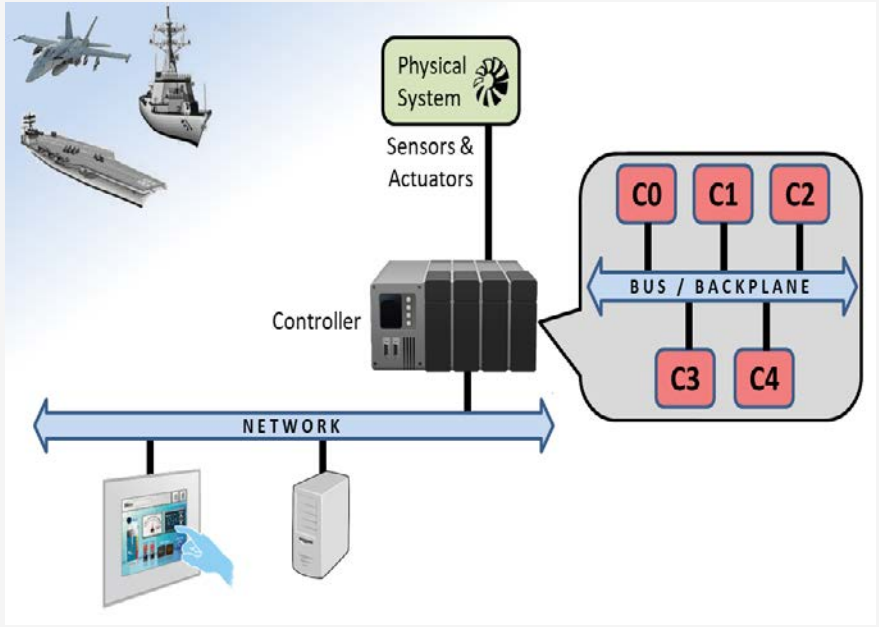
Cross-Layer Detection of Sensor-based Deception Attacks on Cyber-Physical Systems

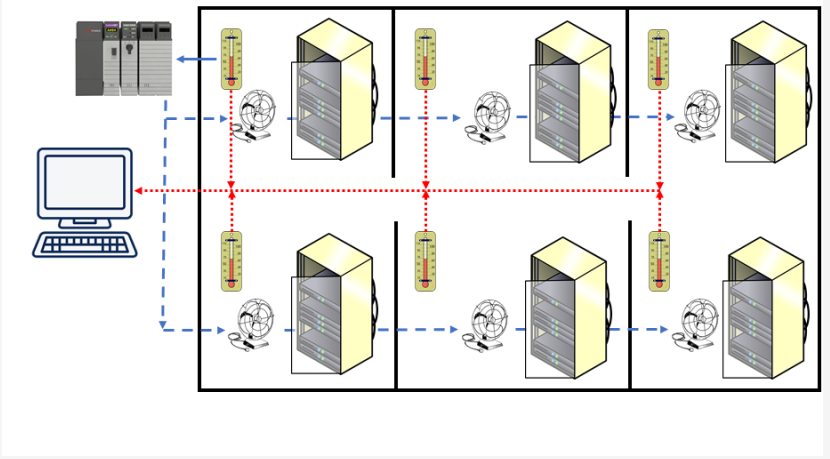
Nilanjan Banerjee

Associate Professor, University of Maryland, Baltimore County

(http://www.csee.umbc.edu/~nilanb)

Cyber-physical systems are powerful systems with applications to several diverse domains





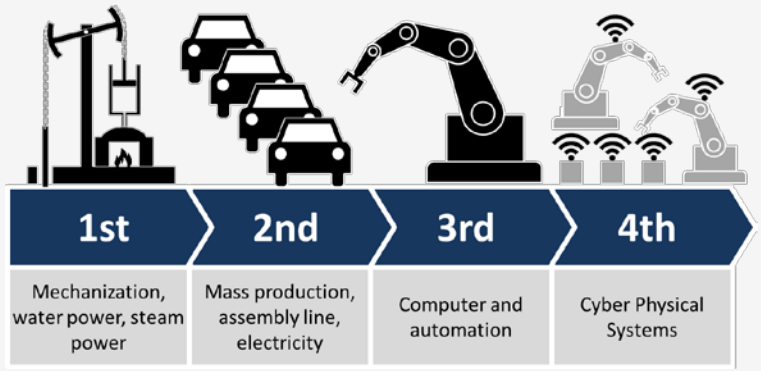
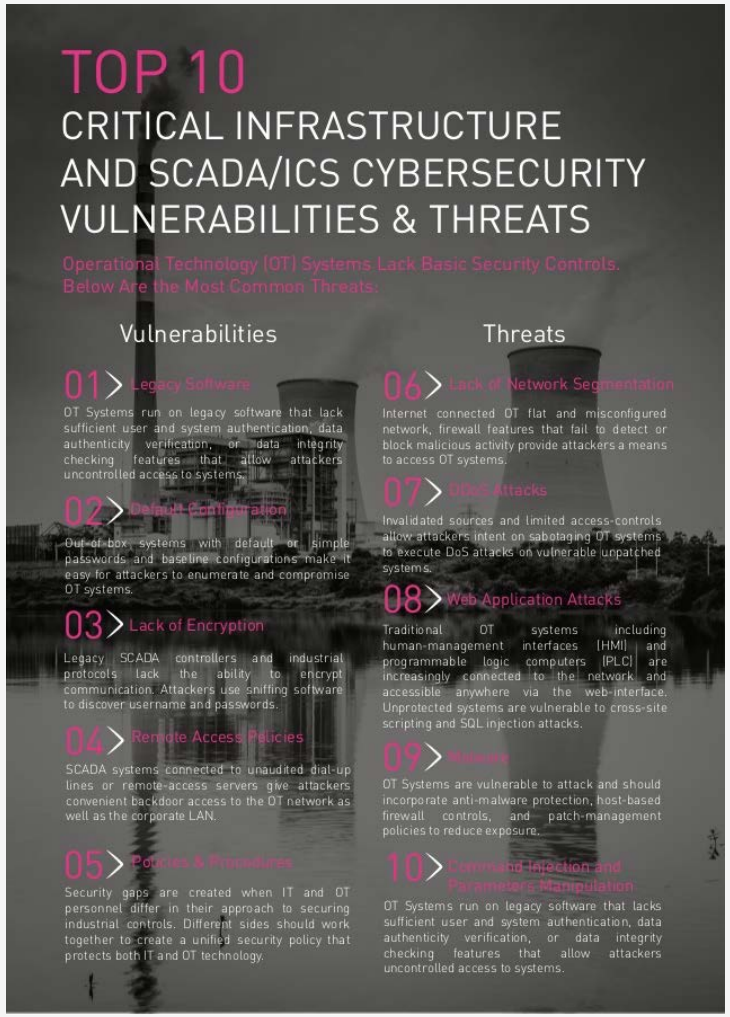
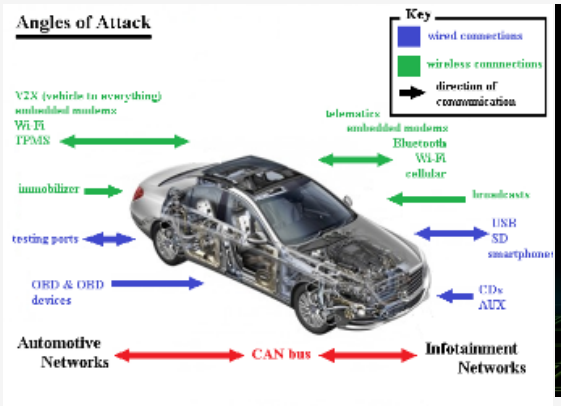


Figure courtesy of Christoph Roser at allaboutlean.com

Alas! with great power comes great responsibility and threat





Traditional CPS defense has focused on only one layer or level

Computer Science

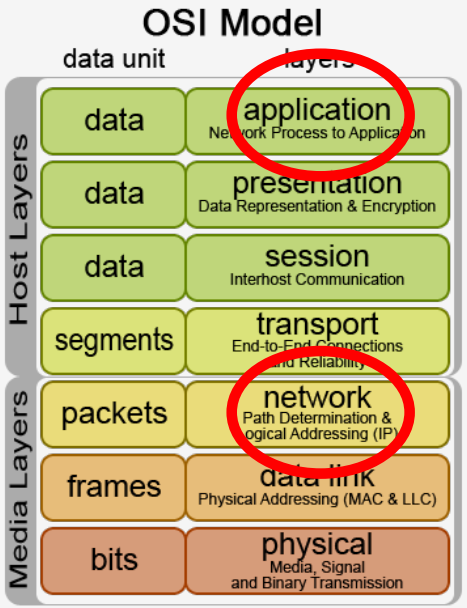


Figure courtesy of Dino.korah (Wikipedia)

Control Engineering

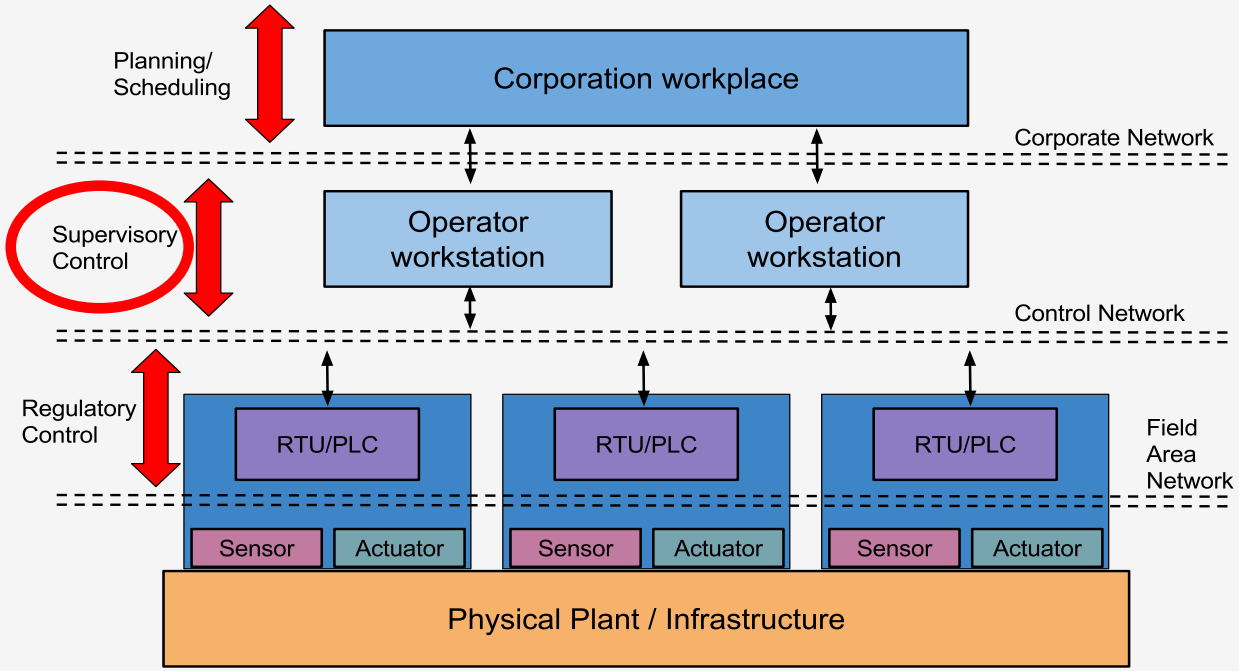
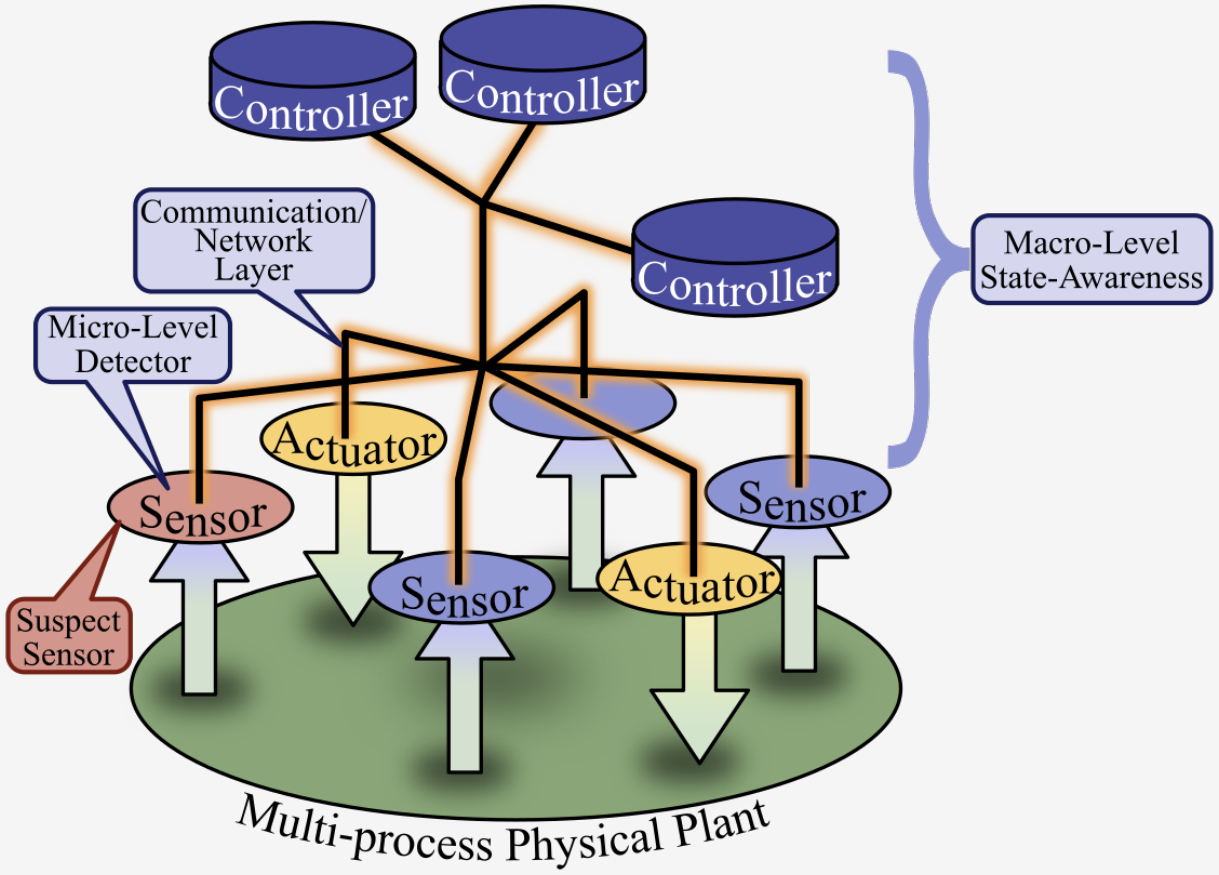


Figure courtesy of Prof. Alvaro A. Cardenas [1]

Cyber defenses for CPS need to work at multiple levels



1. Our method is **holistic** and combines data from sensors at multiple layers

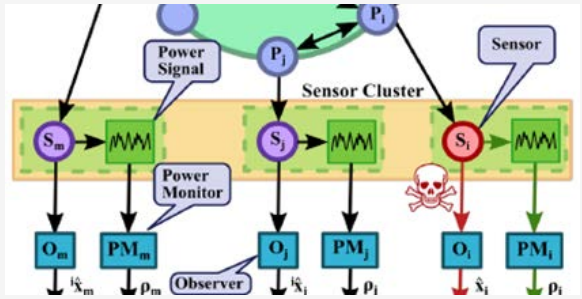
2. Combining sensor information from multiple layers allows the CPS system to **operate in the event of attacks gracefully**

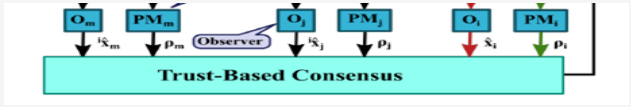
The key idea is to meaningfully combine information from multiple layers

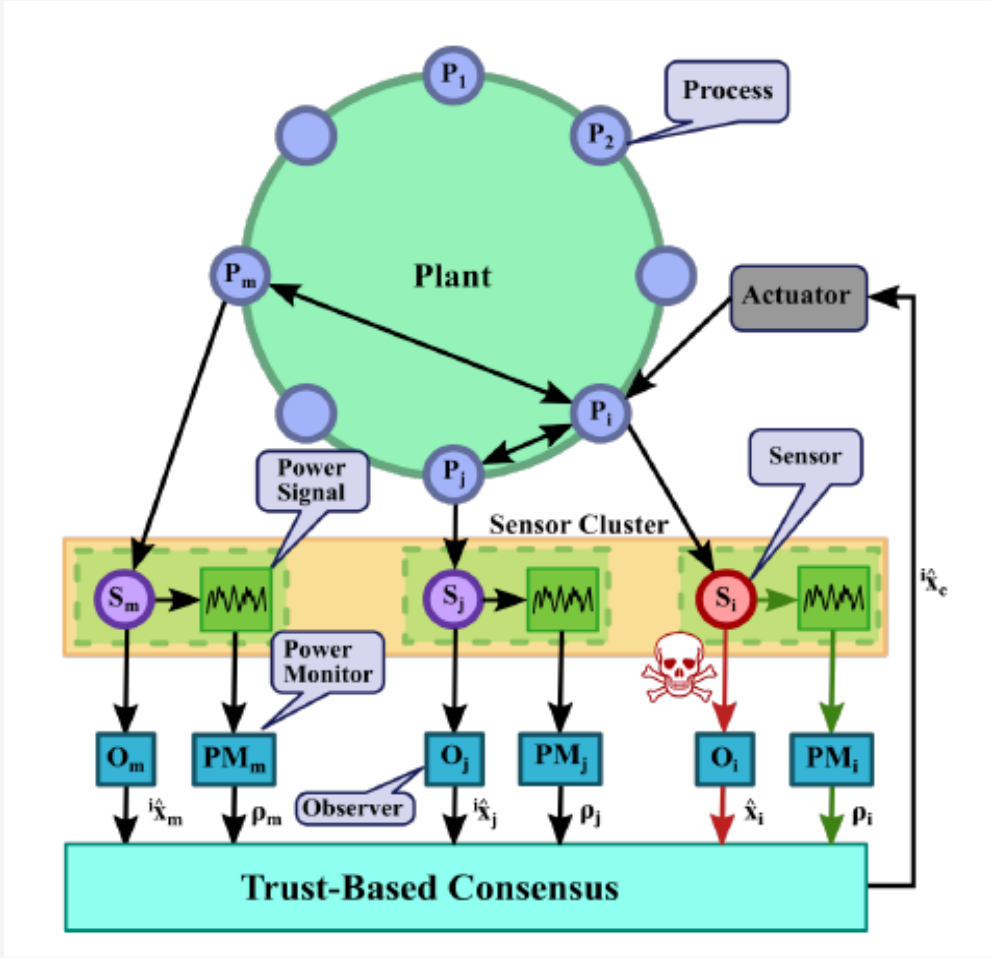
1. **Observers** predict the value that should be emitted by a rogue sensor

2. This value emitted is weighted by a **trust value** that is calculated by using **physical layer side channel analysis**

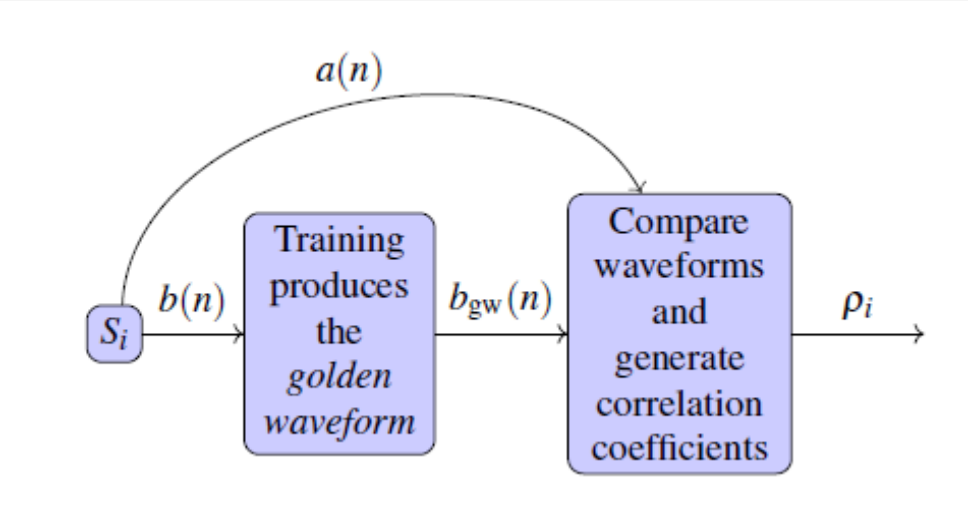
3. A **weighted consensus** of the emitted value is the estimation from the rogue sensor fed into the actuator.



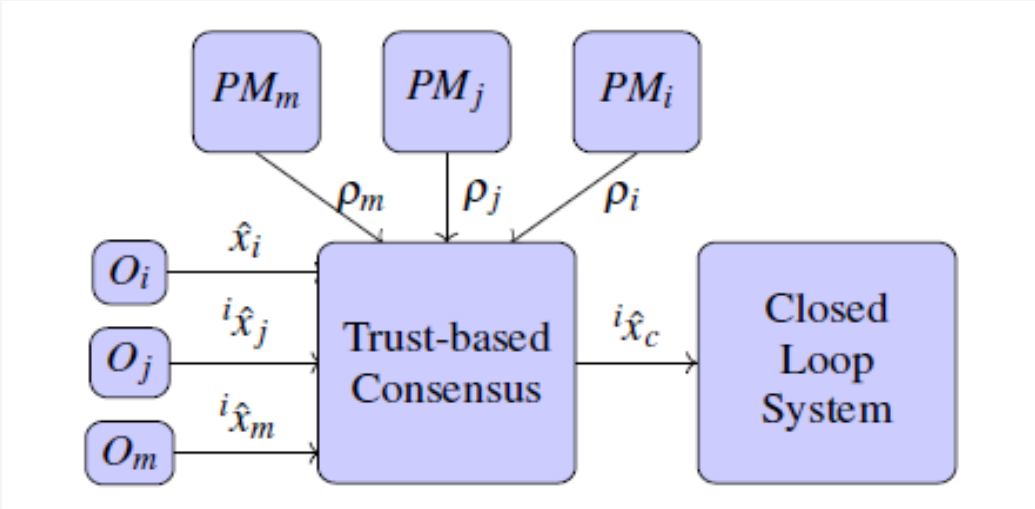




The components for the macro- and micro- detectors work together



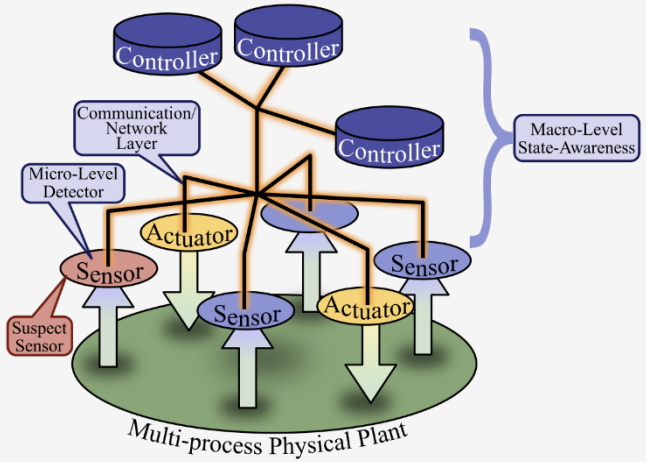
Micro-Level Measurements Block Diagram

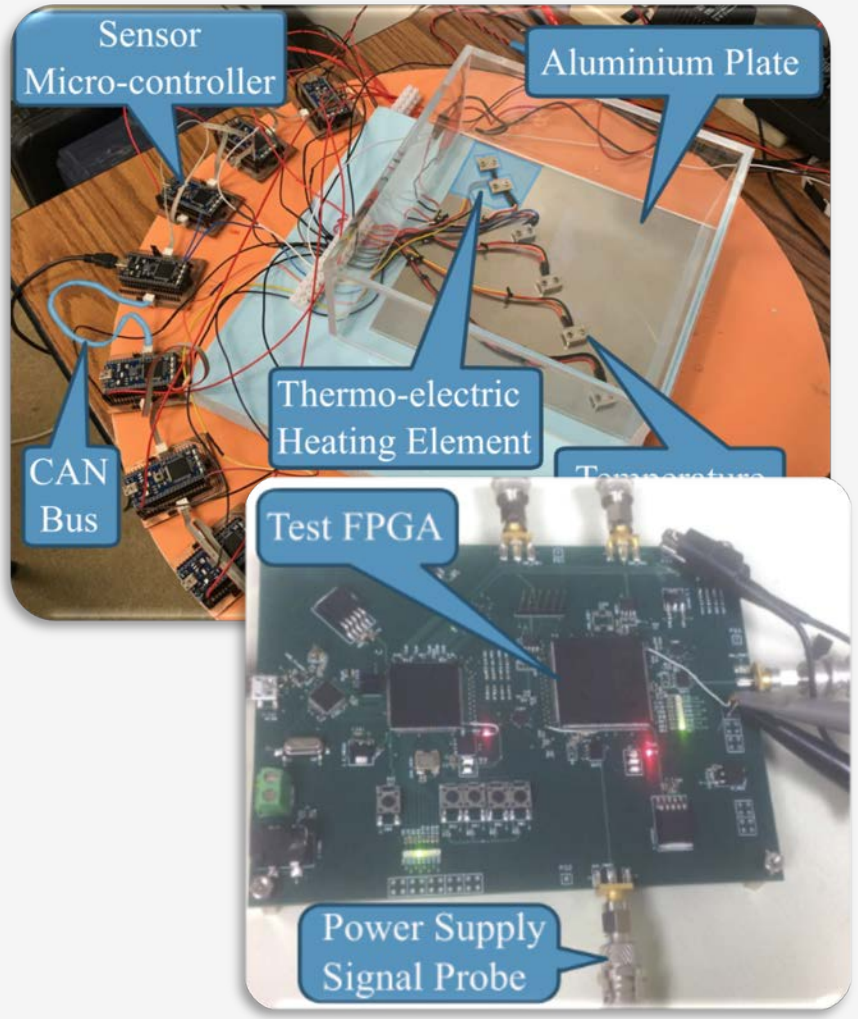


Trust-Based Consensus Block Diagram

Outline of the rest of the presentation

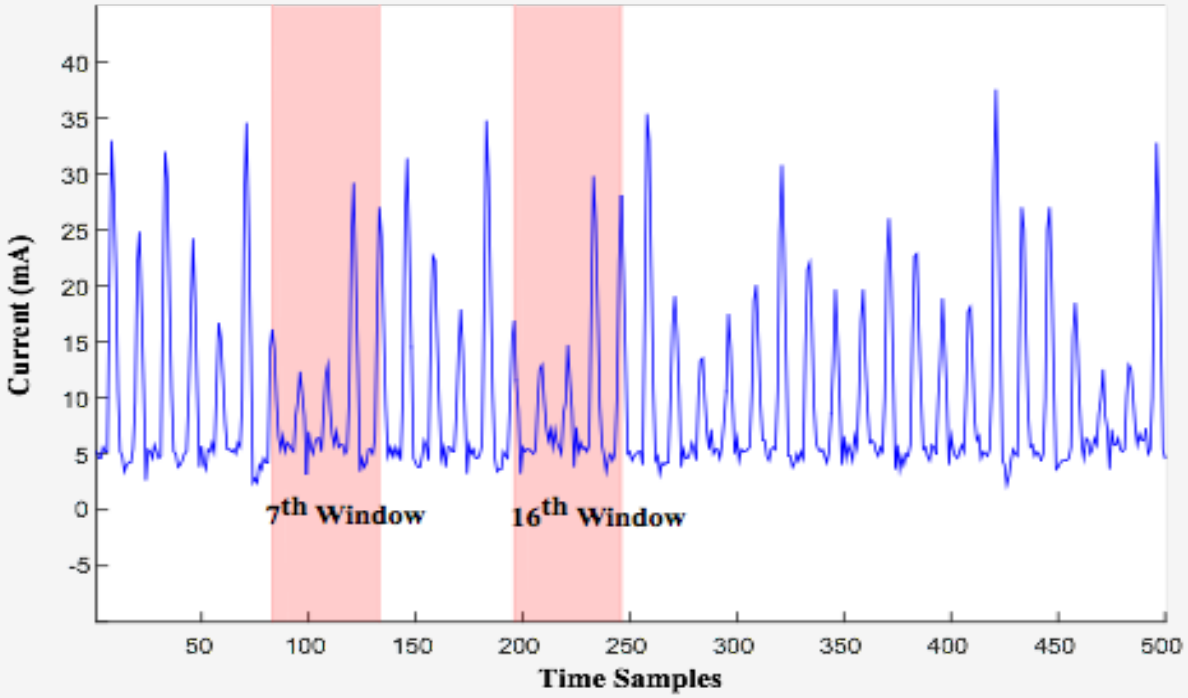
1. Micro-level detector
2. Macro-level consensus algorithm
3. Prototype implementation and evaluation





The micro-level detector is a side-channel monitor that uses power transients to determine if code on the sensors has been modified

#	Instruction Sequence	
-	mov #33, r15	;2 cycles
-	mov.*b* #-1, 0(r15)	;4 cycles
2	mov #304, r15	;2 cycles
4	mov.*b* -6(r4), r14	;3 cycles
7	mov r14, 0(r15)	;4 cycles
11	mov #312, r15	;2 cycles
13	mov.*b* -5(r4), r14	;3 cycles
16	mov r14, 0(r15)	;4 cycles
20	mov #314, r15	;2 cycles
22	mov @r15, -4(r4)	;5 cycles
27	inc.*b* -6(r4)	;4 cycles
31	inc.*b* -5(r4)	;4 cycles
35	mov #33, r15	;2 cycles
37	mov.*b* #0, 0(r15)	;4 cycles
-	jmp $-56	;2 cycles

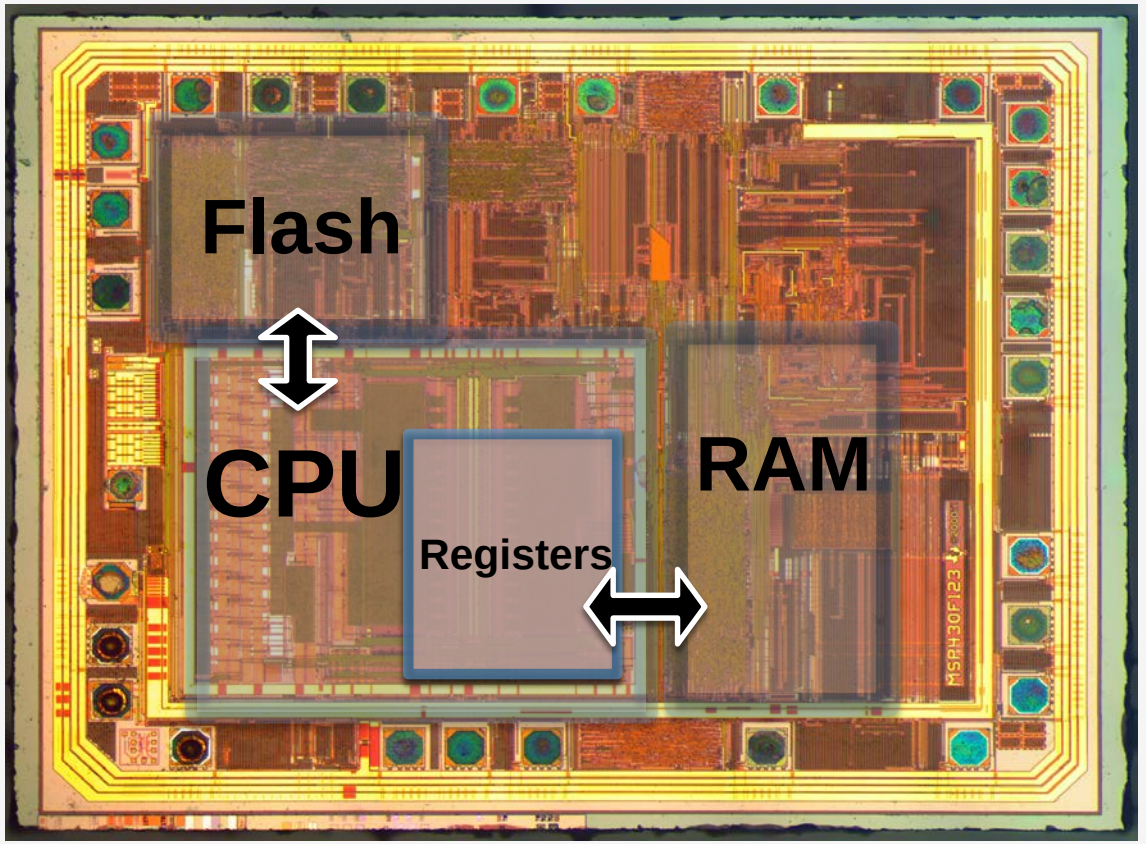


1. The power transient for instruction groups can be unique

Why does using power transients work for instruction identification?

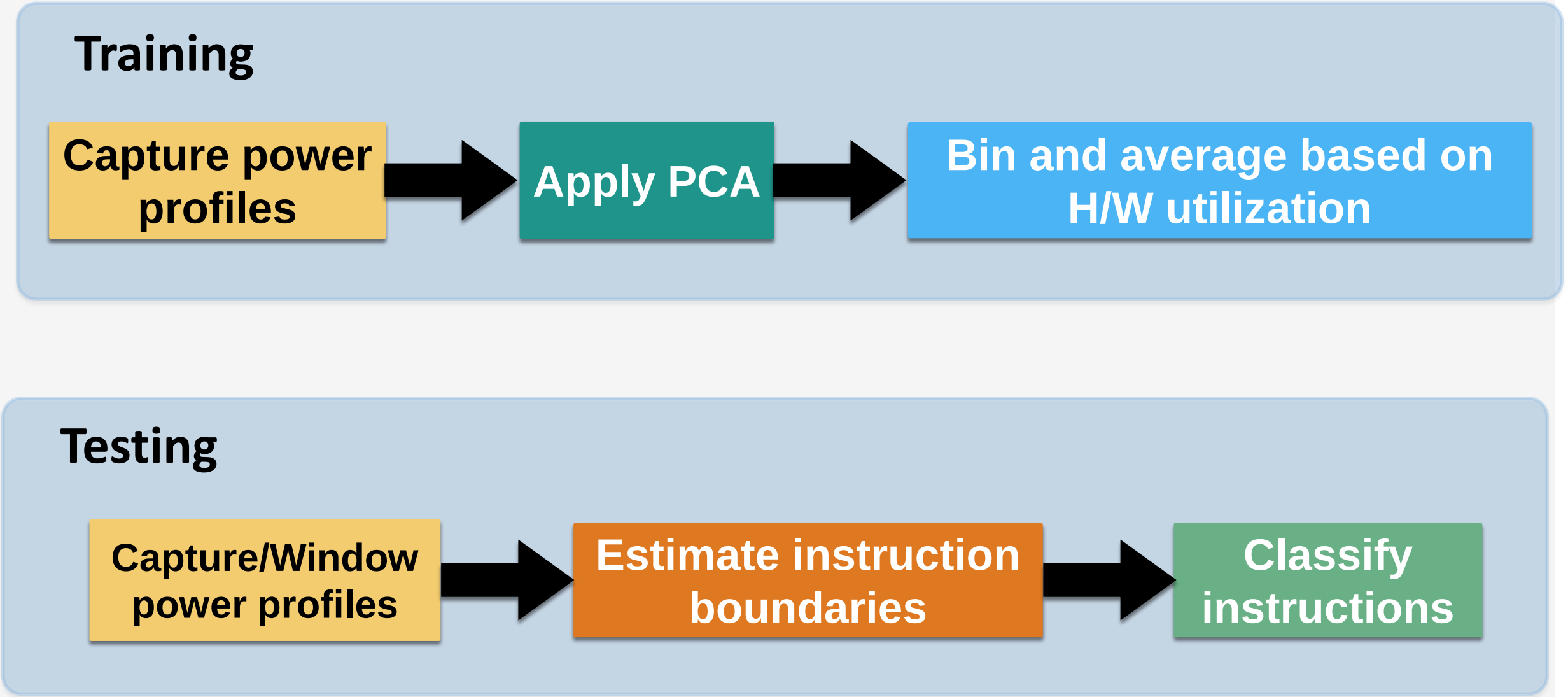
#	Label
1	reg_reg
2	mem_mem_sub
3	mem_mem_nosub
4	reg_const_ind_sub
5	reg_const_ind_nosub
6	ind_reg_sub
7	ind_reg_nosub
8	mem_reg
9	const_reg
10	imm_reg_sub
11	imm_reg_nosub
12	imm_ind_sub
13	imm_ind_nosub
14	other

reg_reg

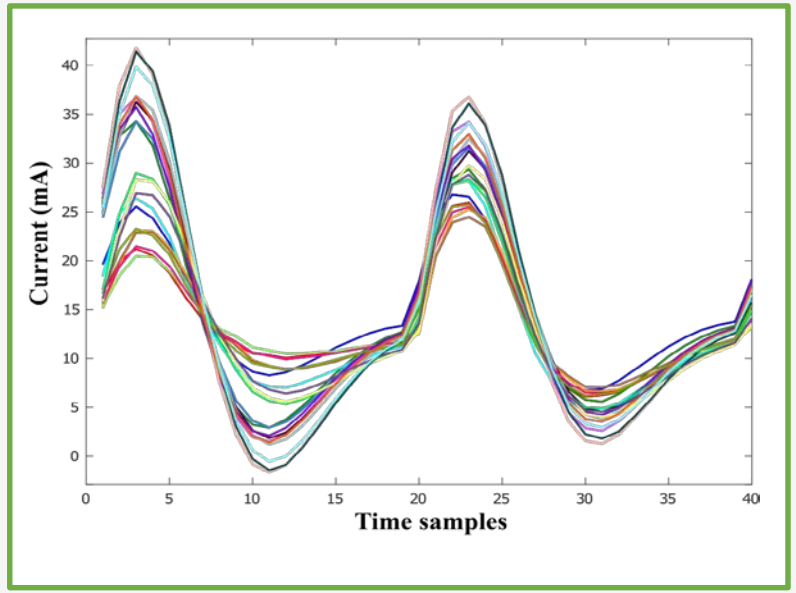


1. The power transient for an instruction group is unique since the hardware utilization for the instruction is unique

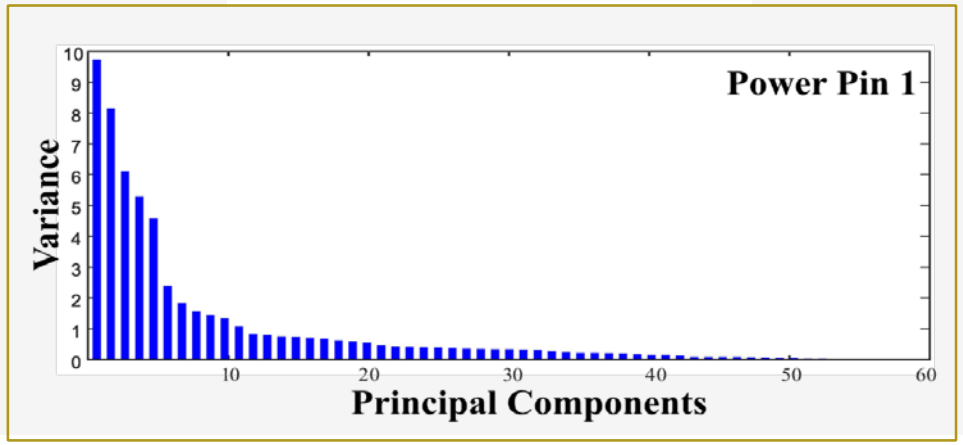
We use a learning approach to reverse engineer instruction sequences

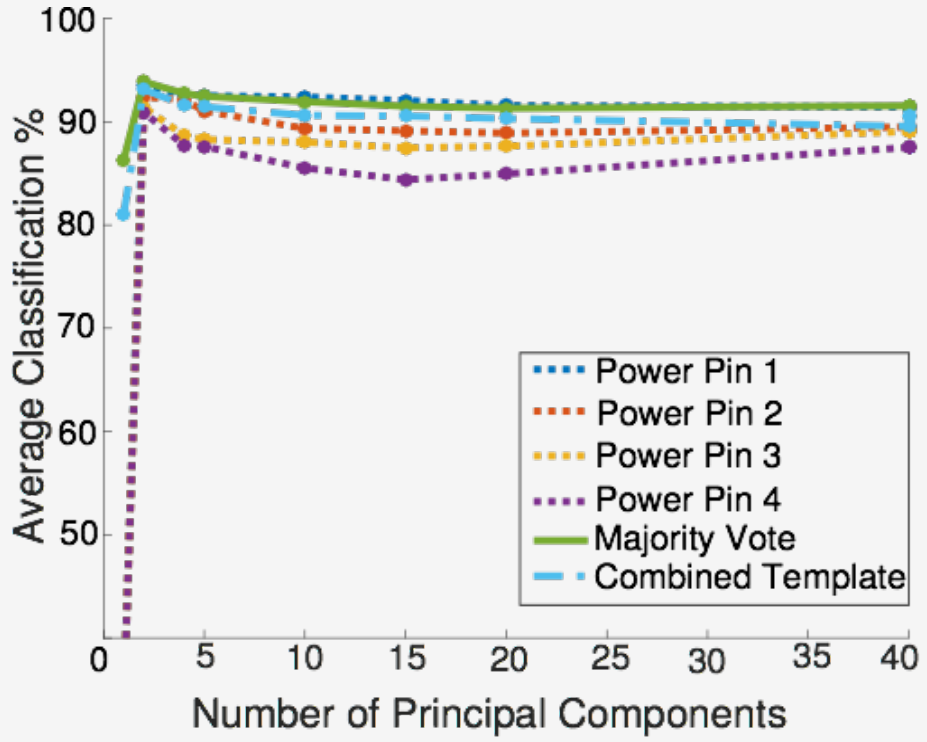


Our trained templates for each instruction class is based on the PCA components



2 Clock Cycle Templates



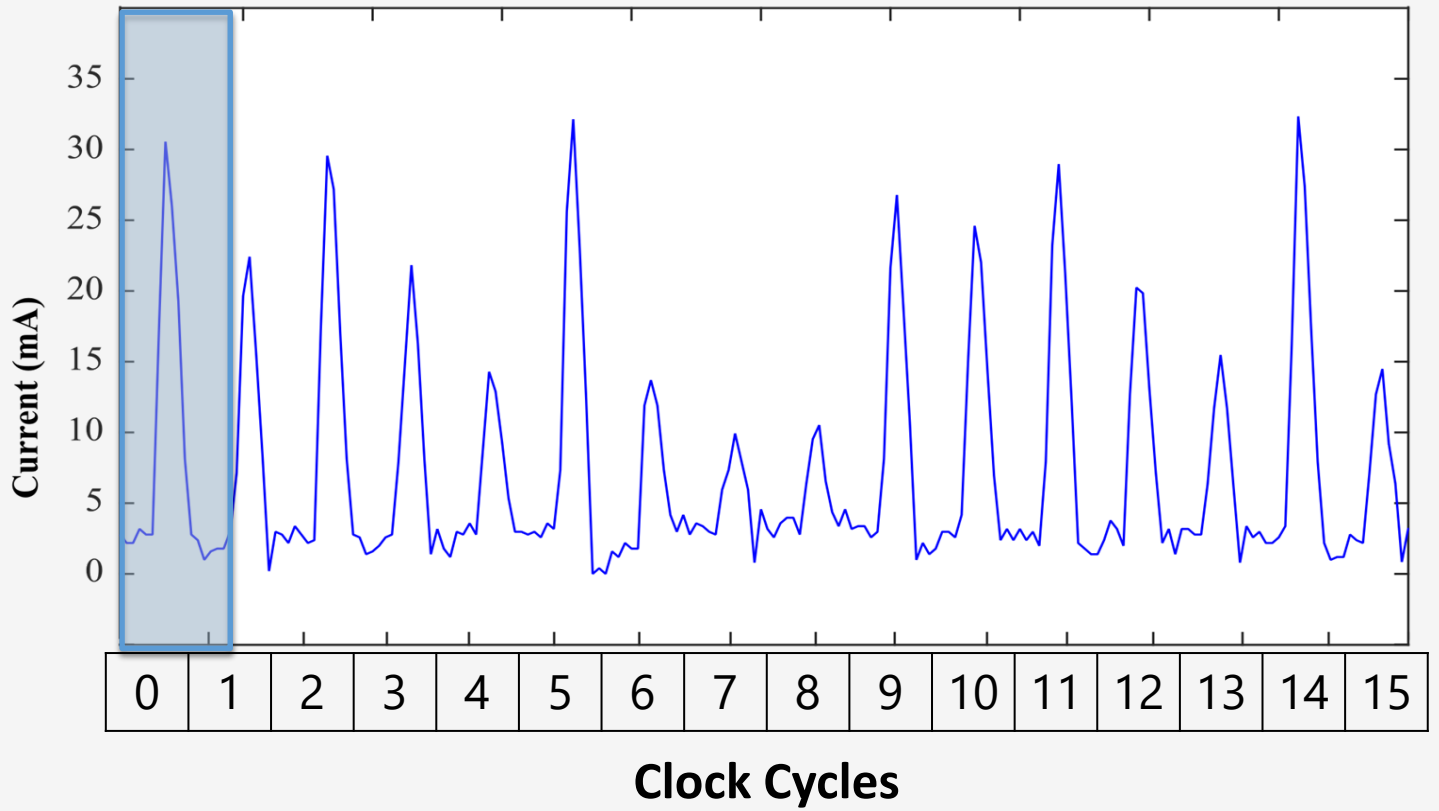


1. The accuracy of classification using **PCA and 1 Nearest Neighbor classifier** is close to 100% with only the top 10 PCA components

2. This leads to **considerable data reduction** over storing the raw power transient signatures/templates.

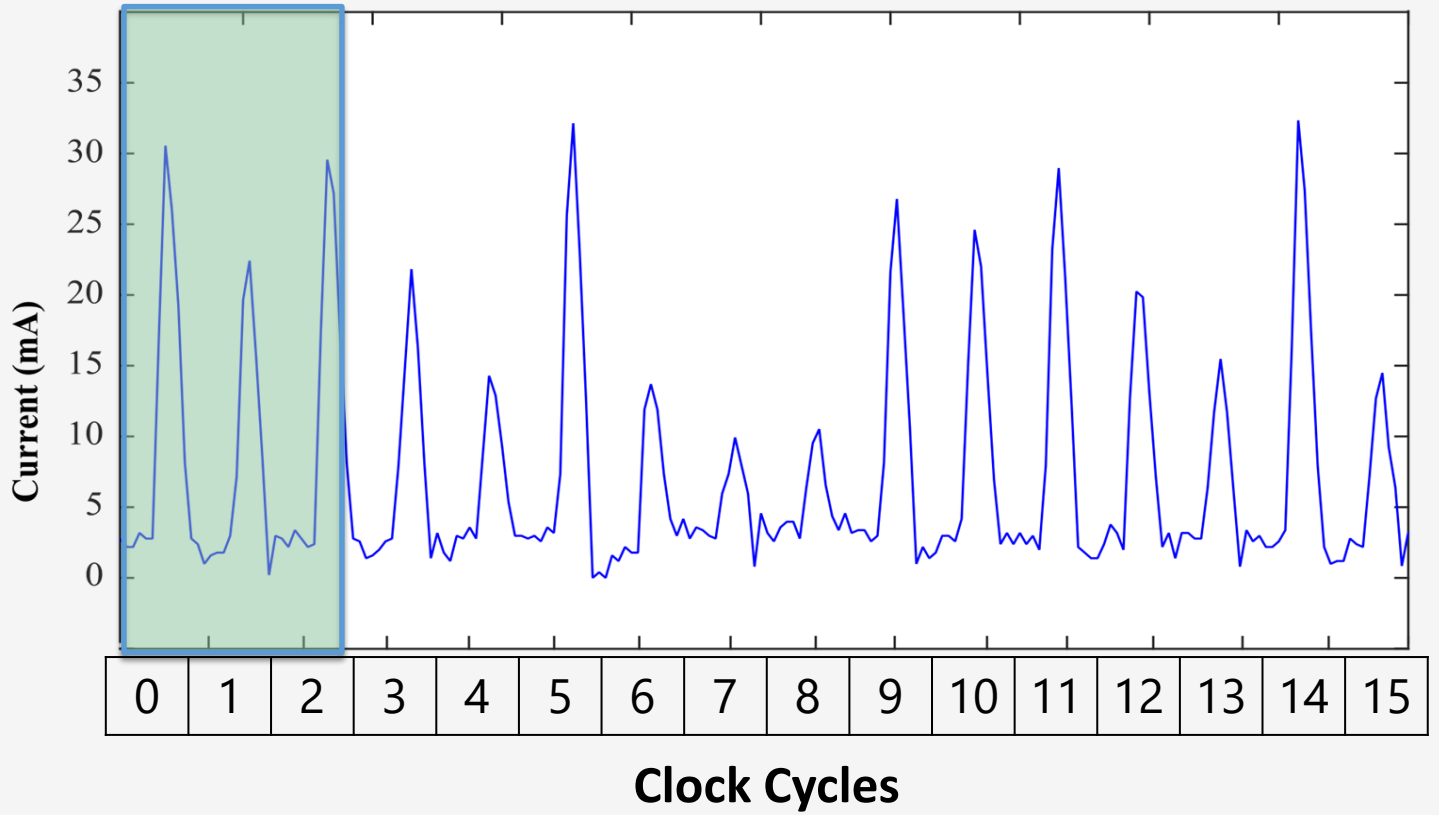
In the testing phase we use a dynamic programming solution to estimate the instruction sequence

Capture/Window power profiles → **Estimate instruction boundaries** → **Classify instructions**



In the testing phase we use a dynamic programming solution to estimate the instruction sequence

Capture/Window power profiles → **Estimate instruction boundaries** → **Classify instructions**



In the testing phase we use a dynamic programming solution to estimate the instruction sequence

Window Size	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	**5**	**4**	**6**	7	2	5	6	1	2	5	6	6	4	2	3
2	5	4	3	3	4	5	7	1	1	1	1	2	2	2	2
3															
4															
5															
6															

Min Distances

Apply **dynamic programming** to estimate the sequence of instructions

Example of instruction classification accuracy.

Instruction Sequence	Predicted Classification Rates			
	Power Pin 1	Power Pin 2	Power Pin 3	Power Pin 4
pop_mem_reg	88	90	93	89
add_mem_mem_nosub	85	84	87	75
inc_reg_const_ind_nosub	100	99	99	100
mov_mem_ind_nosub	98	98	96	98
add_reg_reg	99	99	99	99
sub_mem_mem_sub	92	87	88	82
dec_const_reg	99	99	99	99
mov_ind_reg_nosub	100	100	100	100
subc_imm_reg_sub	95	99	99	95
bit_mem_mem_nosub	96	97	97	97
cmp_mem_mem_sub	95	97	98	98
xor_reg_const_ind_nosub	56	46	67	56
inc_const_reg	99	99	99	99
	92	**81**	**94**	**91**

1

The accuracy of determining the sequence of instructions is close to 94%

2

Certain power pins show better accuracy that others.

3

Majority vote does not always lead to better results

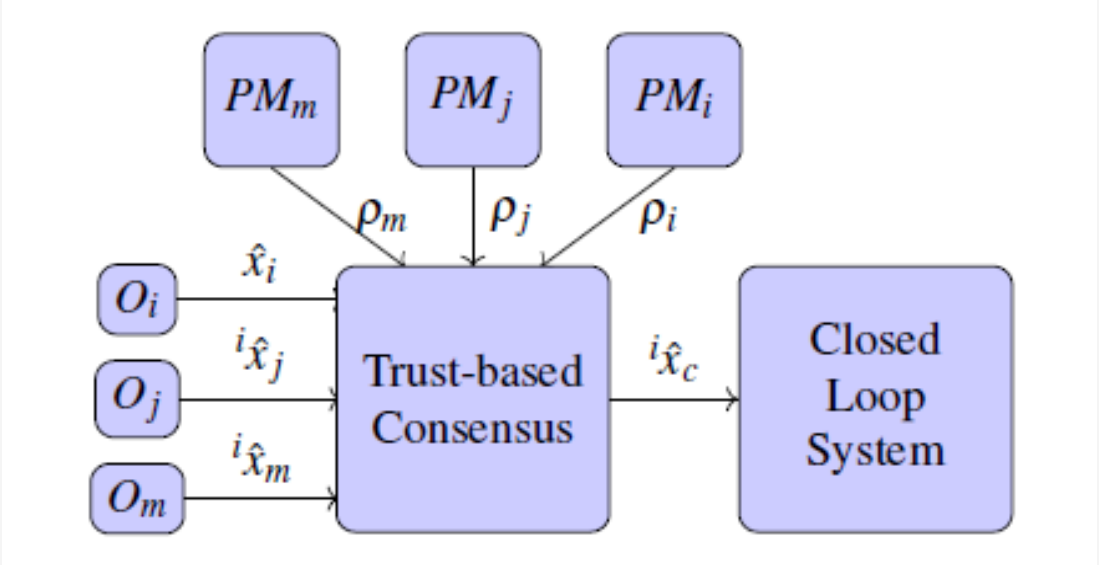
How does this tie to the cross-layer estimation of anomalies?

1. The power monitors determine the degree to which the firmware running on the sensor is different from the one that is supposed to run

2. Based on the above, the values p_m (trustworthiness is estimated)

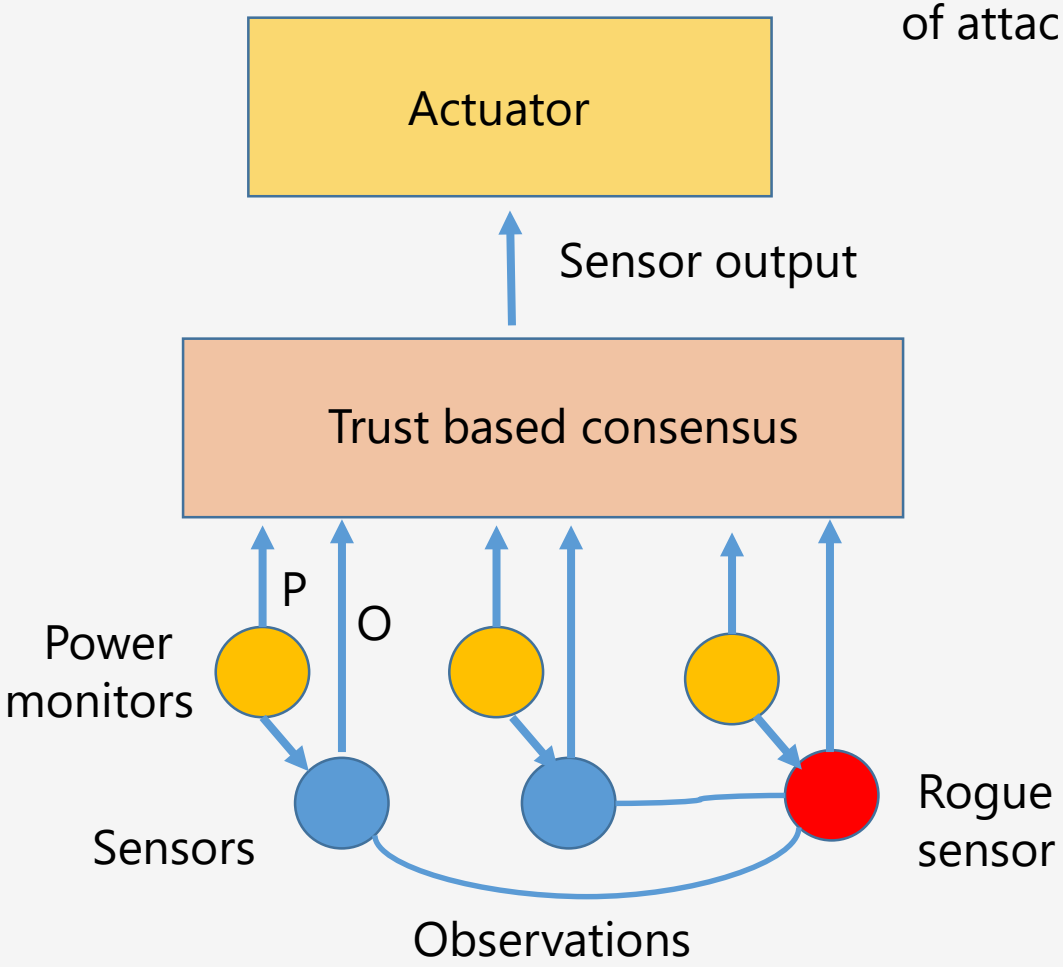
3. The observations for sensor j at sensor i are values that sensor i predicts should be the value sensor j

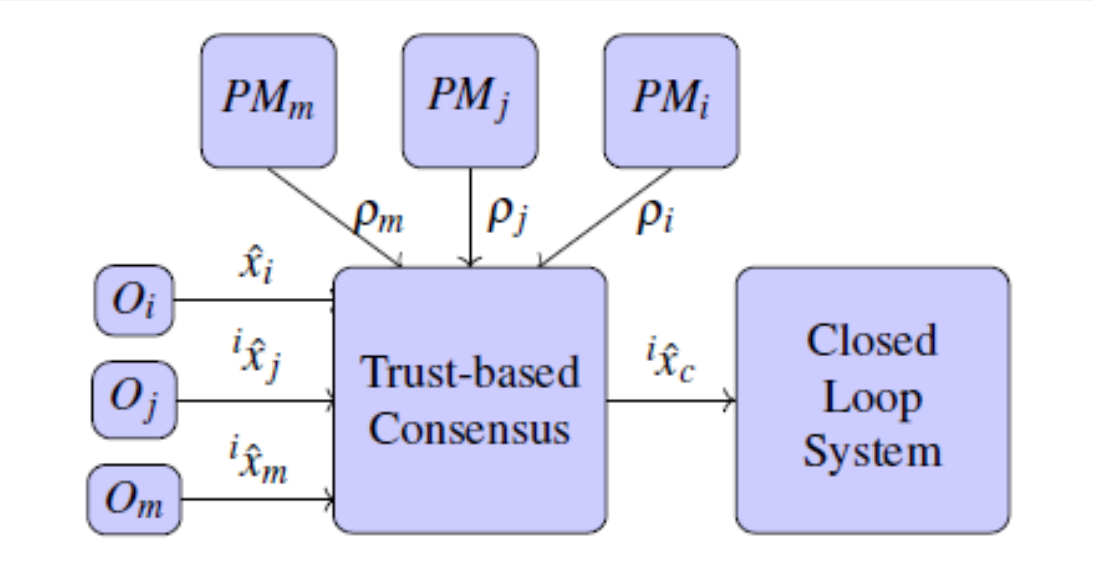
4. System runs a trust-based consensus to estimate the value at sensor j



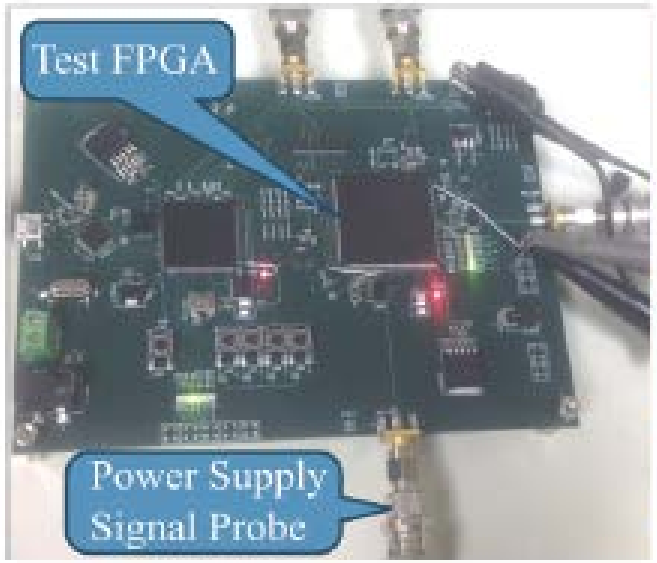
A example to illustrate how this works

System can operate in the presence of attacks

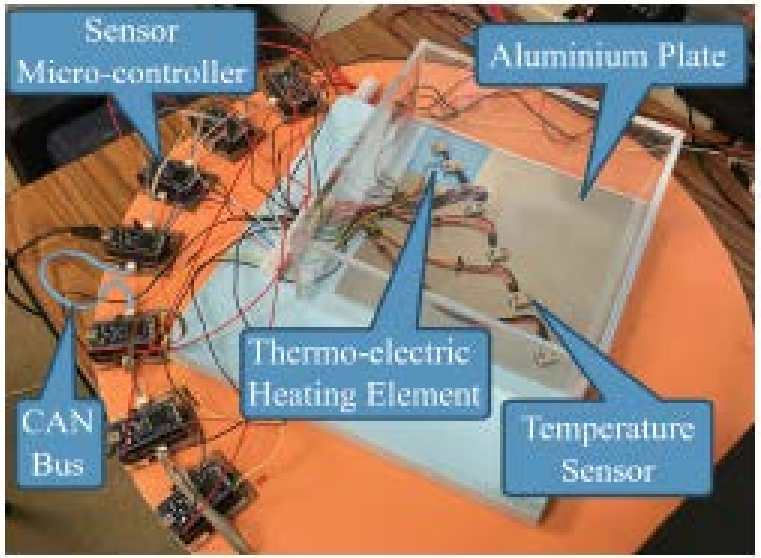




Our experimental setup comprise of temperature sensor testbed



(a) Micro-level Analysis Setup



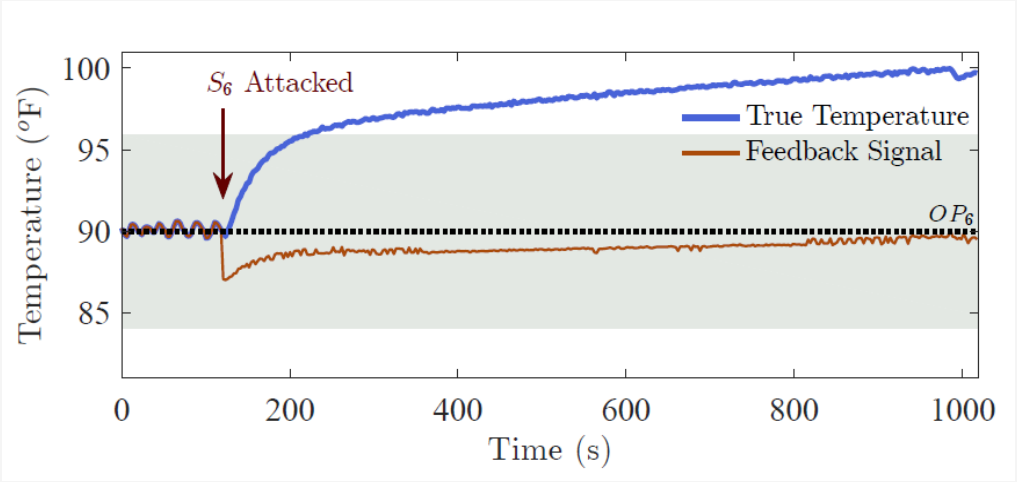
(b) Temperature Control Testbed

SCENARIO TIMELINE

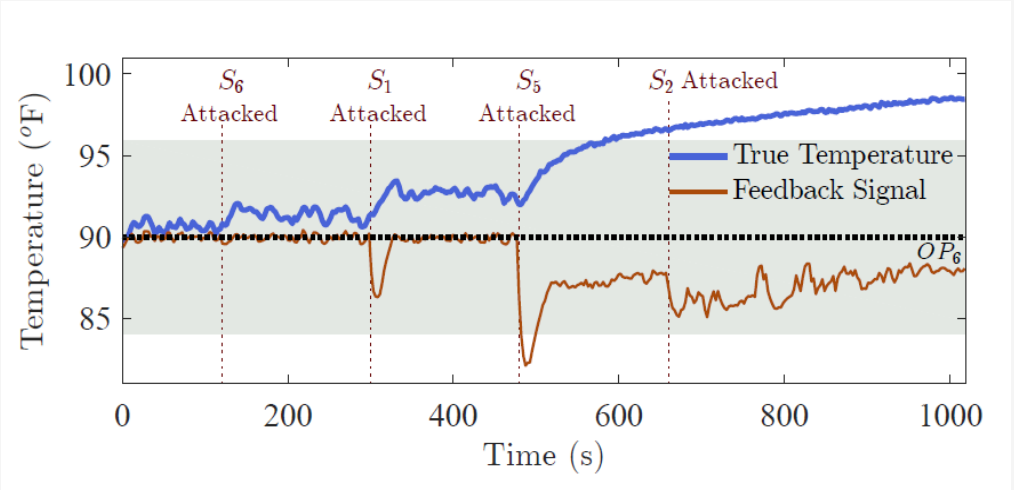
Launch Time (s)	Attack Scenario		
	Sequence # 1	Sequence #2	Sequence #3
120	S_6	S_6	S_6
300	S_1	S_3	S_4
480	S_5	S_5	S_2
660	S_2	S_1	S_3

1. Emulate attack scenarios

If only macro-level inference is used, system is unstable after some time

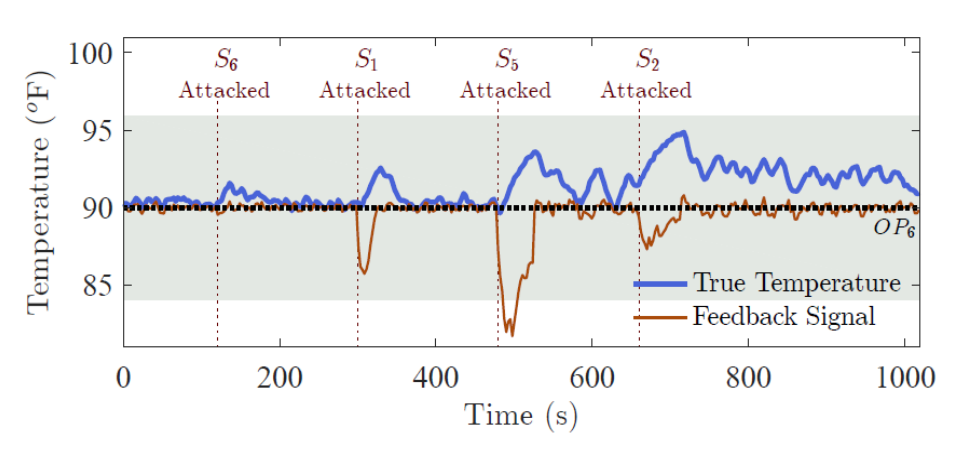


Unstable system, effect of attack if no action taken. Here false temperature is used to close the loop and quickly exits the safe region (gray ±6° F).

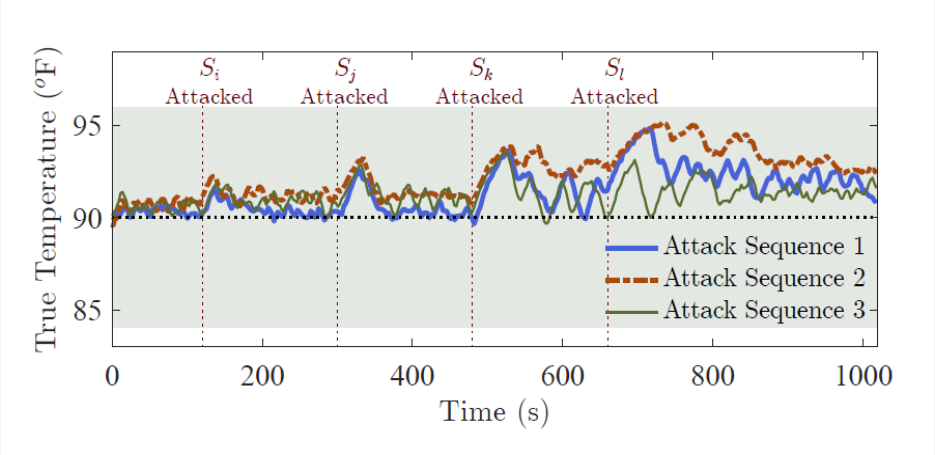


If the consensus value of all six sensors is used without any micro data, once the third sensor is attacked then the temperature control becomes unstable and exits the safe region.

The cross-layer approach can tolerate a large number of compromised sensors



Here the micro-level measurements are used to weight the consensus calculations, even if four sensors are compromised, the temperature stays within the safe region.



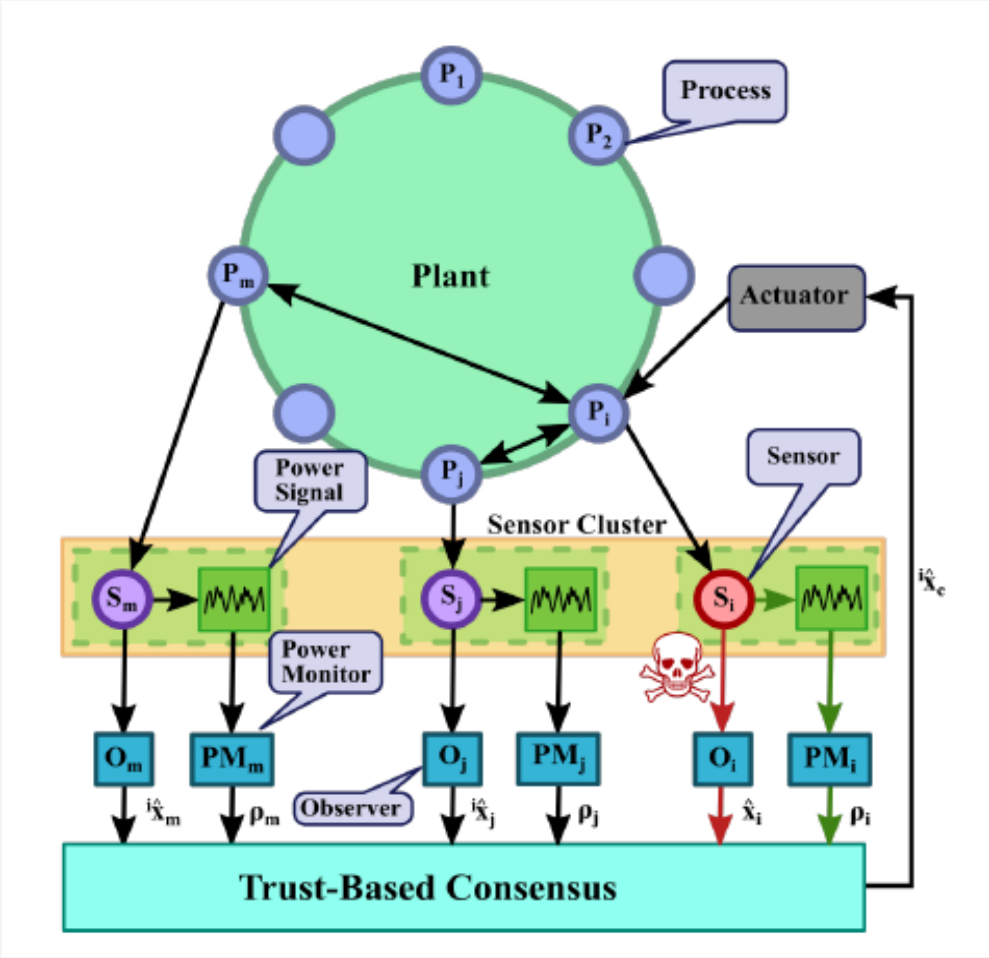
Shows three different experiments all using the weighted trust-based consensus, Here different combinations of sensors were attacked at the same intervals. They all show similar acceptable behavior.

Future work

1. Instruction classification is a still an open problem. Being able to classify individual instructions is complicated

2. Finding differences in code instruction sequences while being cognizant of semantic information in code is challenging

3. Scalability of the system with hundreds of sensors need to be evaluated.



Cyber-Physical Systems (CPS) Security

A COLLABORATION BETWEEN UMBC AND UNITED STATES NAVAL ACADEMY (USNA)

UMBC Collaborators –
Prof. Ryan Robucci, Prof. Chintan Patel,
Prof. Nilanjan Banerjee, Prof. Anupam Joshi
Ph.D. Student Collaborators –
Deepak Krishnankutty, Brien Croteau, Zheng Li

USNA Collaborators –
Prof. Kiriakos Kirikadis, Prof. Tracie Severson,
Prof. Erick Rodriguez-Seda

This work has been supported in part by the U.S. Office of Naval Research under Awards N00014-15-1-2179 and N0001417WX01442

Publications

- Brien Croteau, Deepak Krishnankutty, Kiriakos Kiriakidis, Tracie Severson· Chintan Patel,·Ryan Robucci,·Erick Rodriguez-Seda,·Nilanjan Banerjee "Cross-level Detection Framework for Attacks on Cyber-Physical Systems" Journal of Hardware and Systems Security, Springer International Publishing, Accepted: 10 November 2017, http://dx.doi.org/10.1007/s41635-017-0027-9
- Croteau B, Krishnankutty D, Robucci R, Patel C, Banerjee N, Kiriakidis K, Severson T, Rodriguez-Seda E "Cross-level detection of sensor-based deception attacks on cyber-physical systems." Proceedings of the 7th Annual IEEE International Conference on CYBER Technology in Autonomous, Control, and Intelligent System (2017)
- Krishnankutty D, Robucci R, Banerjee N, Patel C FISCAL: firmware identification using side-channel power analysis." IEEE 35th VLSI Test Symposium (VTS), pp 1–6.997 (2017) https://doi.org/10.1109/VTS.2017.7928948998
- Tracie Severson, Erick J. Rodríguez-Seda, Brien Croteau*, Deepak Krishnankutty, Kiriakos Kiriakidis, Chintan Patel, Nilanjan Banerjee, Ryan Robucci
  Trust-Based Framework for Resilience to Sensor-Targeted Attacks in Cyber-Physical Systems
  Conference: 2018 American Control Conference (accepted for publication)
- Croteau, C. D. R. Brien; Deepak Krishnankutty,. "Cyber-Physical Security Research at Umbc's Eclipse Lab." Mechanical Engineering-CIME. 2017. *HighBeam Research*. (January 23, 2018). https://www.highbeam.com/doc/1P3-4322053665.html

Cross-Layer Detection of Sensor-based Deception Attacks on Cyber-Physical Systems

Nilanjan Banerjee

Associate Professor, University of Maryland, Baltimore County

(http://www.csee.umbc.edu/~nilanb)