

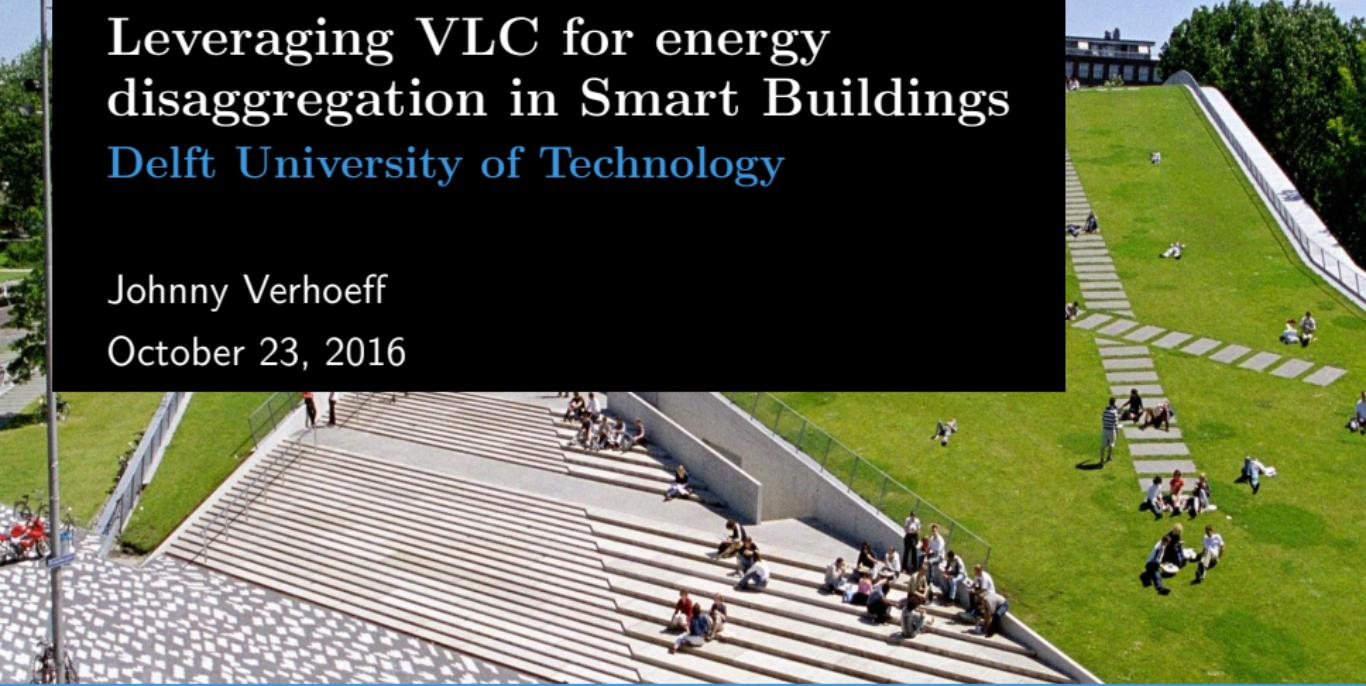


# Leveraging VLC for energy disaggregation in Smart Buildings

## Delft University of Technology

Johnny Verhoeff

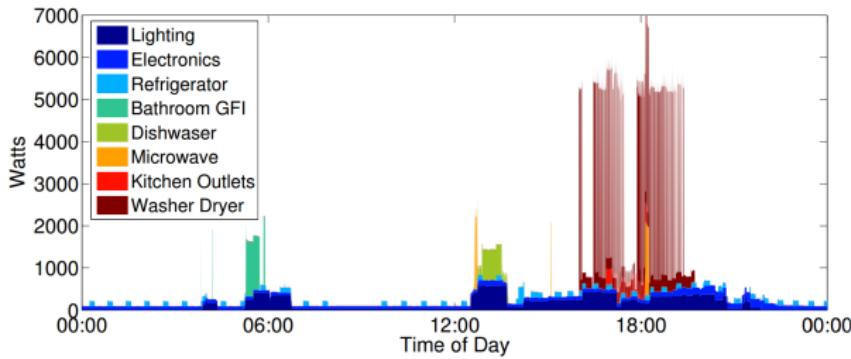
October 23, 2016



# Energy Disaggregation

Energy consumption is a most pressing issue.

- To reduce it, understanding the usage of that energy is needed.
- Smart-meter can disaggregate the energy usage in a household.
- This is done by recognizing the unique signatures of appliances.



# Energy Consumption Lighting

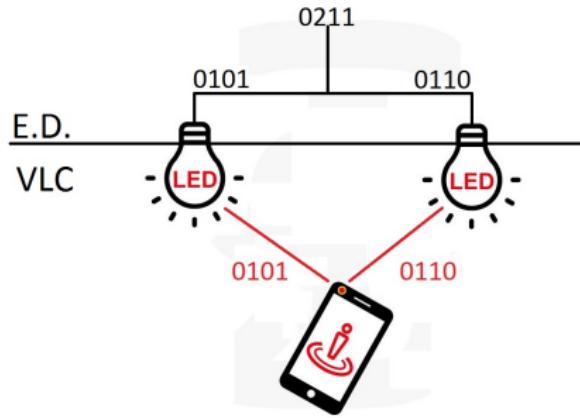
Individual lights cannot be disaggregated (yet).

- Important to be able to disaggregate individual lights: Lighting consumes 19 % of the power in an average household.
- The reason: Lighting does not have a unique signature.
- Instead there are many lights with the same signature.

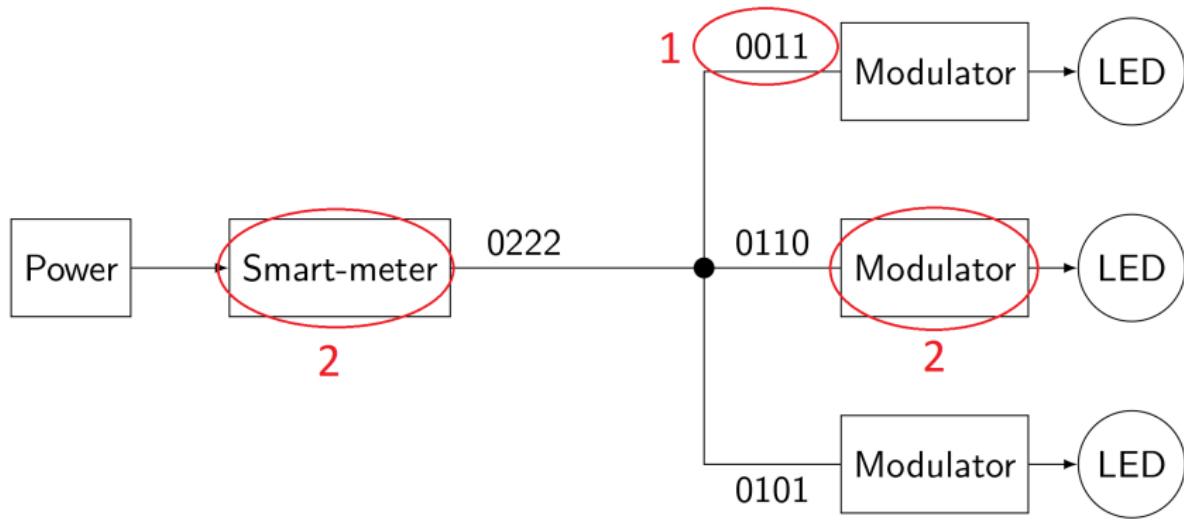
# VLC Piggybacking

VLC is a communication method which uses visible light to transmit data.

- This data can be unique IDs for LED beacons used for indoor localization.
- This data will also propagate through the current draw.



# Contributions



- ① Investigation of codes that can be used.
- ② Design of hardware to modulate and sample the current.
- ③ Evaluate the solutions.

# Code Investigation

- Problem:
  - Multiple transmitters (LEDs) on the same channel at the same time.
  - Only the aggregated energy can be measured.
- Very similar to:
  - Multiple cell phones transmitting to the same base station.
  - Base station receives the combined signals.

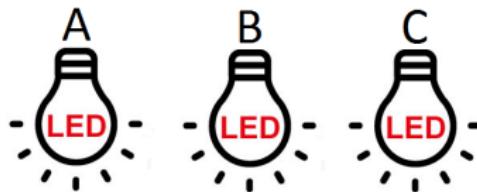
# Requirements for Codes

- Scalability
- Balance
- Synchronous and Asynchronous
- Resilience

# Correlation

Measuring the similarity between a code and a received signal:

$$R(\tau)_{xs} = \sum_{i=0}^{L-1} x(i) \times s(i + \tau) \text{ with } \tau = 0, 1, 2, \dots, L$$



- $S = A + B + C$
- $R_{AS} = R_{AA} + R_{AB} + R_{AC}$

# Orthogonal Codes

- Creation via Hadamard matrix:

$$H_{2n} = \begin{bmatrix} H_n & H_n \\ H_n & \overline{H_n} \end{bmatrix}$$

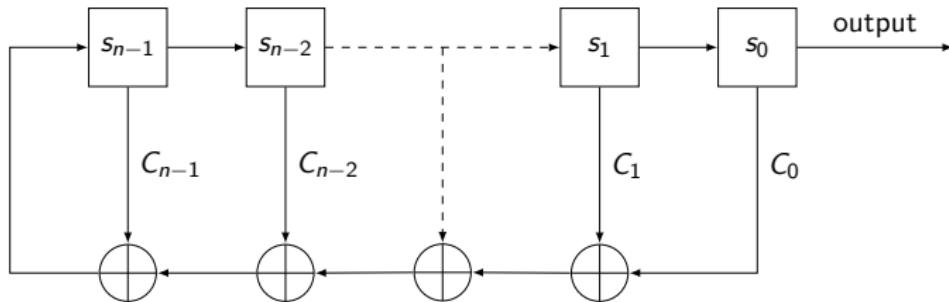
$$H_1 = [1]$$

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

- Properties:
  - Scalable: ✓
  - Balance: ✗
  - Correlation: ✓ (Only synchronous)

# PN Codes

- Creation of a PN code via LFSR:



- Properties:
  - Scalable: ✗
  - Balance: ✓
  - Auto-correlation: ✓
  - Cross-correlation: ✗

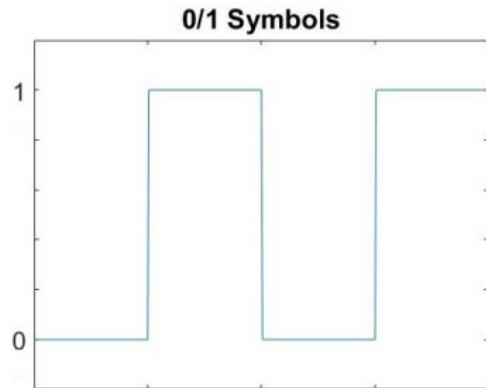
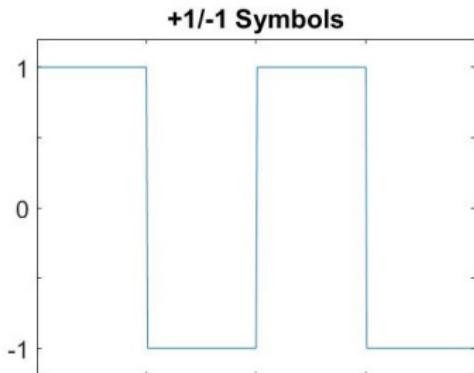
# Gold Codes

- Creation of a Gold code via two LFSRs.
- Properties:
  - Scalable: ✓
  - Balance: ✓
  - Auto-correlation: ✓
  - Cross-correlation: ✓

# Mapping Problem

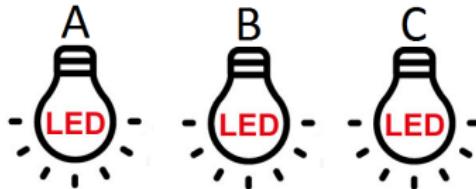
Our Scenario:

Wireless:



- Normal correlation function:  
 $R_{XY}$
- Mapping of symbols:  $b = \frac{1-r}{2}$
- New correlation function:  
 $\hat{R}_{XY}$
- $R_{XY} = -m - 2 \times \hat{R}_{XY}$

# Interference Solution



- $S = A + B + C$
- $R_{AS} = R_{AA} + R_{AB} + R_{AC}$

- Determine max. no. of modulating LEDs  $m = \frac{L}{2 \times \phi}$
- Threshold:  $T = \frac{L}{2}$
- Make sure the no. of modulating LEDs  $\leq m$ :
  - Continuous Method
  - Probabilistic Method

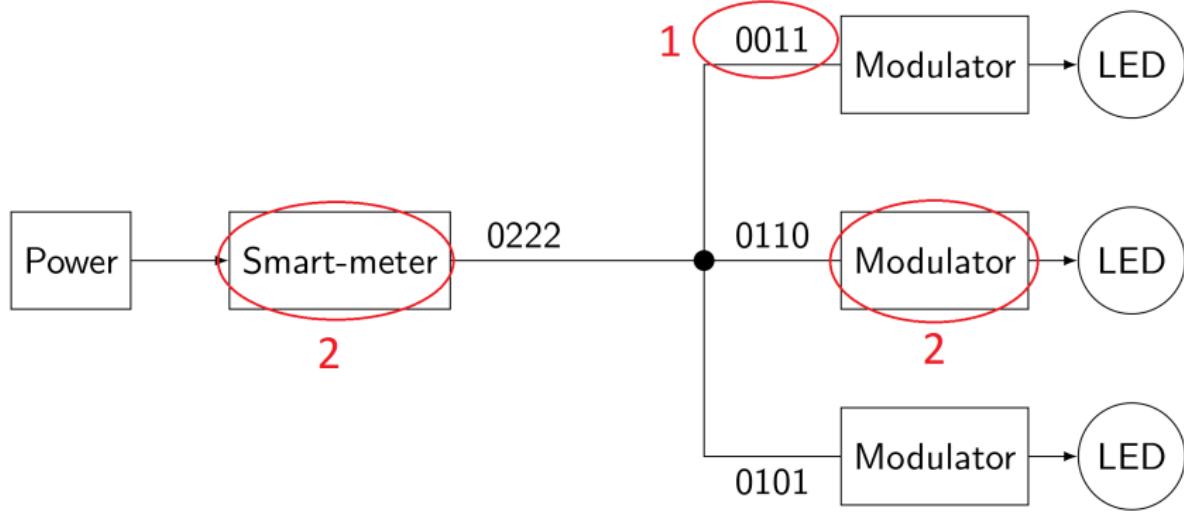
# Continuous Method

- Maximum of  $m$  LEDs in system.
- Appropriate code length:  $L \propto m^2$ .
- Examples with modulating frequency  $f = 10$  kHz:
  - $m = 7$  LEDs  $\rightarrow L = 1023 \rightarrow t = 0.05$  s.
  - $m = 1023$  LEDs  $\rightarrow L = 2^{23} = 8388608 \rightarrow t = 14$  min.

# Probabilistic Approach

- Accuracy  $(1 - \epsilon)$  outputs a probability  $p$ .
- $p$  is chosen such that no. of modulating LEDs  $\leq m$ .
- Example:
  - 1025 LEDs  $\rightarrow L = 1023 \rightarrow t = 53$  s for 99.9 % accuracy or  $t = 22$  s for 99 % accuracy.

# Hardware Components



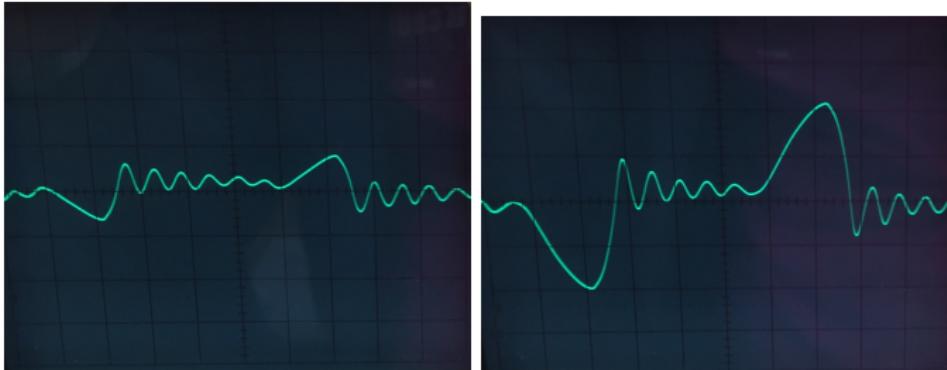
- ① Codes
- ② Hardware

# Modulator Requirements

Modulator must translate ones and zeros of ID into current draw.

- $0 \rightarrow$  no current draw.
- $1 \rightarrow$  some constant current draw.

# Existing Modulator Hardware

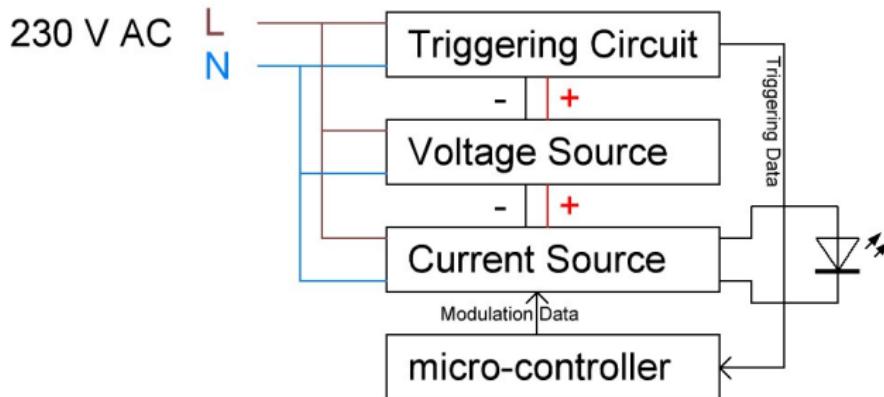


Switching Mode Power Supply  
modulating a '0'

Switching Mode Power Supply  
modulating a '1'

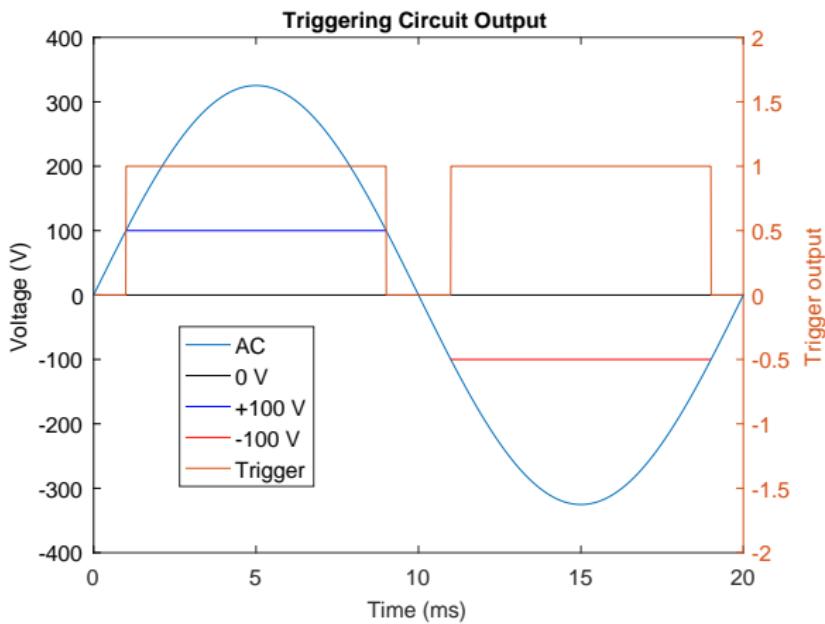
Very hard to distinguish and will not yield nice aggregated results when multiple of these SMPS will be used.

# Custom Modulator



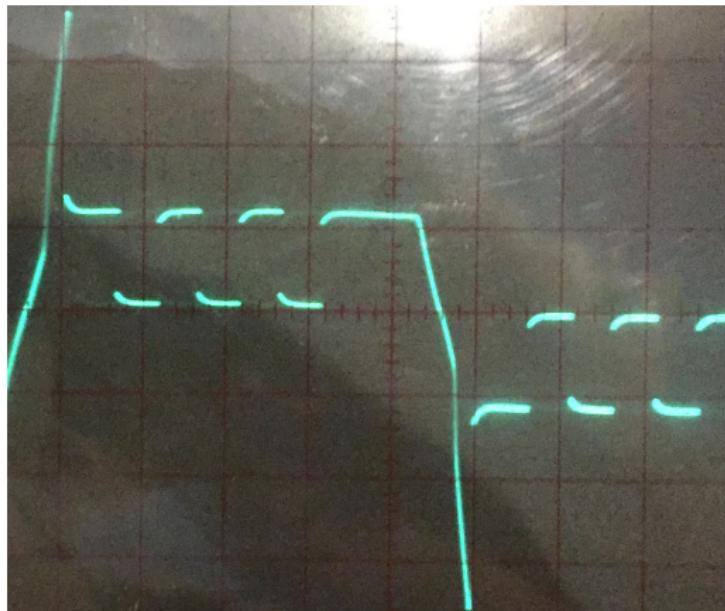
# Detecting When to Modulate

- AC Voltage is zero crossing.
- LEDs require some voltage before the current starts flowing.

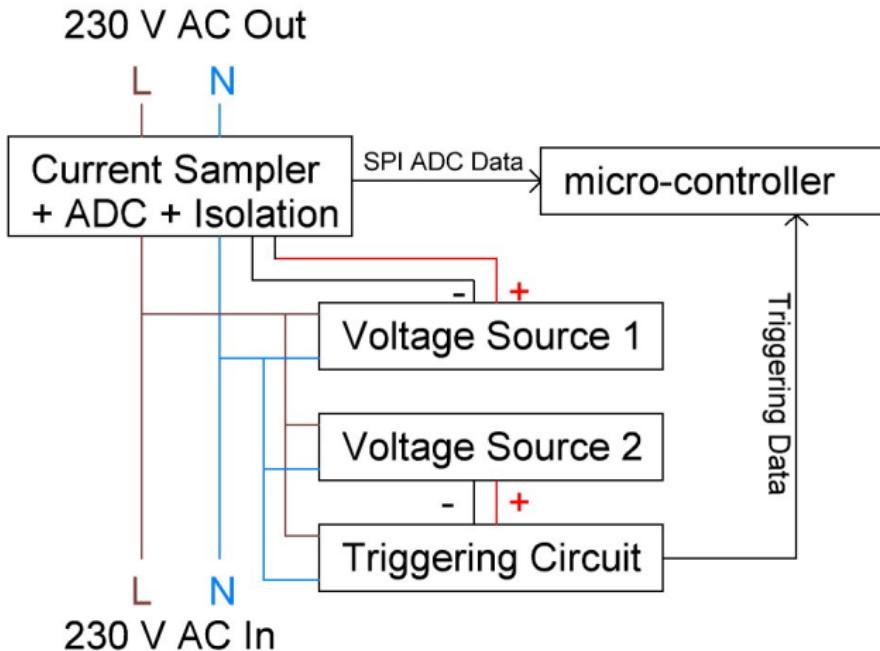


# Constant Current Draw

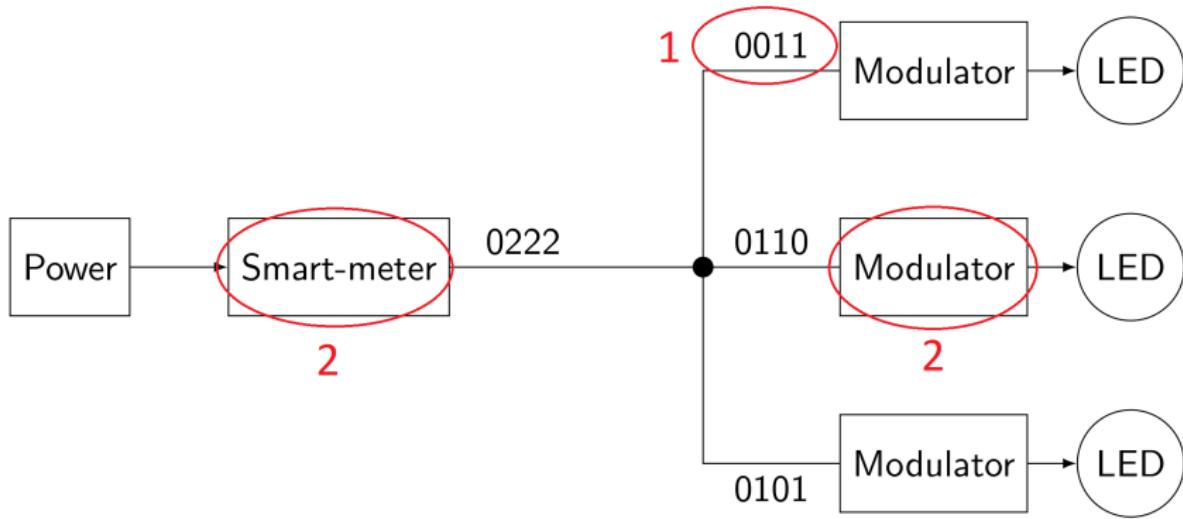
- AC Voltage is not constant.
- For disaggregation a constant current is desired.



# Smart-meter



# Recap

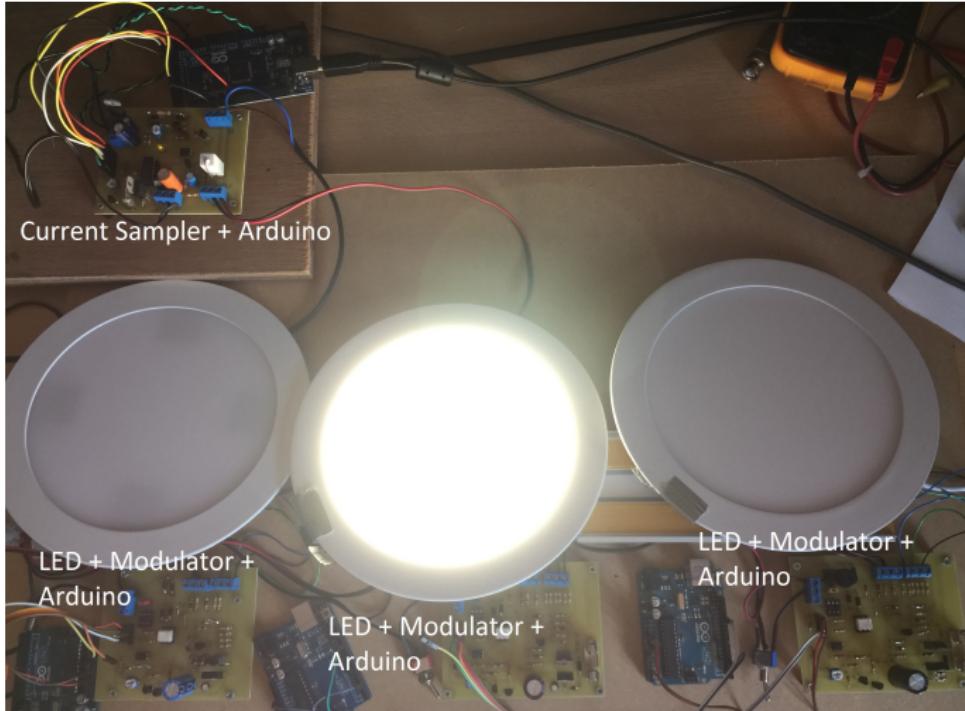


- ① Codes
- ② Hardware
- ③ Evaluate the system

# Evaluation Outline

- Hardware evaluation
- Software simulation

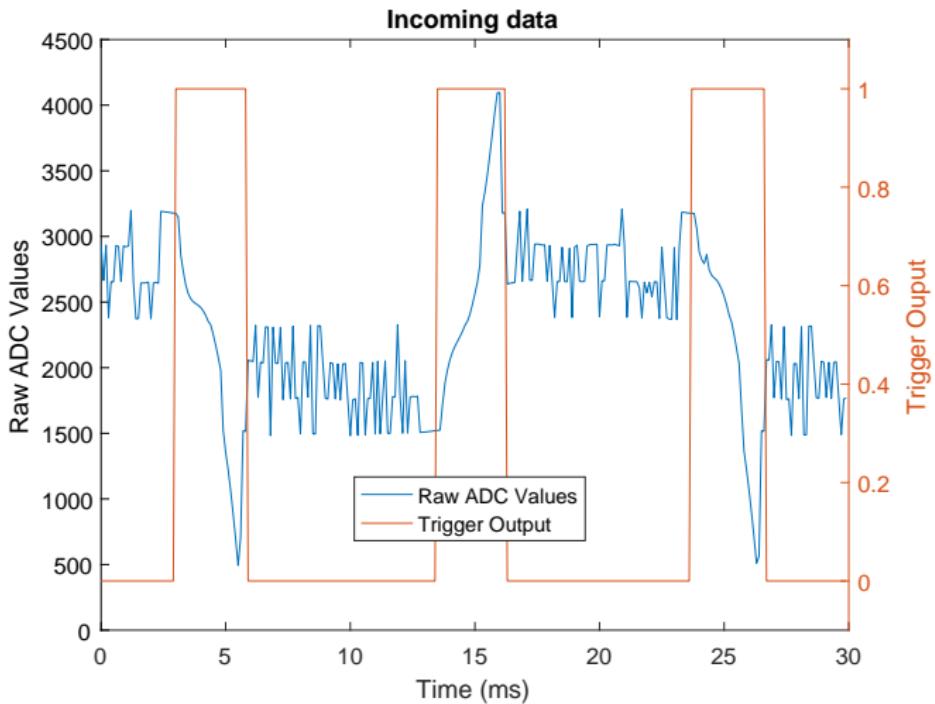
# AC Testbed



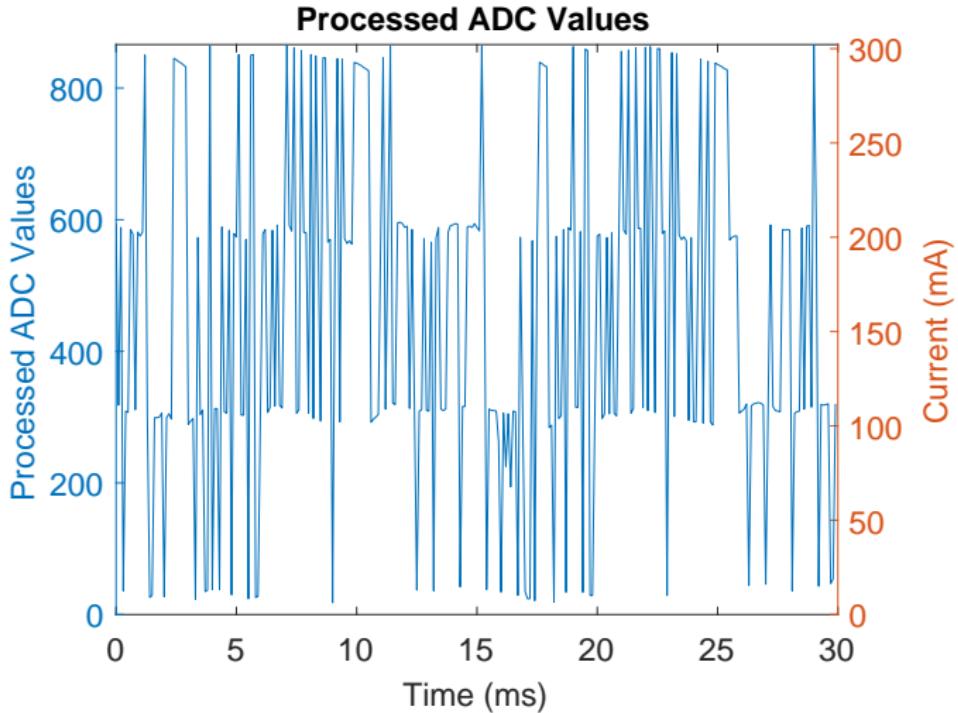
# Setup for Hardware Evaluation

- Setup:
  - 4 distinct codes will be used, 3 for the LEDs and one will represent an LED in an off state.
  - Modulation via continuous method with  $m \geq 3$  code.
- Goals:
  - Correctly identify state of LEDs in a timely manner.

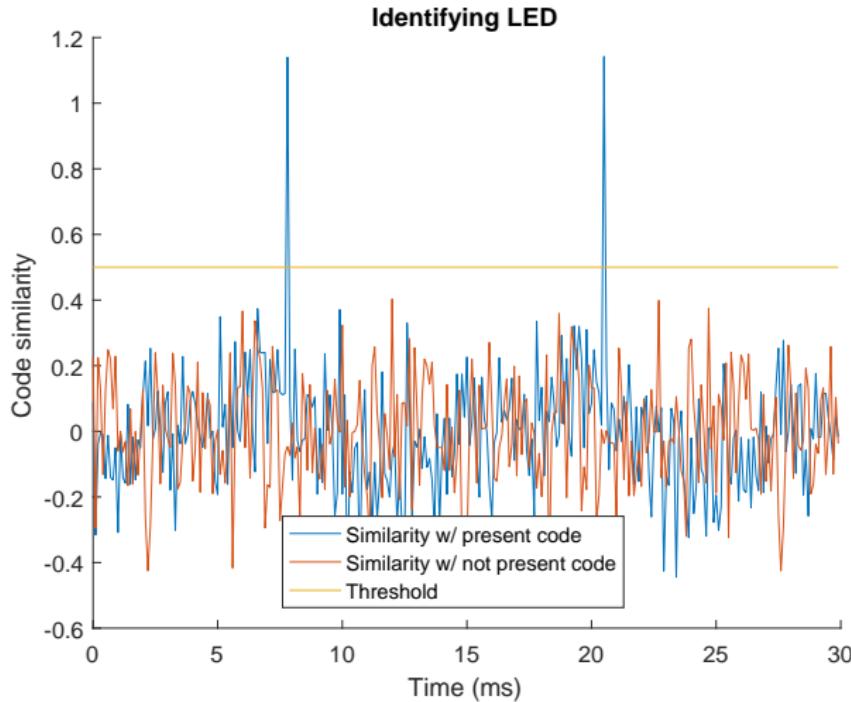
# Raw Data



# Processed Data



# Identifying LEDs

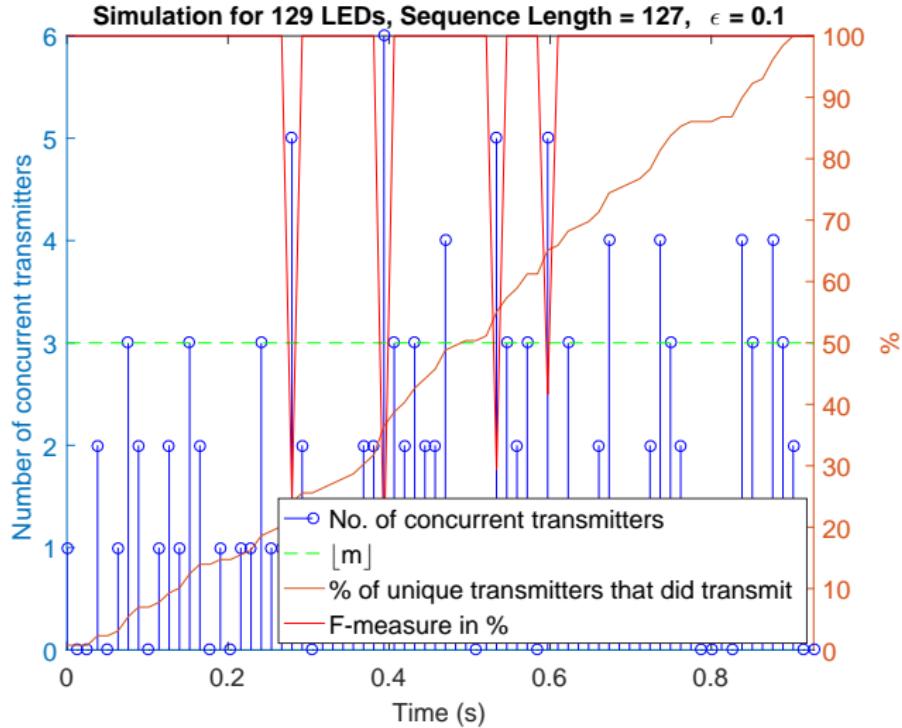


# Simulation

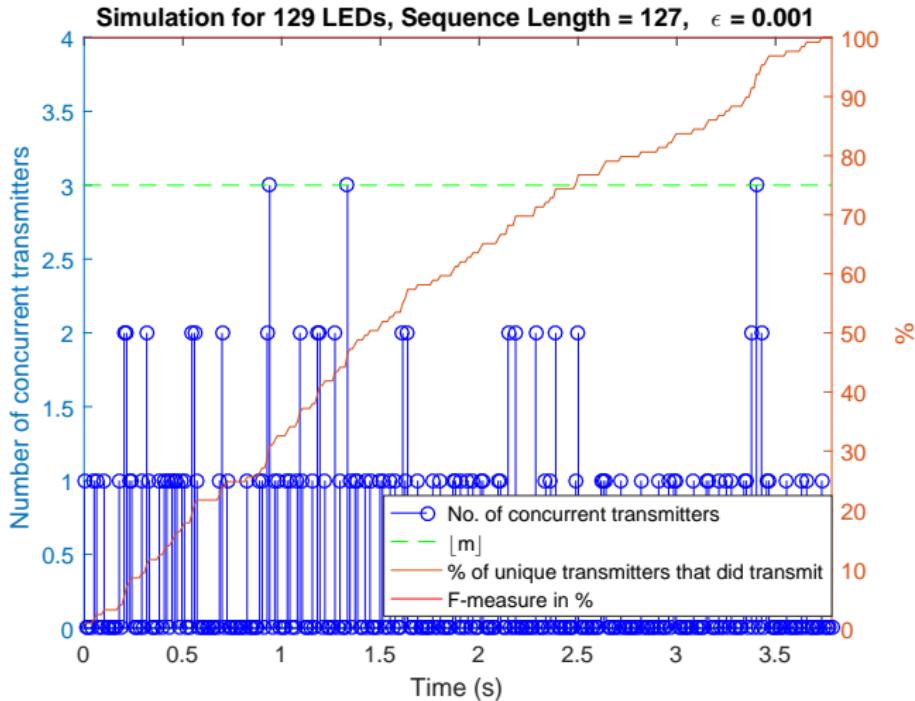
Software simulation:

- 129 LEDs.
- Modulation via probabilistic method.
- Identify state of all LEDs by correlating all IDs with aggregated signal.

# Fast but Inaccurate Simulation



# Slow but Accurate Simulation



# Conclusion and Future Work

- Conclusion:
  - Codes have been investigated.
  - Hardware is designed to encode and decode the IDs.
  - The software and hardware is evaluated along with a simulation for scalability.
- Future Work:
  - Other appliances
  - Dimming lights