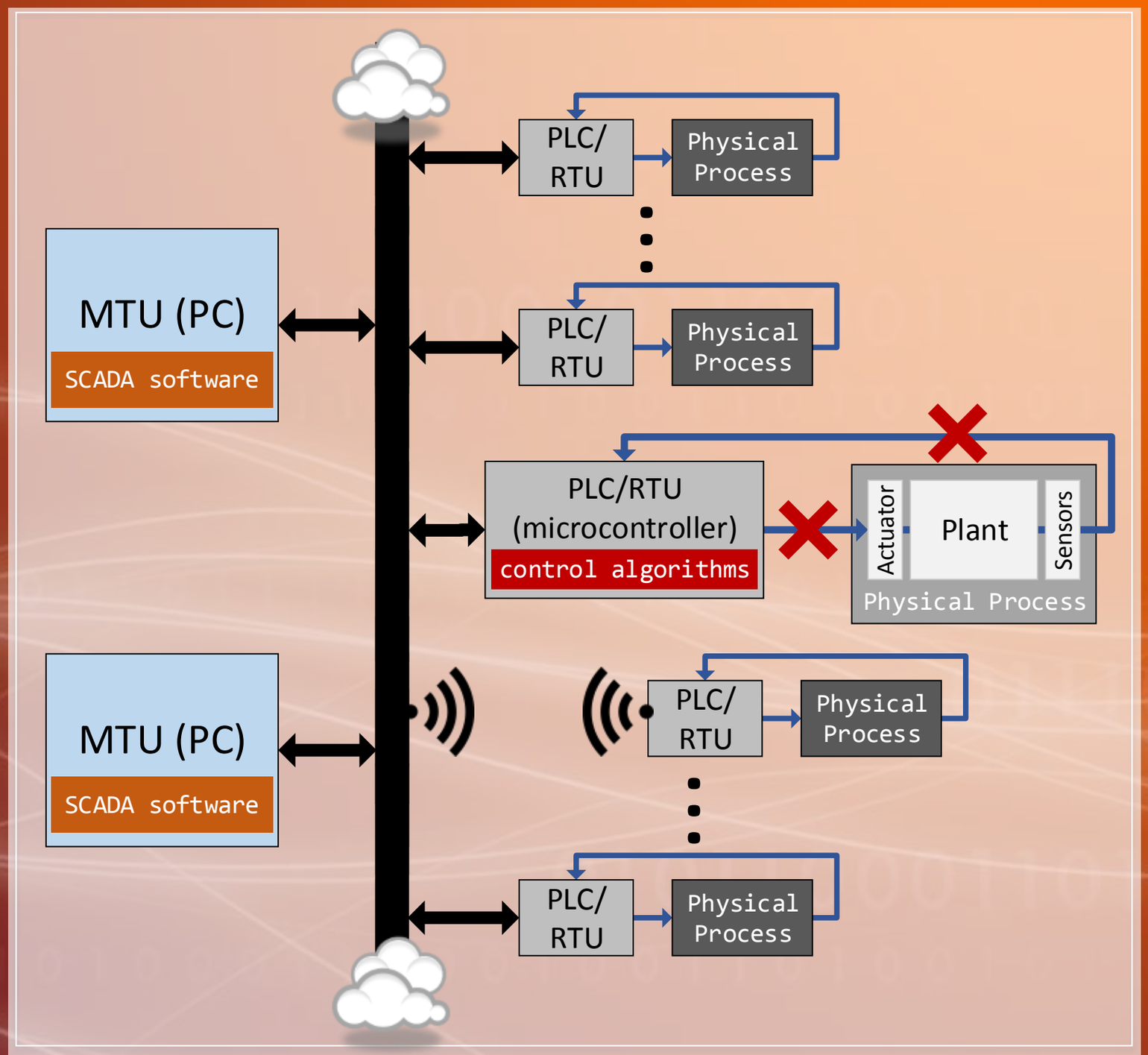
Using High-level Synthesis and Formal Analysis to Predict and Preempt Attacks on Industrial Control Systems

Lee W. Lerner
Zane R. Franklin
William T. Baumann
Cameron D. Patterson

BRADLEY DEPARTMENT of ELECTRICAL & COMPUTER ENGINEERING

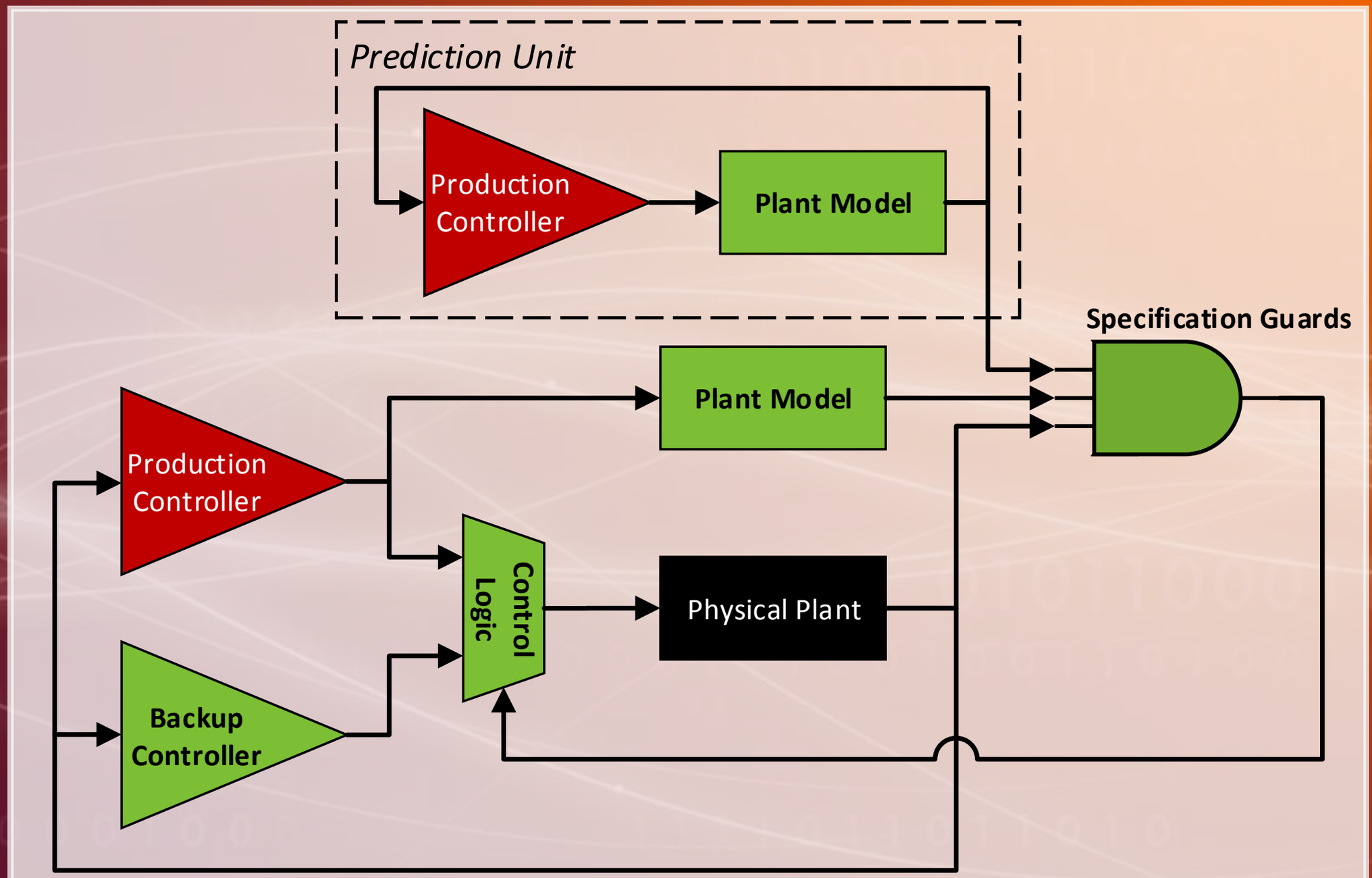
Distributed Control System

- Control loops
  - Control algorithms on microcontroller
  - Physical processes
- Network
  - 100s-1000s control loops
  - Supervised by PCs (SCADA)
- Vulnerabilities
  - Lack of controller security (STUXNET)
  - Malicious software updates
  - Interrupted/malicious actuator and sensor data



Abstract Architecture

- **T**rust **E**hancement of **C**ritical **E**mbedded **P**rocesses (TECEP)
  - Last line of defense for compromised systems
- Prediction Unit: Virtual control loop for speculative execution
- Zynq-7000 SoC: Trusted components isolated in programmable hardware



Formal Verification

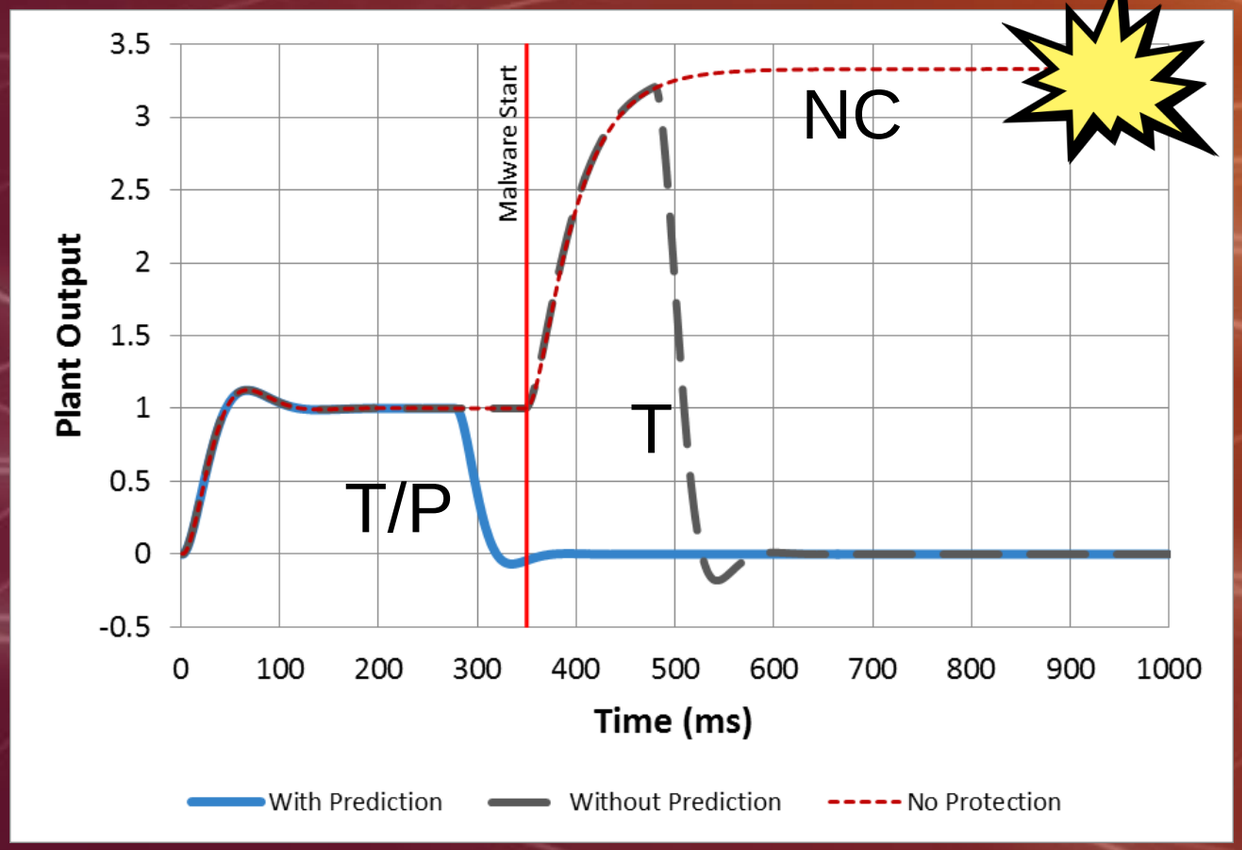
- HLS permits the use of software verification tools on hardware-implemented components
- Preferable to capturing TECEP components in HDL, model checking
- Compatibility of C syntax and semantics for HLS, Frama-C
  - No loops or complex optimization
- Frama-C:
  - Framework of collaborative static analysis techniques

```
behavior verify_any_invalid:
 assumes
    y_physical < y_min ||
    y_physical > y_max ||
    ghost_y_model < y_min ||
    ghost_y_model > y_max ||
    y_accel < y_min ||
    y_accel > y_max;
 ensures \result == ghost_u_hw;
```

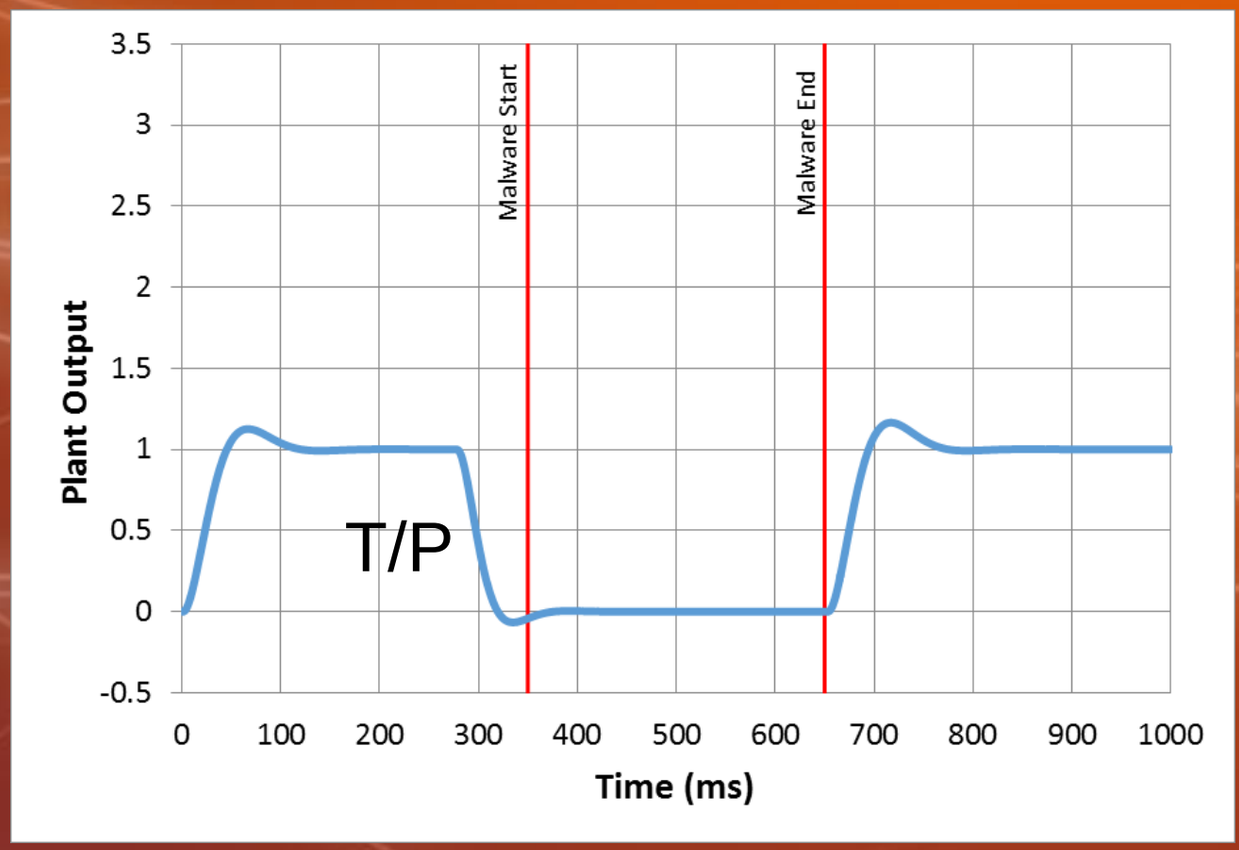
Proof Annotation	Proof Obligations	Behavior Description
`verify_all_valid`	96	Backup not triggered when all units are within spec; production controller output selected
`verify_any_invalid`	96	Backup always triggered when any unit is out of spec; backup controller output selected
`disjoint behaviors`	1	Ensure behavioral proofs are disjoint
`complete behavior`	1	Ensure behavioral proofs are complete

Motor Controller with TECEP

- Latent malicious behavior in production controller software
- Predicted plant behavior allows proactive security measures
- *Left*: Plant output with no countermeasures(NC), TECEP(T), TECEP and prediction(T/P)
- *Right*: TECEP and prediction; resume normal operation after attack ends (DoS)



Stabilization Only



Return to Production Controller

Conclusions

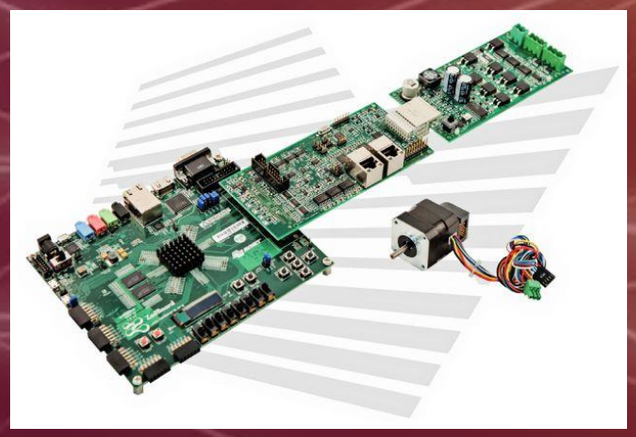
TECEP

- Assumes firewalls can be bypassed, OS compromised, supervisors misled
- Last line of defense for stabilizing a plant under attack
- Targeting hardware reduces vulnerability to malicious software attacks
- Software design/verification flow for hardware components via HLS
- Preserving plant model enables malware prediction and preemption

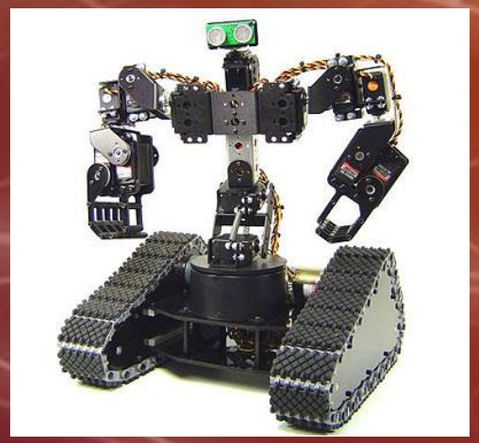
Continuing Work

- Secure updates to spec guards
- Focus on experimental results

Zynq-7020	FF	LUT	DSP	BRAM
Hardware Monitor	677	1046	5	0
HLS AXI Interface	295	78	0	0
Junction Box	70	80	0	0
Prediction Unit	2813	3174	2	4
Total Used	3855	4378	7	4
Available	53200	106400	220	140
Percent Used	7%	4%	5%	2%



GTRI's ICS testbed



Application to robotics

VirginiaTech
Invent the Future

FPGA 2014

This material is based on work supported by the National Science Foundation under Grant Number CNS-1222656. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. ZedBoards and tools were donated by Xilinx, Inc

ece
BRADLEY DEPARTMENT
of ELECTRICAL & COMPUTER ENGINEERING