SMART: Secure and Minimal Architecture for Establishing a Dynamic Root of Trust

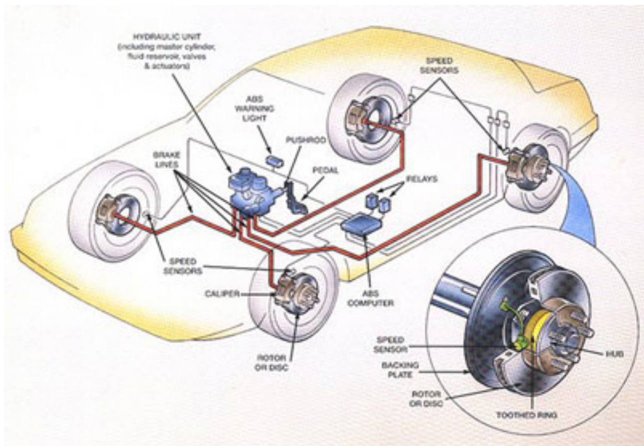
Karim El Defrawy, Aurelien Francillon, Daniele Perito, Gene Tsudik
UCI ETH INRIA UCI

Feb 8 2012

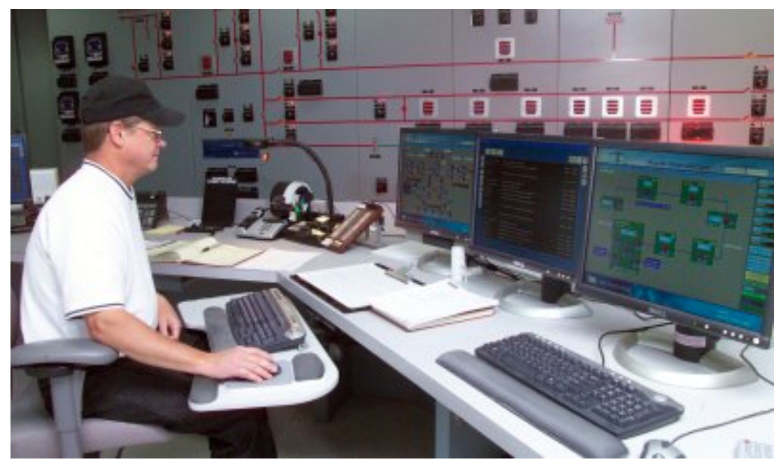
Low-end Embedded Devices

- Low cost, low power devices
- Built around a Micro-controller Unit (MCU)
- Limited:
  - Memory
    - 4 KB Data Memory (SRAM)
    - 128KB Program Memory (Flash)
  - Power
  - computation capabilities
- For example
  - MSP430
  - AVR

Critical Cyber-physical Systems



“A cyber-physical system (CPS) is a system where there is tight coordination of the system’s computational and physical elements, though sensors and actuators”



Why Security Now?

- Cyber-physical systems are built to be reliable
- Security was treated as an afterthought
- Acceptable with very limited connectivity

And

- Ease of management is pushing wireless connectivity
  - Implantable medical devices can be accessed via home readers through an RF channel
  - In car systems are connected via wireless
- Indirectly connected to the Internet

Recent Attacks

- Stuxnet [1]
  - Infected controlling windows machines
  - Changed parameters of the PLC of the centrifuges of Iranian nuclear reactors
- Attacks against automotive controllers [2]
  - Internal controller-area network (CAN)
  - Exploiting one subsystem (e.g., bluetooth) allows access to critical subsystems (e.g., braking)
- Medical devices
  - Insulin pumps hack [3]
  - Implantable cardioverter defibrillator [4]

[1] W32.Stuxnet Dossier. Nicolas Falliere, Liam O Murchu and Eric Chien. Symantec 2011
[2] Comprehensive Experimental Analyses of Automotive Attack Surfaces. S. Checkoway et al. USENIX 2011
[3] Hacking Medical Devices for Fun and Insulin: Breaking the Human SCADA System. Jerome Radcliffe. Blackhat 2011
[4] Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses. S&P 2008

Remote Attestation

Definitions

- Two party protocol between trusted verifier and untrusted prover
- Remotely verify the internal state of the prover

Where

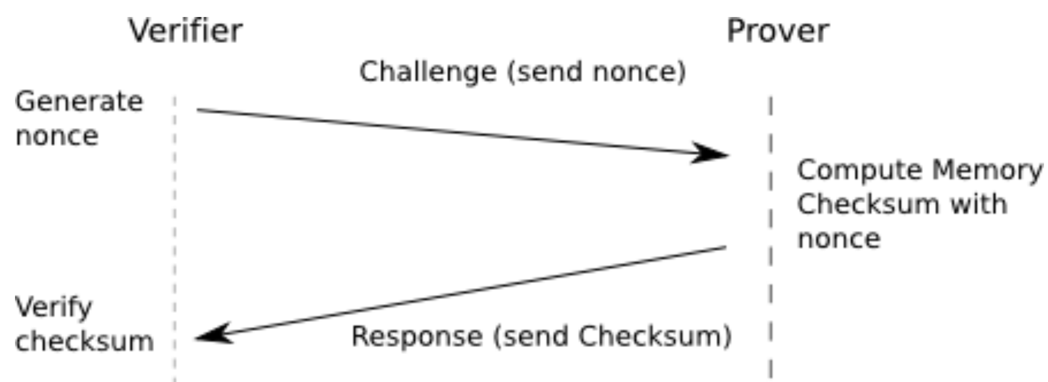
- Prover is the untrusted embedded device
- Verifier is the trusted reader/controller/base station
- Internal state is composed
  - Code
  - Registers
  - Data Memory
  - I/O

Two types of attestation:

- Secure Hardware supported (e.g., TPM)
- Software attestation
  - Does not support multi-hop communication

Remote attestation

- Malicious software will lie about the software state of the prover
- Need to have guarantees that the device is not lying



SMART: Secure and Minimal Architecture for a Root of Trust

Motivation:

- Existing solutions (TPM) are expensive for embedded devices
- What is a minimal set of architectural features to achieve remote attestation?

Desirable features:

- Minimal modifications to existing platforms
  - Fewest additional gates
- Security under a strong attacker model
- Portable to multiple platforms
  - Implemented on AVR and MSP430

Security Goals

Establish a dynamic root of trust on the prover

- "Guarantee untampered execution of a target piece of code, even in the presence of a corrupted platform"

In particular

- Prover authentication
  - Are we are talking with the right prover?
- External verification
  - Do we know the internal state of the prover?
- Guaranteed execution
  - Do we know the execution state?

No tamper resistance/no hardware attacks

Great. How do we do that?

Building Blocks

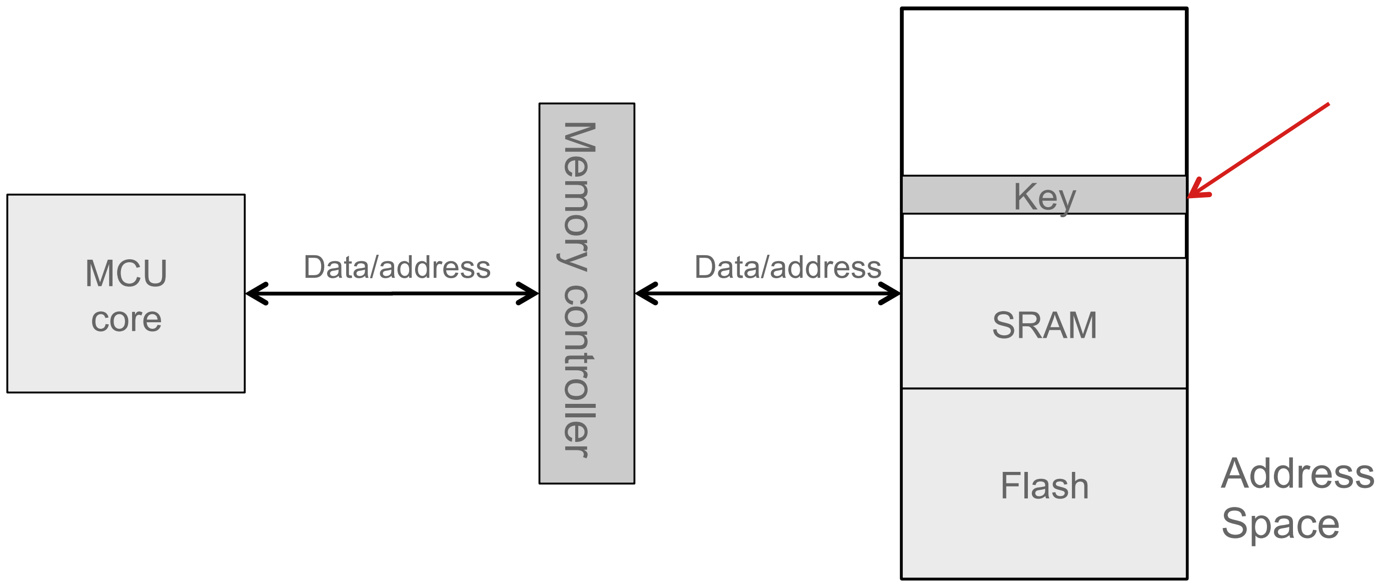
- Secure Key Storage
  - Required for multi-hop authentication
  - Provides prover authentication
- Trusted ROM code memory region
  - Read-only means integrity
  - Accesses and operates on key
- MCU access controls
  - Grants access to key to Trusted ROM

Key storage

- Provides remote prover authentication
- The key cannot be stored in normal memory
  - Malware would steal it
- Need to protect key access

Our approach

- Only a trusted code region can access the key

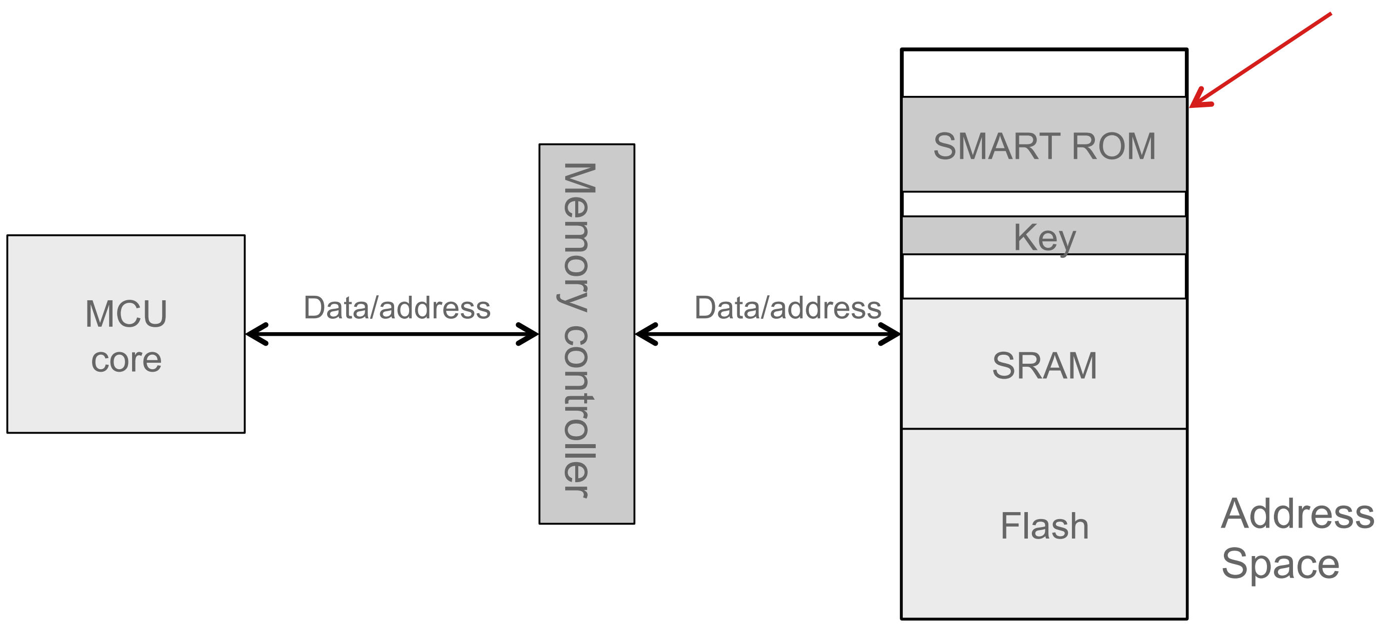


Trusted code region

- Low-end embedded devices do not have support for rings to restrict access to memory
- Adding those would require significant complexity

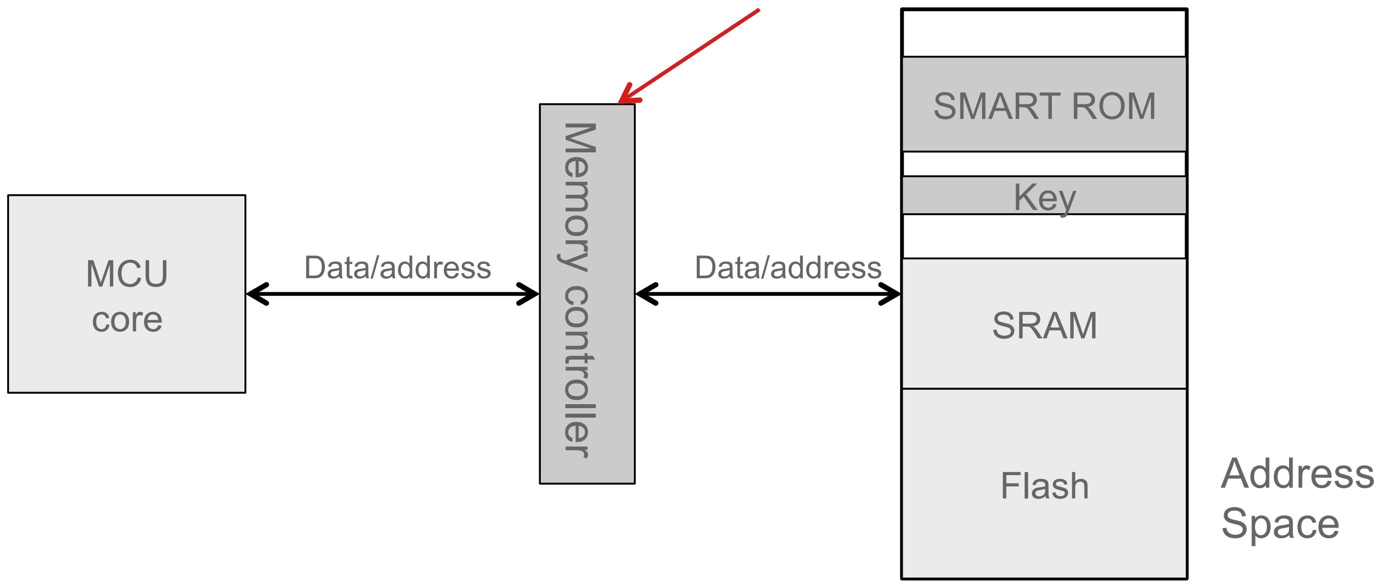
Our approach

- Restrict access to a read-only trusted code region
- Access control can be implemented easily

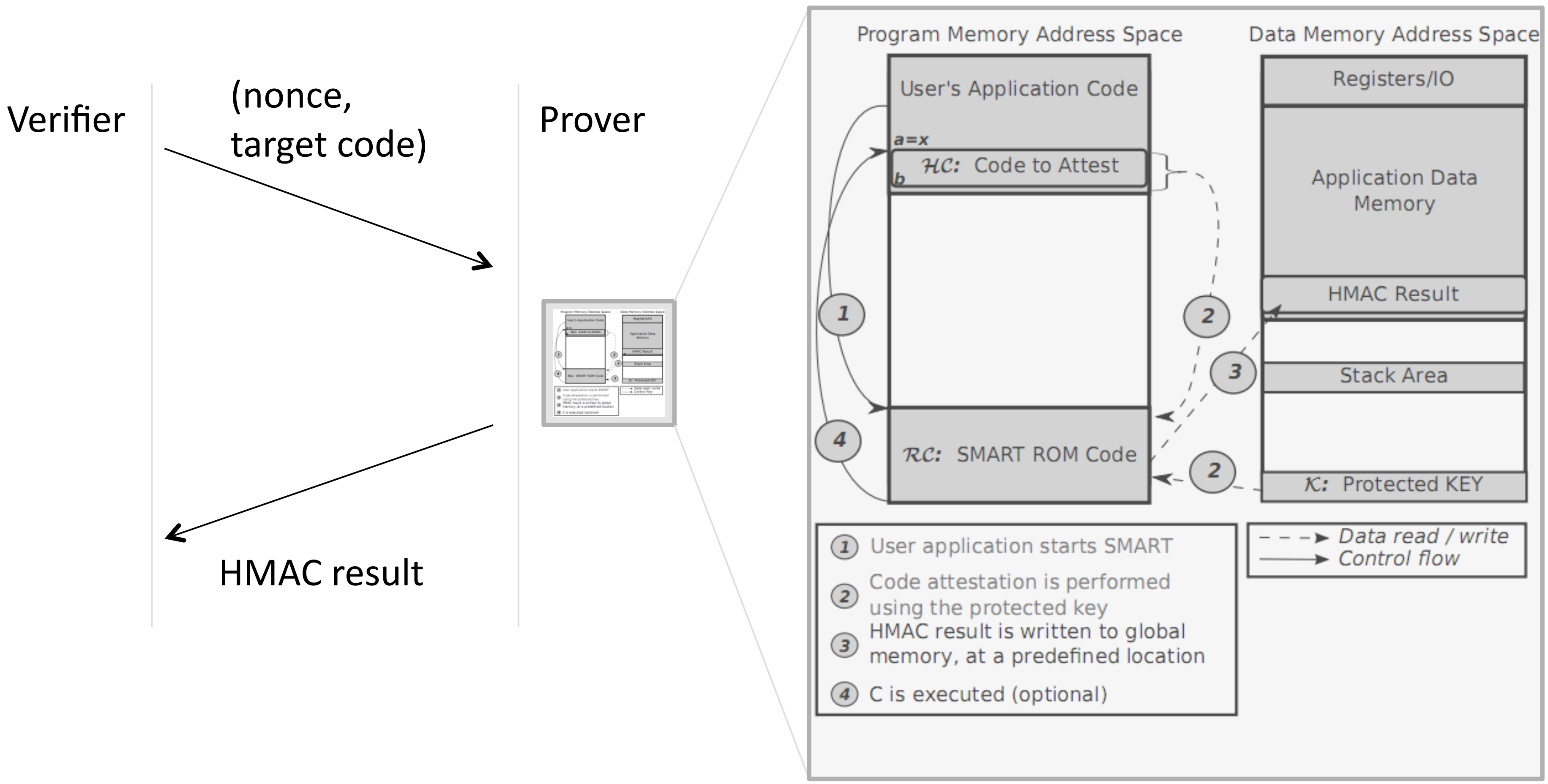


Memory access control

- Only ROM code must be able to access the key
- Control the program counter value



The complete protocol



Problems to solve

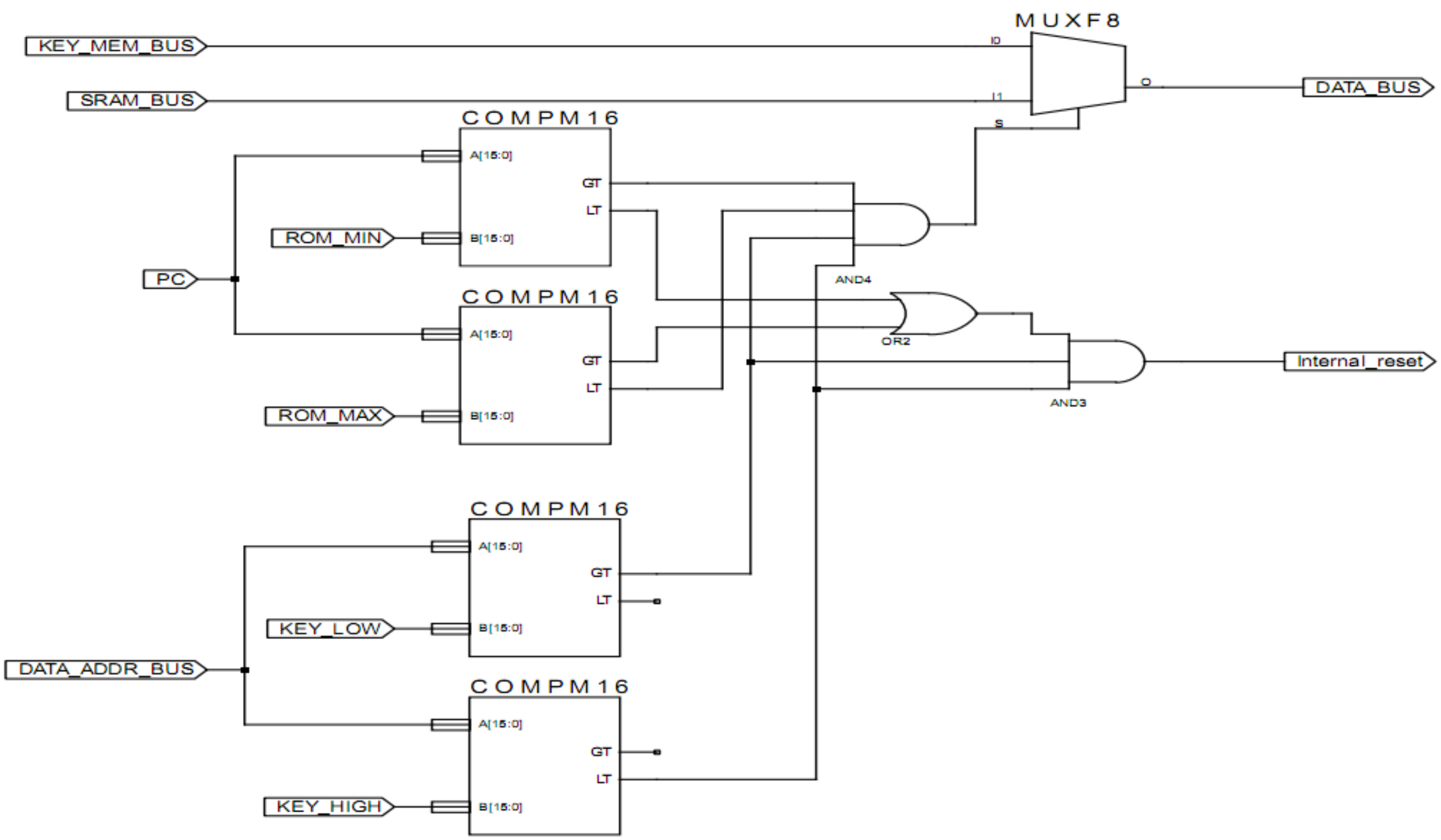
Trusted ROM code and malware share the same resources

- Malware can set up the environment of the execution to compromise trusted code and extract the key
- Interrupts can asynchronously execute while a copy of the key is in main memory
- Malware can use code gadgets in ROM to access the key
  - Return oriented programming
- ROM code might leave traces of the key in memory after execution

Counter Measures

- Atomic ROM code execution
  - Enforce in hardware
  - Enter at first instruction
  - Exit at last instruction
- ROM code is instrumented to check for memory safety
  - Upon detecting error reboot and clean memory
- Interrupts are disabled immediately
  - Before key usage
- Erase key material before end of execution

Schematics



MUXF8
KEY_MEM_BUS
I0
SRAM_BUS
I1
O
DATA_BUS
S
COMPM16
A[15:0]
GT
LT
ROM_MIN
B[15:0]
PC
AND4
COMPM16
A[15:0]
OR2
GT
LT
Internal_reset
AND3
ROM_MAX
B[15:0]
COMPM16
A[15:0]
GT
LT
KEY_LOW
B[15:0]
DATA_ADDR_BUS
COMPM16
A[15:0]
GT
LT
KEY_HIGH
B[15:0]

Cost of adding ROM and access control

- Implemented on two common MCU platforms
  - AVR
  - MSP430

Component		Original Size in kGE	Changed Size in kGE	Ratio
AVR MCU		103	113	10%
Core		11.3	11.6	2.6%
Sram	4 kB	26,6	26.6	0%
Flash	32 kB	65	65	0%
ROM	6 kB	-	10.3	-
MSP430 MCU		128	141	10%
Core		7.6	8.3	9.2%
Sram	10 kB	55.4	55.4	0%
Flash	32 kB	65	65	0%
ROM	4 kB	-	12.7	-

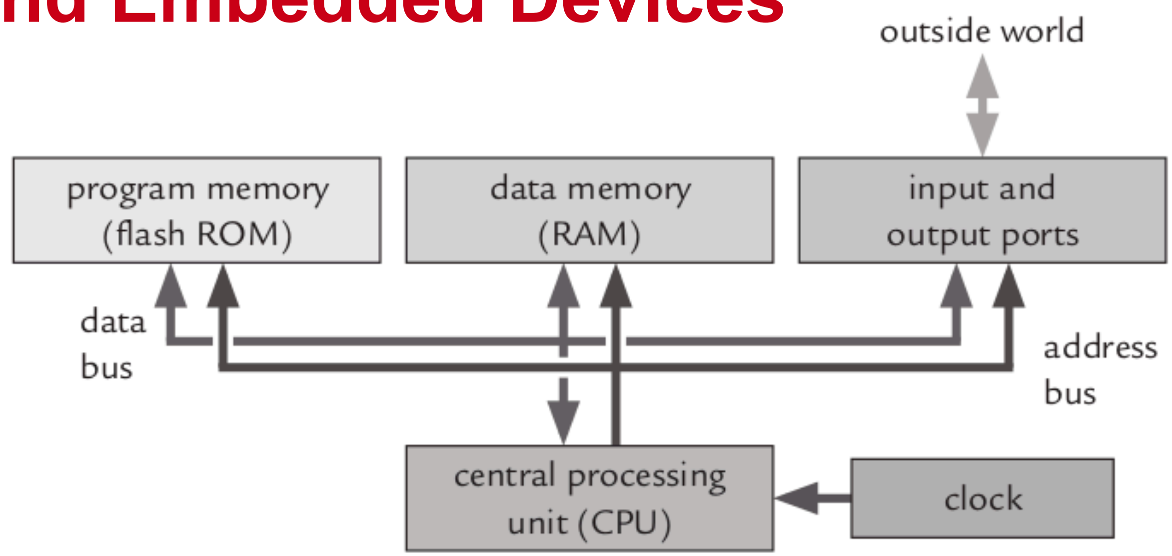
Considerations on SMART

- SMART provides an efficient hardware attestation solution for embedded devices
- Low additional gates required
- No run-time cost

Thanks for your time

Questions?

Low-end Embedded Devices



- Memory for program and data
- CPU
- Integrated clock
- In addition to
  - Communication interfaces (USB, CAN, Serial, Ethernet, etc.)
  - Analog to digital converters
  - ...