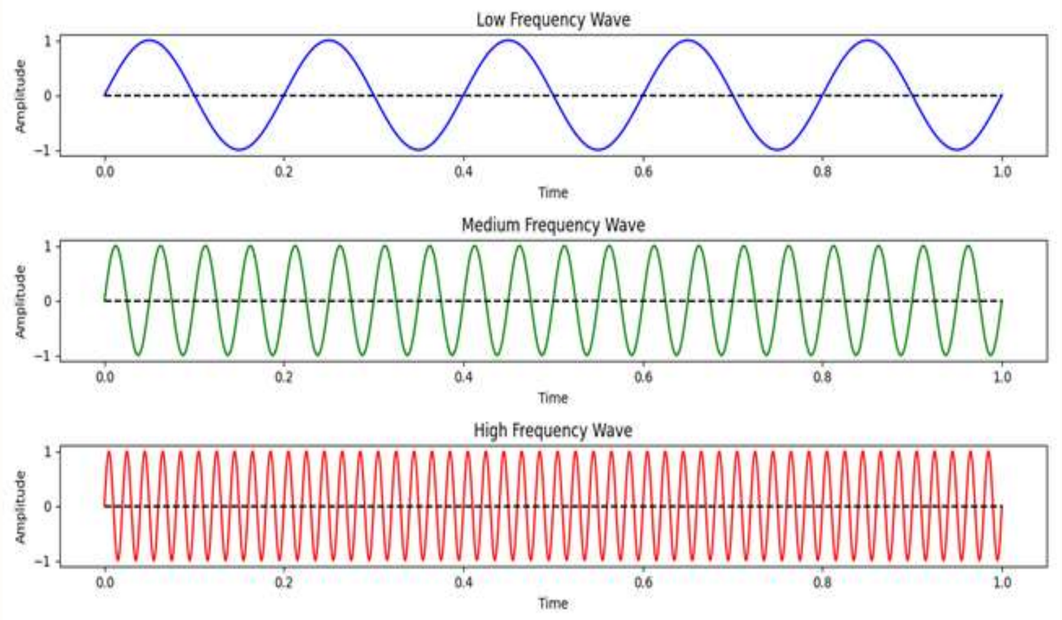


Use a radio to capture NFC signals

Waveform and Properties

- What are waveforms?
  - Visual representations of signals used in telecommunications to convey information.
  - Examples: sound, video, data.
  - Signals are transmitted through mediums like air and cables.
  - Receivers convert the signals back into their original form (e.g., sound, image).

- Types of waveforms:
  - Sine waves: Smooth, periodic oscillations.
  - Square/rectangular waves: Abrupt transitions between voltage levels.
  - Triangular/sawtooth waves: Linear voltage changes with time.



Significance of Waveform Properties

- Understanding waveform properties is crucial for:
  - Signal analysis: Extracting information and meaning from signals.
  - System optimization: Improving efficiency and performance of telecommunication systems.
  - Troubleshooting: Identifying and resolving signal issues.
- Applications:
  - Telecommunications: Ensuring clear transmission of voice, data, and video.
  - Signal processing: Filtering, amplifying, and manipulating signals for various purposes.
  - Data analysis: Understanding patterns and trends in data represented as signals.
  - By analyzing properties like wavelength, amplitude, frequency, and phase, we gain valuable insights into signal behavior and optimize telecommunication systems for better performance.

Equipment and Software

In this session, we will use several essential tools and software:

- Hardware:
  - PN5180 NFC Reader
  - Arduino Uno
  - ISO15693 NFC Tag
  - RTL-SDR Dongle
  - jumper wires
  - a breadboard

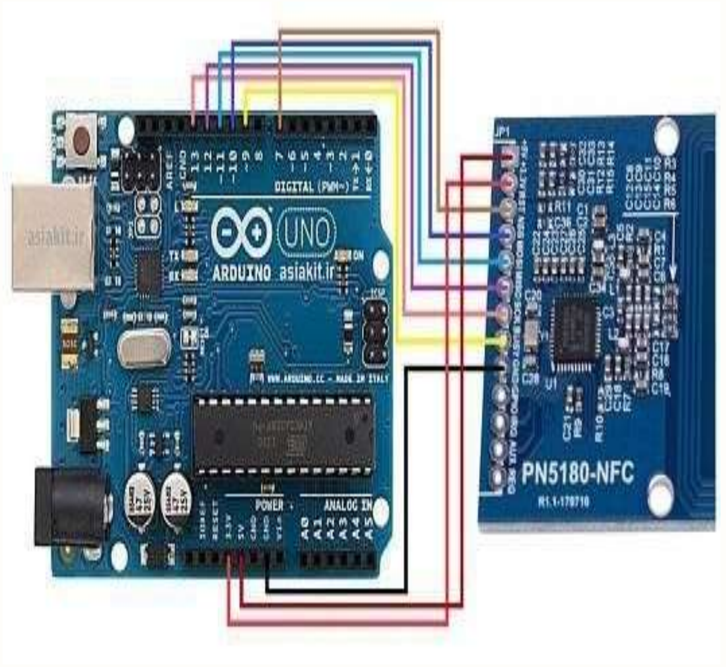
- Software:
  - Arduino IDE for coding
  - PN5180 library
  - SDR Sharp for signal capture
  - Python (Jupyter Lab) for signal processing

Introduction to Software Defined Radio (SDR)

- Overview
  - Transceiver programmable for various signals.
  - Software performs radio signal processing tasks.
- Key Features
  - Versatility and adaptability in radio systems.
  - Functions like a desktop computer with reprogrammable devices.
- Flexibility Through Software:
  - Receives/transmits radio protocols via updates.
  - Efficient spectrum utilization without hardware changes.
- Communication Solutions
  - Addresses diverse needs: data transmission, voice calls, etc.
  - Cost-effective for providers, developers, and users.
- Advantages
  - Unparalleled flexibility, upgradability, and security, enables multifunctionality and adaptability.

Arduino with NFC technology

- Hardware Setup:
  - Connect the PN5180 NFC reader to the Arduino Uno.
  - Ensure the correct wiring setup for power, ground, and data connections.
- Software Installation:
  - Install the necessary libraries in the Arduino IDE.
  - These libraries will provide functions to initialize the NFC reader and detect tags.
- Code Upload:
  - Write or upload the appropriate code into the Arduino IDE.
  - Upload the code to the Arduino, enabling it to scan for ISO15693 tags.
- Operation:
  - The Arduino will read the unique identifiers (UIDs) of detected tags.
  - Display the UID information in the Serial Monitor within the Arduino IDE.
- Outcome:
  - This setup allows for efficient detection of NFC tags and retrieval of their data



The Communication between the Reader and the Tag

- Every transaction is initiated by the reader, and only one participant (either the reader or the transponder) communicates information at any given time.

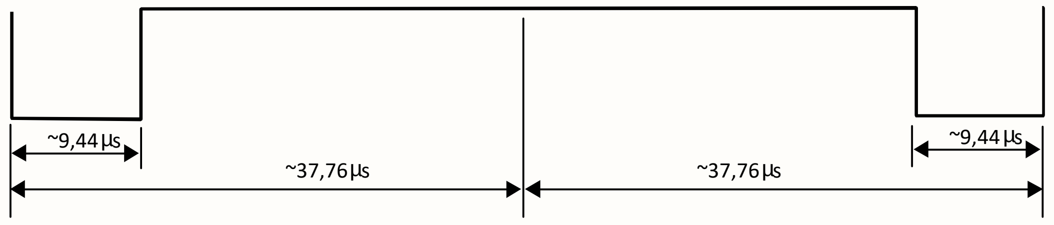
- A data packet starts with a start-of-frame (SOF) signal and ends with an end-of-frame (EOF) signal. A data packet enclosed by an SOF and an EOF is termed a frame.

PN5180 Reader to ISO15396 Tag Communication

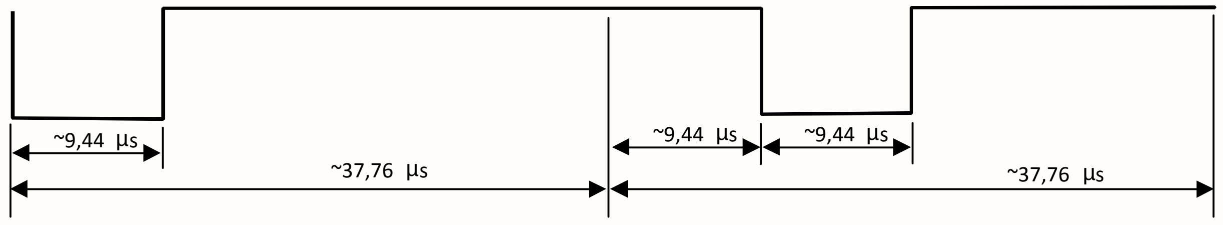
- Transaction Initiation: Reader always initiates communication. Only one participant (reader or transponder) is active at any given time.
- Data Packet Structure: Data packets are enclosed by a Start-of-Frame (SOF) and End-of-Frame (EOF) signal.
- Start-of-Frame (SOF): The SOF pattern utilizes two modulation pauses. The position of the second pause determines whether the frame uses the "1 out of 256" or "1 out of 4" data coding mode.
  - Two Pulse-Position Data Coding Modes:
  - "1 out of 256" mode: 1 byte transmitted in 4.833 milliseconds, resulting in a data rate of 1655 bits per second (bps).
  - The byte's value is conveyed by the position of a modulation pause within the 4.833 milliseconds timeframe.
  - "1 out of 4" mode: 2 bits transmitted within 75.52 microseconds, equivalent to a data rate of 26,484 bps.

PN5180 Reader to ISO15396 Tag Communication

Start of Frame 1 out of 256

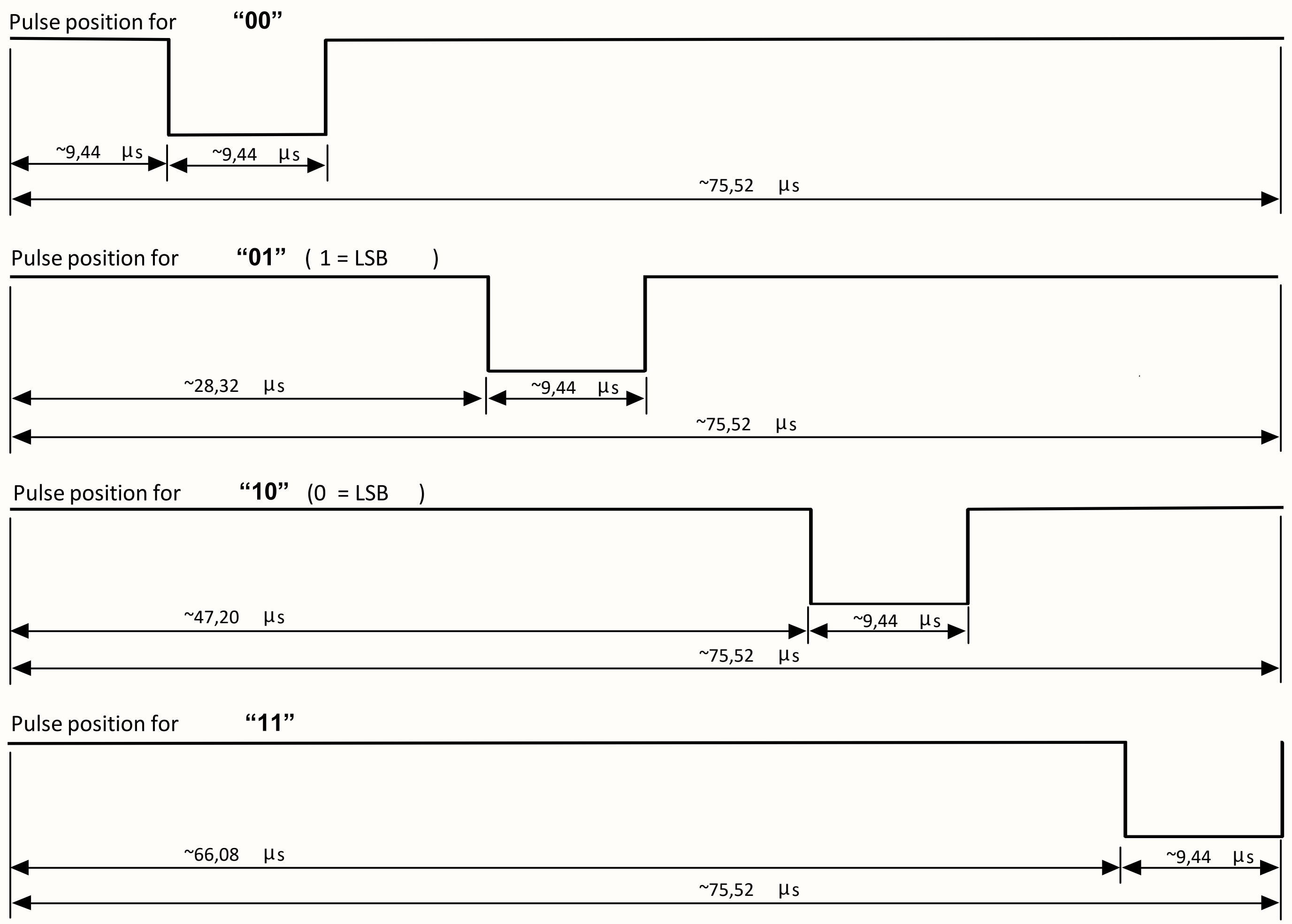


Start of Frame 1 out of 4



PN5180 Reader to ISO15396 Tag Communication

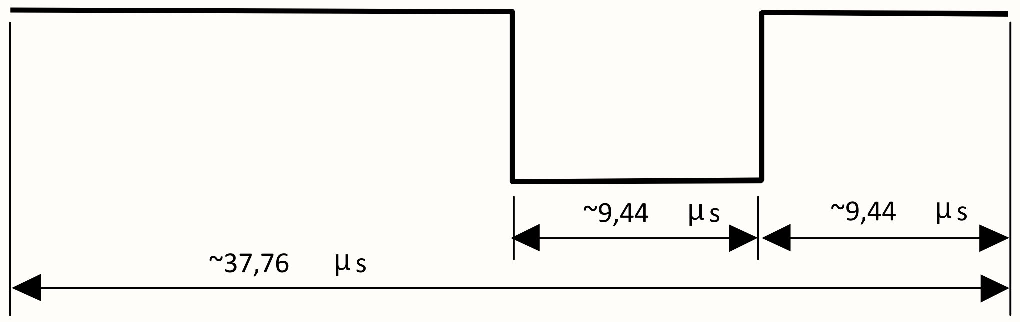
Data Packet in the 1 out of 4 Coding Mode



PN5180 Reader to ISO15396 Tag Communication

End of Frame

- Duration: The EOF pattern transmission lasts for 37.76 microseconds.
- Consistency: The EOF pattern is identical for both "1 out of 256" and "1 out of 4" coding modes.
- Structure: The EOF pattern consists of one modulation pause.



PN5180 ISO15396 Tag to Reader Communication

- Frequency Usage: The transponder communicates with the reader using one or two subcarriers, with frequencies of 423.75 kHz and 484.25 kHz. The choice of using one or two subcarriers is indicated in the request data frame.

- Data Rate: The data rate of the response frame is determined by the data rate flag in the request data frame. A low data rate is around 6600 bps, while a high data rate is approximately 26,500 bps.

PN5180 ISO15396 Tag to Reader Communication

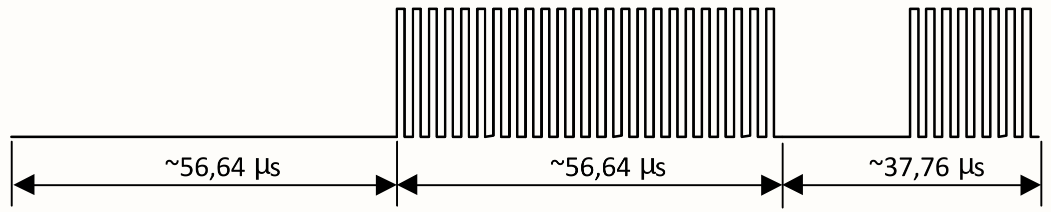
Start of Frame (SOF) Transmission

- Single Subcarrier (High Data Rate):
  - The SOF is sent in roughly 151 microseconds.
  - The transponder doesn't modulate for 768 cycles, then modulates for 16 cycles and repeats this pattern 24 times.
  - This is followed by logic 1.
- Two Subcarriers (High Data Rate):
  - The SOF is transmitted in approximately 149.8 microseconds.
  - Like the single subcarrier case, the transponder doesn't modulate for 14 cycles, then modulates for 16 cycles, repeating this 27 times.
  - This is followed by logic 1

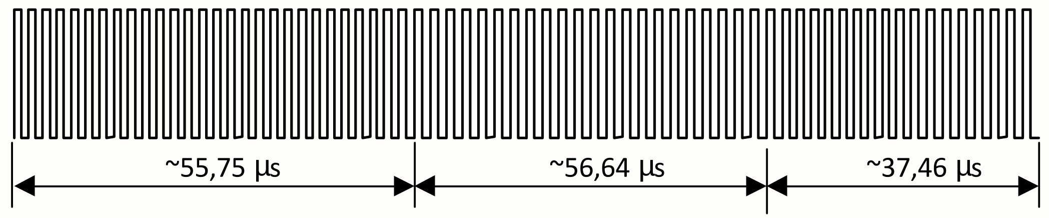
PN5180 ISO15396 Tag to Reader Communication

Start of Frame

Tag to Reader SOF, single subcarrier.



Tag to Reader SOF, two subcarriers.



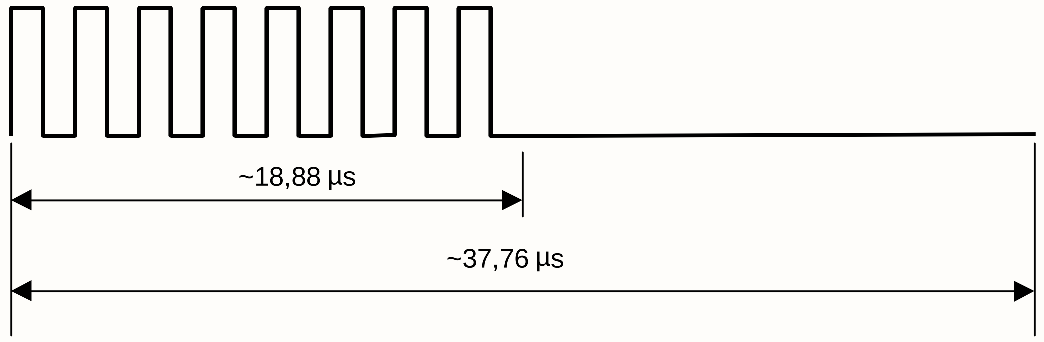
PN5180 ISO15396 Tag to Reader Communication

Data Packet Transmission with Single Subcarrier

It transmits one bit within 37.76 microseconds using a single subcarrier.

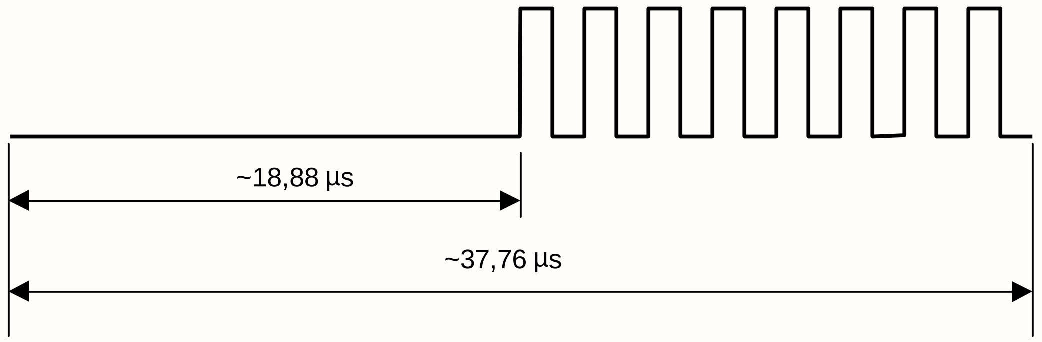
Logic 0 Encoding

- 16 cycles of modulation followed by 256 cycles of no modulation, repeated 8 times.



Logic 1 Encoding

- No modulation for 16 cycles, followed by 16 cycles of modulation, repeated 8 times.



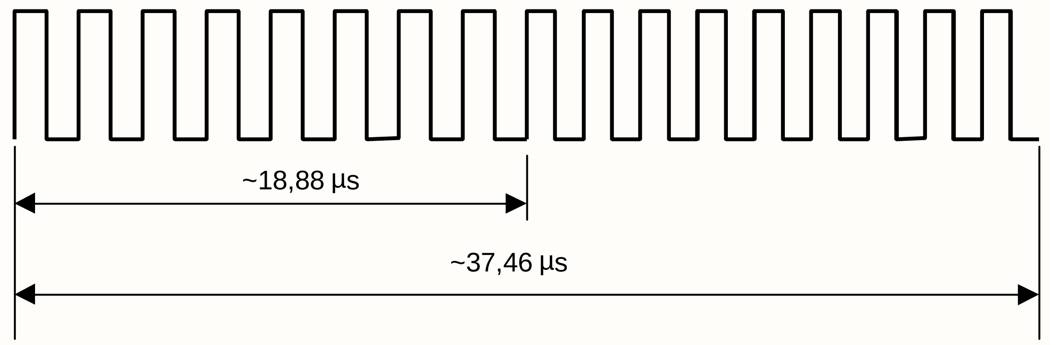
PN5180 ISO15396 Tag to Reader Communication

Data Packet Transmission with Two Subcarriers

- It transmit one bit within 37.76 microseconds using two subcarriers for redundancy.

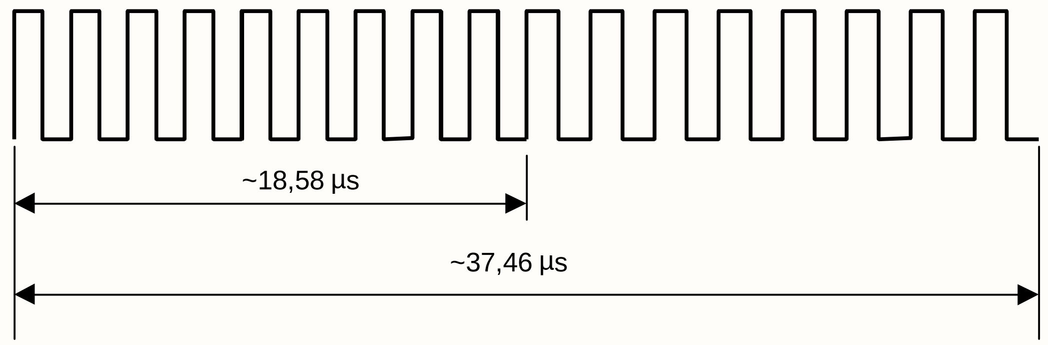
Logic 0 Encoding

- Subcarrier 1: 16 cycles of modulation followed by 16 cycles of no modulation, repeated 8 times.
- Subcarrier 2: 14 cycles of modulation followed by 14 cycles of no modulation, repeated 9 times.



Logic 1 Encoding

- Subcarrier 1: No modulation for 16 cycles, followed by 16 cycles of modulation, repeated 8 times.
- Subcarrier 2: 14 cycles of no modulation followed by 14 cycles of modulation, repeated 9 times.

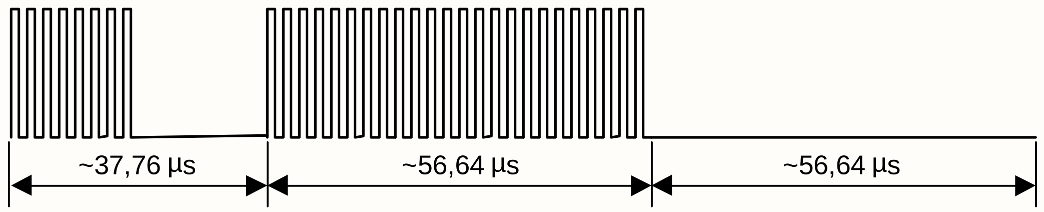


PN5180 ISO15396 Tag to Reader Communication

End of Frame

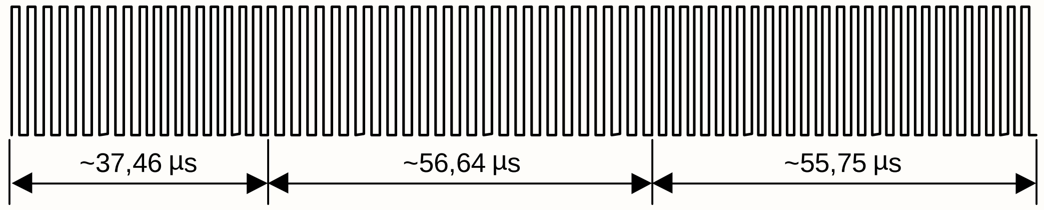
Single Subcarrier

- It transmit a logic 0, alternating between 16 cycles of modulation and 16 cycles of no modulation for a total of 24 times. This is followed by 768 cycles of no modulation.

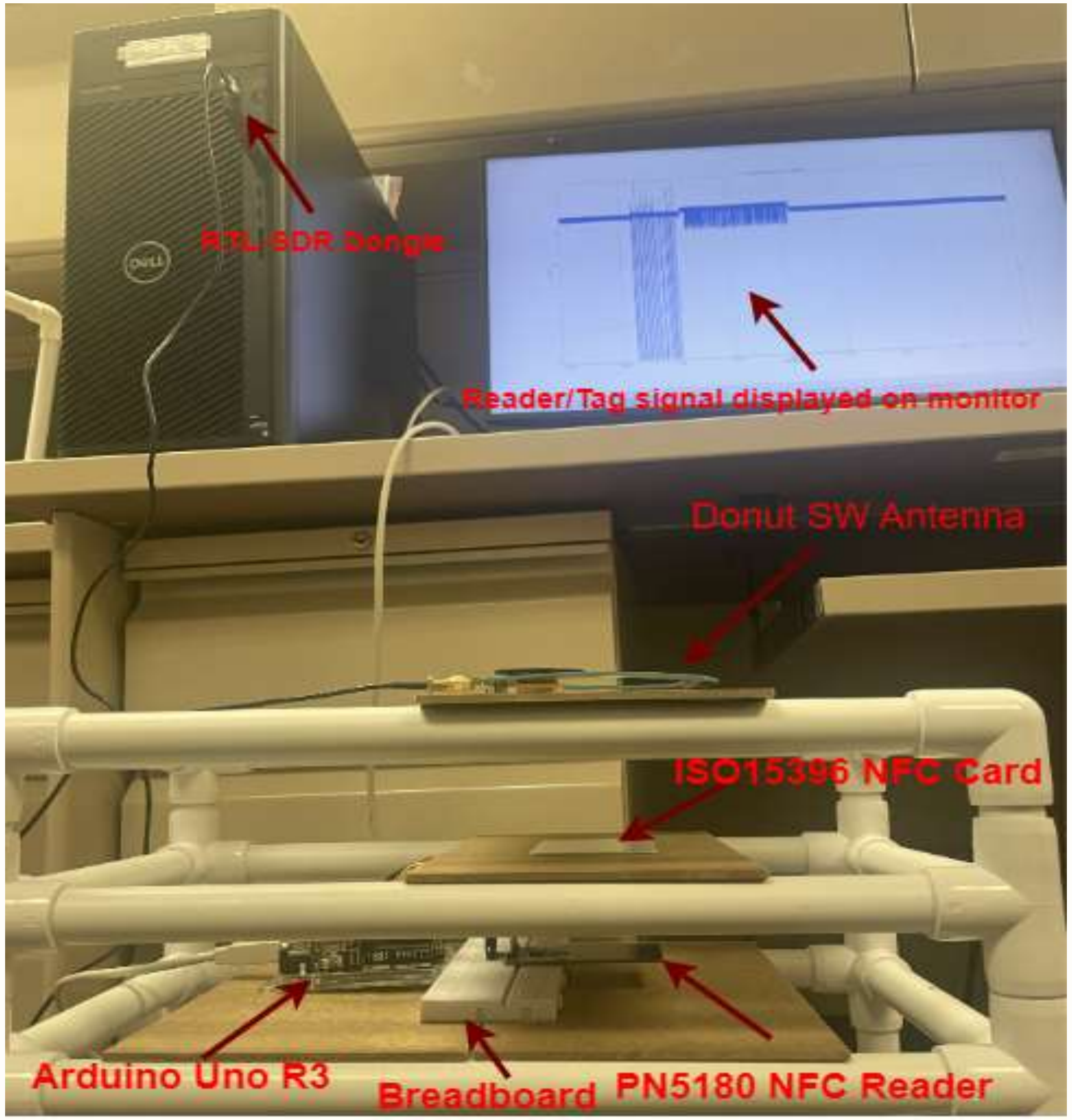


Two Subcarriers

- Subcarrier 1: Transmit a logic 0, alternating between 16 cycles of modulation and 16 cycles of no modulation for a total of 24 times.
- Subcarrier 2: Transmit a logic 0, alternating between 14 cycles of modulation and 14 cycles of no modulation for a total of 27 times.



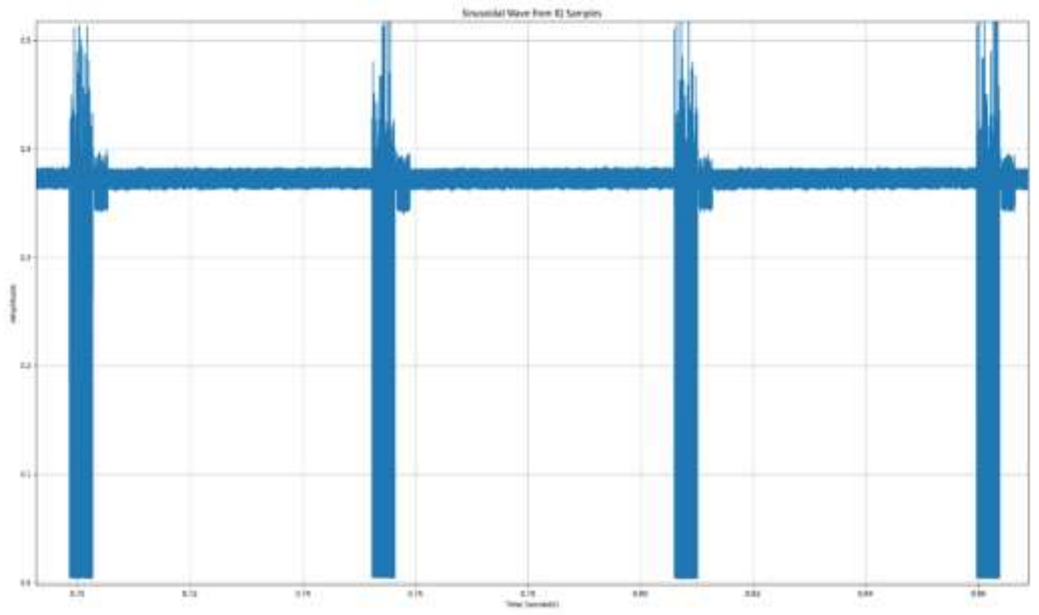
Measurement Setup

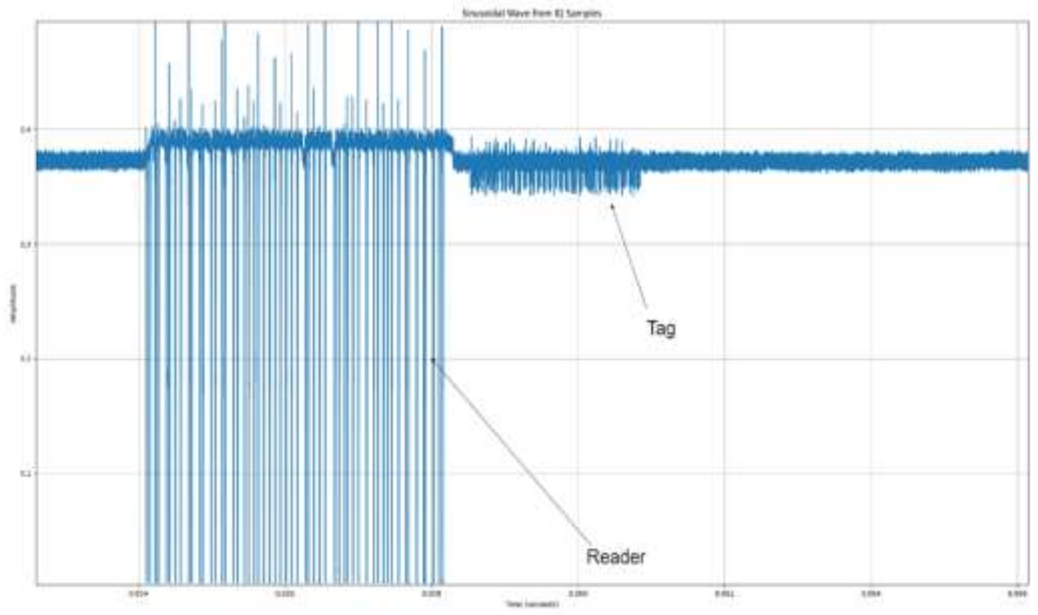


Data Acquisition and Analysis Workflow

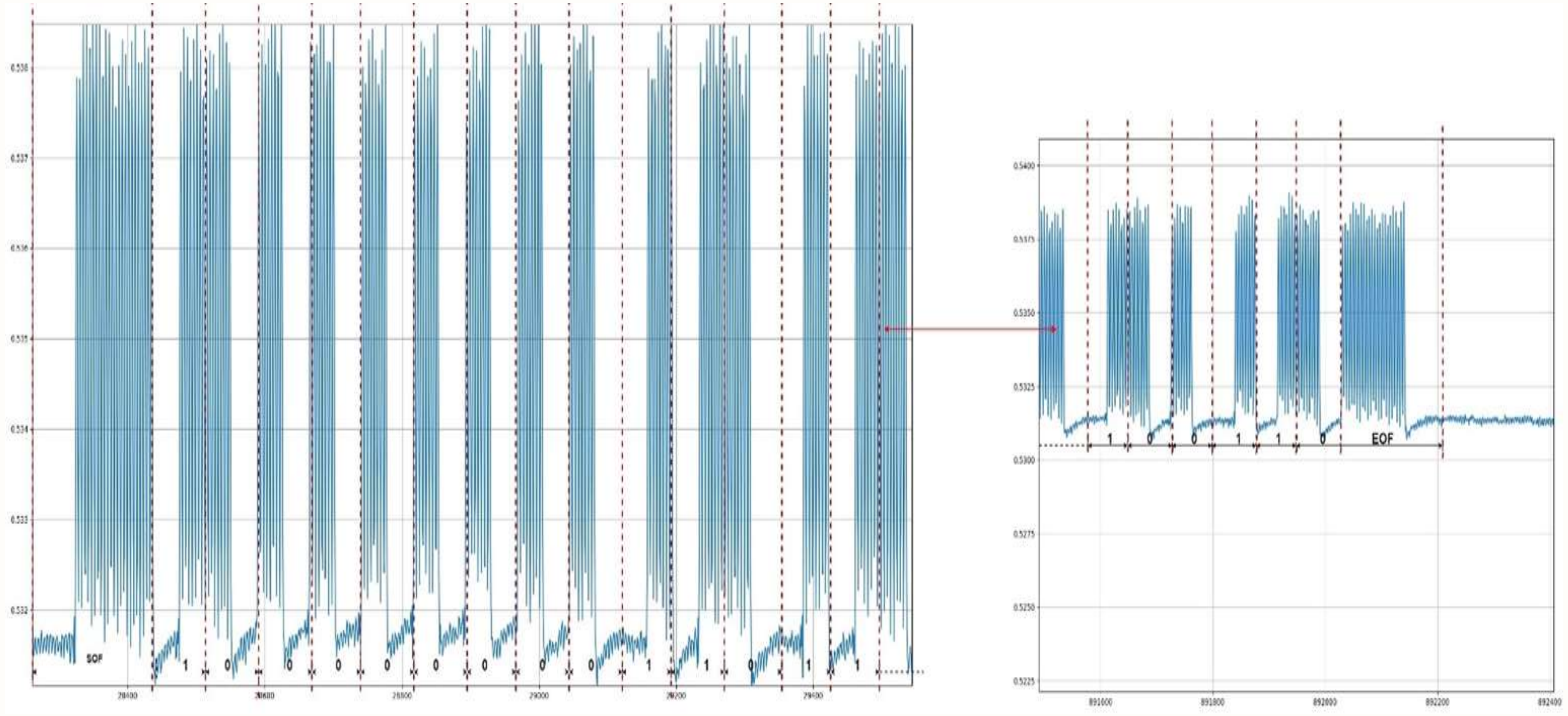
- Signal Generation: The NFC-PN5180 reader transmits a signal to the ISO15396 tag.
- Signal Capture: The SDR dongle receives the signal and captures the raw IQ samples.
- Data Saving: SDR# software saves the captured IQ samples to a file for further processing.
- Data Analysis: Python with libraries is used to read, process, and visualize the IQ samples, allowing for analysis of the NFC signal characteristics (e.g., amplitude, frequency, modulation, framing).
  - Examples of visualized signals can be shown using the provided graphs

Data Acquisition and Analysis Workflow





Bit Representation and Coding



Conclusion

- Demonstrated the use of Near Field Communication (NFC) technology interfaced with Software Defined Radio (SDR) for signal capture and analysis.
- The experiment not only provided hands-on experience with NFC and SDR but also highlighted the importance of signal processing in understanding wireless communication.
- The results confirm that SDR, combined with Python-based signal analysis, is a powerful tool for exploring and understanding NFC technology.
- This module includes hardware and software which can be used for students interested in Computer Science, Computer Engineering, and Electrical Engineering.

Thank you for your attention

This project is supported by National Science Foundation under grant no. 2341846 and 2310856.

University of Nebraska – Lincoln

School of Computing

Department of Computer Science

https://computing.unl.edu/