AN15251

S32N55 cohorts concept and resource management

Rev. 1.0 — 6 November 2026 Application note

Document information

Information	Content
Keywords	Management, Resource, S32N55, Cohort, Cohorts
Abstract	The S32N55 is an early member of NXP’s next-generation 5 nm S32N family.

1 Overview

The S32N55 is an early member of NXP's next-generation 5 nm S32N family. The device is a real-time processor designed for multi-domain control applications and supports multi-application ASIL D control through high-performance real-time processing, hardware isolation, and virtualization support.

This application note provides an overview of the cohort concept on S32N55. It is intended as supplementary material. In case of any inconsistency, the S32N55 Reference Manual [1] and the FSS FW User Manual[2] take precedence. Additionally, Application Note[3], S32N55: Managing Isolation and XRDC Configuration via Domain Manager, provides further guidance on cohort configuration.

2 Concepts

2.1 Cohort

A cohort is a logical grouping of SoC resources that defines an isolated execution and management context, encompassing both hardware and software elements.

It defines a computing environment made up of one or more UENVs that are jointly deployed to fulfill a common functional objective. When multiple UENVs exist within a cohort, they may differ in core architecture and are not required to operate in lockstep mode.

Specifically, cohort boundaries are software-defined and hardware-enforced. This mechanism enables flexible configuration for different use cases while minimizing the latency and cost of isolation.

The S32N55 supports a two-level isolation mechanism, consisting of cohort-level and domain-level isolation. This mechanism enables more complex isolation scenarios by allowing resources to be separated between cohorts and further partitioned among domains within a cohort.

Note: *Cohorts are statically defined during system configuration. Once assigned, resources cannot be migrated from one cohort to another at runtime.*

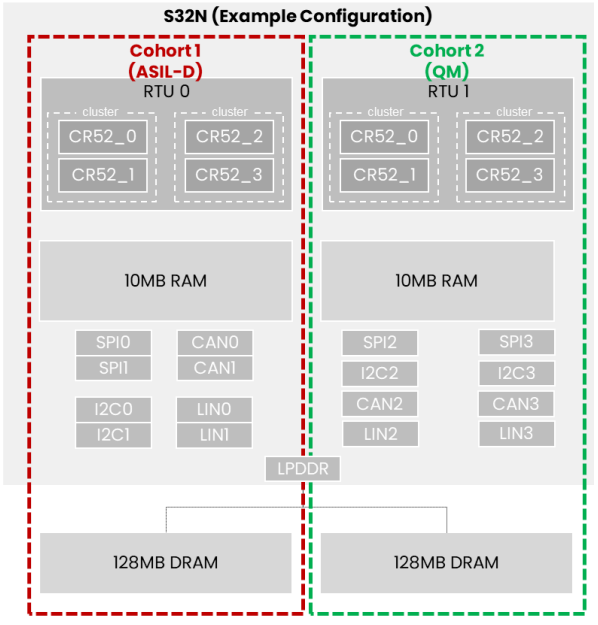


Figure 1. Example cohort configuration

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

Before describing the cohort concept in detail, it is important to clarify two key software entities: the Partition Manager (PM) and the Cohort Manager (CM). Both interact with XRDC to configure and manage resource isolation on the chip.

Table 1. Key concepts related to cohorts

Component/Concept	Description
Partition Manager	Software running in the System Base Cohort partitions device resources into additional cohorts. Within a given cohort, the corresponding Cohort Manager may further refine XRDC settings, but only within the boundaries assigned by the Partition Manager. The Partition Manager is hosted on the HSE core.
Cohort Manager	Software that configures and manages the overall operation of a cohort. Its main functions include: • Allocating UENV(s) within a cohort • Managing XRDC4 access policies to establish isolation and controlled resource sharing among UENVs within the cohort • Communicating with FSS firmware • Performing other cohort-related management tasks A Cohort Manager is dedicated to a single cohort. Its control scope is limited to that cohort and does not extend beyond it.
System Base Cohort (SBC)	A cohort that contains the FSS and is present on every S32N55 device. It is responsible for chip startup and overall system operation, and it hosts the chip's PM. This cohort always corresponds to cohort0.
Uniform Environment (UENV)	A Uniform Environment (UENV) represents a logical execution context composed of one or more cores operating with the same architecture and lockstep configuration, along with the memory regions and peripherals accessible to those cores. A UENV can only be included in at most one cohort.

2.2 Subsystems

At the hardware architecture level, the device is organized into several subsystems, such as FSS, CRS, RTU, and CIS. Subsystems are defined at design time and are not configurable by end users. Each subsystem owns a dedicated set of memory and peripherals, which are closely coupled to a specific bus fabric, with clock and power domains managed by a common control entity.

Figure 2 illustrates the primary subsystems in the S32N55 and their connectivity.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

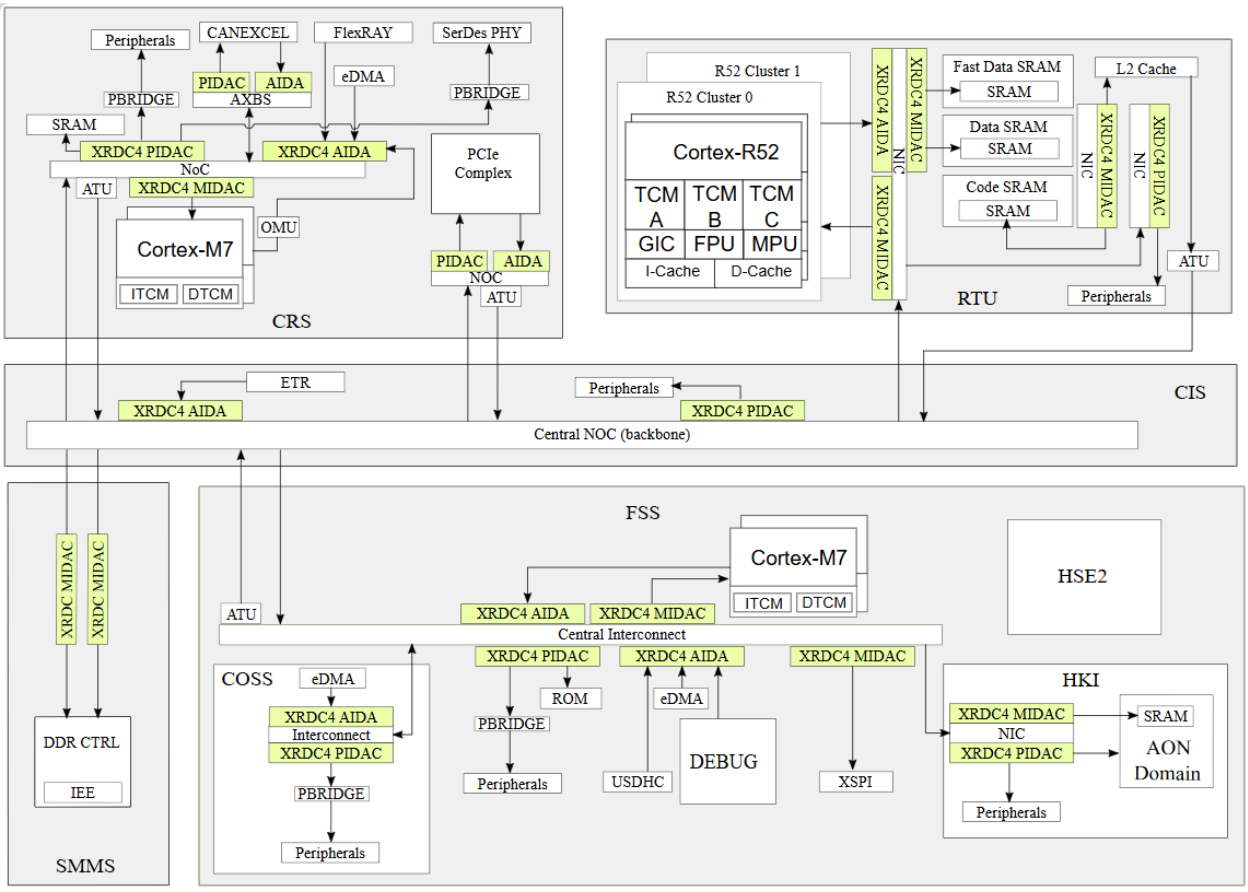


Figure 2. S32N55 subsystem diagram

2.3 ECU consolidation

On the S32N platform, the Software-Defined Vehicle (SDV) concept is enabled by decoupling software from hardware to support flexible deployment and long-term extensibility.

Hardware resources are statically organized into design-time subsystems (such as FSS, CRS, RTU, and CIS), while software functionality is dynamically structured at runtime using cohorts and UENVs.

The S32N platform allows multiple independent applications with different criticality levels to be hosted in SoC partitions, known as cohorts. Cohort boundaries are software-defined and hardware-enforced, enabling flexible resource partitioning while maintaining strong isolation.

Based on this concept, multiple traditional ECUs can be consolidated onto the S32N platform as vECUs. Each cohort can therefore be regarded as a vECU, with a dedicated set of resources assigned to support its functions. Each cohort can be independently booted, reset, and updated, which supports fault containment and prevents interference between cohorts.

Traditional virtualization technologies, such as hypervisors, remain applicable within cohorts. By using VMID and AIDA, different virtual entities running on the same physical core can be assigned different access permissions, enabling fine-grained isolation within a cohort.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

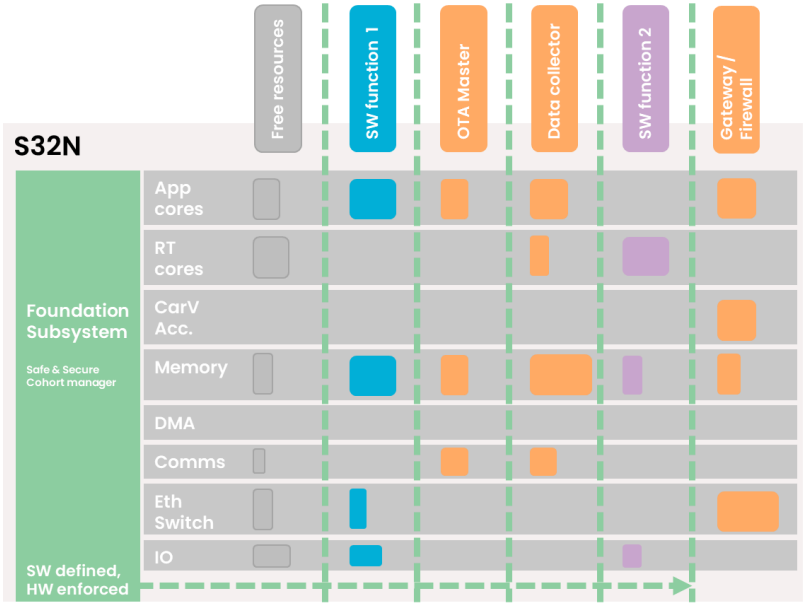


Figure 3. ECU consolidation

3 Resource management

This chapter describes the management of primary system resources, including cores, memory, peripherals, GPIOs, and software-controlled resources.

For most hardware resources, isolation is enforced by dedicated mechanisms such as XRDC and vGPIO. Power, reset, and clock control are managed through subsystem-specific control logic such as DSC.

In addition, software resources are controlled by the FSS. The typical work flow is that a non-SBC entity requests access to a software resource by sending a request to the FSS via SCMI. The FSS then verifies the validity of the requester and the requested resource, and performs the corresponding action accordingly.

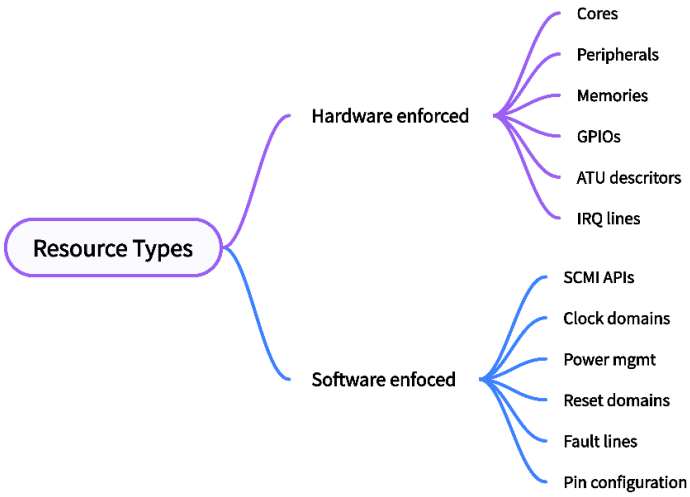


Figure 4. Resource types

The Fss_Rem_Pm module provides the configuration framework for the Partition Manager. Its hierarchical diagram is shown below:

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

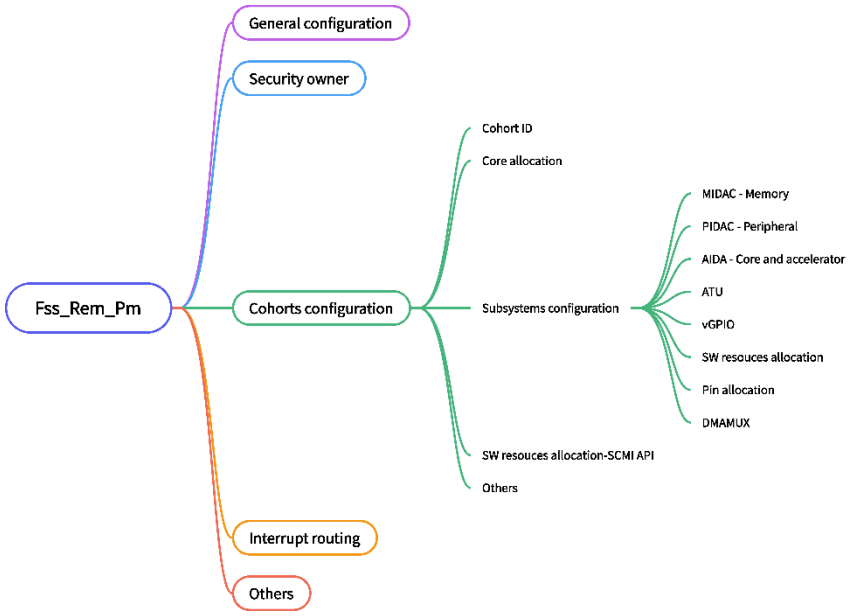


Figure 5. Fss_Rem_Pm module diagram

3.1 Hardware-enforced resources

3.1.1 Cores and memory

Processor cores and memory resources are managed by the on-chip XRDC.

Processor cores on the S32N55 platform can be partitioned into cohorts and domains by XRDC.AIDA_CR, while DMA engines and accelerators are partitioned by XRDC.AIDA_AD. All such entities are treated as initiators for accessing peripheral and memory resources.

Memory resources can be protected by XRDC.MIDAC. Each initiator carries an AID and accesses the corresponding MIDAC instance through a dedicated address path. Based on the initiator's AID and the configured access control rules, the MIDAC either grants or denies the memory access request.

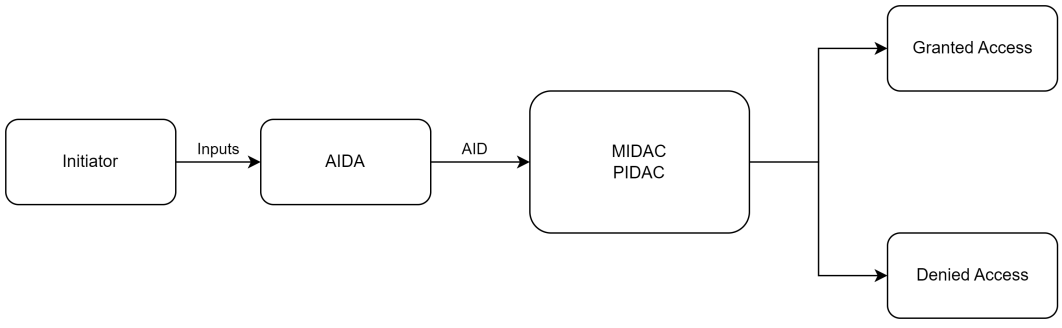


Figure 6. Diagram for XRDC

3.1.2 Peripheral categories

Within the device, peripherals are broadly classified into three categories based on their access and ownership model: dedicated peripherals, hardware-virtualized peripherals, and shared-base peripherals.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

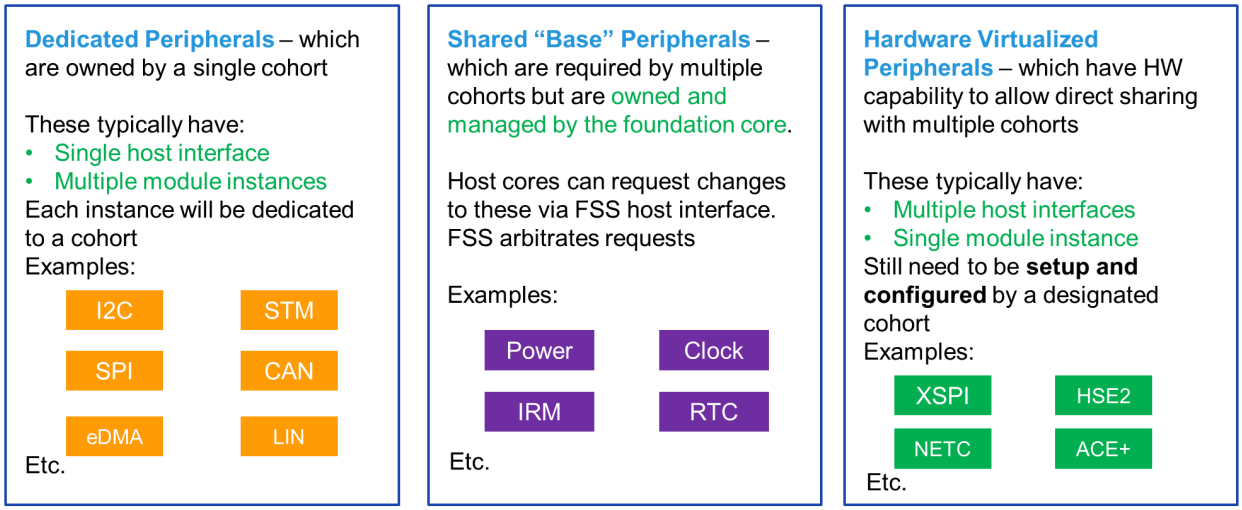


Figure 7. Peripheral categories

- Dedicated Peripherals

Dedicated peripherals are recommended to be exclusively assigned to a single cohort or domain, in order to prevent unintended access or interference from other initiators. Once configured, the hardware enforces access protection, effectively preventing unauthorized usage without introducing additional performance overhead.

- Hardware-virtualized Peripherals

A key characteristic of hardware-virtualized peripherals is that a single hardware instance supports multiple host interfaces. Typically, each host interface is mapped to a distinct cohort or domain.

As part of the system initialization process, these peripherals require a mandatory pre-configuration stage to enable controlled sharing. For example, NETC is initialized by the CRS subsystem. Then, NETC.PSI0 is assigned to the CRS cohort and fully configured before other interfaces, such as VSI, are allowed to access the resource.

- Shared-base peripherals

Shared-base peripherals are owned by the SBC and are centrally managed by the foundation core.

Cohorts other than the SBC must issue access requests to the SBC when interacting with these peripherals. The SBC is responsible for request arbitration and servicing, subject to hardware-defined constraints. Further details are provided in the following section.

3.1.3 GPIO isolation

GPIO access is isolated through vGPIO. The PM uses PaCo to assign GPIO resources to cohorts. The Cohort Manager is responsible for further configuring partitions and allocating pins within its cohort.

GPIO ports on S32N55 support up to 3 banks, with 32 pins on each. vGPIO enables the allocation of GPIO pins to cohorts and partitions (up to 16) within them.

A partition is defined by a set of domain IDs and vGPIO slot IDs that are permitted to control a given set of pins. Access to the register interface which controls the pins is virtualized and accessible via different addresses, called slots. In addition, each vGPIO slot shall be managed by XRDC and shall be treated as an independent peripheral slot from the XRDC perspective.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

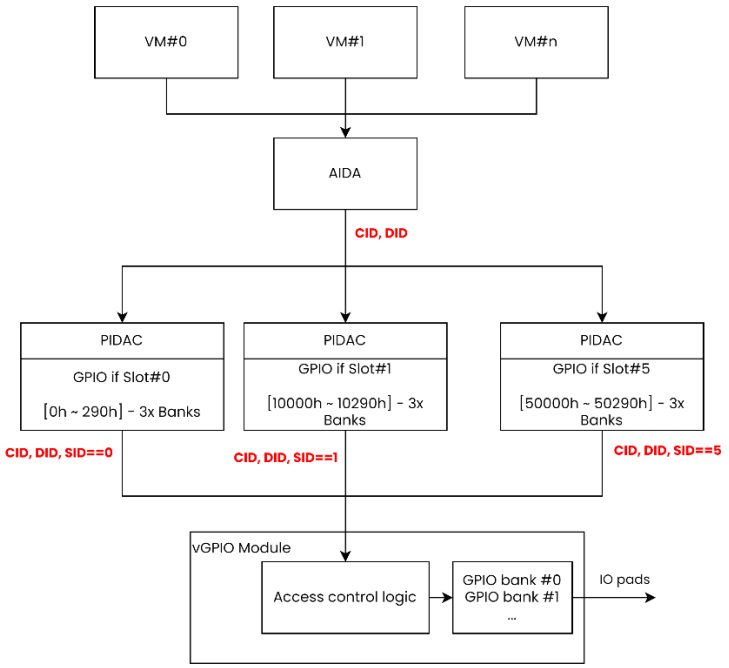


Figure 8. GPIO access

3.2 Software resource management

A software (SW) resource is a system element whose protection is not fully enforced by hardware mechanisms such as XRDC. These resources rely on software-based control, and any configuration or modification is performed by the FSS firmware.

3.2.1 Allocation policies

The allocation of software resources primarily follows the configuration defined in the “SW Resources Allocation” tab within the Fss_Rem_Pm module.

Currently, two policies are defined for the allocation of software resources and for pin muxing.

1. Automatic Allocation Enabled. When a software resource or pin is not explicitly assigned within a cohort, the tool performs an automatic allocation. Ownership is granted to the cohort specified by **“Sw Resources Full Allocation Owner/Mux Full Allocation Owner”**, while all other cohorts are assigned user access. This mechanism is controlled by enabling **“Enable Sw Resources Full Allocation”** and **“Enable Mux Full Allocation”**.
2. Automatic Allocation Disabled. When **“Enable Sw Resources Full Allocation”** and **“Enable Mux Full Allocation”** are disabled, any software resources or pins that are not explicitly assigned to a cohort will not be allocated. As a result, these resources are inaccessible to all cohorts.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

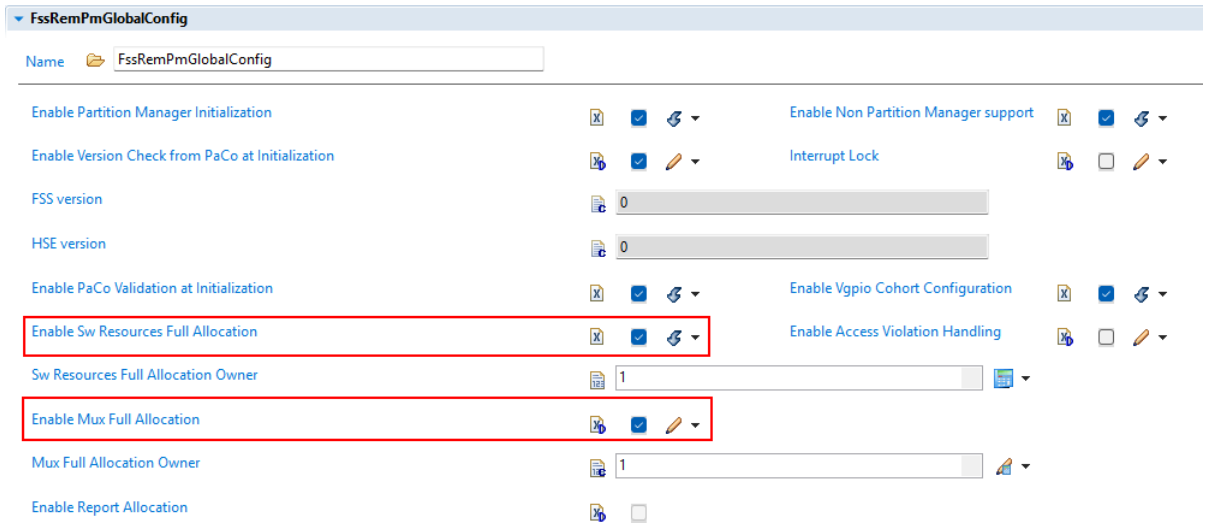


Figure 9. Automatic allocation of software resources

3.2.2 Clock, reset, and power domains

All clock, reset, and power controls exposed through each subsystem's DSC are managed by the FSS.

When any of these resources need to be changed, the software requests the FSS to perform the corresponding operation via a standardized API. The FSS then executes the required steps over the SCB bus to effect the requested change.

However, prior to applying any configuration, the FSS shall verify the requester's permissions and the corresponding software resource assignments.

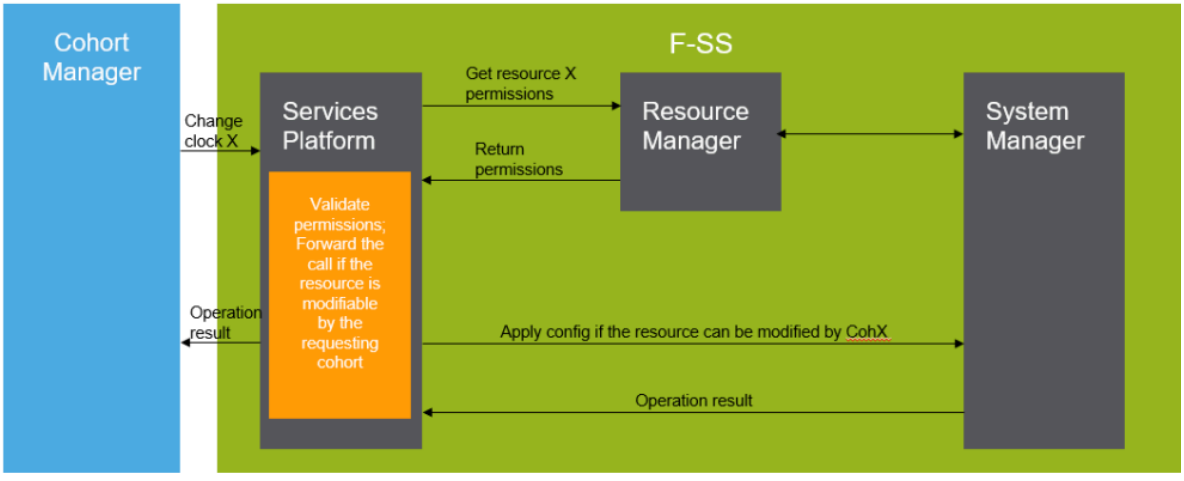


Figure 10. Example of a clock change

For peripherals that provide only a reset domain, the corresponding power domain shall be combined with the main power resource of the subsystem.

To simplify resource preconditions, it is recommended to explicitly assign each peripheral to a cohort. The associated resources then become available when the assigned cohort is in the ready state.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

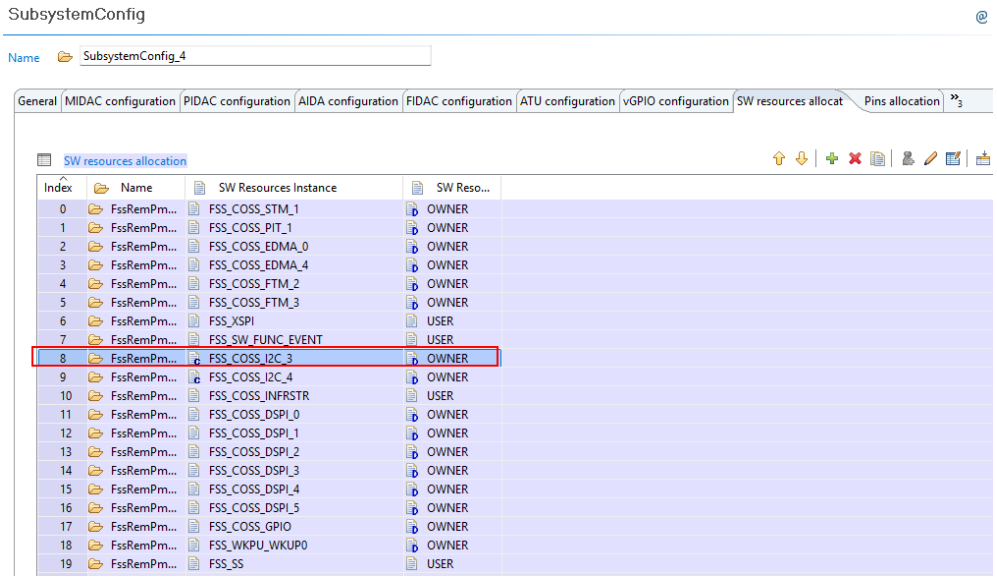


Figure 11. Explicit software resource assignment

For example, based on the default configuration, FSS_COSS_I2C_3 is assigned to cohort2 (RTU0_R52_0) with the **Owner** role. This implies that its reset domain is enabled or disabled in accordance with the start or stop state of cohort2.

In particular, certain resources (such as RAM and <Subsystem>_SS) are mandatory for the FSS to boot or to fulfill safety requirements. Even if these resources are assigned to a non-SBC cohort, they are still enabled by the functions *Fss_Sym_Startup_<Subsystem>_Infrastructure or Fss_Sym_Startup_Cohort0_<Subsystem>.*

3.2.3 Pins allocation

By default, the FSS configures most pins with their expected default functionalities during system initialization. Subsequently, additional pins may be assigned to different cohorts, where their configured functionality may differ from the default settings. For non-SBC cohorts, pin reconfiguration is restricted by the FSS through the "Pins Allocation" configuration.

If a pin is not assigned to a cohort through either explicit pin allocation or automatic allocation, the FSS reports an error when that cohort attempts to configure the pin.

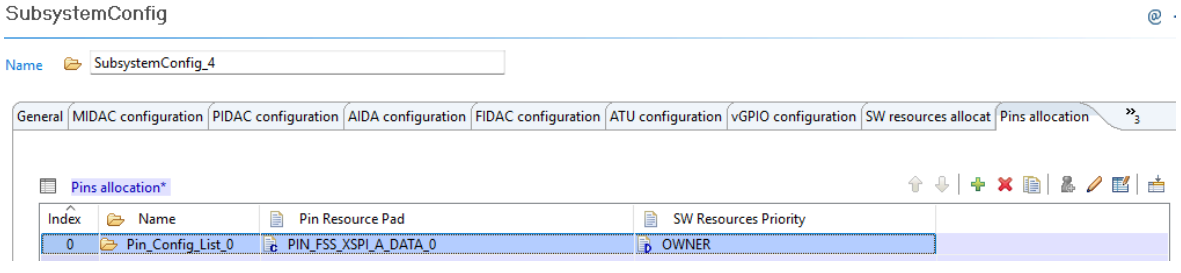


Figure 12. Pins allocation

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

4 Cohort states

4.1 Overview

The Boot Manager is the software entity responsible for tracking and managing cohort states. Cohort states can be initiated either by configuring cohorts for boot through the CoDef configuration or by issuing SCMI requests (via Fss_Services_Platform) to the Boot Manager.

The following diagram illustrates the supported cohort states and the valid transitions between them.

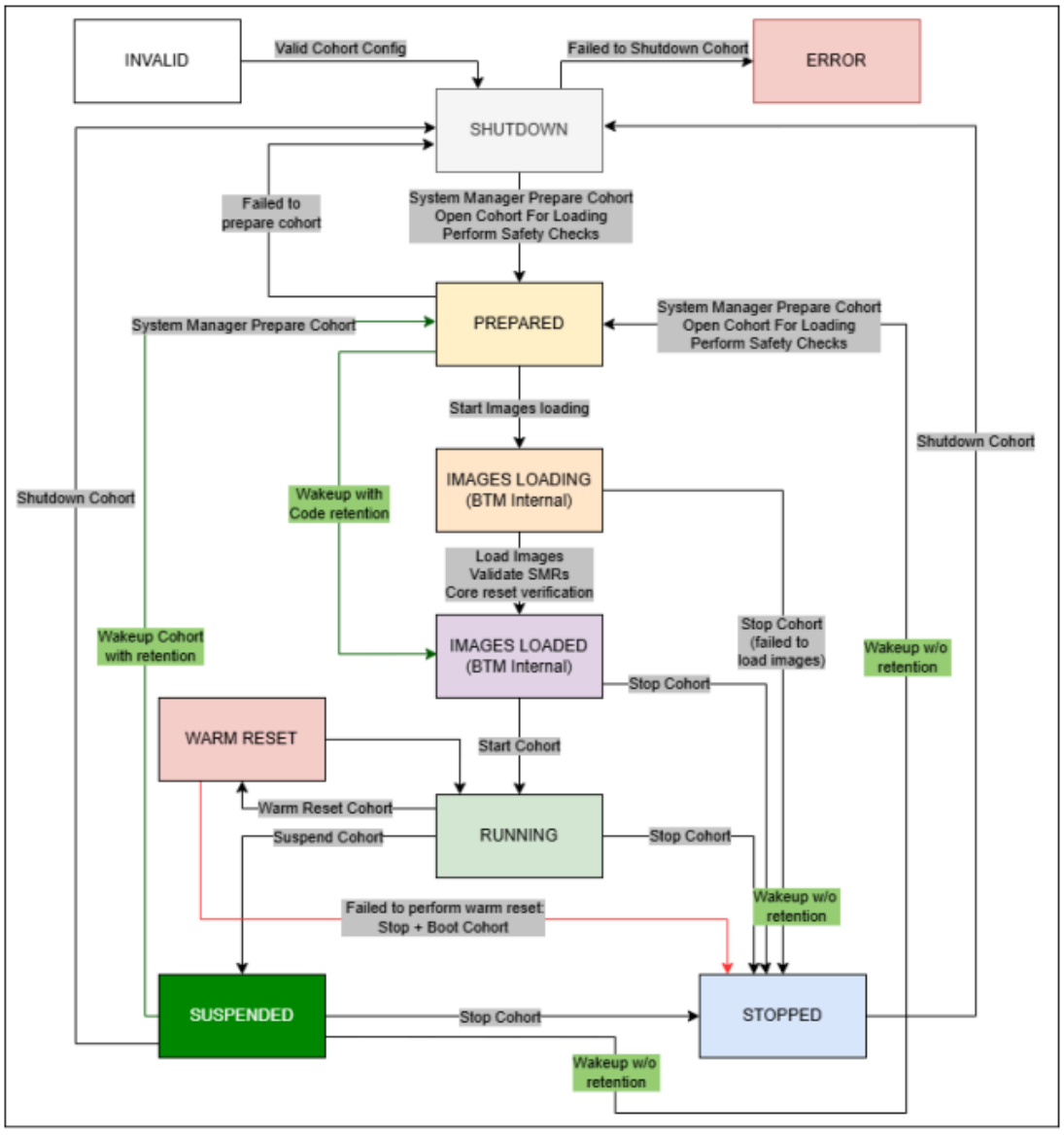


Figure 13. Cohort state transitions

Table 2. Cohort states

State	Description
Shutdown	In this state, all resources associated with the cohort are power-gated and clock-gated, and all cohort resources are held in reset.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

Table 2. Cohort states*...continued*

State	Description
	It is assumed that all cohorts are in this state immediately following a POR.
Prepared	In this state, all cohort resources are powered and clocked, and all resources are released from reset. The CM core remains in reset. If any errors are detected during this preparation state, the cohort will be shut down.
Images Loading	In this state, images are loaded by the Boot Manager. This state does not affect the operational state of the cohort and serves only as an intermediate state during the cohort lifecycle.
Images Loaded	In this state, all cohort images have been loaded, and the HSE has successfully verified the core start address. Only the reset for the CM core remains asserted. If any errors are detected during the Images Loaded state, the cohort will be shut down.
Running	In this state, at least the CM is executing code and is capable of sending requests to the FSS firmware. From the perspective of the FSS firmware, the cohort is considered to be ON, with active processing elements present. This includes cores that are in the WFI state.
Warm Reset	In this state, all individually owned cohort resources are reset, while memory contents are preserved. The CM core undergoes a warm reset. This state is intended to support a fast reset, as the Boot Manager does not reload the cohort images.
Stopped	In this state, the cohort remains powered, but all individually owned cohort resources are held in reset. All cohort cores are shut down.
Suspended	In this state, all cohort cores are shut down and power-gated. Only wake-up sources and memories configured for retention remain powered. Depending on the type of request issued for the cohort (with or without code retention), the cohort may resume to the Running state from this state without reloading the application images.
Error	A cohort enters this state if the transition to the Shutdown state fails. In this case, the cohort is considered to be in an unrecoverable state. Entry into this state is performed by the Boot Manager and cannot be requested via SCMI requests.
Invalid	A cohort is initialized in the Invalid state. During initialization of the Boot Manager module, if the cohort is found to have a valid and authenticated configuration, it is transitioned to the Shutdown state. Otherwise, the cohort remains in the Invalid state, and no requests targeting this cohort are processed.

4.2 Cohort boot

The FSS firmware provides the Fss_Btm module for configuring boot images and cohort states. After a Power-On Reset (POR), the initial cohort state can be configured as one of four values: Disabled, Running, Images Loaded, or Shutdown. This configuration can be found at:

Fss_Btm->Cohort Definition->CodefConfig->BootConfig->BootGeneralConfig->Cohort State after POR.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

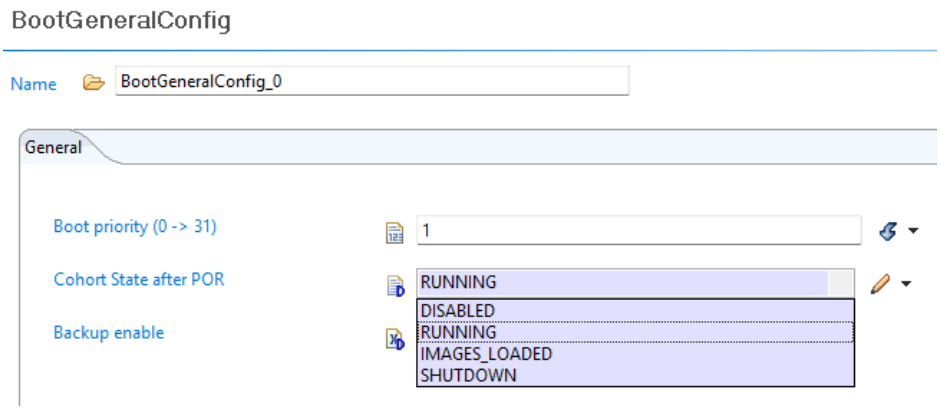


Figure 14. Cohort state after POR

The diagram below shows the different cohort boot settings.

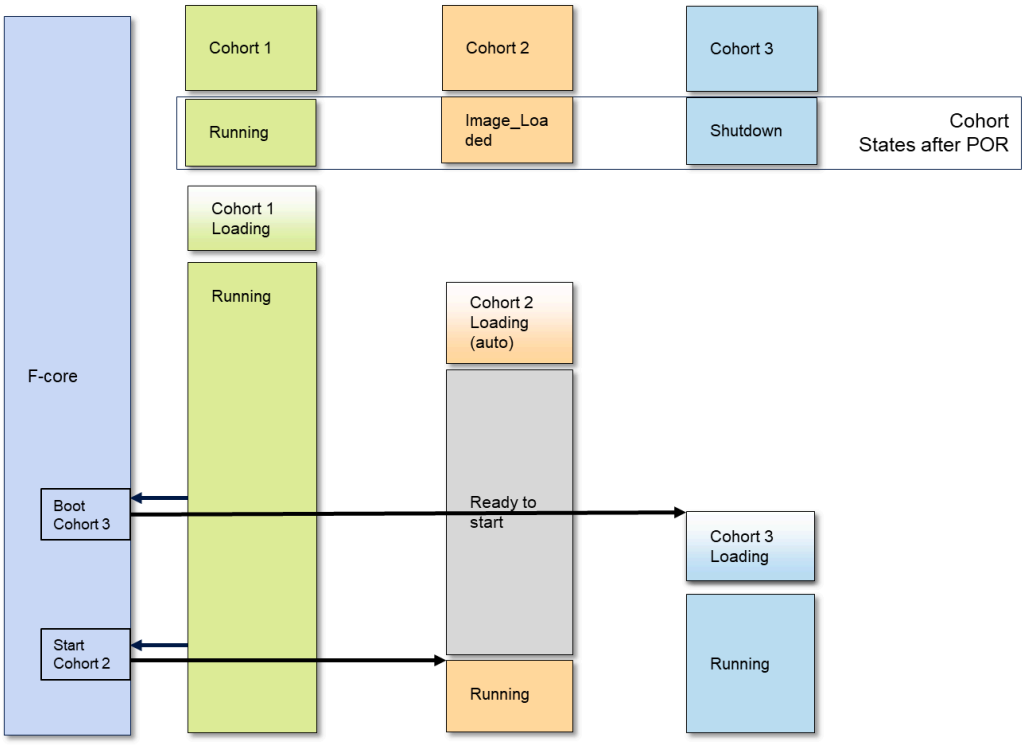


Figure 15. Cohort boot settings

5 CoDef and PaCo structures

The CoDef and PaCo images are generated as part of the Fss_Rem_Pm and Fss_Btm module configuration process.

These images are then applied by the FSS during the initialization phase of system boot.

In the General Configuration of the Fss_Firmware module, the parameter “Fss_Firmware PaCo and CoDef Source” defines the source location of the PaCo and CoDef images, which can be either **FLASH** or **RAM**. By default, this parameter is set to FLASH, indicating that the FSS copies the images from flash memory to RAM during initialization.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

5.1 CoDef

The CoDef contains the boot manager configuration for each cohort. The main components of each cohort definition are the CR table and the Boot Configuration. These define, respectively, the boot information of individual cores and the image-loading behavior of the cohort. Additionally, the CoDef binary contains multiple sub-CoDefs corresponding to each cohort, and each sub-CoDef includes a PaCo hash to ensure the integrity of its associated PaCo.

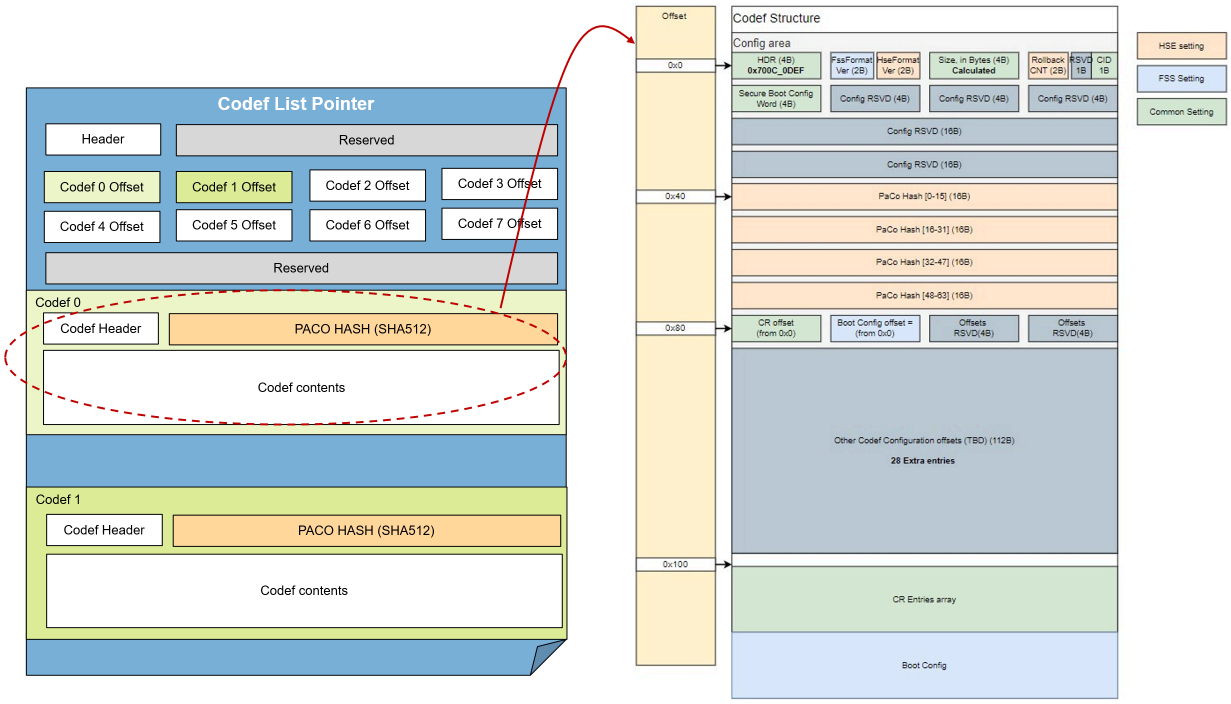


Figure 16. CoDef structure

5.2 PaCo

PaCo defines system resources and their allocation across cohorts.

Several software entities need to be aware of this resource allocation for different purposes:

1. System Control Software
   - Determines what needs to be configured and how during cohort establishment.
   - Enforces access restrictions and performs runtime checks (i.e., which entity is allowed to request which resources).
2. Safety Software
   - Performs safety checks on resource allocation and configuration.
   - Configures fault-to-cohort mapping.
3. Security Software
   - Ensures that the resource allocation policy is correctly implemented and enforced.
   - Configures XRDC and associated domain-aware peripherals.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

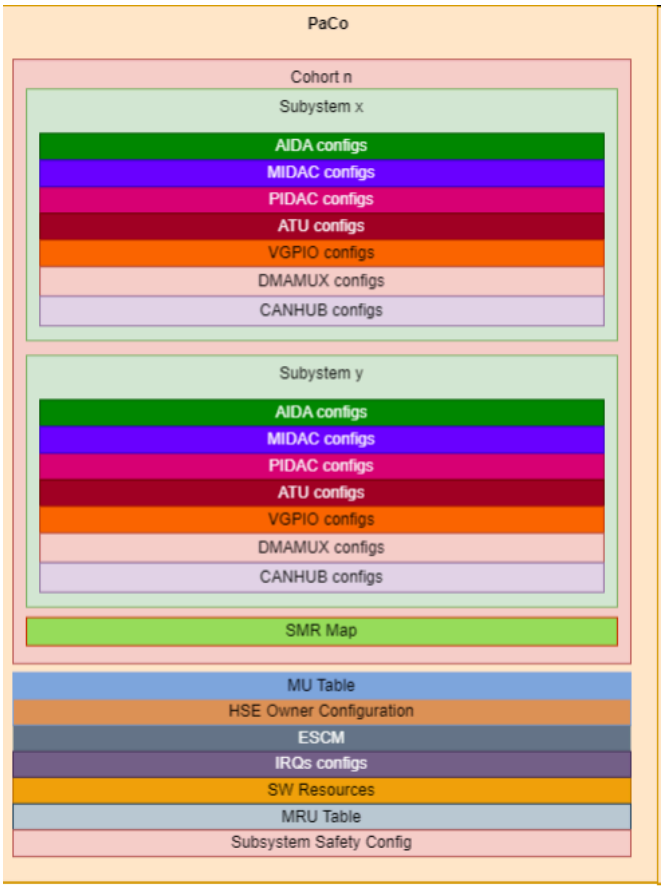


Figure 17. PaCo structure

6 Abbreviations

The following table summarizes the abbreviations and acronyms used throughout this document.

Table 3. Abbreviations and acronyms

Abbreviation	Explanation
AID	Access identifier
AIDA	Access ID assignment
API	Application programming interface
ASIL	Automotive safety integrity level
CM	Cohort manager
CoDef	Cohort definition
DMA	Direct memory access
DSC	Distributed subsystem controller
ECU	Electronic control unit
FSS	Foundation subsystem
GPIO	General-purpose input/output

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

Table 3. Abbreviations and acronyms...*continued*

Abbreviation	Explanation
HSE	Hardware security engine
MIDAC	Memory identifier access control
NETC	Ethernet switch and controller
PaCo	Partition contract
PIDAC	Peripheral identifier access control
PM	Partition manager
POR	Power-on reset
PSI	Physical station interface
RAM	Random access memory
SBC	System base cohort
SCB	System control bus
SCMI	System control and management interface
SoC	System on a chip
UENV	Uniform environment
vECU	Virtual electronic control unit
VSI	Virtual station interface
XRDC	Extended resource domain controllers

7 References

1. S32N55 Reference Manual.pdf
2. FSS_FW_UserManual.pdf
3. AN14892-S32N55: Managing Isolation and XRDC Configuration via Domain Manager

8 Revision history

Document ID	Release date	Description
AN15251 v.1.0	6 November 2026	Initial release

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

Tables

Tab. 1. Key concepts related to cohorts 3
Tab. 2. Cohort states 11
Tab. 3. Abbreviations and acronyms 15

Figures

Fig. 1. Example cohort configuration 2
Fig. 2. S32N55 subsystem diagram 4
Fig. 3. ECU consolidation 5
Fig. 4. Resource types 5
Fig. 5. Fss_Rem_Pm module diagram 6
Fig. 6. Diagram for XRDC 6
Fig. 7. Peripheral categories 7
Fig. 8. GPIO access 8
Fig. 9. Automatic allocation of software resources 9
Fig. 10. Example of a clock change 9
Fig. 11. Explicit software resource assignment 10
Fig. 12. Pins allocation 10
Fig. 13. Cohort state transitions 11
Fig. 14. Cohort state after POR 13
Fig. 15. Cohort boot settings 13
Fig. 16. CoDef structure 14
Fig. 17. PaCo structure 15

All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.

Contents

1 Overview 2
2 Concepts 2
2.1 Cohort 2
2.2 Subsystems 3
2.3 ECU consolidation 4
3 Resource management 5
3.1 Hardware-enforced resources 6
3.1.1 Cores and memory 6
3.1.2 Peripheral categories 6
3.1.3 GPIO isolation 7
3.2 Software resource management 8
3.2.1 Allocation policies 8
3.2.2 Clock, reset, and power domains 9
3.2.3 Pins allocation 10
4 Cohort states 11
4.1 Overview 11
4.2 Cohort boot 12
5 CoDef and PaCo structures 13
5.1 CoDef 14
5.2 PaCo 14
6 Abbreviations 15
7 References 16
8 Revision history 16
Legal information 17

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

© 2026 NXP B.V. **All rights reserved.**