DAVE Embedded Systems

BRX-WP001: Real-timeness, system integrity and TrustZone® technology on AMP configuration

White Paper

DAVE Embedded Systems www.dave.eu info@dave.eu

Table of Contents

White Paper

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

1 Introduction

Because of widely available Internet connectivity, nowadays security concerns are not longer limited to PCs, servers and workstations but have become common to many embedded systems as well. Several hardware and software technologies have been developed to deal with this kind of challenges. ARM® TrustZone® technology is one of these. As stated in [1], Xilinx Zynq-7000 AP SoC natively supports TrustZone® technology, since it integrates dual-core ARM® Cortex™-A9 MPCore™ processor.

Even if this technology has been conceived primarily to address security issues, embedded systems designers can leverage it to implement innovative configurations, satisfying different in nature requirements that typically arise in industrial applications and deep embedded systems[a]. Two of such requirements are real-timeness and system integrity[b].

This White Paper describes the TrustZone-based solution that DAVE Embedded Systems has implemented to meet all these requirements on BORA and BORA Xpress platforms. A technical description of the adopted approach is provided. Also, performance and characterization tests are detailed and considerations about future developments and improvements are included.

It should be remembered that this solution can be considered as a sort of natural evolution on the traditional AMP configuration described in [3]. For this reason, reading of this document is highly recommended.

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

2 Limitations of traditional configurations

Xilinx Zynq AP architecture provides unprecedented possibilities in terms of integration. In industrial world applications, this is often leveraged to combine on a single chip the implementation of real-time tasks with generic software applications and functionalities that don't have specific requirements in terms of real-timeness[c]. In addition, the flexibility offered by the FPGA - known as Programmable Logic or PL for short - allows system designers to implement in hardware custom IPs to add new interfaces and peripherals or to move processing modules from the software to the hardware realm[d].

The following list recaps the typical requirements that such systems must meet. This list has been compiled on the base of real world use cases (specifically medical, transportation, automation and telecom applications):

- REQ1 Real-time and non real-time tasks integration: On the same physical component it is required that non real-time and real-time worlds coexist; this allows overall design simplification, BOM reduction and tighter integration.
- REQ2 The non real-time world - denoted as W2 in the rest of the document - is based on a well-known operating system such as linux; this operating system is also denote as GPOS (general purpose operating system).
- REQ3 The real-time world - denoted as W1 in the rest of the document - is based on a RTOS or a bare metal executable.
- REQ4 Inter-world communication and data sharing: the two worlds must have the capability to:
    1. communicate via asynchronous mechanisms
    2. share data

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

- REQ5 Integrity: W1 - also known as Secure world or Trust world - must guarantee a high reliability level, no matter how the other world behaves; in other words, W1 can not be altered by any kind of actions taken by the code executed in W2 (also called Non-secure world or Non-trust world).
- REQ6 Boot order: W1 must be the first world to come up.
- REQ7 Master-slave relationship: once the system has completed boot process, a master-slave relationship must be established between W1 and W2, in the sense that W1 must have complete control of W2 world (for instance W1 must be able to force the complete reboot of the GPOS).
- REQ8 L2 cache availability on W2 side: Basic AMP configurations does not support L2 cache. Generally speaking, this is not an issue on W1 side because
    - typical code and data footprints are relatively limited, thus performances are generally not affected (this is not true instead for W2 world where, in case L2 is not available, overall GPOS performances may decrease noticeably)
    - L2 may affect predictability of execution time significantly.

Traditional AMP [3] configuration satisfies REQ1 through REQ4. REQ5 through REQ8 are not satisfied instead. About integrity, for example, an application with root privileges could access memory regions that are supposed to be exclusively accessed by code executed in W1. This may lead to unpredictable behaviors and to potentially catastrophic consequences. This is where TrustZone technology comes to help: it establishes a sort of barrier between the two worlds and prevents W2 code from unauthorized accesses to certain regions of the processor's addressing space.

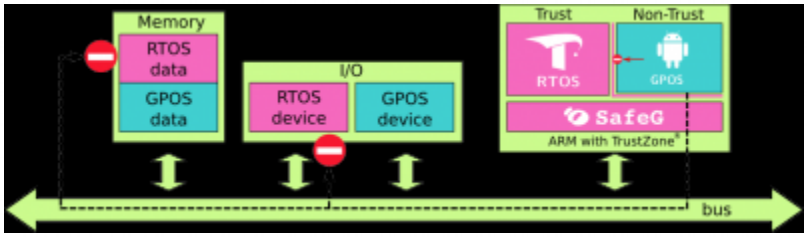
DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

3 TrustZone-based approach

This section describes in detail the solution implemented by DAVE Embedded Systems to overcome the limitations of basic AMP configuration a to satisfy all the requirements listed in section Limitations of traditional configurations.

3.1 Overview

The major difference with respect to the traditional AMP configuration is the use of a software monitor, specifically a customized version of TOPPERS SafeG [4] [5] [6] .



As shown in the picture, the monitor can be viewed as a software layer that lies between Trust/Non-trust worlds and underlying hardware. The monitor is responsible for:

- enabling and initializing TrustZone in order to protect regions that must not be accessible by Non-secure world
- setup data structure and exception handlers needed for context switch and Secure Monitor Call (SMC)
- start the trusted OS

Later, once the trusted OS is ready, it will do a specific SMC that will do the context switch that will start the non-trusted OS.

About operating systems, Linux has been chosen for Non-trust

White Paper

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

world, while FreeRTOS has been selected for the Trust world. At the time of this design, the Linux/FreeRTOS combination has proven to be the most appealing for the majority of applications that this solution addresses. Nevertheless different combinations are possible[e].

About the multi-processing scheme, the two Zynq core are assigned statically to the two world (core0 to Linux, core1 to FreeRTOS). This allows to:

- simplify the whole system implementation
- reduce RTOS latency (because there's never need of non-trusted to trusted context switch)

From the memory point of view:

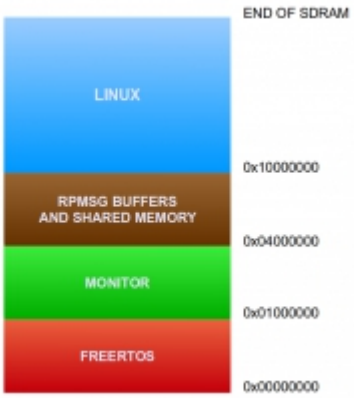
- the main memory is statically partitioned (by the monitor) into tree sections:
  - a non-trusted private area (protected at MMU-only level from trusted access)
  - a trusted private area (protected at TrustZone level by non-trusted access)
  - a shared memory area, marked as non-trusted

These choices lead to the configuration depicted here.

3.2 System memory partitioning

System memory partitioning is shown by the following picture.

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu



3.3 Boot process

The boot process consists of several stages that are detailed in the following list.

1. reset signal is deasserted and core #0's Program Counter is set to reset vector address
2. The first piece of code executed by the processor is BootROM. Depending on bootstrap configuration pins, First Stage Boot Loader (FSBL in the rest of the document) image is retrieved from a specific non-volatile memory by BootROM and stored into on-chip memory (OCM).
3. FSBL performs basic hardware initializations (including SDRAM subsystem) and retrieves U-Boot bootloader image

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

4. U-Boot
- completes hardware initializations
- retrieves the following binary images and store them into SDRAM:
    - monitor
    - trusted code (FreeRTOS image in our case)
    - non-trusted code (linux kernel image and Device Tree Blob in our case).
- gives monitor the control.

5. monitor code
    - initializes TrustZone subsystem
    - enables both cores, setting up all the data structure required by TrustZone
    - gives trusted code the control of the machine.
6. FreeRTOS kernel is initialized and real-time tasks are started. Under the control of the tasks running on top of the RTOS kernel, the non-trusted (NT for short) code is started[f].

3.4 Inter-world communication

Even if it is technically possible to implement a system where W1 and W2 are completely isolated, the majority of real applications need a communication mechanism between the two worlds[g]. Several options are available to implement such a mechanism, each of which having different pros and cons. Exhaustive comparison of all of them is beyond the scope of this paper. Nevertheless some aspects will be briefly discussed and some useful links will be provided to study this notable subject in more depth.

The following criteria have been taken into account to determine how to select the communication mechanism:

1. acceptance into the mainline linux kernel
2. possibility to customize the implementation in order to

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

control the degree of isolation between the two worlds.

The first criteria guarantees future maintainability on Linux side. Generally speaking, it is expected that the GPOS needs relatively frequent maintenance activities in order to add new functionalities or fix vulnerabilities and bugs. In contrast, RTOS side is typically more stable and easier to maintain. Therefore, selecting a communication mechanism that is included in mainline Linux kernel is preferable from the point of view of overall system maintainability.

The second criteria is very important because, as discussed in following sections, it is absolutely crucial that the communication channel - that affects directly the degree of isolation between W1 and W2 - is very flexible in order to adjust it to application-specific requirements as needed.

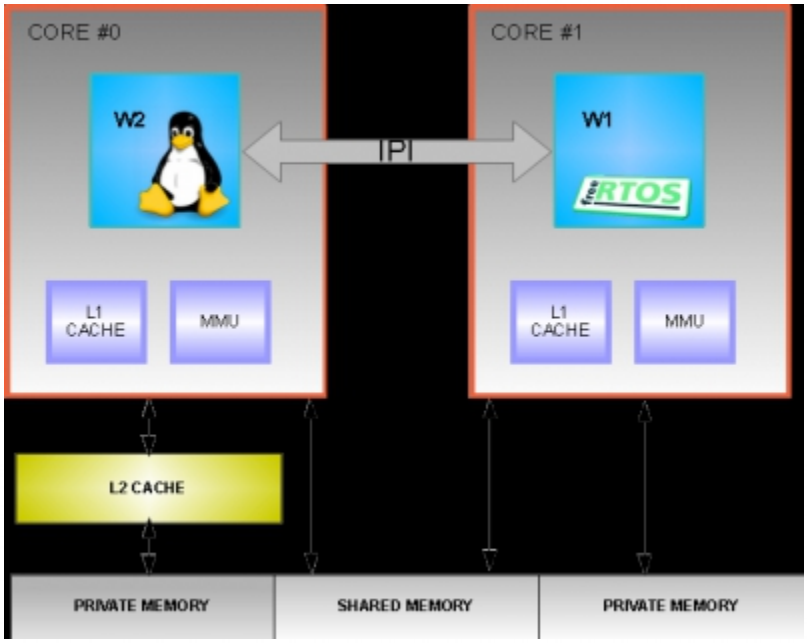
Different solutions have been evaluated, including but not limited to OP-TEE [7], dualoscom [8], RPMsg [9], [10] and OpenAMP [11]. The choice has fallen on RPMsg that has been considered the best compromise among the available options. It should be recalled that this choice is reversible, in the sense that if application-specific requirements can not be met by RPMsg, it can be replaced by a different communication scheme.

3.5 L2 cache management

As specified by REQ8, L2 cache must be able to be enabled (at least) on W2 side. As stated in [12] and [13], L2 cache - in contrast to L1 - is a shared resource in Zynq implementation. As such, in case both cores have to use it, specific techniques have to be implemented to handle it properly in dual-OS AMP configuration. In principle the approach that has been adopted allows to implement different strategies related to L2 cache management. The actual configuration that has been used to

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

conduct the tests described in this section assigns the whole L2 cache to the W2 world, as depicted in the following picture.



This configuration has been chosen for several reasons:

- Linux performance are not affected by this dual-OS solution when accessing it's private memory
- FreeRTOS determinism is granted, because is using only non-shared resources (excluding the SOC L3 bus)
- access to shared memory is uncached or only L1 cached (the latter forces SCU SMP mode usage).

For sake of completeness, it is recalled that it is also possible to enable L2 cache in W1 too, without breaking REQ5, because ARM PL310 L2 cache controller support the TrustZone technology and does not allow the non-trusted OS (W2) to access trusted OS (W1) cached data. However, enabling L2

DAVE Embedded Systems
DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

cache for W1 may improve its computational performance but, in general, reduces real time determinism as well. In case L2 cache unavailability on W1 side is unacceptable, on-chip memory (OCM) can come to help to mitigate this issue.

3.6 Leveraging OCM

The use of L1 and L2 is strictly related to performance of ARM cores. There's also another precious resource that can increase core performance without affecting determinism, which is the OCM.

In SOCs, OCM is usually tightly coupled to ARM core, providing access performances that are comparable to L2 cache. This allows to leverage it to compensate for the unavailability of L2 memory.

The most common scenario is to:

- restrict OCM access to W1 only (again, this can be done with TrustZone)
- move the most latency sensitive code - usually vectors and ISRs - inside its memory ranges
- prevent this memory range to be L1-cached (because this usually does not increase the performance significantly but may waste precious L1 memory lines).

4 Characterization and performance tests

Some basics tests have been conducted to characterize the system configured as described above. The figure the tests

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

focus on is the interrupt latency on W1 realm. This value has been measured under different system load conditions to verify if and how the non real-time world may influence the real-time world.

About Linux side, two load conditions have been considered:

- idle
- Google stressapptest (SAT for short) [14] running to stress SDRAM memory and SD I/O.

About RTOS side:

- idle
- memory intensive task; two subcases, in turn, have been considered to evaluate the impact of the L2 cache unavailability.

The RTOS memory task access an array in main memory of two different sizes:

- the smaller is half of L1 size (16KiB)
- the larger is 4 times the L1 size (128KiB)

The main task on the RTOS side is the latencystat demo provided with AN-BELK-001 [3], which:

- programs PS TTC timer as freerun, triggering an interrupt on overflow
- inside the overflow ISR the TTC counter is read: this counter reports the number of ticks elapsed between the event (overflow) and the handler itself, in other words the interrupt latency
- after a while the TTC is reprogrammed and interrupt is enabled again, to trigger another event
- those latency counters are collected into an array
- the Linux-side application, by default after 10 seconds, stops the RTOS task which sends the array data over RPMsg
- the Linux-side application collects the data and display the mininum, maximum and average latency measured

DAVE Embedded Systems
DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

The following table summarize the test results (all timing are given in ns)

Latency	Linux idle	Linux SAT		
	RTOS idle	RTOS idle	RTOS 16k	RTOS 128k
min	287	287	287	1268
avg	287	296	205	2024
max	548	539	575	3050

5 Conclusions and future work

The following conclusions can be drawn from the test results:

- Real-timeness of W1 realm is preserved in any condition, since Linux activity on CPU/memory/SD virtually has no influence on RTOS latency.
- Moderate RTOS activity has no impact on latency.
- As expected, in case intensive memory activity is performed on RTOS side, data/instruction cache misses increase significantly resulting in higher latency.

5.1 Future work

Future work will first focus on an additional feature that has not been included in the requirement list but that is undoubtedly useful in several applications. We are referring to the possibility of performing a complete reboot of the GPOS under the control of the RTOS, while this keeps operating normally. For instance this can be exploited when the RTOS needs to work as software watchdog for W2 activity: in case no activity is detected for a certain period of time, GPOS can be shutdown and rebooted.

DAVE Embedded Systems
DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

Another aspect that should be investigated in more depth refers to the effects of the communication between W1 and W2 on the IRQ latency and the integrity of the real-time world. This matter is strictly related to the degree of isolation between the two worlds. In this work a strong-isolation approach has been adopted, meaning that

- no data is exchanged during the execution of the IRQ latency measurement
- it has been implicitly assumed that data sent from W2 to W1 can not compromise the integrity of the trust domain.

These assumption may be not verified in real applications, however specific techniques can be implemented to manage these situations (see for example [8] and [15]).

1. These kind of requirements are often totally independent of Internet connectivity
2. In this context, for the terms **integrity** and **security** the definitions provided by [2] are in use:
   - **security** refers to a system's immunity to data disclosure or loss as a result of the unlawful electronic penetration of the system's protections and defenses
   - **integrity** denotes the certainty that a system cannot be improperly altered.

For the sake of completeness, definition of security is provided as well:

- **security** refers to a system's immunity to data disclosure or loss as a result of the unlawful electronic penetration of the system's protections and

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

defenses.

3. Network connectivity is an example of such functionalities.
4. Powerful tools have been introduced in the market recently that facilitate this process significantly. One of these is SDSoC. BORA and BORA Xpress are two of the supported hardware platforms.
5. For example TOPPERS project makes use of different RTOSes.
6. This is done via a Secure Monitor Call (referred as SMC in the rest of the document) that is handled by the monitor.
7. From a different point of view, this is a sort of break in the barrier between the two worlds. As such it poses non-trivial issues in term of integrity and timeliness in the Trust world. This matter will be covered in more detail in following sections.

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu

6 References

References

1: Yashu Gosain and Prushothaman Palanichamy, Xilinx WP429 - TrustZone Technology Support in Zynq-7000 All Programmable SoCs (v1.0), May 20, 2014
3: DAVE Embedded Systems, AN-BELK-001: Asymmetric Multiprocessing (AMP) on Bora – Linux FreeRTOS, -
4: -, TOPPERS SafeG home page (English), -, https://www.toppers.jp/en/safeg.html
5: -, TOPPERS SafeG home page (Japanese), -, https://www.toppers.jp/safeg.html
6: -, TOPPERS SafeG (Nagoya University), -
7: -, Open Portable Trusted Execution Environment, -, https://www.linaro.org/blog/core-dump/op-tee-open-source-security-mass-market/
8: Daniel Sangorrin Lopez, Advanced integration techniques for highly reliable dual-os embedded systems, 27th July 2012, http://ir.nul.nagoya-u.ac.jp/jspui/bitstream/2237/16907/1/k9888.pdf
9: -, -, -, http://omappedia.org/wiki/Category:RPMsg
10: -, -, -, https://lwn.net/Articles/464391/
11: -, -, -, https://github.com/OpenAMP/open-amp
12: John McDougall, XAPP1078 (v1.0) Simple AMP Running Linux and Bare-Metal System on Both Zynq SoC Processors, 14th February 2013
13: John McDougall, XAPP1079 (v1.0.1) Simple AMP: Bare-Metal System Running on Both Cortex-A9 Processors, 24th January 2014
14: -, -, -, https://code.google.com/p/stressapptest/
15: J. Regehr, U. Duongsaa, Preventing Interrupt Overload, 2nd May 2005, http://www.cs.utah.edu/~regehr/papers/lctes05/regehr-lctes05.pdf
2: Ed Hallett, Giulio Corradi, Steven McNeil, Xilinx WP461 - Xilinx Reduces Risk and Increases Efficiency for IEC61508 and ISO26262 Certified Safety Applications (v1.0), April 9, 2015

DAVE Srl, via Talponedo, 29/A
33080 Porcia (PN), Italy
Tel. +39.0434921215
Fax +39.04341994030
E-mail: info@dave.eu
http://www.dave.eu