

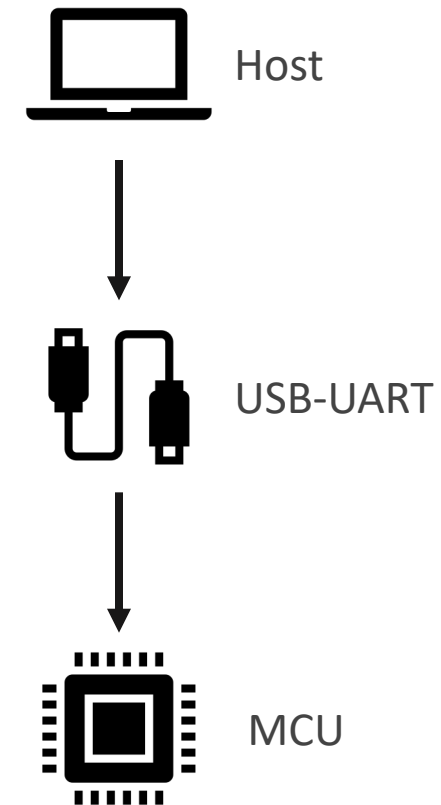
Glitching Firmware Readout Protection in an EV Charger and Instrument Cluster

Case Studies on GD32 and RH850 MCUs

Niclas Kühnapfel (TUB), Mahsa Kholghi (RUB), Dr. Kai Jansen (PHYSEC GmbH), Prof. Dr.
Christian Zenger (RUB), Prof. Dr. Jean-Pierre Seifert (TUB)

Serial Flash Programming Interfaces

- Bootloader interfaces support device programming and recovery
- Serial commands can request access to internal flash
- Readout protection restricts these operations after deployment
- A fault during authorization may enable a normally rejected read



Threat Model & Evaluated Hardware

- Attacker capabilities:
 - Physical possession or prolonged access to the target
 - Access to boot-mode signals, UART, reset, and a supply node
 - No valid unlock credential
- GD32
 - GD32F303RCT6 used in the go-e Charger
 - Experiments on a development board
- RH850
 - R7F701401 (RH850/D1L) in a VW instrument cluster
 - Experiments directly on the production cluster

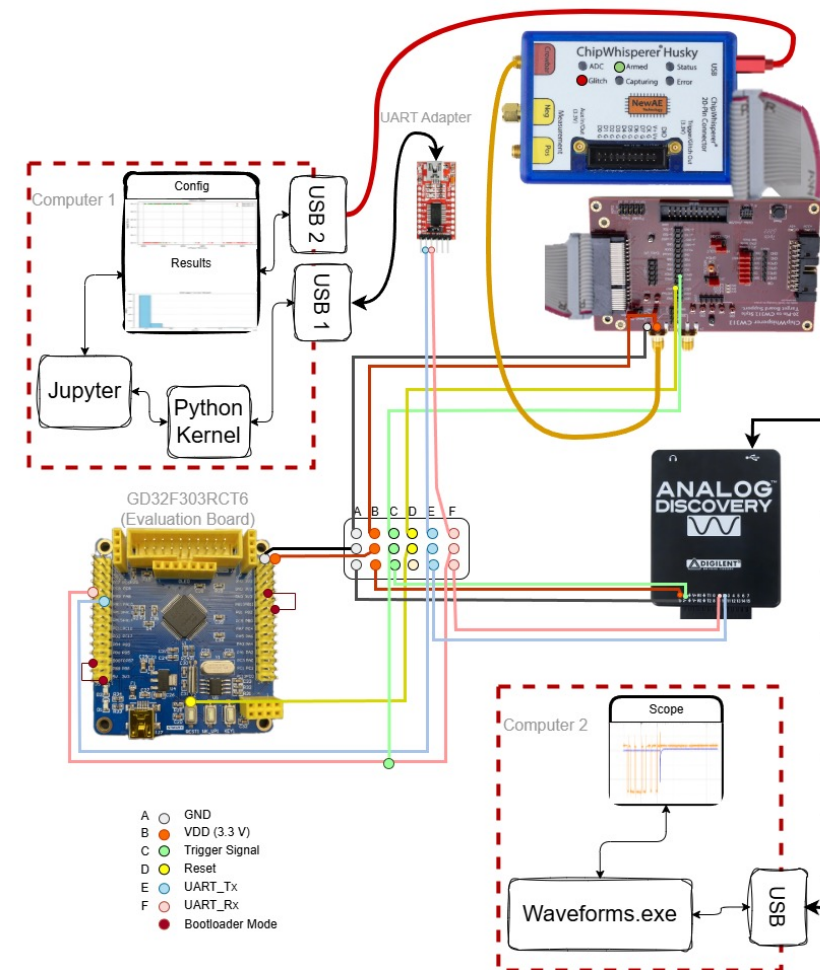


Common Experimental Method



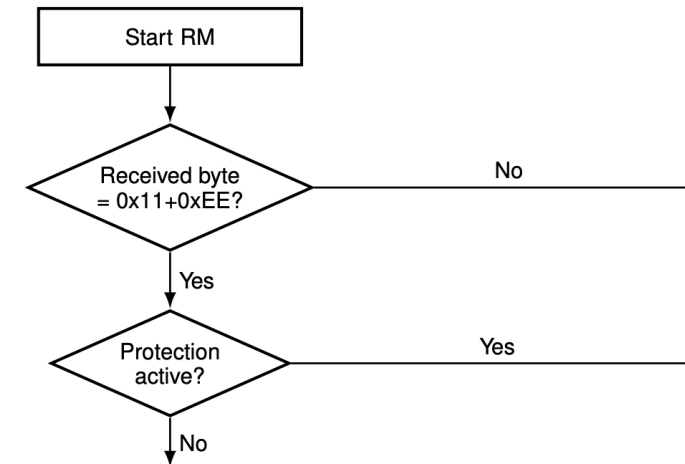
GD32: Experimental Setup

- GD32F303RCT6 development board with 256 KiB flash
- Boot straps select the ROM-based UART bootloader
- Host controls serial communication and reset
- Local decoupling adjustments enable short supply disturbances
- CW NANO supports initial parameter search
- CW HUSKY supports the UART trigger for bootloader attack



GD32: Read Memory Protocol

- Read Memory begins with command bytes 0x11 0xEE
- RDP Level 1 normally rejects the command
- An accepted transaction continues through address and length stages
- A valid read requires three ACKs and the complete payload
- Each request transfers up to 256 bytes



GD32: Bootloader Protection Check

```

1fffb330 00 25      movs      r5,#0x0
1fffb332 ff f7 2b    bl        FMC_GetOB_security
                ff
1fffb336 f0 b9      cbnz      Var_1,LAB_1fffb376
1fffb338 79 20      movs      Var_1,#0x79
1fffb33a ff f7 9d    bl        usart_write_byte
                fe
1fffb33e ff f7 71    bl        usart_get_char
                fe
1fffb342 04 46      mov       r4,Var_1
1fffb344 45 ea 00      orr.w     r5,r5,Var_1,lsr,#0x18

```

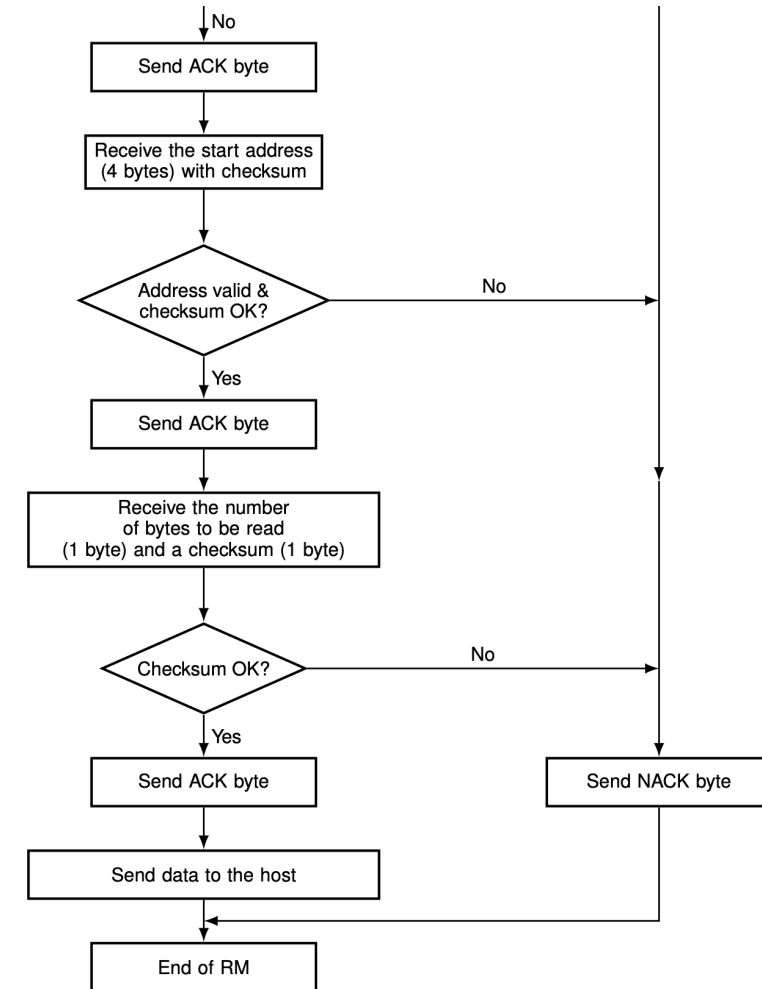
```

9  uint uVar4,
10  uint uVar5;
11
12  Var_1 = FMC_GetOB_security();
13  if (Var_1 == 0) {
14      usart_write_byte(0x79);
15      uVar1 = usart_get_char();
16      uVar2 = usart_get_char();
17      uVar3 = usart_get_char();
18      uVar4 = usart_get_char();
19      uVar5 = usart_get_char();

```

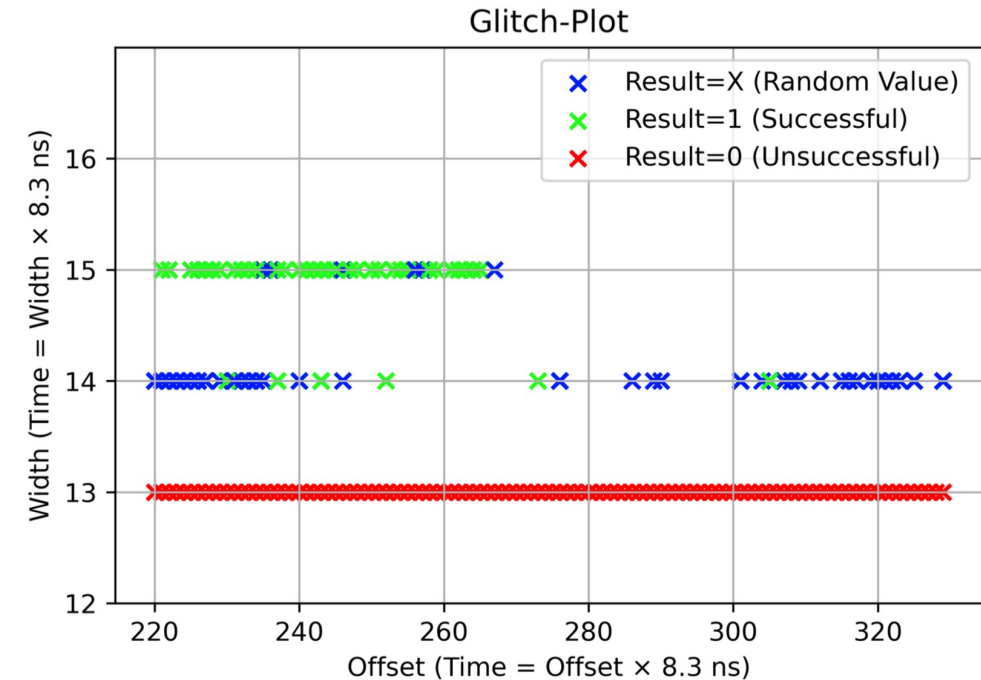
GD32: Read Memory Protocol

- Read Memory begins with command bytes 0x11 0xEE
- RDP Level 1 normally rejects the command
- An accepted transaction continues through address and length stages
- A valid read requires three ACKs and the complete payload
- Each request transfers up to 256 bytes



GD32: Calibration & UART Triggering

- Custom firmware exposes an observable security branch
- A GPIO trigger marks the calibration interval
- **385 / 2,200 trials reach the target branch: 17.5%**
- HUSKY triggers on the final command byte, 0xEE
- Approximately 8 μ s separate the command-byte end and first response



GD32: Attack Results

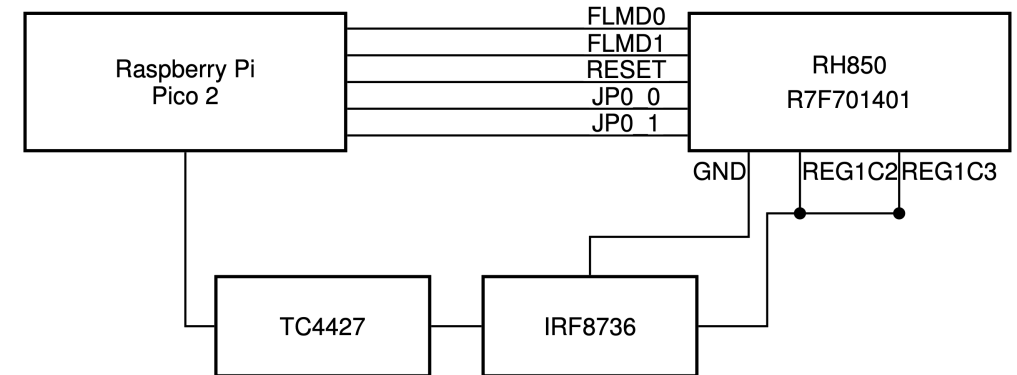
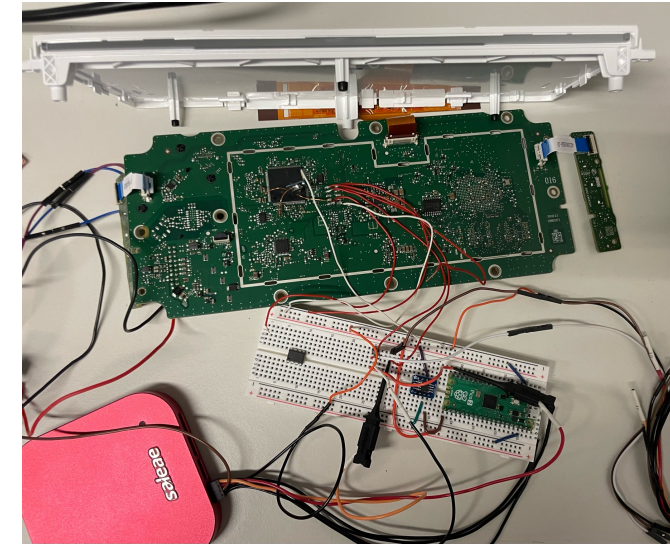
- Faulted transactions return protected flash data
- Repeated reads must agree before accepting a block
- Reset restores normal read rejection
- Full 256 KiB extraction under Level 1 remains future work
- Idealized extraction time: approximately **14.2 hours**, before retries and validation

$256 \text{ KiB} / 256 \text{ bytes} = 1,024 \text{ requests}$

$1,024 \times 50 \text{ s} \approx 14.2 \text{ hours}$

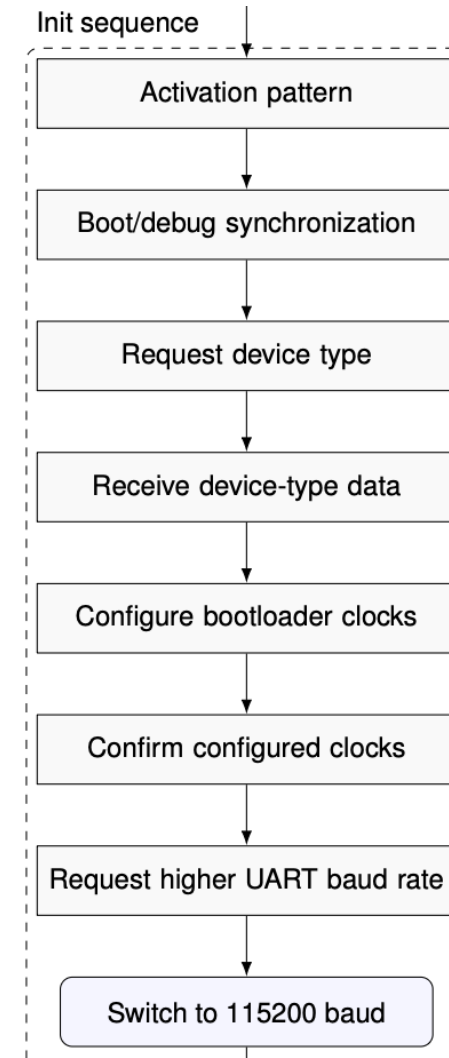
RH850: Experimental Setup

- Volkswagen instrument cluster with R7F701401 (RH850/D1L)
- UART captures support black-box protocol reconstruction
- Pico 2 PIO controls fault timing
- An external driver and MOSFET apply the supply disturbance
- Experiments run directly on the cluster PCB



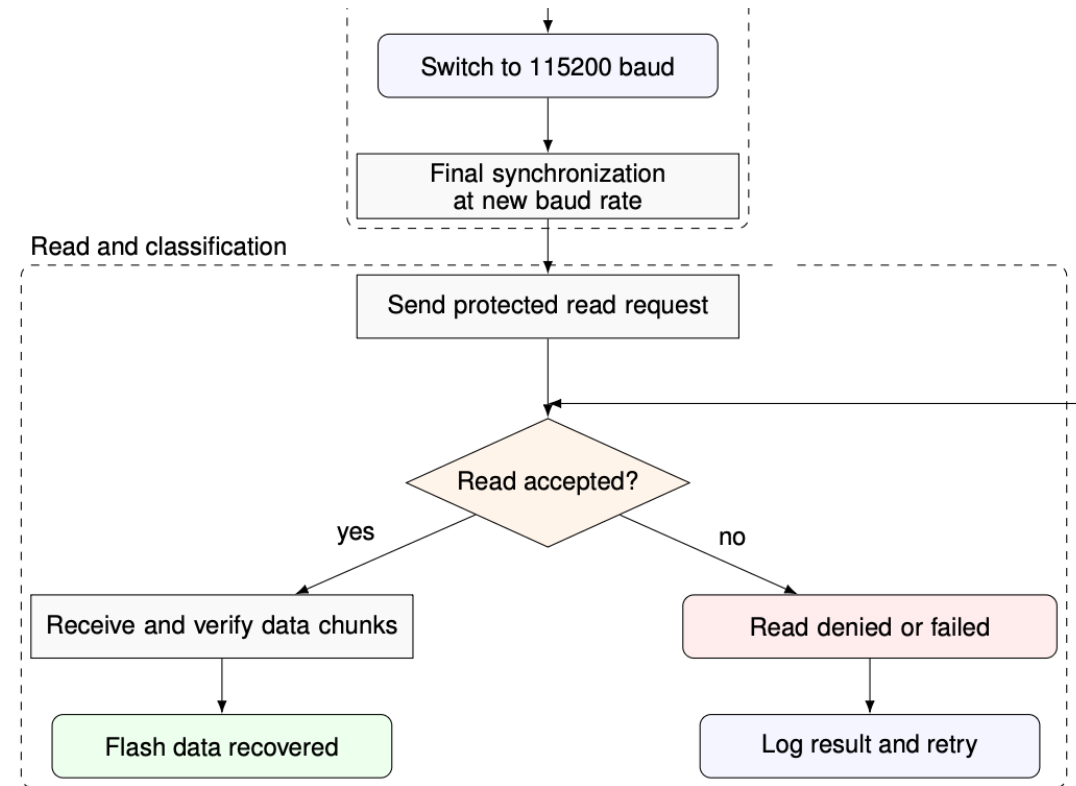
RH850: Range-Read Protocol

- Initialization establishes the serial programming session
- Command 0x15 requests a start-to-end address range
- The protected device normally returns a protection error
- An accepted range enters a continuing transfer state
- Subsequent requests retrieve 1 KiB blocks without another fault in that session



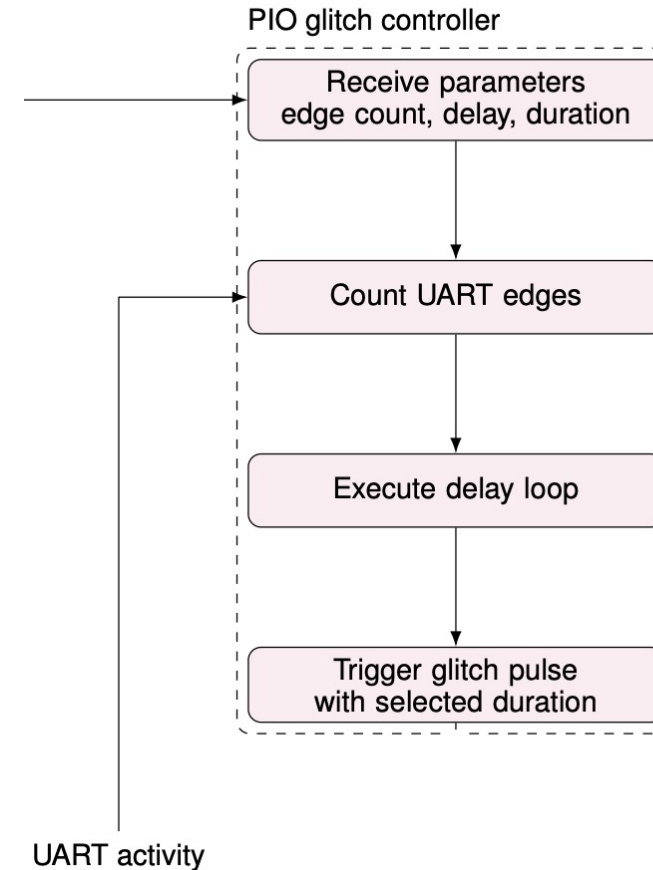
RH850: Range-Read Protocol

- Initialization establishes the serial programming session
- Command 0x15 requests a start-to-end address range
- The protected device normally returns a protection error
- An accepted range enters a continuing transfer state
- Subsequent requests retrieve 1 KiB blocks without another fault in that session



RH850: Synchronization & Parameter Search

- Reset-relative timing includes variable initialization delays
- PIO counts falling UART edges before starting a fine delay
- Address bytes change the command's edge pattern
- The host adjusts the edge count for each constructed request
- The search varies delay and pulse duration



RH850: Attack Results

- **7 / 3,462** search attempts return valid 1 KiB blocks: **0.202%**
- After narrowing down the timing window **7 / 131** successes: **5.34%**
- One successful fault enables the tested **2 MiB code-flash range**.
- Protection rejects reads again after reset

Outcome	Count	Fraction
Malformed packet	3,157	91.1901%
Protection error	268	7.7412%
Unknown response	14	0.4044%
Packet error	9	0.2600%
Valid 1 KiB bypass	7	0.2022%
Address error	4	0.1155%
Unsupported command	3	0.0867%

Discussion

- **Authorization scope:** GD32 checks each block request; RH850 authorizes a complete range
- **Validation:** complete transactions and repeat reads distinguish bypasses from misleading responses
- **Impact:** RH850 firmware analysis enabled diagnostic access and device reconfiguration
- **Limits:** one sample per case, unresolved fault mechanism
- **Proposed Countermeasures:** harden authorization and protocol state, recheck permissions, and monitor physical faults

Conclusion

- Command-relative voltage faults bypassed the tested read-protection decisions
- GD32 yielded repeat-validated protected block reads
- RH850 yielded the complete flash contents
- Authorization lifetime determines how much data one bypass exposes
- Bootrom protection must remain effective under physical faults

Questions?



Niclas Kühnapfel

kuehnapfel@tu-berlin.de



<https://www.tu.berlin/en/sect>

