Cryo-Mechanical RAM Content Extraction Against Modern Embedded Systems

Yuanzhe Wu
Red Balloon Security
New York, NY, USA
hans@redballoonsecurity.com

Grant Skipper
Red Balloon Security
New York, NY, USA
grant@redballoonsecurity.com

Ang Cui
Red Balloon Security
New York, NY, USA
ang@redballoonsecurity.com

***Abstract*—Cryogenic mechanical memory extraction provides a means to obtain a device's volatile memory content at runtime. Numerous prior works has have demonstrated successful exploitation of the Memory Remanence Effect on modern computers and mobile devices. While this approach is arguably one of the most direct paths to reading a target device's physical RAM content, several significant limitations exist. For example, prior works were done either on removable memory with standardized connectors, or with the use of a custom kernel/bootloader.**

We present a generalized and automated system that performs reliable RAM content extraction against modern embedded devices. Our cryo-mechanical apparatus is built using low-cost hardware that is widely available, and supports target devices using single or multiple DDR1|2|3 memory modules. We discuss several novel techniques and hardware modifications that allow our apparatus to exceed the spatial and temporal precision required to reliably perform memory extraction against modern embedded systems that have memory modules soldered directly onto the PCB, and use custom memory controllers that spread bits of each word of memory across multiple physical RAM chips.

***Index Terms*—cold-boot, side-channel, memory extraction, reverse engineering, embedded security**

I. Introduction

Modern high-performance embedded systems typically store firmware code in nonvolatile flash memory, and load the code into volatile dynamic random access memory (DRAM) during boot-up. The code and data contents of the device is often useful for security analysis. Firmware binaries can usually be extracted by directly reading the flash chip via physical probes, or through hardware debugging interfaces like Serial Wire Debug (SWD) [1] and Joint Test Action Group (JTAG) [2]. However, device manufactures can store the firmware as encrypted binaries at rest, and remove all hardware debugging interfaces. While these practices can be good for device security, it also makes the extraction of firmware for security analysis very difficult via traditional methods.

We present a generalized system for performing a "blind" RAM content extraction against modern embedded devices that does not require any hardware debugging interface, and is compatible with all embedded devices using double data rate (DDR)1, DDR2, and DDR3 memory modules. As shown in Figure 1, our system consists of a modified low-cost commercial off-the-shelf (COTS) computer numerical control (CNC) machine, a memory reader device implemented with an Field-Progammable-Gate-Array (FPGA), and controller implemented using an ESP32 [3] module and microPython [4]. This process involved cooling the memory chip, booting up the target device, physically transferring the chip to the readout platform, and recovering data. The entire apparatus can be built with widely available parts costing approximately $2,000 USD. The remainder of this paper discusses novel techniques and hardware modifications we developed to enable the described apparatus to perform reliable cryo-mechanical RAM content extraction on single and multi-memory-chip embedded devices. We demonstrate that our system can successfully perform memory extraction and reconstruction against an embedded device that uses a custom black-box memory controller and five physical RAM chips.

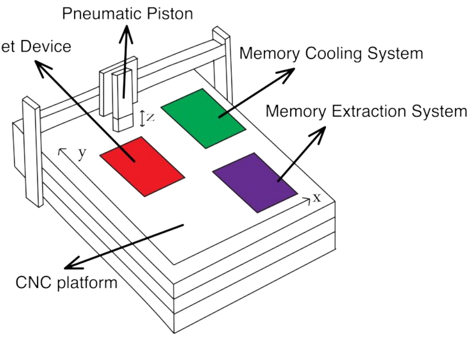

Fig. 1. New apparatus for performing cryo-mechanical attacks against hardware security of generic embedded devices. The CNC platform is used to transfer a memory chip between target device, memory cooling system and FPGA-based memory extraction system. This memory chip is mounted on the bottom of the pneumatic piston which outputs regulated pressure to push the memory chip against our custom-designed electrical elastomer socket.

In the context of the paper, we refer to LPDDR1, DDR2, and DDR3 memory as DRAM, which has been demonstrated to be vulnerable to cold boot attacks. These attacks exploit the Memory Remanence Effect [5], which enables the contents

This material is based in part upon work supported by the United States Air Force and DARPA under Contract No. FA8750-18-C-0127. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force and DARPA. Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

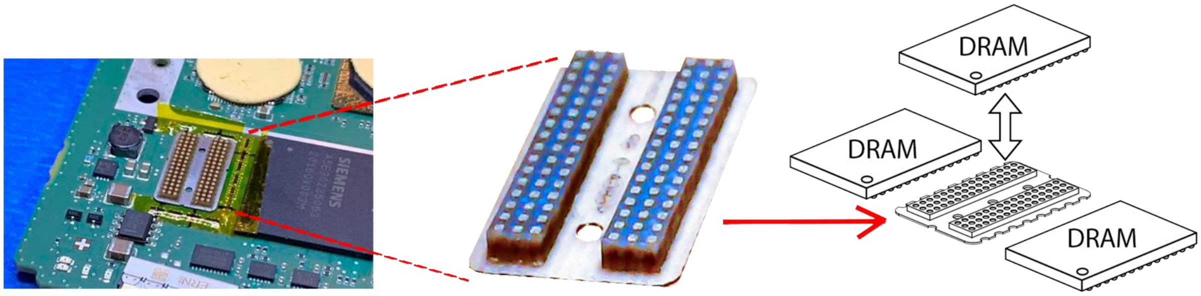

Fig. 2. New method for transferring embedded memory chips, such as BGA DRAM chips that are soldered on the embedded device printed circuit board (PCB) board. Our designed conductive rubber can make all the memory removable at runtime without affecting the function of the target device.

of a DRAM module to be retained for short durations after power-off when the module is significantly cooled below room temperature. Traditionally, cold boot attacks have been focused on personal computers rather than embedded systems, and their goal has been to recover disk encryption keys used in established schemes like TrueCrypt [6] and BitLocker [7]. These attacks have required code execution on the target device, but the proposed cryogenic mechanical apparatus presented in this paper overcomes this limitation by using a separate FPGA-based memory readout platform that generalizes different types of DRAM extraction. Furthermore, as shown in Figure 2, the apparatus introduces methods to physically transfer individual DRAM chips between the target device and readout platform, allowing attacks to be applied to a wider range of physical device form-factors without the need for removable memory like Dual In-line Memory Module (DIMM) on the target device.

Alternative techniques for memory extraction include directly memory bus probing, which involves intercepting electronic signals on a device's memory bus. However, this method requires specialized and expensive hardware capable of capturing high-frequency signals [8]–[10].

Our presented cryogenic mechanical memory extraction approach offers a direct path to accessing physical RAM content at runtime, and can be applied to embedded systems with custom memory configurations. The low-cost hardware used is widely available, making it a cost-effective solution. This approach provides a more generalized and automated system for extracting RAM content from modern embedded devices, compared to traditional cold boot attacks and prior works.

II. Background

At the physical level, DRAM is structured as a grid of capacitors, addressable by row and column. Data is stored by charging or discharging the capacitors. The charge stored in capacitors decays due to leakage, so during normal operation the capacitors must be periodically refreshed to prevent data loss. Upon system power-down, all capacitor cells will decay towards a ground state, which varies depending on the cell's address.

Both the traditional cold boot method and our cryo-mechanical attack exploit the fact that (1) DRAM exhibits some degree of data remanence - data does not instantly decay upon power loss, and (2) physical data decay rates may be significantly slowed by cooling of the DRAM chips to a very low temperature. A typical refresh rate for DDR2 memories operating at standard temperatures is 64 milliseconds, which is specified by manufacturers to virtually eliminate all possibility of data decay. Halderman *et al.* [11] found that DDR1 and DDR2 chips retain roughly 50% of their data a few minutes after power loss at room temperature. When the chips were cooled to -50° Celsius, the retention rate increased to well over 99%.

From DDR1 to DDR5, each generation of DDR memory has aimed to improve memory bandwidth while reducing memory cell capacitance to increase speed and storage density. This development has impacted observable memory decay rate shown in previous research [11]; which has found that new generations of DDR memories typically decay faster than previous ones. Interestingly, DDR4 and DDR3 exhibit the most similar decay rates between generations. Yitbarek *et al.* [12] hypothesizes this indicates a practical limit in minimal memory cell capacitance, suggesting that there might be a boundary to further minimizing capacitance.

A. DRAM in Embedded Systems

Many modern embedded systems are built around a System-on-Chip (SoC), which combines CPU core(s), peripherals, and data storage into one package. Systems demanding higher performance will typically integrate the SoC with external DRAM. We have encountered several examples of embedded systems with external DRAM, including industrial controllers, printers, gaming consoles, and quadcopter drones.

In contrast to PCs, which use standardized and removable DIMM memory modules [13], the DRAM chips on embedded devices are usually soldered directly to a custom-designed PCB. Modern DRAM chips use Ball Grid Array (BGA) packages, which raise several difficulties with physical manipulation and circuit rework. (We will discuss this further in section IV). Designers of embedded systems also have far more flexibility in the architecture of their external DRAM. Options such as the number of chips used, types and packages of chips used, interleaving of data between several chips, and use of error correction codes (ECC) are limited only by the feature set supported by the SoC memory controller. Thus, there is a great deal of variability in DRAM designs across different embedded platforms.

For manufacturers of embedded devices, designing a circuit board with external DRAM is an intensive undertaking. The DRAM I/O interfaces and their high-speed signal lines demand strict routing requirements, stable power supplies, and signal terminations [14]. In this work, we highlight one particular layout technique: pin swapping. DRAM interfaces allow for the pins within a particular data byte to be freely reordered by the board design engineer in order to simplify layout [14], [15]. Without knowing the exact permutation used, an attacker who has physically captured data from the DRAM chip would observe the bits within each data byte to be shuffled. We explore this issue in Section V-C and present our solution to efficiently recover the original data.

B. Prior Research

The cold boot attack is a physical hardware exploit pioneered by Halderman *et al.* in 2008, who successfully extracted the contents of DRAM memory on personal computers using SDRAM, DDR1, and DDR2 memories [11]. The researchers developed two attack procedures applicable towards PCs: rebooting the target computer with a custom bare-metal memory capture program, and removing and transferring a cooled-down DIMM memory module to a separate computer running the memory capture program. Later researchers have replicated this work and extended it to DDR3 and DDR4 memories [16], which may make use of scrambling to improve signal integrity and obfuscate contents [12], [17]. In 2019, Cojocar *et al.* [18] employed a unique combination of a custom-built hardware probe, Rowhammer [19] bit flips, and a cold-boot attack to reverse engineer ECC functions on commodity AMD and Intel processors with removable memory. While manufacturers of memory controllers have enhanced the complexity of memory scrambling on commodity CPUs, it remains not cryptographically secure and does not provide adequate protection against cold boot attacks.

It is notable that all the aforementioned work was conducted on PCs. Relatively little attention has been paid to the feasibility of cold boot attacks on other platforms, despite the ubiquity of DRAM in modern computing systems. In 2013, Müller and Spreitzenbarth [20] performed cold boot attacks on Android phones by using the Android recovery to capture memory contents. In 2020, the first cold boot attack on an IoT device was performed by Won *et al.* [21], who targeted the Raspberry Pi model B+. In 2021, Won *et al.* [22] recovered data from an encrypted machine learning accelerator, the Intel Neural Compute Stick, when used in conjunction with a Raspberry Pi platform vulnerable to cold boot attacks.

All of these attack methods rely on either executing custom code on the target system or removable memory, and would be defeated by modern embedded device secure boot processes that cryptographically authenticate device code at bootup. To the best of our knowledge, we present the first cryo-mechanical attack generalizable to all embedded systems with discrete DRAM chips, without any prior need for code execution on the target system.

III. Embedded Device Target: S7-1500 CPU

Embedded devices with secure boot implementation are becoming prevalent in most applications scenarios. These devices are widely deployed in critical industrial environments. The use of encrypted firmware with secure boot makes security analysis of such devices difficult, especially when no debugging interface is available. Therefore, there are few studies on related devices. In our research, we selected a typical embedded device — a programmable logic controller (PLC) — with encrypted firmware and secure boot implementation as our target. We demonstrated that our presented new cryo-mechanical attack can overcome the limitations of previous schemes, and be applied to almost all embedded devices.

PLCs are embedded devices critical to operating machinery for modern industrial manufacturing processes. Siemens AG commands more than 30% of PLC market share [23]. Based on its prevalence in critical infrastructure, it is desirable to perform security analysis of SIMATIC S7-1500 PLC devices. Our particular target for this study was the 1511-1 PN main CPU module, model number 6ES7511-1AK02-0AB0, as shown in Figure 3.

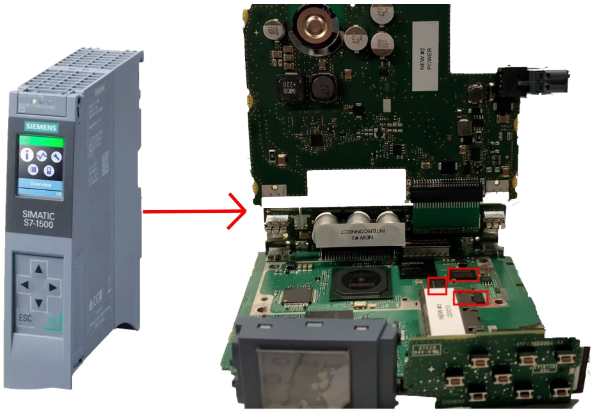

Fig. 3. The Siemens SIMATIC S7-1511-1 PN main CPU module. The right side is the same device with case removed to make Dynamic random-access memory (DRAM) chip exposed. The DRAM chips are labelled in the figure.

The Siemens S7-1500 PLC CPU module consists of two sets of microcontroller subsystems on different parts of the PCB. The first subsystem consists of one NOR flash with one LPDDR1 memory and an ARM based MCU. The content extracted from NOR flash chip is plain text, and later we identified its function is only for PLC LCD screen and input button control. The second subsystem consists of two NAND flash with five DDR2 memory chips and Siemens proprietary MCU.

IV. Implementation for Memory Extraction

The implementation steps for conducting our new cryo-mechanical memory extraction method is shown in Figure 4. Initially, we prepared an FPGA-based platform for readout of individual DRAM chips by desoldering [24] the chip from the target device and attached integrated circuit test sockets to both the device and the FPGA platform, allowing us to install and remove the DRAM chip quickly. Then we built the cryo-mechanical apparatus to transfer the physical DRAM chip between target device, memory cooling system and the FPGA-based memory readout platform. The attack procedure involved cooling the memory chip, booting up the target device with it, physically transferring the chip to the readout platform, and recovering data on the chip. We will discuss the implementation in this section. For devices with multiple discrete DRAM chips, we repeated the procedure separately for each chip and reconstructed the memory image in a later procedure.

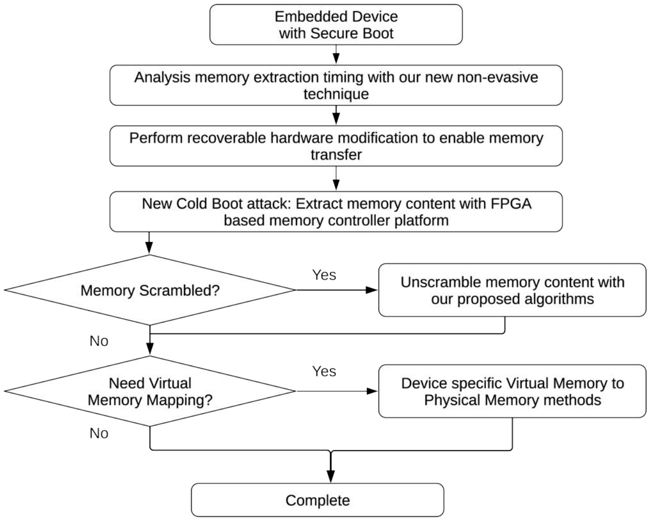
Embedded Device
with Secure Boot
Analysis memory extraction timing with our new non-evasive technique
Perform recoverable hardware modification to enable memory transfer
New Cold Boot attack: Extract memory content with FPGA based memory controller platform
Memory Scrambled?
Yes
Unscramble memory content with our proposed algorithms
No
Need Virtual Memory Mapping?
Yes
Device specific Virtual Memory to Physical Memory methods
No
Complete

Fig. 4. This flow chart describes the implementation steps for conducting our new cryo-mechanical memory extraction method. This method may be used on most generic embedded devices, including those with secure boot enabled. After the memory content was extracted from the device, we also introduced our methods to unscramble the bit shuffling memory and map virtual memory to physical memory.

A. FPGA-based Memory Extraction Platform

DRAM chips have highly complex timing requirements that must be satisfied in order to read and write data. In both PCs and embedded systems, they are interfaced through memory controllers, specialized digital logic circuits that mediate between CPU memory accesses and the DRAM physical layer interface. Similar to the work of Trikalinou 2017 [25] which uses FPGA boards as memory controllers to interface with DIMM removable memory, we decided that an FPGA was a flexible and cost-effective solution to interfacing with irremovable embedded DRAM chips. As shown in Figure 5, our research used a MicroPhase Zynq-7000 board (XC7Z020 [26]) for LPDDR1 and a Digilent Nexys A7 (XC7A100T [27]) board for DDR2; each costs between $100 USD and $200 USD.

We implemented custom memory controller logic for LPDDR1 and used Xilinx Memory Interface Generator (MIG) [28] for DDR2. The LPDDR1 memory controller operates the same way as DDR1 memory controller and the only difference are the I/O interface voltage and impedance. LPDDR1 operates at 1.8V as opposed to 2.5V for DDR1 [29]. Xilinx does not provide MIG for DDR1/LPDDR1 memory. Thus, We implemented Verilog LPDDR controller with AXI4 [30] interface as well as custom-designed LPDDR1 to FPGA breakout board. The AXI4 protocol is commonly utilized as a high-speed bus protocol for system-on-chip designs, facilitating effective communication among hardware components. Adopting AXI4 in our FPGA design enables a general hardware design that can quickly accommodate different types of DDR memory.

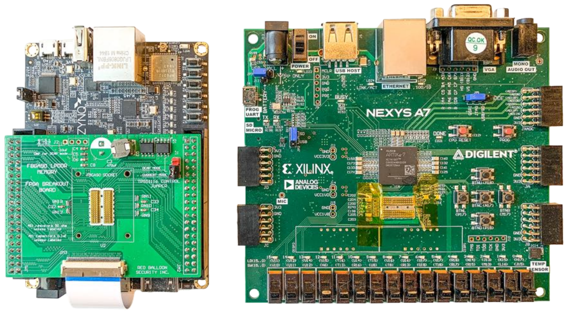

Fig. 5. FPGA-based memory readout platforms. The left side shows the Zynq-7000 development board with custom-designed DDR to FPGA breakout board for LPDDR1 memory. The right side shows the Digilent Nexys A7 board with our conductive elastomer for DDR2 memory.

B. Automated Cryo-Mechanical Apparatus

In our cryo-mechanical attack procedure, the DRAM chip had to be transferred across devices and positioned accurately above a test socket. We developed an apparatus for this purpose using a small desktop CNC machine and a pneumatic piston. As shown in Figure 1, with the help of the CNC platform, the pneumatic piston can move around between target device, memory cooling system, and FPGA-based memory extraction system.

The machine body is a generic type 3040 CNC [31], typically sold as engraving machines for wood and soft metals with work area measures 300 × 400 mm. While this machine is very low-cost, and not built for high-precision movement, it does have a fairly rigid chassis and quality ball screw actuators [32] on both X and Y axis. To enhance the precision and control of the CNC machine's X and Y axes, we replaced the stock stepper motors and external motor driver that originally came with the machine, which had low accuracy and closed-source controlling software, with ClearPath MCPV [33] stepper motors. MCPV motors come integrated with a high resolution encoder system and microprocessor controller capable of achieving rotational resolution of 0.057 degrees [34]. This simple modification converted the low-cost open-loop controlled CNC into a closed-loop controlled system capable of high resolution repeatable motion control. Lastly, the CNC's Z axis drive system and motor was removed and replaced with the SMC CXSM15-100 [35] pneumatic linear actuator. This pneumatic actuator was chosen to eliminate unwanted lateral wobble caused by the machine's original leadsscrew-based actuator. Actuation using air also provided the additional benefit of cushioning the downward force applied to the target device and FPGA reader socket when engaged. The total applied force and travel speed can also be carefully controlled by adjusting the input air pressure and air manifold exhaust port valves.

The apparatus was controlled through an ESP32 microcontroller board. I/O pins on the board were used to signal the stepper motors and toggle a solenoid valve for compressed air. As shown in Figure 6, we programmed a sequence of motions for performing the cryo-mechanical attack: submerging the chip into the dry ice cooling bath, pressing the chip against tissue paper to wipe excess isopropyl alcohol, positioning the chip onto the target device, and transferring the chip to the FPGA readout platform.

For alignment, the CNC machine was programmed to follow specific coordinates as a normal procedure for CNC operation. Proper coordinate adjustments were made to ensure the target device could boot up correctly and the FPGA-based readout platform could access data with the correct alignment while the memory chip was attached. The CNC automation ensured accurate, repeatable movements, eliminating the need for manual alignment and guaranteeing a consistent and reliable outcome in the extraction procedure.

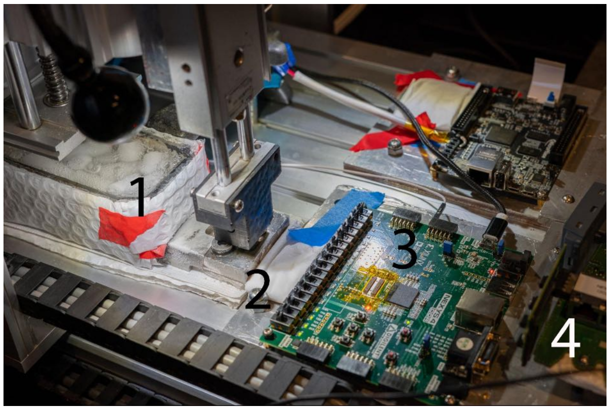

Fig. 6. The apparatus setup for the new cold attack. (1) Memory cooling system (2) Tissue paper to wipe liquid (3) FPGA-based Memory extraction system (4) Target device

C. Integrated Circuit Test Sockets

Our methodology requires a DRAM chip to be transferred between the target device and a readout device. To minimize memory decay, the physical movement had to be completed quickly while the chip remained cold. We investigated three different types of integrated circuit (IC) test sockets that allow chips to be attached and removed to a circuit board without soldering. Shown in Figure 7, the tested items were (1) a large pre-built socket with through-hole pins, (2) a small pre-built socket with BGA balls, and (3) a custom-designed socket made of electrically conductive elastomer.

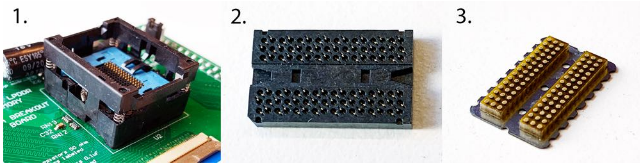

Fig. 7. DDR memory sockets we tested for the new cryo-mechanical attack. Socket (3), our custom-designed electrically conductive elastomer, met the requirements for our research.

Socket (1) could not be installed on our target device, as the socket uses through-hole pins and our device had surface-mount BGA pads for the DRAM chip. We were unable to modify the target device's circuit board, but we designed the custom LPDDR1 breakout board to be compatible with this socket. Sockets (2) and (3) were compatible with the target device, since they both use a grid of contacts in the same dimensions as the DRAM chip itself. Our testing revealed that socket (2) was very difficult to use. We struggled to solder it onto a circuit board without melting the plastic parts of the socket, and we found that its contacts for the DRAM chip were easily contaminated with dirt and failed to deliver a reliable electrical connection.

However, the conductive elastomer socket (3) avoided these problems. The socket consists of a flexible rubber-like material with gold-conductive polymer composite embedded inside portions where electrical connectivity is desired. In this gold-conductive polymer composite a base polymer, such as silicone rubber or polyurethane, is combined with gold particles (e.g., gold nanoparticles or gold powder) to create a conductive matrix. The gold particles are uniformly distributed within the polymer, which allows for effective electrical conductivity when the material is compressed.

When pressed between a chip and a circuit board, the particles inside the polymer composite form a reliable signal pathway, without the need to solder the socket with either the board or the chip. This socket is usually manufactured for IC testing which can reach 14GHz@1-dB and less than 100mΩ for each contact point which is within the operating range for DDR memory. Our proposed IC socket is a custom-designed component, which ensures optimal alignment and connectivity with the target memory module. Based on the DRAM datasheet for the target device, we designed and provided the mechanical drawing of the conductive elastomer socket, ensuring proper alignment with the BGA positions corresponding to the target device. This detailed design was then supplied to the manufacturer for production. The IC socket can be sourced from various suppliers [36]–[38], and its design files will be made available for replication.

D. Cooling Target Memory Chips

In previous research, cold boot attacks were typically performed by cooling DRAM chips by evaporating the fluorocarbons in compressed-air dusters or purpose-made freeze spray [39]. In addition to freeze spray, we also investigated the use of liquid nitrogen and dry ice.

Freeze spray is easy to obtain and can cool to roughly -50° Celsius. When testing freeze spray, we found that this cooling method tends to condense and freeze atmospheric water vapor around the device. We determined that these side effects were undesirable for a cryo-mechanical attack procedure that involves physically moving DRAM chips: The fine-pitch grid of electrical contacts could be easily short-circuited by ambient water vapor rapidly condensing due to the low temperature. The spray may also accidentally cover nearby electronics on the PCB and may contribute to thermal shock and damage.

Liquid nitrogen has the advantage of reaching an extremely low temperature of -196° Celsius. However, this method has several drawbacks due to liquid nitrogen being a cryogenic fluid: It is relatively expensive to obtain, difficult to work with, cannot be stored for long periods due to evaporation, and carries considerable safety risks.

Dry ice (solid carbon dioxide) has a sublimation temperature of -78.5° Celsius at 1 standard atmosphere. Although not easily stored, dry ice has the advantage of being low-cost and easily obtained. As a solid material, we found that it requires a heat transfer fluid to effectively cool a memory chip. Acetone and isopropyl alcohol are suitable for this purpose, since they have freezing points lower than the temperature of dry ice.

Based on these considerations, the cooling method we found most effective was a bath of dry ice and isopropyl alcohol. In addition, by attaching a small block of aluminum or copper to the chip with thermally conductive epoxy, it was possible to substantially increase its thermal mass and retain the cold temperature for much longer time period. To cool down the DRAM chip, the CNC platform utilizes a pneumatic piston described in seciton IV-B to submerge it into the bath of dry ice and isopropyl alcohol.

V. Engineering Challenges

A significant challenge we encountered in our initial testing was physical damage to the target device. After a number of successful runs, the device would often cease to function, but with no obvious or visible problems. The memory chip itself would remain operational, which could be verified by writing and reading data from the FPGA alone. It is likely that this phenomenon occurred due to our using the hardware beyond its intended operating procedures and conditions.

One suspected issue was mechanical damage caused by the downward pressure of the pneumatic piston. Pushing on the board slightly deflects a portion of it, which can lead to a warping or bending effect across the board. Repeated pressings may lead to fatigue in thin copper traces, and uneven bending may induce shearing of internal layers.

Mitigations for mechanical damage first involved reducing the pressure of air supplied to the pneumatic piston. It was possible to operate the piston well below its rated pressure. Pressures of 10 PSI or less were sufficient to actuate the piston and press the chip with enough force to reliably make electrical contact.

To better support the board under a pressing force, we made castings of the reverse side of the board in epoxy resin. The casting forms a negative image of the board's components, conforming to the contours of its exact shape and distributing pressure evenly across its surface. By milling the other side of the casting flat, it was possible to manufacture a custom support structure that could be fixed between the board and the CNC machine platform. As shown in Figure 8, the board warping conditions was enormously improved after we adopted the epoxy resin solution.

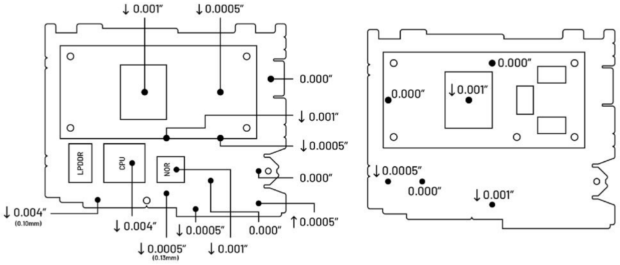

Fig. 8. Measurements of board deflection (in inches). The left side was pressing down without epoxy resin support structure. The piston was operated with 10 PSI of air pressure. Note that the bottom-right corner moved upward while other locations moved downward, indicating warping of the board. The right side was pressing down with epoxy resin support structure. Note the smaller numbers and lack of warping.

Another source of PCB damage may be due to temperature changes. The DDR memory chip was cooled using dry ice, which has a temperature of -78° Celsius. Although the chip itself withstands this temperature without apparent issues, the board may be more sensitive to extreme cold. Thermal shock, from pressing a cold chip onto the board, could damage narrow PCB traces. Different amounts of thermal expansion and contraction between PCB layers may also lead to delamination. Using a K-type thermocouple affixed to a ground plane region next to the chip, we measured the temperature of the PCB as the cold chip was pressed onto it.

The data revealed that the PCB temperature would drop by roughly 3-4° Celsius within the span of a few seconds. For comparison, we measured the temperature of the same location on the board with a room-temperature chip, which gave information on ordinary thermal fluctuations due to heat generated by the chips when powered on. The results indicated that the total temperature fluctuation was well within the amounts seen in ordinary operation.

Furthermore, the lowest temperature measured on the PCB with a cold chip was approximately 5° Celsius below ambient temperature, which is much warmer than the temperature of the cold chip itself. These signs indicated that the conductive elastomer socket was a good thermal insulator. With our mitigation strategies in place, further operation of the cryo-mechanical attack apparatus did not lead to any more instances of device failure.

A. Timing Analysis for Memory Extraction

A primary obstacle when extracting memory content is determining when to remove the DRAM IC from the PCB and begin extraction. If removed too early, or late, the chip may not contain useful information or code used in the boot process. In 2021, Van den Herrewegen *et al.* [40] used side-channel information for finding critical timing of bootloader for fault injections. Following a similar approach, we developed a non-invasive side channel-based technique for determining optimal timing values for extracting memory data. Conventional side channel-baded analysis methods extract sensitve information from systems by analyzing their unintentional emissions. These emissions may include power consumption [41]–[43], electromagnetic radiation [44], or even acoustic signals [45], which can be analyzed to infer secrets such as encryption keys, passwords, or other sensitive information. Our approach uses electromagnetic analysis to determine the critical timing of the booting up process of this device.

As shown in Figure 9 and 10, using a near-field probe, RF amplifier, and a spectrum analyzer [46], one can observe the intensity of electromagnetic emanations [47] from each chip on the device over the course of device operation, which gives insight into relative levels of electrical switching activity. For DRAM chips, strong electromagnetic emanations would imply large changes in memory contents, and weak emanations would imply minimal changes.

To ensure the reliability of our observations, we repeated the experiment multiple times and compared the emanations result of the boot-up process. We found that on our target device, the pattern of emanations was deterministic from the initial boot-up, and they occurred in bursts with gaps over time. Result of one experiment is shown in Figure 10. We identified periods of consistently low electrical activity during the bursts, which minimized the impact of timing variation on the captured contents of memory. Subsequently, we utilized the timestamp that was determined to be the appropriate timing for the Cryo-mechanical memory extraction process.

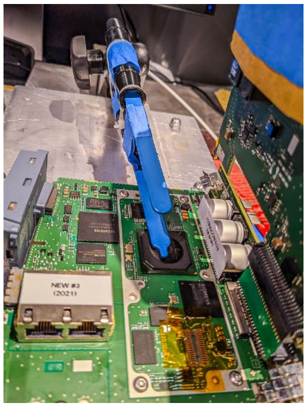

Fig. 9. A stack of two near-field probes, used for observing electromagnetic emanations, can be seen placed on top of Siemens S7-1500 PLC main SoC. Using this setup, CPU bond emanations during the bootup process can be observed.

B. Pre-extraction DRAM Decay Analysis

In order to use the memory remanence effect for the DRAM chip, we must determine the memory decay time during the transfer of the cooled DRAM chip. Before performing live memory extraction, we first analyzed the decay patterns and rates in LPDDR1 memory at ambient room temperature. Two experiments were performed using the FPGA readout platform, writing either all 0s or all 1s to memory and reading out after 2000ms. Both experiments were conducted at room temperature without auto-refreshing. Decay rates at ambient temperature are extremely fast. Approximately 40% of bits decay in 2 seconds, the shortest duration that could be reliably tested.

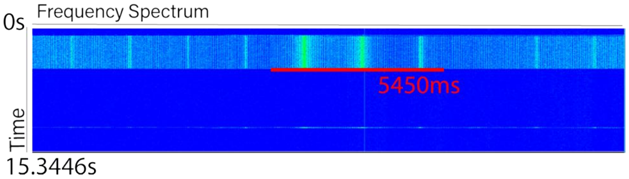

Fig. 10. Electromagnetic emission spectrum of DDR2 chip as seen on a Keysight PXA N9030B spectrum analyzer. The X-axis shows frequency, and Y-axis shows time. Specific timing can be retrieved according to the spectrum.

We repeated the procedures for memory decay analysis on DDR2. The results showed significant differences in decay properties between LPDDR1 and DDR2 chips. At room temperature, approximately 13% of bits decay in 2 seconds. The bit error rate was significantly less than that of LPDDR memory, so we believed that obtaining a high quality memory dump from the DDR2 memory should be easier than for the LPDDR memory.

Transferring the super-cooled memory chip quickly is crucial, as the chip gradually gains heat until it reaches ambient room temperature. Throughout our research, we continuously developed and improved our cryo-mechanical memory extraction process. We performed trial testing on the LPDDR1 chip memory dump, and based on our research findings, we discovered that after cooling the DRAM with our dry-ice bath cooling system, there was a 12-second window for reliable memory content extraction. This was evidenced by the absence of bit flips in human-readable strings and consistently repeatable memory extraction results. DDR2 exhibits an even better memory remanence effect based on our decay analysis at room temperature. This improved retention property of DDR2 memory chip ensures the 12-second window provides enough time to transfer and extract memory content successfully.

C. Reconstructing Memory Contents

Since embedded systems designers have a great deal of flexibility in DRAM layout, physical data retrieved via cryo-mechanical attack may not directly correspond to data logically seen by the device as shown in Figure 11. This posits another major challenge for us in making sense of the extracted memory content. In this section, we show how data interleaving across multiple chips can be identified and reverse-engineered, and how the bit permutation effect of pin swapping can be determined and corrected for.

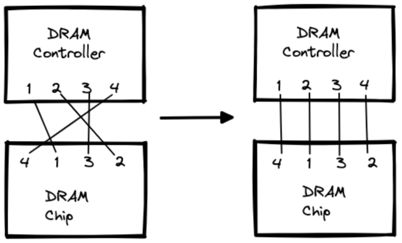

Fig. 11. PCB routing pin swapping optimizations are used in Escape Interface Routing, such as memory controller with multiple data lines, which can be interchangeable for routing.

1) Preliminary Investigation of Extracted Data: As introduced in Section III, the SIMATIC S7-1500 DDR2 subsystem uses five DRAM chips in parallel with an 8-bit data width per chip, accessing a 40-bit word per memory address. This layout suggests that the system uses an ECC scheme with 32-bit data words, 7 bits of Hamming ECC, and 1 unused bit, a common implementation for performing single-error correction and double-error detection (SECDED). The statistical distribution of zeros and ones for each of the 40 bits is shown in Figure 12. This distribution reveals a data bit with a value of zero across nearly all memory addresses, which we interpreted as the unused bit. Entropy analysis also shows that the chip containing this zero bit exhibits a slightly different entropy distribution relative to the remaining four chips. Hence, we hypothesized that the seven ECC bits are stored exclusively on one chip, and the other four chips together make up the 32-bit data word.

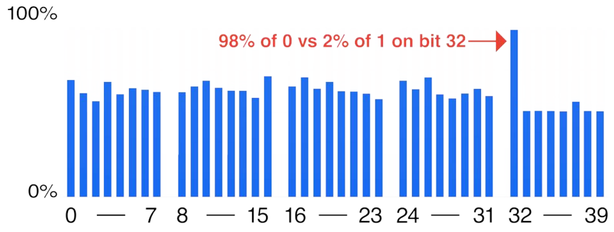

Fig. 12. Statistical distribution of zeros and ones for each of the 40 bits. Bit No.32 on the DDR2 chip (5) have significantly higher zeros. The total of 40 data bits line across 5 memory chips. (Each have 8 bits data line). Usually modern embedded system is 32-bit or 64-bit system which means the number data bits needed is either 32 or 64. Hamming ECC have 7 bits of ECC data for 32-bit data words. Total 32+7=39 bits data line is needed as opposed to total 40 bits present. The 1 unused bit can be seen from this statistical distribution graph, which indicate the Hamming ECC is used in this Embedded system and the 5th DRAM chip contains the ECC data.

We first observed that no bytewise permutation of the four data chips resulted in recognizable code or strings, leading us to suspect an additional bitwise permutation. Such an effect arises from pin swapping, commonly used by hardware designers during the board layout process to simplify routing of data signals [14]. Because DRAM uses bidirectional data lines shared for reading and writing, a designer can arbitrarily reorder the data signals between the CPU and the DRAM chip without affecting the data as seen by the CPU. When the DRAM chip was transferred to our FPGA readout device, however, we retrieved a data readout with an unknown permutation applied to the data bits. We noted that DRAM standards restrict pin swapping to within the data bits within a byte, due to certain bytewise signals such as DQS and DM. This restriction worked to our advantage by limiting the permutation search space, as we could conclude that bits would not be permuted across multiple chips.

2) Unshuffling Memory with Known Plaintext: Suppose that we are able to identify the plaintext information that corresponds to some portion of bit-shuffled data extracted from the five DDR2 chips. Then, due to the nature of the bit shuffling, we can search $5 \times (8!)$ permutations of the shuffled data to find the closest match against the plaintext. This value is feasible to brute-force, and the search technique is robust against bit flips arising from memory decay.

The problem is thus reduced to identifying plaintext information contained in some part of our memory dumps. Using this technique, we found cryptographic constants in an unencrypted stub of the NAND storage of the target PLC device. Given the cryptographic features known to be supported by the device, we predicted that SHA256 hash constants would be present in the unshuffled memory data. Cryptographic constants are highly suitable for our unshuffling strategy, as they form a distinct sequence of values that would be extremely unlikely to occur by coincidence.

3) Finding Matching Hamming Weight Sequences: To efficiently search for likely matches, we made use of an invariant — some significant property of the data that did not change with bit-shuffling. Our investigation found that Hamming weight (i.e. popcount), the number of bits with value one, was a simple and effective invariant.

First, we computed the Hamming weight for each shuffled 40-bit word formed by interleaving the five DDR2 chips. With 1-Gbit chips, the result was a sequence of length 2^{27}, roughly 130 million, which formed our search space. For each of our four ECC variants((1) little-endian, (2) little-endian + bit reverse, (3) big-endian, and (4) big-endian + bit reverse), we computed the Hamming weight of each 40-bit word. The result was a target sequence of 64 values, one for each cryptographic hash constants. The problem then reduced to finding the closest match for one of the four target sequences inside the search space, which was feasible to perform by brute force.

Computing the Hamming weight sequence on a bytewise basis allowed us to quickly identify the byte ordering of the four data chips. A brute force search on all $5 \times (8!)$ possible permutations identified the particular bit shuffling used by the SIMATIC S7-1500.

D. Correlating Physical to Virtual Memory

The results of unshuffled cryo-mechanical attack were DRAM content in physical address space. Most relatively high performance embedded processors, including the main SoC on the SIMATIC S7-1500, use a memory management unit (MMU). Properly reverse-engineering any code retrieved from DRAM requires knowledge of the virtual address space being mapped to the physical location of the code. This leads to our next challenge: correlating physical memory to virtual memory.

The memory mapping for this MIPS [48] architecture is specified in the ISA documentation [49]. The kernel mode addresses have a fixed mapping to physical addresses, while the rest can be mapped by the MMU. The bootloader loads the OS and runs in kernel mode. By analyzing the bootloader, we found the physical to virtual address mapping, which allowed us to access the plaintext memory content for further security analysis.

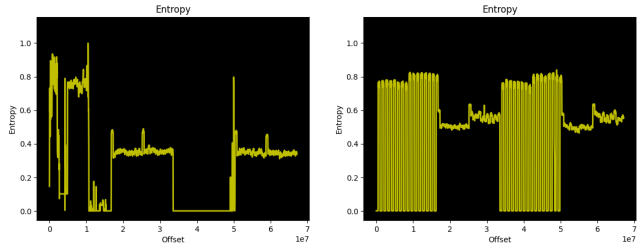

Fig. 13. Left, entropy graph of good LPDDR1 memory extraction. Right, entropy graph of bad LPDDR1 memory extraction. By comparing these two entropy graphs, we can see the entropy abnormality regions on the right, and quickly rule out this memory extraction.

VI. Experimental Results

We performed cryo-mechanical attacks on both the LPDDR1 memory chip and the five DDR2 memory chips on the Siemens S7-1500 PLC. We evaluated our results by analyzing the accuracy of memory extractions for both chipsets.

A. LPDDR1 Extraction Results

For capturing LPDDR1 memory, our experimental setup was as follows: We desoldered the LPDDR1 chip from the Siemens S7-1500 PLC and placed a socket over the contact pads. The device and the FPGA platform were both mounted onto the CNC machine with their LPDDR1 chip pads oriented identically. For each memory extraction, we first immersed the chip into the dry ice bath until it achieved thermal equilibrium. We placed the chip onto the S7-1500 PLC using the CNC platform and pneumatic piston, powered on the device, and waited 5000 ms for the display controller to fully boot up. Then, we powered off the device, immediately transferred the chip to the FPGA-based memory extraction system, and performed a readout of the memory contents.

In total, we conducted 17 trials of the LPDDR1 memory extraction process. During the first 10 trials, we made iterative adjustments to the CNC alignment and motion planning, memory cooling time, and memory transfer duration to achieve a stable state of the extraction process. This state was determined by the identity of the entropy [50] graphs of the extracted contents across multiple runs as shown in Figure 13. Once we identified the optimal parameters, we utilized them for the remaining trials, which resulted in valid outcomes. Furthermore, we combined multiple valid memory extractions from the final seven trials to produce a flawless memory extraction. We found that no further processing was necessary to interpret the data because only PLC screen controller ARM code appeared in the LPDDR1 memory images.

B. DDR2 Extraction Results

The arrangement of five dedicated DDR2 memory chips on the device's PCB is shown in Figure 14. For capturing DDR2 memory, we desoldered each of the five DDR2 chips on the Siemens S7-1500, and installed a socket for each one. The device and the FPGA platform were mounted onto the machine with the appropriate orientation. The memory extraction procedure was identical to that for the LPDDR1 chip, except for the bootup times used. We performed extractions for each of the DDR2 chips at two times: 1000 ms and 5450 ms after power-on, with the timings chosen according to the methods described in Section V-A. After each memory chip extraction was finished, the chip was soldered back on to prepare the device for extracting next DDR2 chip. To ensure accurate results, we performed three trial runs for each DRAM chip to iteratively adjust the CNC platform alignment. We then conducted five identical runs for data extraction, separately obtaining valid extraction data for the 5450ms and 1000ms timings. Since we had precise control of the timing for memory extractions, reconstruction memory image from five dedicated chip content was feasible.

With the live memory extraction containing the decrypted firmware content, we were able to perform reverse engineering and security assessment of the Siemens S7-1500 PLC. Architectural Root-of-Trust vulnerabilities (CVE-2022-38773 [51], [52]) were discovered with the help of this extracted memory content. Siemens confirmed these vulnerabilities and disclosed them in January 2023. For the sake of responsible disclosure, we chose not to include any technical details about these vulnerabilities, focusing instead on presenting our apparatus working on real-world devices. The main emphasis of the paper is our overall approach, not individual vulnerabilities.

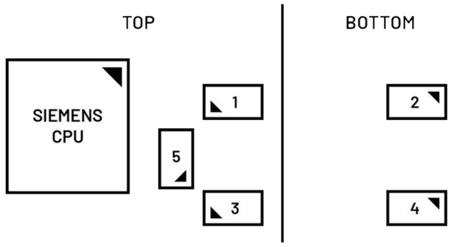

Fig. 14. Relative locations of the five DDR2 memory chips on the Siemens S7-1500 main PCB. (1)-(5) are the DDR2 memory chips with the orientation markers. The new cryo-mechanical memory extractions were performed for one memory chip at a time.

VII. Discussion

In the case of the Siemens S7-1500 PLC, we performed cryo-mechanical attacks on LPDDR1 and DDR2 DRAM chips. The main firmware is encrypted in the nonvolatile memory and decrypted during the runtime across five individual DDR2 DRAM chips. The data line for each physical DDR2 chip is shuffled. In our procedure for reconstructing the full memory image from the contents of individual chips, we assumed that the memory contained SHA256 hash constants, which can be found in the unencrypted stub in NAND storage. To generalize the dataline unshuffle process for other embedded device other than Siemens S7-1500, it is necessary to obtain alternative samples of known plaintext. Fortunately, acquiring such plaintext samples is relatively straightforward, as they can be derived from constants used by popular encryption algorithms, encrypted ciphertext, or known text strings, making the unshuffle process more adaptable and efficient.

The success rate of individual cryo-mechanical memory extraction operations can be influenced by factors such as accurate alignment, temperature control and ambient temperature during the extraction process, and the specific memory module's susceptibility to the Memory Remanence Effect. However, the objective is to achieve deterministic memory extraction through a combination of multiple repeatable and automated memory extraction. By adopting our presented cryo-mechanical memory extraction procedure above, the overall success rate of the procedure can be consistently maintained at 100%.

The chip alignment, both before and after the transfer between the target device and the FPGA-based readout platform, is a crucial aspect of the process. Common BGA pitches for DRAM vary from 0.4mm to 1.27mm. The presence of more pins and smaller pads may increase the difficulty of manually aligning the chip by hand when it is cooled down. Using a simple manual vacuum suction pen for alignment is nearly impossible due to the need for a reliable and repeatable procedure. Moreover, the moisture condensation on the supercooled chip can compromise the vacuum suction. Therefore, we employ a CNC machine to automate the alignment procedures, ensuring precise and consistent positioning of the chip during the transfer process.

Additionally, to perform our new cryo-mechanical memory extraction, there is a physical requirement on the target device — the DRAM chip must be unobstructed by other components. In the Siemens S7-1500 PLC case, the assembly of the module is a sandwich structure of two PCBs. The DRAM chip is on the middle layer, blocked by the PCB. We designed an adapter board to access the DRAM chip without affecting the PLC's operation. Base on the research, we recommend placing sensitive chips in more difficult-to-access locations on the PCB to increase security.

The hardware setup for our new cryo-mechanical attack is inexpensive. In total, we estimate that the attack can be performed with roughly $2000 USD worth of hardware. This attack demonstrates the feasibility of extracting encrypted firmware without the need for exotic lab equipment. Extending our presented method to a new target primarily involves adapting the cryo-mechanical apparatus to accommodate the target's specific memory module type and configuration. This effort is minimal, as our approach is designed to be modular and easily adjustable. Only new conductive elastomer socket and CNC operation programming are needed. Our attack suggests that even small cybercrime groups or amateur security enthusiasts may have access to capabilities once thought exclusive to well-funded adversaries.

VIII. Conclusion

The cold boot attack is a long-known physical vulnerability of DRAM, but existing attacks generally depend on factors such as code execution on the target device or standardized form factors like DIMMs. We introduced a cryo-mechanical methodology that allows such attacks to be performed on nearly any embedded system with DRAM without previous cold boot attack limitations of either removable memory or code execution with debugging, done through physically transferring discrete memory chips, reading out data with an FPGA, and reconstructing a memory image from the results. Our experimental results on a Siemens SIMATIC S7-1500 PLC demonstrated that we could recover the contents of encrypted firmware binaries which allowed us to perform security analysis of the encrypted firmware. We are actively working on adapting the cryo-mechanical memory extraction procedure to the DDR3 chip for Cisco IP phones [53]. The setup process closely resembles our work with LPDDR1 and DDR2, so we've omitted the description from the main content. Based on our current experimental results, we were able to identify code and data from ARM TrustZone [54] memory in extracted data and further analysis will be performed in subsequent research. We believe that performing cryo-mechanical attacks on DDR4 and DDR5 is similarly feasible, but with one caveat — it requires expensive FPGA-based memory readout platform (around $10,000 USD). However, we expect the cost of those FPGA boards to come down as DDR4 and DDR5 memory becomes more widely used in embedded systems in the future.

Modern embedded devices increasingly leverage cryptographic features, such as secure boot and firmware encryption, to prevent unauthorized code execution, maintain user privacy, and protect valuable intellectual property. By their very nature, however, embedded systems tend to be deployed in situations more vulnerable to physical attacks. In light of our findings, even secure boot and firmware encryption may not truly provide protection against a sufficiently dedicated attacker. We hope that our work will motivate others to rethink embedded security and develop more advanced security techniques.

Acknowledgments

The authors would like to thank Jack Zheng and Aleksey Nogin from Red Balloon Security for their contributions to this research.

References

[1] L. Bogdanov, "Multiple microcontroller programming using the swd interface," in *2020 XXIX International Scientific Conference Electronics (ET)*, 2020, pp. 1–4.

[2] IEEE 1149.1 Working Group and others, "IEEE standard for test access port and boundary-scan architecture," *IEEE Std 1149.1-2013 (Revision of IEEE Std 1149.1-2001)*, pp. 1–444, 2013.

[3] A. Kurniawan, *Internet of Things Projects with ESP32: Build exciting and powerful IoT projects using the all-new Espressif ESP32*. Packt Publishing Ltd, 2019.

[4] N. H. Tollervey, *Programming with MicroPython: embedded programming with microcontrollers and Python*. O'Reilly Media, Inc., 2017.

[5] P. Gutmann, "Data remanence in semiconductor devices." in *USENIX Security Symposium*, 2001, pp. 39–54.

[6] Ubuntu Privacy Remix Team, "Security analysis of TrueCrypt 7.0 a with an attack on the keyfile algorithm," Tech. Rep., 2011, [Online; accessed 20-March-2023]. [Online]. Available: https://digi77.com/software/public/truecrypt/truecrypt_7.0a-analysis-en.pdf

[7] S. Türpe, A. Poller, J. Steffan, J.-P. Stotz, and J. Trukenmüller, "Attacking the bitlocker boot process," in *International Conference on Trusted Computing*. Springer, 2009, pp. 183–196.

[8] Teledyne Lecroy, "Ddr2 test solutions qphy-ddr2," https://cdn.teledynelecroy.com/files/pdf/qphy-ddr2-datasheet.pdf, 2020, [Online; accessed 19-August-2022].

[9] Keysight Technologies, "W3630a series ddr3 bga probes for logic analyzers and oscilloscopes," https://www.keysight.com/us/en/assets/7018-01987/data-sheets/5990-3179.pdf, 2017, [Online; accessed 20-March-2023].

[10] ——, "U4154a axie-based logic analyzer module," https://www.keysight.com/us/en/assets/7018-02901/data-sheets/5990-7513.pdf, 2017, [Online; accessed 20-March-2023].

[11] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J. Appelbaum, and E. W. Felten, "Lest we remember: cold-boot attacks on encryption keys," *Communications of the ACM*, vol. 52, no. 5, pp. 91–98, 2009.

[12] S. F. Yitbarek, M. T. Aga, R. Das, and T. Austin, "Cold boot attacks are still hot: Security analysis of memory scramblers in modern processors," in *2017 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE, 2017, pp. 313–324.

[13] JEDEC, "JEDEC MO-269J whitepaper," 2014, [Online; accessed 20-March-2023]. [Online]. Available: https://www.jedec.org/sites/default/files/docs/MO-269J.pdf

[14] Freescale Semiconductor, "Hardware and layout design considerations for DDR memory interfaces," 2004, [Online; accessed 20-March-2023]. [Online]. Available: https://www.nxp.com/docs/en/application-note/AN2582.pdf

[15] M. Catrambone, "Routing DDR4 interfaces quickly and efficiently," 2016, [Online; accessed 20-March-2023]. [Online]. Available: https://www.cadence.com/content/dam/cadence-www/global/en_US/documents/tools/pcb-design-analysis/pcb-west-2016-47-rte-ddr4-interfaces-cp.pdf

[16] M. Gruhn and T. Müller, "On the practicability of cold boot attacks," in *2013 International Conference on Availability, Reliability and Security*. IEEE, 2013, pp. 390–397.

[17] J. Bauer, M. Gruhn, and F. C. Freiling, "Lest we forget: Cold-boot attacks on scrambled DDR3 memory," *Digital Investigation*, vol. 16, pp. S65–S74, 2016, dFRWS 2016 Europe. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1742287616300032

[18] L. Cojocar, K. Razavi, C. Giuffrida, and H. Bos, "Exploiting correcting codes: On the effectiveness of ecc memory against rowhammer attacks," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 55–71.

[19] M. Seaborn and T. Dullien, "Exploiting the dram rowhammer bug to gain kernel privileges," *Black Hat*, vol. 15, p. 71, 2015.

[20] T. Müller and M. Spreitzenbarth, "Frost," in *International Conference on Applied Cryptography and Network Security*. Springer, 2013, pp. 373–388.

[21] Y.-S. Won, J.-Y. Park, D.-G. Han, and S. Bhasin, "Practical cold boot attack on IoT device-case study on raspberry pi," in *2020 IEEE International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA)*. IEEE, 2020, pp. 1–4.

[22] Y.-S. Won, S. Chatterjee, D. Jap, A. Basu, and S. Bhasin, "DeepFreeze: Cold boot attacks and high fidelity model recovery on commercial EdgeML device," in *2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*. IEEE, 2021, pp. 1–9.

[23] Statista Research Department, "Global PLC market share as of 2017, by manufacturer," 2017, [Online; accessed 25-July-2022]. [Online]. Available: https://www.statista.com/statistics/897201/global-plc-market-share-by-manufacturer/

[24] Z. Weitao, Z. Haibing, K. Xiaole, and M. Dehong, "Study on rework process of bga components," in *2015 16th International Conference on Electronic Packaging Technology (ICEPT)*. IEEE, 2015, pp. 681–684.

[25] A. Trikalinou and D. Lake, "Taking DMA attacks to the next level," *BlackHat USA*, 2017.

[26] Xilinx, "Zynq-7000 SoC data sheet: Overview," 2018, [Online; accessed 23-July-2022]. [Online]. Available: https://docs.xilinx.com/v/u/en-US/ds190-Zynq-7000-Overview

[27] ——, "7 series FPGAs data sheet: Overview," 2020, [Online; accessed 23-July-2022]. [Online]. Available: https://docs.xilinx.com/v/u/en-US/ds180_7Series_Overview

[28] ——, "7 series FPGAs memory interface solutions user guide UG586," 2012, [Online; accessed 23-July-2022]. [Online]. Available: https://docs.xilinx.com/v/u/1.4-English/ug586_7Series_MIS

[29] J. Romo, "DDR memories comparison and overview," [Online; accessed 14-October-2022]. [Online]. Available: https://www.nxp.com/docs/en/supporting-information/BeyondBits2article17.pdf

[30] S. S. Math, R. Manjula, S. Manvi, and P. Kaunds, "Data transactions on system-on-chip bus using AXI4 protocol," in *2011 International Conference on Recent Advancements in Electrical, Electronics and Control Engineering*. IEEE, 2011, pp. 423–427.

[31] John, "Best 3040 CNC routers: Affordable & quality," 2023, [Online; accessed 20-March-2023]. [Online]. Available: https://mellowpine.com/cnc/best-3040-cnc-routers/

[32] D. Y. Goswami, *The CRC handbook of mechanical engineering*. CRC press, 2004.

[33] TEKNIC, "Clearpath - integrated servo system," [Online; accessed 14-October-2022]. [Online]. Available: https://teknic.com/model-info/CPM-MCPV-3411P-RLN/?model_voltage=75VDC

[34] ——, "Clearpath user manual rev. 3.22," https://www.teknic.com/files/downloads/clearpath_user_manual.pdf, 2022, [Online; accessed 19-August-2022].

[35] SMC, "Dual rod cylinder series CXSJ/CXS," [Online; accessed 20-March-2023]. [Online]. Available: https://www.smcpneumatics.com/pdfs/CXS.pdf

[36] All In Win Technology & Trade Co., Ltd., "Conductive elastomer IC test sockets for BGA," 2023, [Online; accessed 26-March-2023]. [Online]. Available: https://www.all-in-win.com/products/ic-socket

[37] DediProg, "IC test socket," 2021, [Online; accessed 16-March-2023]. [Online]. Available: https://www.dediprog.com/page/semiconductor-ic-test-socket

[38] Ironwood Electronics, "GHz BGA & QFN/MLF Sockets," 2021, [Online; accessed 16-March-2023]. [Online]. Available: https://www.ironwoodelectronics.com/products/ghz-elastomer-sockets/

[39] Chemtronics, "Ultimate guide to diagnostic freeze spray," https://www.chemtronics.com/ultimate-guide-to-diagnostic-freeze-spray, 2022, [Online; accessed 19-August-2022].

[40] J. Van den Herrewegen, D. Oswald, F. D. Garcia, and Q. Temeiza, "Fill your boots: Enhanced embedded bootloader exploits via fault injection and binary analysis," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 56–81, 2021.

[41] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE transactions on computers*, vol. 51, no. 5, pp. 541–552, 2002.

[42] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO'99: 19th Annual International Cryptology Conference Santa Barbara, California, USA, August 15–19, 1999 Proceedings 19*. Springer, 1999, pp. 388–397.

[43] S. Mangard, E. Oswald, and T. Popp, *Power analysis attacks: Revealing the secrets of smart cards*. Springer Science & Business Media, 2008, vol. 31.

[44] J.-J. Quisquater and D. Samyde, "Electromagnetic analysis (ema): Measures and counter-measures for smart cards," in *Smart Card Programming and Security: International Conference on Research in Smart Cards, E-smart 2001 Cannes, France, September 19–21, 2001 Proceedings*. Springer, 2001, pp. 200–210.

[45] D. Genkin, A. Shamir, and E. Tromer, “Rsa key extraction via low-bandwidth acoustic cryptanalysis,” in *Advances in Cryptology–CRYPTO 2014: 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I 34*. Springer, 2014, pp. 444–461.

[46] T. Sugawara, Y.-i. Hayashi, N. Homma, T. Mizuki, T. Aoki, H. Sone, and A. Satoh, “Spectrum analysis on cryptographic modules to counteract side-channel attacks,” in *EMC*, vol. 9, 2009, pp. 21–24.

[47] D. Agrawal, B. Archambeault, J. R. Rao, and P. Rohatgi, “The EM side—channel (s),” in *International workshop on cryptographic hardware and embedded systems*. Springer, 2002, pp. 29–45.

[48] G. Kane and J. Heinrich, *MIPS RISC architectures*. Prentice-Hall, Inc., 1992.

[49] MIPS, “Memory map and the MIPS privileged resource architecture,” 2010, [Online; accessed 21-July-2022]. [Online]. Available: https://training.mips.com/basic_mips/PDF/Memory_Map.pdf

[50] K. S. Han, J. H. Lim, B. Kang, and E. G. Im, “Malware analysis using visualized images and entropy graphs,” *International Journal of Information Security*, vol. 14, no. 1, pp. 1–14, 2015.

[51] National Vulnerability Database, “CVE-2022-38773 Detail,” 2022, [Online; accessed 20-March-2023]. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2022-38773

[52] Y. Wu, G. Skipper, and A. Cui, “Uprooting trust: Learnings from an unpatchable hardware root-of-trust vulnerability in Siemens S7-1500 PLCs,” in *2023 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. IEEE, 2023, to appear.

[53] A. Cui and R. Housley, “{BADFET}: Defeating modern secure boot using {Second-Order} pulsed electromagnetic fault injection,” in *11th USENIX Workshop on Offensive Technologies (WOOT 17)*, 2017.

[54] S. Pinto and N. Santos, “Demystifying arm trustzone: A comprehensive survey,” *ACM computing surveys (CSUR)*, vol. 51, no. 6, pp. 1–36, 2019.