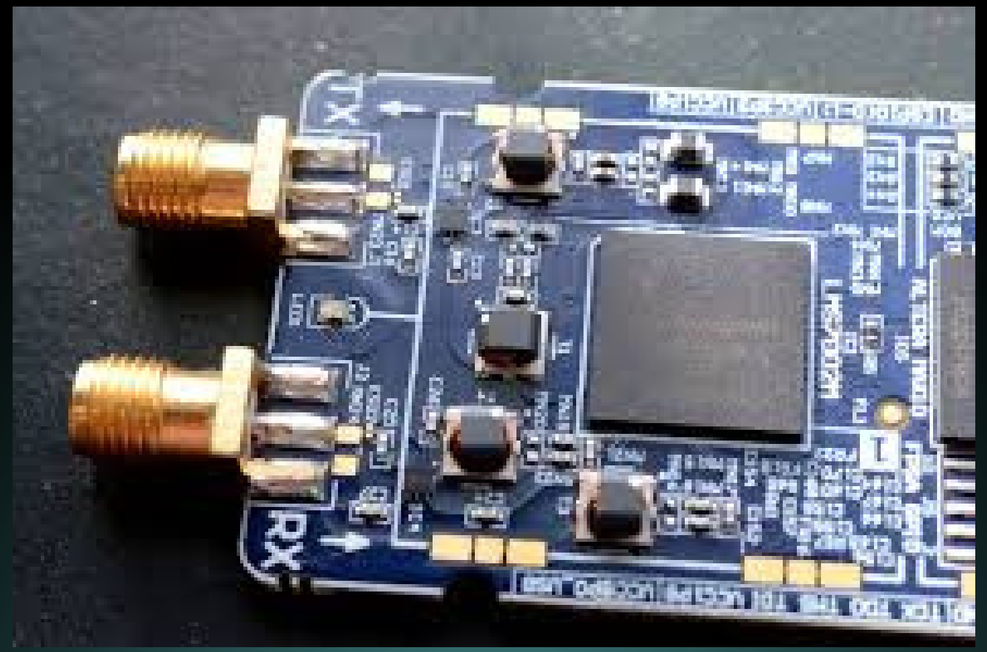
# Smart Sensors

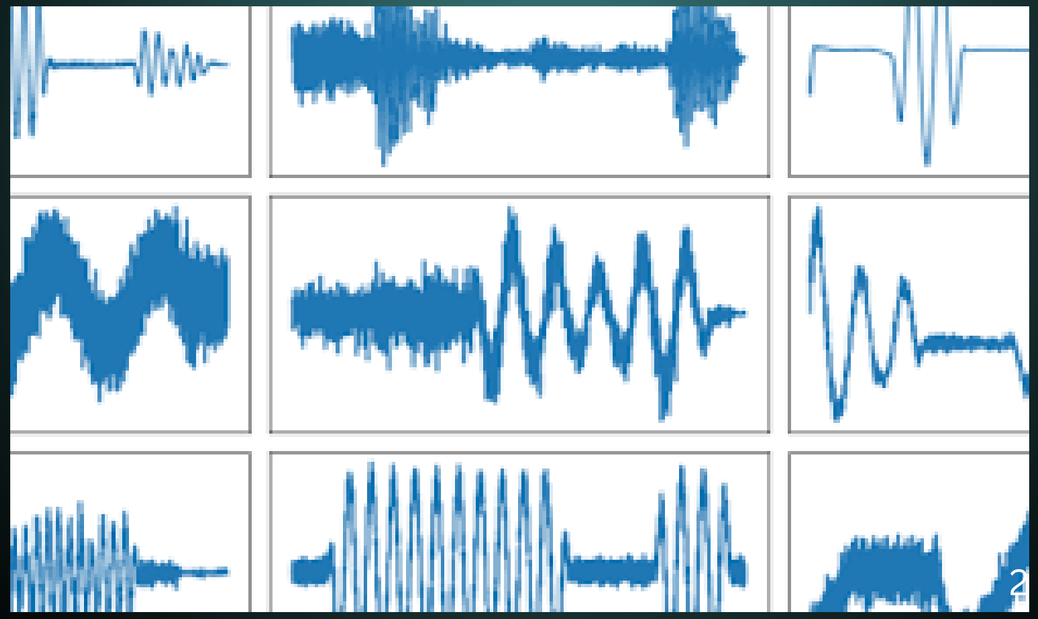
Presented by
Derrick Boston, III
Timothy Ford
Lamont Syrreal Watson

Collaborators:
Boeing
Intel
Howard University CEA

# Background

- RF signal classification is critical for defense, telecommunications, IoT
- Traditional setups are bulky, power intensive
- Traditional setups rely on GPUs, large hardware
- Current systems are inefficient for applications where SWAP constraits are important (size, weight, and power)
- What if we could create a compact, efficient, and real time systems to classify RF signals with the accuracy and adoptability needed for dynamic environments?



Existing RF signal classification systems are too large, power-hungry, and unsuitable for real-time, portable applications due to reliance on GPUs and bulky hardware. We are designing a compact, power-efficient RF signal classification system using Intel NUC AND FPGA integration, optimized with OpenVINO and a CNN model for efficient classification. This solution will enable real-time high-accuracy RF signal classification, meeting SWAP constraints and opening new opportunities for scalable and adaptable signal detection applications.

# Problem Statement

# Research Overview

- Hardware software co-design using Intel Nuc and OpenVINO
- Integration of CNNs, RNNs, and Transformation for real time signal classification
- Benchmarking Intel NUC performance for speed, latency, and scalability

Key Techniques

- Statistical feature detection
- Frequency domain analysis
- Image-based edge detection

# Design Ideas

## Design Idea 1

- Leveroges Intel NUC CPU and GPU for efficient time processing
- CNN model classifies RF signals based on spectrogram features (frequency/amplitude changes)
- Model runs on NUC's GPU to accelerate processing with low power usage

## Design Idea 2

- FPGA added to assist Intel NUC with feature extraction/initial signal preprocessing
- Transformer model classifies signals based on temporal patterns in RF data
- Can adjust to changes in signal environment
- Suitable for noisy conditions

## Design Idea 3

- Offloads real-time preprocessing and initial feature extraction to FPGA
- Preprocessed data is sent to CNN on Intel NUC for classification
- Provides high-speed processing by using FPGA's parallel processing capabilities
- Reduces latency and improves speed for real-time applications

# Top 2 Designs

## Design Idea 2

- FPGA added to assist Intel NUC with feature extraction/initial signal preprocessing
- Transformer model classifies signals based on temporal patterns in RF data
- Can adjust to changes in signal environment
- Suitable for noisy conditions

### Pros

- Significant for noisy environments
- Adaptable to changing signal conditions

### Cons

- Higher complexity
- Requires additional FPGA integration

## Design Idea 3

- Offloads real-time preprocessing and initial feature extraction to FPGA
- Preprocessed data is sent to CNN on Intel NUC for classification
- Provides high-speed processing by using FPGA's parallel processing capabilities
- Reduces latency and improves speed for real-time applications

### Pros

- High Processing Speed
- Optimized for Real-Time Applications
- Efficient Data flow
- Flexibility with CNN Model
- Real-Time Adaptability

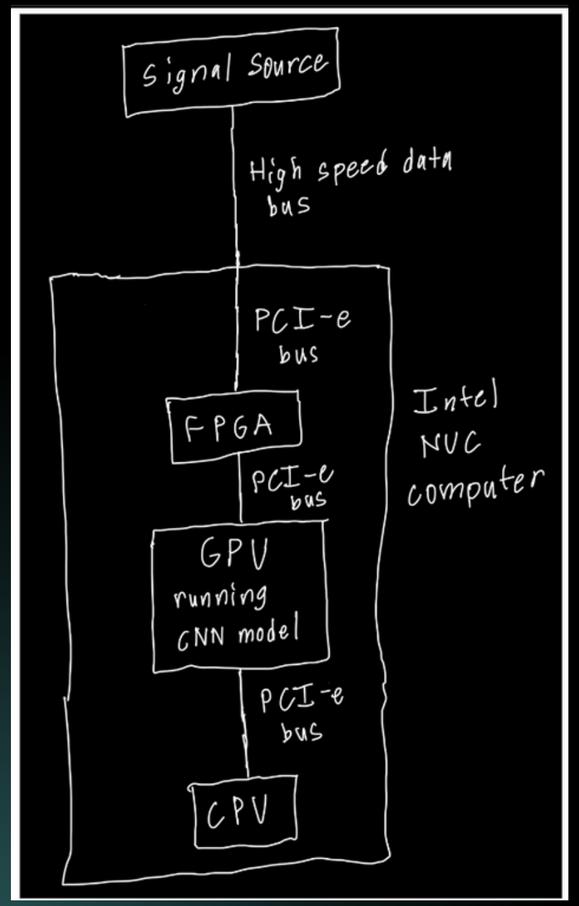
### Cons

- Increased Complexity
- Higher Initial Cost
- Power consumption
- Integration Challenges

# Design Matrix

| Attribute | Weight | FPGA-Assisted CNN Model | Weighted Score (CNN) | FPGA - Enhanced Transformer Model | Weighed Score (Transformer) |
|---|---|---|---|---|---|
| Processing Speed | 5 | 5 | 25 | 4 | 20 |
| Adaptability | 4 | 4 | 16 | 5 | 20 |
| Power Efficiency | 3 | 3 | 9 | 2 | 6 |
| Ease of Integration | 2 | 4 | 8 | 3 | 6 |
| Cost | 1 | 3 | 3 | 2 | 2 |
| Total Score | - | - | 61 | - | 54 |

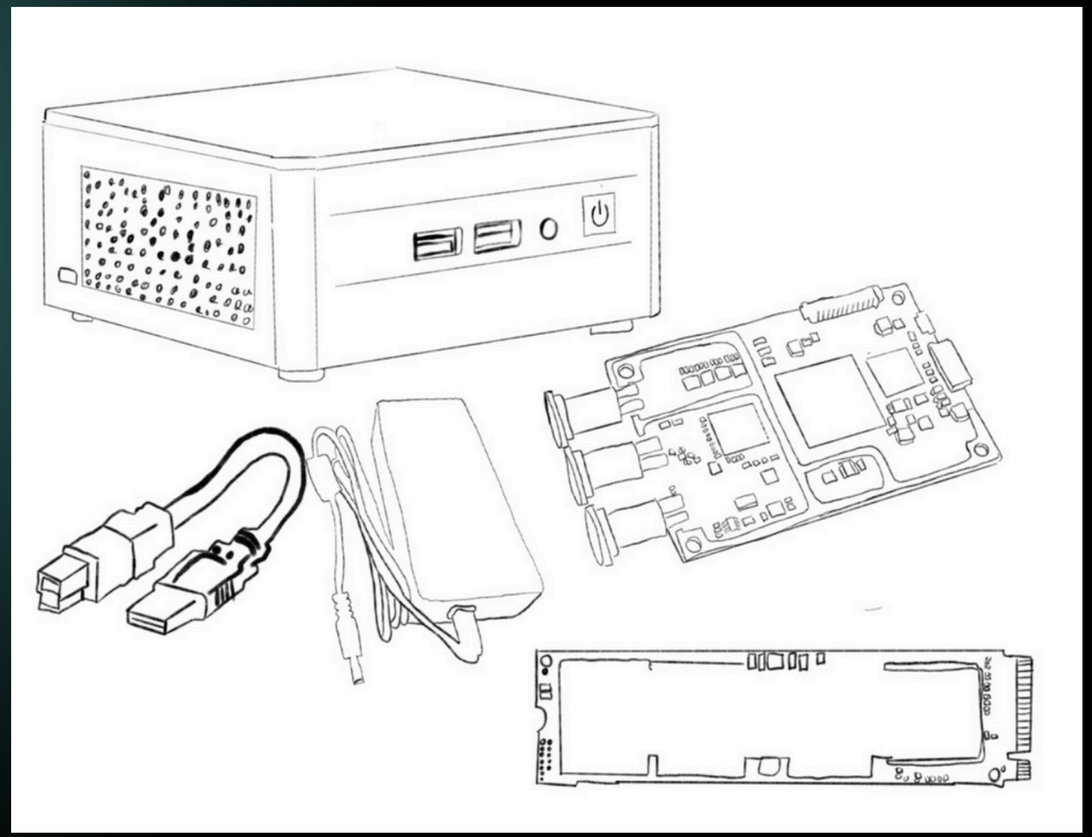
# Solution Generation




## Behavior

- Intel NUC computer receives signal data from radar via high speed, high-capacity data bus
- Data is sent to FPGA for preprocessing
- FPGA forwards processed data to NUC GPU running a CNN model
- GPU sends results from CNN model to CPU
- CPU processes data using software

# Solution Design



| Component | Specifications | Part Store Web Link | Unit Price | Quantity | Total Price |
|---|---|---|---|---|---|
| Intel NUC 11th Gen | Core i7, 16GB Ram | https://www.intel.com/content/www/us/en/homepage.html | $650 | 1 | $650 |
| Ettus USRP B205mini-i | 70 MHz – 6GHz, USB 3.0 | https://www.ettus.com/ | $900 | 1 | $900 |
| Mean Well 19V Adapter | 65W | https://www.digikey.com/ | $30 | 1 | $30 |
| Samsung 970 EVO Plus | 1 TB | https://www.samsung.com/us/ | $110 | 1 | $110 |
| USB 3.0 Cable | 3 feet | https://www.amazon.com/ | $10 | 1 | $10 |

# Research

- Initial benchmarking shows potential for high performance inferencing tasks

# Deliverables

- Benchmarked performance of Intel NUC hardware in real time RF signal processing, focusing on accuracy, speed, and latency
- Hardware based test algorthims using OpenVINO for feature detection and classification
- Key metrics:
- Scalability, Speed/Latency, SWAP-C( Size, Weight, Power, Cost).

# Motivation

- Current RF classification systems rely on bulky and power-intesive setups
- Putting an entire GPU on a rader. How can we use a SWAP tradeoff to use this in real life? GPU isn't used to support SWAP
- Intel NUC hardware offers a compact alternative with the potential to drastically reduce size and power needs while improving real-time signal

# Challenges

- Translate the Machine Learning Algorithm from one system to another and then test it
- Benchmarking
- Testing

# Broader Impact

- Improve SWAP
- Enhance real-time RF signal detection in noisy environents

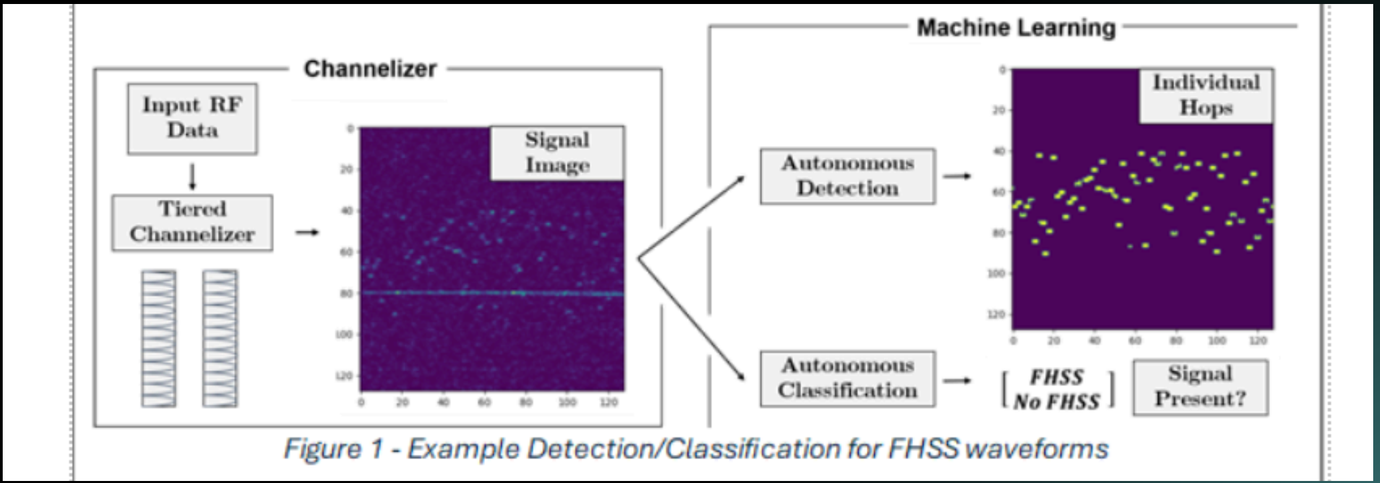


Figure 1 - Example Detection/Classification for FHSS waveforms

# Research Goals/Future Work

- Explore AI architectures (CNNs, RNNs, Autoencoders) for robust modulation
- Benchmark hardware performance, including accuracy, speed, and latency
- Expand dataset to include more modulation types for better model generalization
- Investigate additional hardware platforms and scaling possibilities

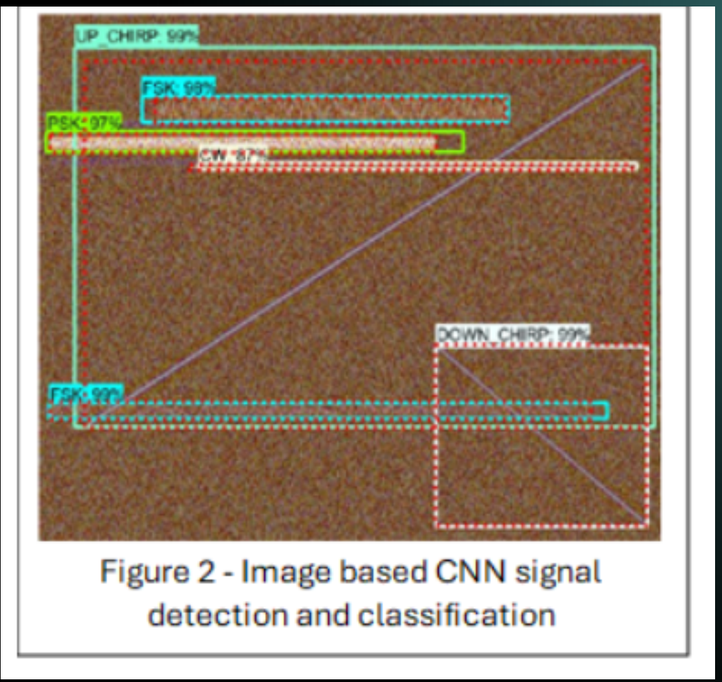


Figure 2 - Image based CNN signal detection and classification

# Thank You