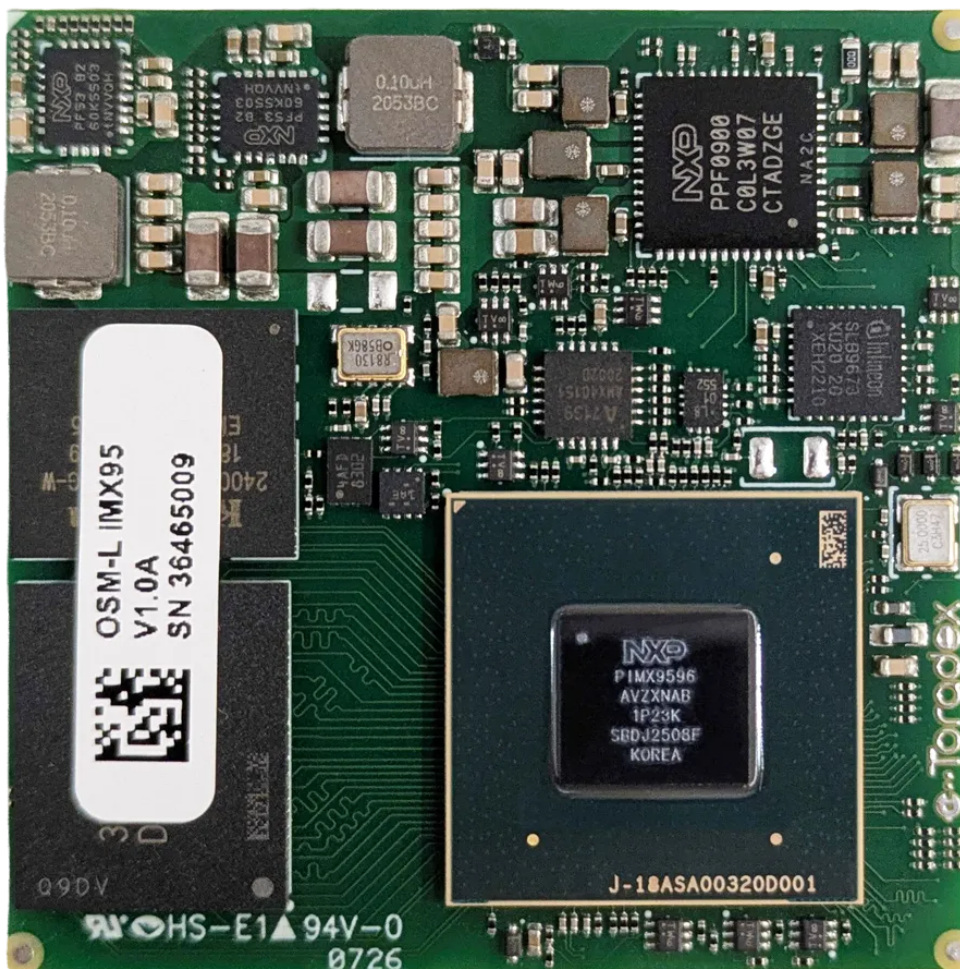


OSM iMX95

Datasheet

Preliminary – Subject to Change



Revision History

Document Revisions

Date	Doc. Revision	Product Version	Changes
27-Jul-2026	Rev. 0.1	V1.0	Initial documentation

Contents

1	Introduction	8
1.1	Purpose of the Datasheet	8
1.2	OSM iMX95 System on Module	8
1.3	NXP i.MX 95 System on Chip	8
1.4	Security	9
1.4.1	EdgeLock Secure Enclave (ELE)	9
1.4.2	Post-Quantum Cryptography (PQC)	9
1.4.3	Tamper Detection	10
1.4.4	Secure Boot	10
1.5	OSM Compliance and Design Guidelines	10
1.6	Module Interface Classification	11
1.7	Interfaces Overview	11
1.8	Reference Documents	12
1.8.1	Toradex Developer Center	12
1.8.2	OSM iMX95 Product Page	12
1.8.3	Carrier Board Design Guides	12
1.8.4	Toradex Pinout Designer	13
1.8.5	NXP i.MX 95 Documentation	13
1.9	Naming Convention	13
1.10	Part Number Nomenclature	13
1.11	Configuration Options	15
1.12	Configure-To-Order (CTO) Options	17
2	Architecture Overview	18
2.1	Block Diagram	19
3	OSM Interface	20
3.1	Carrier Board Design Considerations	20
3.2	Contact Assignment	20
4	I/O Pins	54
4.1	Function Multiplexing	54
4.2	Pin Multiplexing and Pad Control	61
4.3	Pin Reset Status	61
5	Interfaces Description	62
5.1	ADC – Analog-to-Digital Converter	62
5.2	Camera Interfaces	62
5.2.1	MIPI CSI – Camera Serial Interface	63
5.2.2	MIPI CSI/DSI – Multiplexed Camera Serial Interface	64
5.3	CAN – Controller Area Network	64
5.3.1	FlexCAN Features	65
5.3.2	Timing and Synchronization	65
5.3.3	Physical Layer Requirements	65
5.4	Digital Audio	67
5.4.1	SAI – Serial Audio Interface	67
5.4.2	I ² S Interfaces – Inter-IC Sound Interfaces	67
5.4.3	eARC – enhanced Audio Return Channel	70
5.5	Displays Interfaces	71
5.5.1	LVDS Interface – Low-Voltage Differential Signaling Interface	72
5.5.2	MIPI-DSI – Display Serial Interface	73
5.6	eMMC – embedded MultiMediaCard Storage Device	74
5.7	Ethernet Interfaces	75
5.7.1	ETH_A Interface	75
5.7.2	ETH_B Interface	76
5.7.3	Shared MDIO Management Interface	76

5.7.4	NETC Ethernet Features	76
5.7.5	ETH_C USXGMII Interface	76
5.7.6	Design Considerations	77
5.8	GPIO – General-Purpose Input/Output	79
5.8.1	GPIOs with Defined Functions	82
5.9	I ² C – Inter-Integrated Circuit - OSM iMX95	83
5.9.1	Exposed I ² C Interfaces by Default	83
5.9.2	Internal I ² C Interfaces	84
5.9.3	I ² C Addresses of Internal Devices	85
5.9.4	Additional I ² C Interfaces	86
5.10	JTAG – Joint Test Action Group	86
5.11	PWM – Pulse Width Modulation	87
5.11.1	PWM Signal Availability	88
5.11.2	Additional PWM Interfaces	89
5.12	SPI – Serial Peripheral Interface	90
5.13	Tamper Detection	93
5.14	UART – Universal Asynchronous Receiver/Transmitter	94
5.14.1	Key Features and Supported Capabilities	94
5.14.2	UART Recommended Usage	96
5.15	USB – Universal Serial Bus	98
5.15.1	Recovery Mode	99
6	Low Power Modes	101
7	Recovery Mode	102
7.1	Boot Mode Selection and Pin Mapping	102
7.1.1	Pin Mapping	102
7.1.2	Recovery Mode Control	103
7.2	Recovery Procedure	103
8	Known Issues	104
9	Technical Specification	105
9.1	Electrical Characteristics	105
9.1.1	Absolute Maximum Ratings	105
9.1.2	Recommended Operating Conditions	105
9.1.3	PMIC – Power Management IC	105
9.1.4	Power Sequence	106
9.1.5	Watchdog Signal	106
9.2	Mechanical Characteristics	106
9.2.1	SoM Outline Dimensions	106
9.2.2	Thermal Specifications	107
10	Soldering and Assembly	108
11	OSM Size-L Adapter	109
11.1	Ground and Power Net Mapping Considerations	110
11.1.1	Ground (GND)	110
11.1.2	Power Rails	110
11.2	OSM to Development Board Adapter Pin Assignment	110
12	Product Compliance	135
13	Device and Documentation Support	136
13.1	Trademarks	136

List of Tables

1	Interface overview	11
2	Toradex naming conventions	13
3	Part number nomenclature options	14
4	Part number nomenclature example	15
5	Configuration options	15
6	OSM Size-0 contact assignment	22
7	OSM Size-S additional contact assignment	31
8	OSM Size-M additional contact assignment	38
9	OSM Size-L additional contact assignment	45
10	Alternate functions	55
11	ADC interfaces	62
12	MIPI CSI interface 1	63
13	MIPI CSI DSI interface	64
14	CAN interfaces available as default	66
15	CAN interfaces available as alternate functions	66
16	I ² S interfaces	68
17	eARC interface	71
18	LVDS interface signals	72
19	MIPI-DSI interface	74
20	Ethernet A interface	77
21	Ethernet B interface	78
22	Ethernet C interface	79
23	GPIO A interface	81
24	GPIO B interface	81
25	GPIO C interface	82
26	GPIO D interface	82
27	Defined functions from mandatory GPIO A (IC16)	82
28	I ² C interfaces	84
29	PMIC I ² C interface	85
30	Internal I ² C interface	85
31	I ² C addresses	85
32	JTAG interface	87
33	PWM interfaces	88
34	PWM signals available as function multiplexing	89
35	SPI interfaces	91
36	Tamper detection	94
37	UART interfaces	95
38	USB overview	98
39	USB interfaces	99
40	Boot modes	102
41	Absolute maximum ratings	105
42	Recommended operating conditions	105
43	OSM adapter connector M1 row A	110
44	OSM adapter connector M1 row B	114
45	OSM adapter connector M1 row C	117
46	OSM adapter connector M1 row D	120
47	OSM adapter connector M2 row A	123
48	OSM adapter connector M2 row B	126
49	OSM adapter connector M2 row C	129
50	OSM adapter connector M2 row D	132

List of Figures

1	Block diagram	19
2	OSM iMX95 pad layout top view	21
3	Displays block diagram	71
4	GPIO block diagram	80
5	USB block diagram	100
6	Mechanical reference view (left: top view, right: bottom view)	109

Abbreviations

Abbreviations

Abbreviation	Explanation
ADC	Analog to Digital Converter
BTB	Board To Board
CAN	Controller Area Network, a bus that is mainly used in the automotive and industrial environment
CAN FD	Controller Area Network Flexible Data-Rate, an extension to the original CAN bus protocol which allows higher data rates and larger message sizes.
CEC	Consumer Electronic Control, HDMI feature that allows controlling CEC compatible devices
CPU	Central Processor Unit
CSI	Camera Serial Interface
DAC	Digital to Analog Converter
DDC	Display Data Channel, interface for reading out the capability of a monitor. In this document DDC2B (based on I2C) is always meant.
DFP	Downstream Facing Port, USB Type-C port that acts as a host
DRP	Dual-Role Port, USB Type-C port that can operate as power sink and source
DSI	Display Serial Interface
DVI	Digital Visual Interface, digital signals are electrically compatible with HDMI
EDID	Extended Display Identification Data, timing setting information provided by the display in a PROM
EMI	Electromagnetic Interference, high-frequency disturbances
ESD	Electrostatic Discharge, high voltage spike or spark that can damage electrostatic-sensitive devices
FPD-Link	Flat Panel Display Link, high-speed serial interface for liquid crystal displays. In this document is also called the LVDS interface.
GBE	Gigabit Ethernet, Ethernet interface with a maximum data rate of 1000Mbit/s
GND	Ground
GND_CHASSIS	Chassis Ground
GPIO	General Purpose Input/Output, pin that can be configured as an input or output
GSM	Global System for Mobile Communications
HDA	High-Definition Audio (HD Audio), the digital audio interface between CPU and audio codec
I2C	Inter-Integrated Circuit, the two-wire interface for connecting low-speed peripherals
I2S	Integrated Interchip Sound, serial bus for connecting PCM audio data between two devices
I/O	Input-Output
JTAG	Joint Test Action Group, widely used debug interface
LCD	Liquid Crystal Display
LSB	Least Significant Bit
LVDS	Low-Voltage Differential Signaling, electrical interface standard that can transport high-speed signals over twisted-pair cables. Many interfaces like PCIe or SATA use this interface. Since the first successful application was the Flat Panel Display Link, LVDS became a synonymous for this interface. In this document, the term LVDS is used for the FPD-Link interface.
MAC	Medium Access Control is part of the second layer (data link layer) in the Ethernet stack
MIPI	Mobile Industry Processor Interface Alliance
MDI	Medium Dependent Interface, the physical interface between Ethernet PHY and cable connector
MDIO	Management Data Input/Output, an interface that is used for controlling the Ethernet PHY. The bus consists of the MDC clock and the MDIO bidirectional data signal.

Continued on next page

Abbreviations (Continued)

Abbreviation	Explanation
mini PCIe	PCI Express Mini Card, the card form factor for internal peripherals. The interface features PCIe and USB 2.0 connectivity
MMC	MultiMediaCard, flash memory card
MSB	Most Significant Bit
NC	Not Connected
OD	Open-Drain
OTG	USB On-The-Go, a USB host interface that can also act as USB client when connected to another host interface
PCB	Printed Circuit Board
PCI	Peripheral Component Interconnect, parallel computer expansion bus for connecting peripherals
PCIe	PCI Express, a high-speed serial computer expansion bus, replaces the PCI bus
PCM	Pulse-Code Modulation, digitally representation of analog signals, standard interface for digital audio
PD	Pull-Down Resistor
PHY	The physical layer of the OSI model
PU	Pull-up Resistor
PWM	Pulse-Width Modulation
PWR	Power
QSPI	Quad SPI, SPI interface with four bidirectional data signals
RGMI	Reduced Gigabit Media-Independent Interface, the interface between Ethernet MAC and PHY for up to 1Gb/s
RJ45	Registered Jack, common name for the 8P8C modular connector that is used for Ethernet wiring
RS232	The single-ended serial port interface
RS485	Differential signaling serial port interface, half-duplex, multi-drop configuration possible
R-UID	Removable User Identity Module, identifications card for CDMA phones and networks, an extension of the GSM SIM card
SD	Secure Digital, flash memory card
SDIO	Secure Digital Input Output, an external bus for peripherals that uses the SD interface
SIM	Subscriber Identification Module, an identification card for GSM phones
SMBus	System Management Bus (SMB), a two-wire bus based on the I ² C specifications, is used in x86 designs for system management.
SoC	System on a Chip, IC which integrates the main component of a computer on a single chip
SoM	System on a Module, PCB which integrates the main component of a computer on a single board
SPI	Serial Peripheral Interface Bus, synchronous four-wire full-duplex bus for peripherals
TIM	Thermal Interface Material, thermally conductive material between CPU and heat spreader or heat sink
TMDS	Transition-Minimized Differential Signaling, serial high-speed transmitting technology that is used by DVI and HDMI
TVS Diode	Transient-Voltage-Suppression Diode, a diode that is used to protect interfaces against voltage spikes
UFP	Upstream Facing Port, USB Type-C port that acts as a client
UART	Universal Asynchronous Receiver/Transmitter, serial interface, in combination with a transceiver an RS232, RS422, RS485, IrDA or similar interface can be achieved
USB	Universal Serial Bus, serial interface for internal and external peripherals

1 Introduction

1.1 Purpose of the Datasheet

This document describes the hardware characteristics and capabilities of the OSM iMX95 System-on-Module (SoM). It is intended as a technical reference for hardware designers, system integrators, and developers working with this module.

For information regarding currently supported software features and integration details, please refer to the OSM iMX95 product page on the Toradex Developer Center:

<https://developer.toradex.com/hardware/osm-som-family/modules/osm-imx95>

1.2 OSM iMX95 System on Module

The **OSM iMX95 SoM** simplifies the design of embedded systems by integrating high-density, high-speed circuitry within a compact, production-ready module. By encapsulating complex layout requirements, such as high-speed interfaces, memory routing, and power management. The module enables carrier board development to focus on application-specific functionality, reducing overall design complexity and accelerating time to market.

The module is based on the **Open Standard Module (OSM) Size-L (45 × 45 mm)** form factor and uses an **LGA (Land Grid Array)** interface for direct soldering onto the carrier board. This approach enables a compact system design without the need for board-to-board connectors.

The OSM iMX95 provides a comprehensive set of interfaces, ranging from low-speed signals such as CAN FD, GPIO, I²C, SPI, and UART, to high-speed interfaces including multi-gigabit Ethernet, PCIe Gen3, and USB 3.2.

In addition, the module supports multimedia capabilities such as MIPI CSI-2 camera interfaces inputs, and MIPI DSI and LVDS display outputs.

Designed for industrial and embedded applications, the module is suitable for use cases including industrial automation, medical devices, transportation systems, smart infrastructure, and human-machine interfaces. It provides a balanced combination of performance, power efficiency, and integration, enabling scalable and cost-optimized system designs.

1.3 NXP i.MX 95 System on Chip

The OSM iMX95 SoM is based on the **NXP i.MX 95 applications processor family**, featuring a heterogeneous multicore architecture. The main processing cluster includes up to six Arm® Cortex®-A55 cores operating at up to 1.8 GHz, supporting the Arm®v8.2-A 64-bit architecture with backward compatibility for 32-bit applications. Additional real-time processing capability is provided by integrated Cortex®-M7 and Cortex®-M33 microcontrollers, enabling deterministic execution for time-critical and safety-related tasks.



Security

A dedicated security subsystem (EdgeLock® Secure Enclave) provides hardware-based security features such as secure boot, key management, and runtime protection.

The SoC integrates dedicated hardware accelerators, including:

- A Graphics Processing Unit (GPU) for advanced graphics rendering,
- A Neural Processing Unit (NPU) for machine learning workloads,
- A Video Processing Unit (VPU) for video encode and decode, and
- An Image Signal Processor (ISP) for camera processing pipelines.

1.4 Security

The OSM iMX95 integrates a hardware security architecture based on the NXP i.MX 95 System-on-Chip. The security subsystem combines secure boot capabilities, hardware cryptographic accelerators, secure key management, tamper-detection mechanisms, and the EdgeLock Secure Enclave (ELE) to provide a trusted foundation for embedded applications requiring strong protection against unauthorized access, software manipulation, and data compromise.

The security features of the i.MX 95 support the implementation of secure device provisioning, authenticated firmware updates, trusted execution environments, secure communications, and long-term cryptographic protection suitable for applications with extended deployment lifetimes.

1.4.1 EdgeLock Secure Enclave (ELE)

The i.MX 95 integrates the NXP EdgeLock Secure Enclave (ELE), an isolated security subsystem responsible for managing the device root of trust and executing security-sensitive operations independently from the application processors.

The EdgeLock Secure Enclave provides:

- Secure boot and image authentication
- Secure key storage and lifecycle management
- Cryptographic services and hardware acceleration
- Device identity management
- Secure provisioning and manufacturing support
- Tamper monitoring and security event handling

The ELE operates as the trusted security anchor of the platform and provides hardware-isolated execution of security-critical functions.

1.4.2 Post-Quantum Cryptography (PQC)

The OSM iMX95 supports the transition towards **Post-Quantum Cryptography (PQC)** through the security capabilities integrated into the NXP i.MX 95. These capabilities are designed to help protect systems against attacks enabled by large-scale quantum computers while maintaining compatibility with current cryptographic infrastructures.

Conventional public-key cryptography algorithms such as RSA and Elliptic Curve Cryptography (ECC) rely on mathematical problems that are considered computationally infeasible for classical computers. Future quantum computers may be capable of solving these problems efficiently, potentially compromising long-term confidentiality and authenticity guarantees.

To address this challenge, the i.MX 95 security subsystem supports the NIST-standardized post-quantum cryptographic algorithms:

- **ML-KEM (Module-Lattice Key Encapsulation Mechanism)** for secure key establishment and key exchange
- **ML-DSA (Module-Lattice Digital Signature Algorithm)** for authentication, firmware signing, and secure boot

These algorithms provide quantum-resistant alternatives to traditional public-key cryptography and can be deployed in combination with classical cryptographic algorithms as part of hybrid security architectures.

Potential applications include:

- Quantum-resistant secure boot chains
- Authenticated firmware and software updates
- Secure device provisioning
- Secure cloud and edge connectivity
- Long-term protection of sensitive information
- Critical infrastructure and industrial control systems

By integrating support for post-quantum cryptographic algorithms within the EdgeLock Secure Enclave, the i.MX 95 enables the implementation of future-proof security architectures while minimizing the performance impact on the application processors.

The Post-Quantum Cryptography capabilities of the platform also help mitigate “harvest-now, decrypt-later” attack scenarios, where encrypted data captured today may be stored and decrypted in the future using quantum computing technologies.

1.4.3 Tamper Detection

The i.MX 95 security subsystem includes tamper-detection capabilities that allow the system to monitor external signals and detect physical intrusion attempts or unauthorized access events. Tamper events can be configured to generate interrupts, status indications, secure key invalidation, or other application-specific security responses.

Refer to the [Tamper Detection](#) section for details on the tamper signals exposed by the OSM iMX95 module.

1.4.4 Secure Boot

The i.MX 95 implements a hardware Root of Trust and secure boot architecture that verifies the authenticity and integrity of software before execution. During the boot process, cryptographic signatures are validated by the security subsystem, preventing the execution of unauthorized or modified software images.

Secure boot support enables the creation of trusted software chains extending from the boot ROM through the bootloader, operating system, and application software.

1.5 OSM Compliance and Design Guidelines

The OSM iMX95 SoM is designed in accordance with the **OSM Specification V1.2**, ensuring compatibility with standard OSM carrier board designs. The module uses an LGA (Land Grid Array) interface, allowing direct soldering onto the carrier PCB without the need for board-to-board connectors.

Key design considerations include:

- **Flexible MIPI interface configuration**
The i.MX 95 provides one dedicated MIPI CSI-2 interface and one shared interface that can be configured as either CSI-2 or DSI. An on-module analog switch enables assignment of these signals to the corresponding OSM pins, ensuring compatibility with different module configurations.
- **Bottom-side component placement**
Components are located on the underside of the module, requiring carrier board cut-outs defined by the OSM specification. This placement helps improve power delivery integrity and contributes to a more stable system with less noise and better EMC performance.
- **Thermal management**
The module requires an external cooling solution. As the OSM specification does not define a standardized heatsink interface, a reference mounting concept is provided via the OSM SoM Carrier

Board Adapter. Carrier board designs should consider compatible mounting features for thermal solutions.

1.6 Module Interface Classification

Interfaces provided by the OSM iMX95 are categorized according to their compliance level with the OSM specification, using the terms **SHALL**, **SHOULD**, and **MAY**.

- **SHALL**

Interfaces defined as SHALL are mandatory according to the OSM specification. These signals are required to be implemented and electrically compatible across all compliant OSM modules. They ensure a consistent baseline of functionality and enable full interchangeability between modules without modification to the carrier board design.

- **SHOULD**

Interfaces defined as SHOULD are recommended by the OSM specification. While not mandatory, they are typically implemented on most modules to enhance functionality and compatibility. Absence of a SHOULD interface does not violate compliance, but its presence improves portability and reduces the need for carrier board redesign when migrating between modules.

- **MAY**

Interfaces defined as MAY are optional and module-specific. These signals are not guaranteed to be present or electrically compatible across different modules. Their implementation depends on the capabilities of the underlying SoC and design choices. Use of MAY interfaces can limit cross-module compatibility and should be carefully evaluated in system design.

In addition to these categories, many pins support **alternate functions**, allowing multiplexing of multiple signal types depending on the selected pin configuration.

For maximum portability and long-term scalability across different OSM modules, designs should prioritize interfaces classified as **SHALL**, and where possible, also leverage **SHOULD** interfaces. Use of **MAY** interfaces should be limited to application-specific requirements.

1.7 Interfaces Overview

The table below shows the interfaces that are supported on the OSM iMX95 SoM:

Table 1: Interface overview

Feature	SHALL	SHOULD	MAY	Vendor Specific	Total
Analog					
Analog Input (ADC)	0	0	2	6	8
Audio					
I ² S	0	0	2	0	2
Camera					
MIPI CSI-2	0	0	2	0	2
Display					
MIPI DSI ¹	0	0	1	0	1
LVDS (dual channel)	0	0	2	0	2
Low-Speed					
I ² C	1	1	0	3	5
I ³ C	0	0	0	1	1 ²

Continued on next page

Table 1: Interface overview (Continued)

Feature	SHALL	SHOULD	MAY	Vendor Specific	Total
SPI	0	1	1	0	2
UART	0	1	3	0	4
UART console	1	0	0	0	1
PWM	0	0	5	0	5
CAN FD	0	0	2	0	2
JTAG	0	1	0	0	1
OSPI	0	0	0	1	1 ²
GPIO	0	16	14	0	30 ³
Network					
RGMII	1	0	1	0	2
USXGMII (for 2.5 and 10 Gigabit Ethernet)	0	0	0	1	1
High-Speed					
PCIe Gen3	0	0	2	0	2
Storage					
SD/SDIO/MMC	0	2	0	0	2
USB					
USB 2.0 OTG	0	1	0	0	1
USB 3.2 OTG (SS Signals for USB 2.0 OTG)	0	0	1	0	1
USB 2.0 Host	0	1	0	0	1

¹ Multiplexed with a second MIPI CSI interface.

² Available as an alternate function.

³ Additional GPIOs are available through pin multiplexing.

1.8 Reference Documents

The following documents provide additional technical information required to design, validate, and maintain carrier boards and end products based on the OSM iMX95.

1.8.1 Toradex Developer Center

The Toradex Developer Center is the primary source for module documentation, software enablement, and integration guidelines. Verify that the selected content applies specifically to the OSM family and the OSM iMX95 module variant.

<https://developer.toradex.com/>

1.8.2 OSM iMX95 Product Page

The product page provides a consolidated entry point for hardware and software collateral, ordering information, and ecosystem resources for the OSM iMX95.

<https://www.toradex.com/computer-on-modules/osm-arm-family/nxp-imx95>

1.8.3 Carrier Board Design Guides

Toradex provides carrier board design guides and supporting references (including layout guidance, checklists, and best practices) intended to reduce risk during custom carrier board development.

<https://developer.toradex.com/carrier-board-design/carrier-board-design-guides/>

<https://developer.toradex.com/carrier-board-design/>

1.8.4 Toradex Pinout Designer

The Toradex Pinout Designer is used to configure and review pin multiplexing and to compare interface availability across Toradex modules. It is recommended during schematic capture to confirm pin function selections and compatibility.

<https://developer.toradex.com/carrier-board-design/pinout-designer/>

1.8.5 NXP i.MX 95 Documentation

For detailed SoC-level electrical characteristics, programming model, and peripheral descriptions, refer to the NXP i.MX 95 documentation.

<https://www.nxp.com/products/i.MX95>

1.9 Naming Convention

This document follows a consistent naming convention to avoid ambiguity between **SoM-level** and **SoC-level** signals and features.

- **OSM signal names** refer to the module-level naming used in pin tables and interface descriptions
- **SoC ball names** refer to NXP i.MX 95 package ball identifiers
- **Alternate functions** refer to i.MX 95 `IOMUX` selections (`ALT` modes) associated with each SoC pad

Pay close attention to punctuation and spacing in names. Do not confuse the NXP i.MX 95 SoC with the Toradex OSM iMX95 SoM. [Table 2](#) shows the differences in naming convention between the NXP and Toradex products.

Table 2: Toradex naming conventions

Name	Description
i.MX 95	NXP i.MX 95 System-on-Chip family
i.MX 9596	Specific variant of the NXP i.MX 95 family featuring six Cortex®-A55 cores and a Cortex®-M33 MCU
OSM iMX95	OSM module based on the i.MX 95 SoC the term OSM iMX95 refers to all versions of the module
OSM-IMX9596-2G-32G-T-IT	OSM module based on the full featured i.MX 95 SoC with 6 Cortex®-A cores, 2 GB of RAM, 32 GB of eMMC, with TPM assembled, and support for the industrial temperature range

1.10 Part Number Nomenclature

The part number nomenclature consists of a structured sequence of seven fields, each representing a specific configurable attribute of the device within the family. These fields encode key parameters such as:

- Family name
- i.MX 95 segment
- A-Core quantity
- RAM density
- eMMC storage capacity

- TPM security module inclusion
- Operating temperature range

This arrangement enables precise identification and differentiation of variants based on their feature sets and functional capabilities. For clarification and detailed visualization, refer to Tables 3 and 4, which provides the significance of each individual field and an illustrative example of the part number formation.

Table 3: Part number nomenclature options

Field	Options
Family name	
OSM-IMX95	OSM iMX95 system-on-module
i.MX 95 Segment	
9	Full-featured segment with ISP, GPU, NPU, VPU and Displays ¹
5	HMI with GPU, NPU, VPU and Displays ¹
3	Compute with NPU
Cortex®-A core Quantity	
6	Six Arm Cortex®-A55 core
4	Four Arm Cortex®-A55 cores
RAM Density	
-2G	2 GB LPDDR5
-4G	4 GB LPDDR5
-8G	8 GB LPDDR5
-16G	16 GB LPDDR5
eMMC Capacity	
-4G	4 GB eMMC
-8G	8 GB eMMC
-16G	16 GB eMMC
-32G	32 GB eMMC
-64G	64 GB eMMC
-128G	128 GB eMMC
-256G	256 GB eMMC
TPM	
-T	TPM 2.0 assembled
-N	TPM 2.0 not assembled
Temperature	
-IT	Industrial temperature range (-40 °C to +85 °C)
-ET	Extended temperature range (-25 °C to +85 °C)
-CT	Commercial temperature range (0 °C to +70 °C)

¹ The MIPI-DSI and LVDS interfaces are disabled. The display controller is enabled.

Table 4: Part number nomenclature example

Family name	i.MX 95 segment	A-Core qty	RAM	eMMC	TPM	Temperature
OSMIMX95	9	6	-8G	-32G	-T	-IT

For the configuration of [Table 4](#), the NPI part number is `OSMIMX9596-8G-32G-T-IT`, and represents

- A fully featured CPU with 6 Cortex®-A55 cores, with
- 8 GB of RAM,
- 32 GB of eMMC storage,
- A Trusted Module Platform, and
- Suitable for industrial temperature range.

If your desired configuration is not available please get in touch with your Toradex sales representative mentioning the NPI part number desired.



Customization Availability

Please note, the NPI part number nomenclature shown below does not reflect volume or sellable part numbers from Toradex. They should only be used to communicate your desired configuration.

1.11 Configuration Options

[Table 5](#) shows off-the-shelf configurations for the OSM iMX95.

Table 5: Configuration options

Feature	OSM iMX95 Hexa 8 GB 32 GB T IT PN: OSM-IMX9596-8G-32G-T-IT
CPU Details	
CPU Name	NXP MIMX9596
CPU Type	6x Arm® Cortex®-A55
Microcontroller	1x Arm® Cortex®-M7 1x Arm® Cortex®-M33
CPU Clock	1800 MHz (A55) 800 MHz (M7) 333 MHz (M33)
Memory	
RAM	8 LPDDR5 (32 Bit)
Flash	32 GB eMMC

Continued on next page

Table 5: Configuration options (Continued)

Feature	OSM iMX95 Hexa 8 GB 32 GB T IT PN: OSM-IMX9596-8G-32G-T-IT
Connectivity	
Analog Input	8x
CAN FD	2x
10-Gigabit Ethernet	1x
GPIO	30x
I ² C	5x
I ³ C	1x
JTAG	1x
OSPI	1x
PCIe	2x (x1 Gen3)
PWM	5x
RGMII	2x
SDIO/SD/MMC	2x
SPI	2x
UART	5x
USB 2.0	1x Host
USB OTG (USB 2.0 + USB 3.2 Gen1)	1x OTG (Gen1)
Multimedia	
Display Controllers	Single (Dual Engine)
Video Decoder	Yes (H.264/AVC and H.265/HEVC)
Video Encoder	Yes (H.264/AVC and H.265/HEVC)
Display Serial Interface	1x Quad Lane MIPI DSI (up to 4k)
LVDS	1x 8-lane or 2x 4-lane (up to 1920x1080)
Neural Processing Unit (NPU)	Yes (2 TOPS)
Digital Audio	2x I ² S
2D Acceleration	Yes
3D Acceleration	Yes
Camera Serial Interface	2x Quad Lane MIPI CSI-2
Integrated ISP	Yes
Operating System	
Torizon	Coming Soon
Embedded Linux	Coming Soon
Preinstalled OS	Toradex Easy Installer (Coming Soon)
Android	Contact Us
QNX	Coming Soon
FreeRTOS	Contact Us

Continued on next page

Table 5: Configuration options (Continued)

Feature	OSM iMX95 Hexa 8 GB 32 GB T IT PN: OSM-IMX9596-8G-32G-T-IT
Physical	
Size	45 x 45 mm
Temperature	-40°C to 85°C
Shock / Vibration	EN 60068-2-6/50g 20 ms
Power Dissipation	TBD
Intended Use, Pricing, Availability	
Intended Use	For End Products
Pricing	Contact Us
Minimum Product Commitment	2039



Interfaces Availability

- Interface availability on the module depends on **pin multiplexing and configuration choices**.
- Use the [Toradex Pinout Designer](#) to validate interface usage for your design.
- Other concept configurations are intended for **guidance only** and may not become off-the-shelf products.

1.12 Configure-To-Order (CTO) Options

In addition to standard products, configuration variants may be available through Toradex Configure-To-Order (CTO) programs depending on platform maturity and productization status. Refer to the Toradex sales channel for feasibility and ordering details.

2 Architecture Overview

The **OSM iMX95 System-on-Module** integrates the NXP i.MX 95 SoC memory, storage, connectivity subsystems, and supporting components on a compact PCB. This high level of integration reduces carrier board design complexity while ensuring robust signal integrity, thermal stability, and compliance with relevant industry standards.

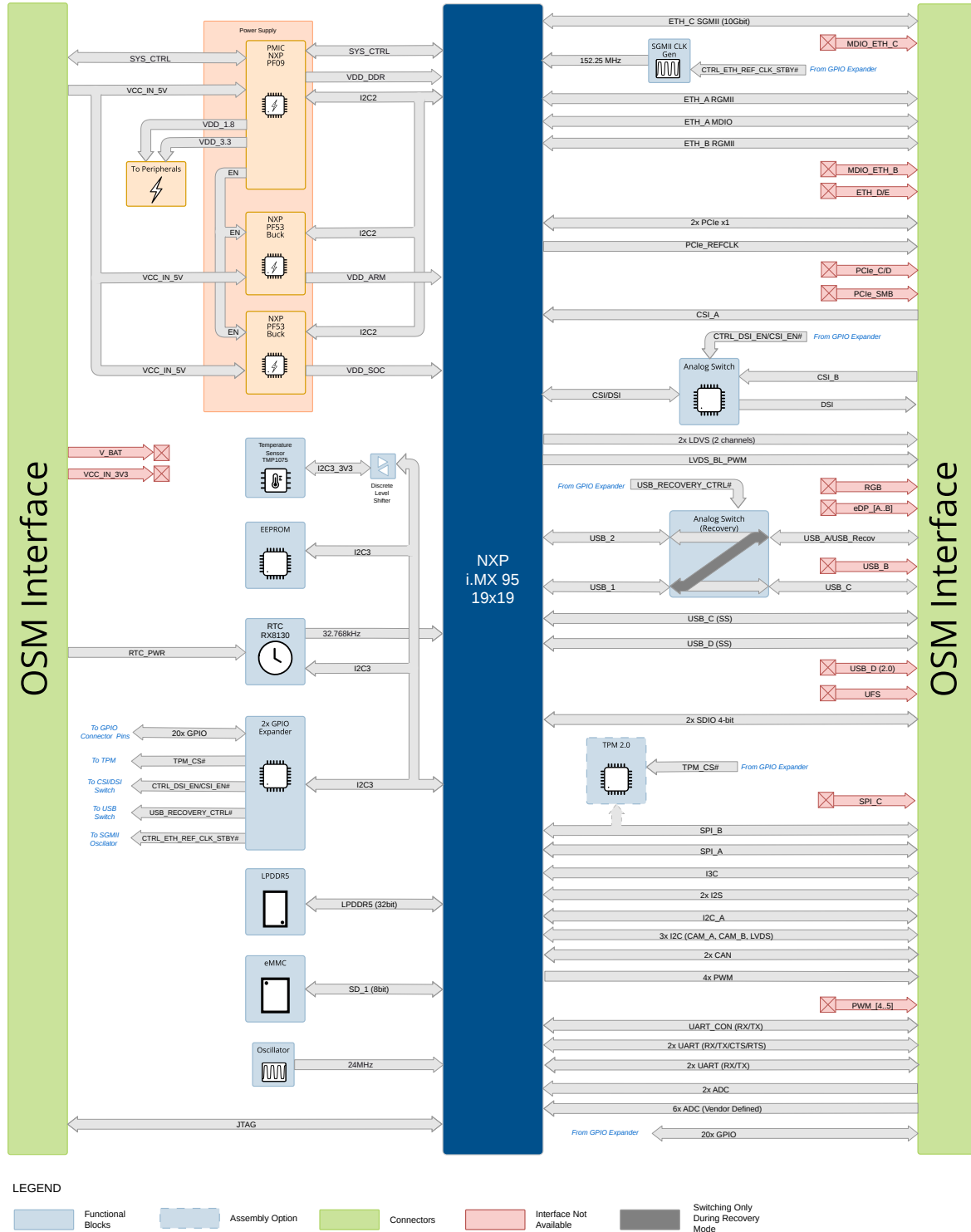
The overall architecture combines:

- **Application Processing:** up to six Arm® Cortex®-A55 cores optimized for efficient application execution
- **Real-Time Processing:** Integrated Arm® Cortex®-M33 core for low-latency control and real-time tasks
- **Graphics & Multimedia:** Integrated GPU and multimedia accelerators supporting display and video processing
- **AI Acceleration:** On-chip NPU for neural processing capabilities and efficient machine learning inference at the edge
- **Connectivity:** Integrated Ethernet, USB, CAN FD, and SD/SDIO interfaces, with support for external wireless connectivity
- **Security:** EdgeLock® security subsystem with support for secure boot, cryptographic acceleration, and TPM 2.0 options
- **System Infrastructure:** EEPROM, I²C-based peripherals, Power management, RTC, and on-module thermal monitoring

The functional block diagram of the OSM iMX95 SoM is shown in [Figure 1](#).

2.1 Block Diagram

Figure 1: Block diagram



3 OSM Interface

The OSM iMX95 system-on-module is implemented as a **Size-L Open Standard Module (OSM)** and is designed for **direct surface-mount soldering to the carrier board**. The mechanical and electrical interface is realized through a standardized array of solder pads on the underside of the module, in accordance with the OSM Size-L specification.

This solderable interface exposes the full feature set of the NXP i.MX 95 system-on-chip, including high-speed interfaces, low-speed control signals, and multiple power domains, while eliminating the need for board-to-board connectors. This approach enables a **reduced overall system height**, improved mechanical robustness, and enhanced suitability for volume production.

The OSM pad layout supports controlled-impedance routing for high-speed signals and is optimized for compact carrier board designs. Proper PCB layout, footprint definition, and assembly process control are essential to ensure signal integrity, mechanical reliability, and long-term performance.

3.1 Carrier Board Design Considerations

The carrier board must implement the **OSM Size-L land pattern** as defined by the Open Standard Module specification. Designers should follow the recommended guidelines for:

- Pad geometry and solder mask definition
- Placement accuracy and coplanarity
- Reflow soldering profile
- Keep-out areas and component clearance

Adhering to the OSM specification and recommended PCB and assembly practices is critical to achieving reliable electrical connections and robust mechanical attachment between the OSM iMX95 module and the carrier board.

For more information, refer to the [Section 9](#).

3.2 Contact Assignment

[Table 6](#) to [Table 9](#) describe the **contact assignment of the OSM Size-L interface** and highlight the compatibility of each contact's electrical function with the OSM Family Specification. The physical location and designation of each **OSM contact** within the module's solderable interface are shown in [Figure 2](#). A detailed explanation of the compatibility groups defined in the specification is available in [Section 1.6](#).



Naming Convention

On the OSM iMX95 system-on-module, all external interfaces are exposed exclusively through the **OSM Size-L contact array**, as defined by the Open Standard Module specification (Size-L, Rev. 1.2). Signals are identified by their **OSM contact designators**, which correspond directly to the carrier board **land pattern** shown in [Figure 2](#).

Figure 2: OSM iMX95 pad layout top view

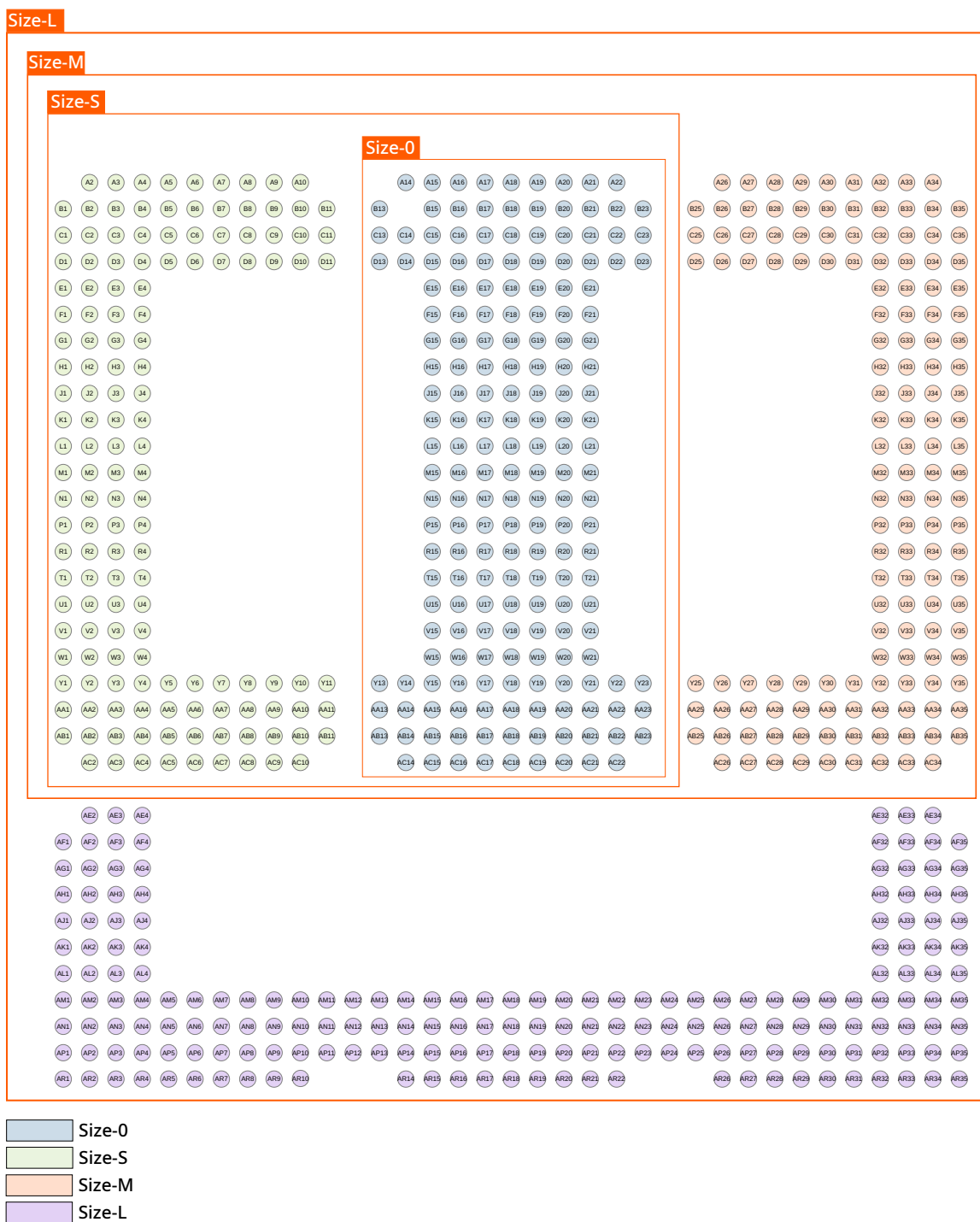


Table 6: OSM Size-0 contact assignment

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
A14	UART_A_RX	SHOULD	NVCC_GPIO	GPIO_IO05	L45	–	Input with PD	–
A15	CH1_RX_N	MAY	–	–	–	–	–	Not connected
A16	CH1_LINK	MAY	–	–	–	–	–	Not connected
A17	CH1_TX_N	MAY	–	–	–	–	–	Not connected
A18	CH0_SYNC_TRIGGER	MAY	–	–	–	–	–	Not connected
A19	CH0_RX_N	MAY	–	–	–	–	–	Not connected
A20	CH0_LINK	MAY	–	–	–	–	–	Not connected
A21	CH0_TX_N	MAY	–	–	–	–	–	Not connected
A22	UART_C_RX	SHOULD	NVCC_GPIO	GPIO_IO15	P44	–	Input with PD	–
B13	UART_A_TX	SHOULD	NVCC_GPIO	GPIO_IO04	K46	–	Input with PD	–
B15	CH1_RX_P	MAY	–	–	–	–	–	Not connected
B16	CH1_ACT	MAY	–	–	–	–	–	Not connected
B17	CH1_TX_P	MAY	–	–	–	–	–	Not connected
B18	CH1_SYNC_TRIGGER	MAY	–	–	–	–	–	Not connected
B19	CH0_RX_P	MAY	–	–	–	–	–	Not connected
B20	CH0_ACT	MAY	–	–	–	–	–	Not connected
B21	CH0_TX_P	MAY	–	–	–	–	–	Not connected
B22	Vendor Defined	MAY	AVDD_1P8_ADC	ADC_IN2	C39	–	–	–
B23	UART_C_TX	SHOULD	NVCC_GPIO	GPIO_IO14	N51	–	Input with PD	–
C13	UART_A_RTS	SHOULD	NVCC_GPIO	GPIO_IO07	L51	–	Input with PD	–
C14	UART_A_CTS	SHOULD	NVCC_GPIO	GPIO_IO06	L49	–	Input with PD	–
C15	CH1_RXC	MAY	–	–	–	–	–	Not connected
C16	Vendor Defined	MAY	AVDD_1P8_ADC	ADC_IN3	B40	–	–	–

Continued on next page

Table 6: OSM Size-0 contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
C17	CH1_TXC	MAY	–	–	–	–	–	Not connected
C18	TEST_GENERIC	SHOULD	–	–	–	–	–	–
C19	CH0_RXC	MAY	–	–	–	–	–	Not connected
C20	SDIO_A_IOPWR	SHOULD	V3.3_1.8_SD	–	AD36	–	–	–
C21	CH0_TXC	MAY	–	–	–	–	–	Not connected
C22	UART_D_RX	SHOULD	NVCC_AON	UART2_RXD	E51	–	Input with PD	–
C23	UART_D_TX	SHOULD	NVCC_AON	UART2_TXD	F48	–	Input with PD	with 10 kΩ pull-down ¹
D13	UART_B_TX	SHOULD	NVCC_GPIO	GPIO_IO00	J49	–	Input with PD	–
D14	UART_B_RX	SHOULD	NVCC_GPIO	GPIO_IO01	J51	–	Input with PD	–
D15	UART_B_RTS	SHOULD	NVCC_GPIO	GPIO_IO03	K52	–	Input with PD	–
D16	UART_B_CTS	SHOULD	NVCC_GPIO	GPIO_IO02	K48	–	Input with PD	–
D17	GPIO_A_0	SHOULD	NVCC_WAKEUP	XSPI1_DATA0	AJ45	–	Input with PD	–
D18	GND	SHALL	–	–	–	–	–	–
D19	GPIO_B_0	SHOULD	NVCC_WAKEUP	XSPI1_SS1_B	AH42	–	Input with PD	–
D20	SDIO_A_WP	MAY	–	–	–	–	–	Not connected
D21	SDIO_A_PWR_EN	SHOULD	NVCC_SD2	SD2_RESET_B	AD52	–	Input with PD	with 4.7 kΩ pull-up to +V3.3_1.8_SD
D22	UART_CON_RX	SHOULD	NVCC_AON	UART1_RXD	E49	–	Input with PD	–
D23	UART_CON_TX	SHOULD	NVCC_AON	UART1_TXD	F52	–	–	–
E15	GND	SHALL	–	–	–	–	–	–
E16	ETH_A_(R)(G)MII_CRS	MAY	–	–	–	–	–	Not connected
E17	GPIO_A_1	SHOULD	NVCC_WAKEUP	XSPI1_DATA1	AH46	–	Input with PD	–
E18	DISP_BL_PWM/ PWM_0	SHOULD	NVCC_GPIO	GPIO_IO12	N45	–	Input with PD	–
E19	GPIO_B_1	SHOULD	NVCC_WAKEUP	XSPI1_DQS	AK44	–	Input with PD	–

Continued on next page

Table 6: OSM Size-0 contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
E20	SDIO_A_CMD	SHOULD	NVCC_SD2	SD2_CMD	AB52	–	Input with PD	–
E21	GND	SHALL	–	–	–	–	–	–
F15	ETH_A_(R)(G)MII_COL	MAY	–	–	–	–	–	Not connected
F16	GND	SHALL	–	–	–	–	–	–
F17	GPIO_A_2	SHOULD	NVCC_WAKEUP	XSPI1_DATA2	AJ47	–	Input with PD	–
F18	PWM_1	SHOULD	NVCC_GPIO	GPIO_IO13	N49	–	Input with PD	–
F19	GPIO_B_2	SHOULD	NVCC_AON	PDM_CLK	F46	–	Input with PD	–
F20	GND	SHALL	–	–	–	–	–	–
F21	SDIO_A_CLK	SHOULD	NVCC_SD2	SD2_CLK	AB48	–	Input with PD	–
G15	ETH_A_(S)(R)(G)MII_TXD1	SHOULD	NVCC_ENET	ENET1_TD1	AG35	–	Input with PD	–
G16	ETH_A_(S)(R)(G)MII_TXD3	SHOULD	NVCC_ENET	ENET1_TD3	AG37	–	Input with PD	–
G17	GPIO_A_3	SHOULD	NVCC_WAKEUP	XSPI1_DATA3	AK48	–	Input with PD	–
G18	PWM_2	SHOULD	NVCC_GPIO	GPIO_IO18	P52	–	Input with PD	–
G19	GPIO_B_3	SHOULD	NVCC_AON	PDM_BIT_STREAM1	H46	–	Input with PD	–
G20	SDIO_A_D0	SHOULD	NVCC_SD2	SD2_DATA0	AC51	–	Input with PD	–
G21	SDIO_A_D1	SHOULD	NVCC_SD2	SD2_DATA1	AC49	–	Input with PD	–
H15	ETH_A_(S)(R)(G)MII_TXD0	SHOULD	NVCC_ENET	ENET1_TD0	AG33	–	Input with PD	–
H16	ETH_A_(S)(R)(G)MII_TXD2	SHOULD	NVCC_ENET	ENET1_TD2	AF36	–	Input with PD	–
H17	GPIO_A_4	SHOULD	NVCC_WAKEUP	XSPI1_SCLK	AJ43	–	Input with PD	–
H18	PWM_3	SHOULD	NVCC_GPIO	GPIO_IO19	R45	–	Input with PD	–
H19	GPIO_B_4	SHOULD	–	–	–	GPIO Expander IC16 - P0_4 (5)	Input without PU/PD	–
H20	SDIO_A_D2	SHOULD	NVCC_SD2	SD2_DATA2	AA51	–	Input with PD	–
H21	SDIO_A_D3	SHOULD	NVCC_SD2	SD2_DATA3	AA49	–	Input with PD	–

Continued on next page

Table 6: OSM Size-0 contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
J15	ETH_A_(R)(G)MII_TX_CLK	SHOULD	NVCC_ENET	ENET1_TXC	AG31	–	Input with PD	–
J16	GND	SHALL	–	–	–	–	–	–
J17	GPIO_A_5	SHOULD	NVCC_WAKEUP	XSPI1_SS0_B	AJ41	–	Input with PD	–
J18	PWM_4	MAY	–	–	–	–	–	Not connected
J19	GPIO_B_5	SHOULD	–	–	–	GPIO Expander IC16 - P0_5 (6)	Input without PU/PD	–
J20	GND	SHALL	–	–	–	–	–	–
J21	SDIO_A_CD#	SHOULD	NVCC_SD2	SD2_CD_B	AD48	–	Input with PD	with 10 kΩ pull-up to +V3.3_1.8_SD
K15	ETH_A_(S)(R)(G)MII_RXD0	SHOULD	NVCC_ENET	ENET1_RD0	AJ35	–	Input with PD	–
K16	ETH_A_(R)(G)MII_TX_EN(ER)	SHOULD	NVCC_ENET	ENET1_TX_CTL	AF32	–	Input with PD	–
K17	SPI_A_CS1#/ GPIO_A_6	SHOULD	NVCC_GPIO	GPIO_IO33	W45	–	Input with PD	–
K18	PWM_5	MAY	–	–	–	–	–	Not connected
K19	GPIO_B_6	SHOULD	–	–	–	GPIO Expander IC16 - P0_6 (7)	Input without PU/PD	–
K20	SDIO_B_CLK	SHOULD	NVCC_WAKEUP	SD3_CLK	AG51	–	Input with PD	–
K21	SDIO_B_CMD	SHOULD	NVCC_WAKEUP	SD3_CMD	AF48	–	Input with PD	–
L15	ETH_A_(S)(R)(G)MII_RXD1	SHOULD	NVCC_ENET	ENET1_RD1	AK36	–	Input with PD	–
L16	ETH_A_(R)(G)MII_RX_ER	MAY	–	–	–	–	–	Not connected
L17	SPI_B_CS1#/ GPIO_A_7	SHOULD	NVCC_AON	SAI1_TXFS	G49	–	Input with PD	with 10 kΩ pull-down ¹
L18	GND	SHALL	–	–	–	–	–	–
L19	GPIO_B_7	SHOULD	–	–	–	GPIO Expander IC16 - P0_7 (8)	Input without PU/PD	–
L20	SDIO_B_D0	SHOULD	NVCC_WAKEUP	SD3_DATA0	AG49	–	Input with PD	–
L21	SDIO_B_D1	SHOULD	NVCC_WAKEUP	SD3_DATA1	AH52	–	Input with PD	–
M15	ETH_A_(R)(G)MII_RX_DV(ER)	SHOULD	NVCC_ENET	ENET1_RX_CTL	AH34	–	Input with PD	–
M16	GND	SHALL	–	–	–	–	–	–

Continued on next page

Table 6: OSM Size-0 contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
M17	ETH_IOPWR	MAY	–	–	–	–	–	Connected to +V1.8 rail
M18	ADC_0	SHOULD	AVDD_1P8_ADC	ADC_IN0	A37	–	–	–
M19	VCC_2_TEST2	MAY	–	–	–	–	–	–
M20	GND	SHALL	–	–	–	–	–	–
M21	SDIO_B_D2	SHOULD	NVCC_WAKEUP	SD3_DATA2	AE49	–	Input with PD	–
N15	ETH_A_(R)(G)MII_RXD2	SHOULD	NVCC_ENET	ENET1_RD2	AJ37	–	Input with PD	–
N16	ETH_A_SDP	SHOULD	NVCC_CCM_DAP	CCM_CLKO2	AF20	–	Output low	–
N17	JTAG_TCK(SWCLK)	SHOULD	NVCC_CCM_DAP	DAP_TCLK_SWCLK	AG21	–	Input with PD	–
N18	ADC_1	SHOULD	AVDD_1P8_ADC	ADC_IN1	B38	–	–	–
N19	JTAG_TMS(SWDIO)	SHOULD	NVCC_CCM_DAP	DAP_TMS_SWDIO	AH22	–	Input with PU	–
N20	SDIO_B_D3	SHOULD	NVCC_WAKEUP	SD3_DATA3	AF52	–	Input with PD	–
N21	SDIO_B_D4	MAY	–	–	–	–	–	Not connected
P15	ETH_A_(R)(G)MII_RXD3	SHOULD	NVCC_ENET	ENET1_RD3	AH38	–	Input with PD	–
P16	Vendor Defined	MAY	–	–	–	EEPROM (2 kB) $\overline{WC}$	–	with 10 k Ω pull-up to +V1.8
P17	JTAG_TDI	SHOULD	NVCC_CCM_DAP	DAP_TDI	AK24	–	Input with PU	–
P18	GND	SHALL	–	–	–	–	–	–
P19	JTAG_RTCK	MAY	–	–	–	–	–	Not connected
P20	SDIO_B_D5	MAY	–	–	–	–	–	Not connected
P21	SDIO_B_D6	MAY	–	–	–	–	–	Not connected
R15	ETH_A_(R)(G)MII_RX_CLK	SHOULD	NVCC_ENET	ENET1_RXC	AJ33	–	Input with PD	–
R16	GND	SHALL	–	–	–	–	–	–
R17	JTAG_TDO(SWO)	SHOULD	NVCC_CCM_DAP	DAP_TDO_TRACESWO	AJ23	–	Input without PU/PD	–
R18	BOOT_SEL1#	MAY	–	–	–	–	–	Not connected

Continued on next page

Table 6: OSM Size-0 contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
R19	JTAG_nTRST	MAY	–	–	–	–	–	Not connected
R20	GND	SHALL	–	–	–	–	–	–
R21	SDIO_B_D7	MAY	–	–	–	–	–	Not connected
T15	ETH_MDIO	SHOULD	NVCC_ENET	ENET1_MDIO	AJ39	–	Input with PD	–
T16	ETH_MDC	SHOULD	NVCC_ENET	ENET1_MDC	AK40	–	Input with PD	–
T17	FORCE_RECOVERY#	SHOULD	–	–	–	Recovery circuit set BOOT_MODE0 high	–	with 10 kΩ pull-up to +V1.8
T18	I2S_B_LRCLK	SHOULD	NVCC_WAKEUP	XSPI1_DATA5	AK50	–	Input with PD	–
T19	I2S_B_BITCLK	SHOULD	NVCC_WAKEUP	XSPI1_DATA6	AJ51	–	Input with PD	–
T20	SDIO_B_IOPWR	MAY	–	–	–	–	–	Not connected
T21	SDIO_B_CD#	MAY	–	–	–	–	–	Connected to +V1.8 rail
U15	SPI_A_SDI(IO1)	SHOULD	NVCC_GPIO	GPIO_IO35	W51	–	Input with PD	–
U16	SPI_A_SCK	SHOULD	NVCC_GPIO	GPIO_IO37	Y52	–	Input with PD	–
U17	RESET_IN#	MAY	–	–	–	Reset circuit	–	with 10 kΩ pull-up to +V1.8_BB5M
U18	VCC_OUT_IO	MAY	–	–	–	–	–	Connected to +V1.8 rail
U19	BOOT_SELO#	MAY	–	–	–	–	–	Not connected
U20	SDIO_B_WP	MAY	–	–	–	–	–	Not connected
U21	SDIO_B_PWR_EN	MAY	–	–	–	–	–	Not connected
V15	SPI_A_SDO(IO0)	SHOULD	NVCC_GPIO	GPIO_IO36	Y48	–	Input with PD	–
V16	GND	SHALL	–	–	–	–	–	–
V17	CARRIER_PWR_EN	SHALL	–	–	–	PMIC circuit	–	with 120 Ω pull-up to +V1.8
V18	I2S_MCLK	SHOULD	NVCC_GPIO	GPIO_IO17	P48	–	Input with PD	–
V19	I2S_B_DATA_IN	SHOULD	NVCC_WAKEUP	XSPI1_DATA7	AH50	–	Input with PD	–
V20	GND	SHALL	–	–	–	–	–	–

Continued on next page

Table 6: OSM Size-0 contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
V21	I2S_A_DATA_IN	SHOULD	NVCC_GPIO	GPIO_IO20	R49	–	Input with PD	–
W15	SPI_A_HOLD_(IO3)	MAY	–	–	–	–	–	Not connected
W16	SPI_A_WP_(IO2)	MAY	–	–	–	–	–	Not connected
W17	RTC_PWR	MAY	–	–	–	–	–	Not connected
W18	I2S_A_LRCLK	SHOULD	NVCC_GPIO	GPIO_IO26	U45	–	Input with PD	–
W19	I2S_B_DATA_OUT	SHOULD	NVCC_WAKEUP	XSPI1_DATA4	AJ49	–	Input with PD	–
W20	I2S_A_BITCLK	SHOULD	NVCC_GPIO	GPIO_IO16	P46	–	Input with PD	–
W21	I2S_A_DATA_OUT	SHOULD	NVCC_GPIO	GPIO_IO21	R51	–	Input with PD	–
Y13	CARRIER_STBY#	MAY	–	–	–	–	–	Not connected
Y14	RESET_OUT#	MAY	–	–	–	PMIC reset circuit	–	–
Y15	SPI_A_CS0#	SHOULD	NVCC_GPIO	GPIO_IO34	W49	–	Input with PD	–
Y16	VCC_3_TEST	MAY	–	–	–	–	–	–
Y17	VCC_IN_5V	MAY	–	–	–	–	–	–
Y18	GND	SHALL	–	–	–	–	–	–
Y19	VCC_IN_3V3	MAY	–	–	–	–	–	Not connected
Y20	VCC_4_TEST	MAY	–	–	–	–	–	–
Y21	SPI_B_SCK	SHOULD	–	–	–	IC10 SPI_CLK (D4)	–	with 10 kΩ pull-up to +V1.8 ¹
Y22	SPI_B_SDI	SHOULD	–	–	–	IC10 SPI_MISO (B4)	–	–
Y23	SPI_B_SDO	SHOULD	–	–	–	IC10 SPI_MOSI (C4)	–	–
AA13	RESERVED	MAY	–	–	–	–	–	Not connected
AA14	GND	SHALL	–	–	–	–	–	–
AA15	I2C_A_SCL	SHOULD	NVCC_AON	I2C1_SCL	D48	–	Input with PD	with 2.2 kΩ pull-up to +V1.8
AA16	I2C_A_SDA	SHOULD	NVCC_AON	I2C1_SDA	D52	–	Input with PD	with 2.2 kΩ pull-up to +V1.8

Continued on next page

Table 6: OSM Size-0 contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AA17	GND	SHALL	–	–	–	–	–	–
AA18	V_BAT	MAY	–	–	–	–	–	Not connected
AA19	GND	SHALL	–	–	–	–	–	–
AA20	I2C_B_SCL	SHOULD	NVCC_GPIO	GPIO_IO23	T46	–	Input with PD	with 2.2 kΩ pull-up to +V1.8
AA21	I2C_B_SDA	SHOULD	NVCC_GPIO	GPIO_IO22	T44	–	Input with PD	with 2.2 kΩ pull-up to +V1.8
AA22	GND	SHALL	–	–	–	–	–	–
AA23	SPI_B_CS0#	SHOULD	NVCC_AON	PDM_BIT_STREAM0	G45	–	Input with PD	–
AB13	USB_A_D_N	SHOULD	VDD_USB_3P3	USB2_D_N	A25	USB 2.0 480-Mbps 2:1 multiplexer	–	Default switch to USB2_D_N
AB14	USB_A_ID	SHOULD	VDD_USB_1P8	USB2_ID	F24	–	–	with 10 kΩ pull-up to +V1.8
AB15	GND	SHALL	–	–	–	–	–	–
AB16	USB_A_VBUS	SHOULD	VDD_USB_3P3	USB2_VBUS	E27	–	–	with 30 kΩ series resistor
AB17	CAN_A_RX	SHOULD	NVCC_GPIO	GPIO_IO27	U49	–	Input with PD	–
AB18	V_BAT	MAY	–	–	–	–	–	Not connected
AB19	CAN_B_RX	SHOULD	NVCC_CCM_DAP	CCM_CLKO4	AJ21	–	Input with PD	–
AB20	USB_B_VBUS	MAY	VDD_USB_3P3	–	–	–	–	Not connected
AB21	GND	SHALL	–	–	–	–	–	–
AB22	USB_B_ID	MAY	–	–	–	–	–	Not connected
AB23	USB_B_D_N	MAY	VDD_USB_3P3	–	–	–	–	Not connected
AC14	USB_A_D_P	SHOULD	VDD_USB_3P3	USB2_D_P	B24	USB 2.0 480-Mbps 2:1 multiplexer	–	Default switch to USB2_D_P
AC15	USB_A_OC#	SHOULD	–	–	–	GPIO Expander IC16 - P0_1 (2)	Input without PU/PD	with 10 kΩ pull-up to +V1.8
AC16	USB_A_EN	SHOULD	–	–	–	GPIO Expander IC16 - P0_0 (1)	Input without PU/PD	–
AC17	CAN_A_TX	SHOULD	NVCC_GPIO	GPIO_IO25	T52	–	Input with PD	–
AC18	DEBUG_EN	MAY	–	–	–	–	–	Not connected

Continued on next page

Table 6: OSM Size-0 contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AC19	CAN_B_TX	SHOULD	NVCC_CCM_DAP	CCM_CLK03	AK20	–	Input with PD	–
AC20	USB_B_EN	MAY	–	–	–	–	–	Not connected
AC21	USB_B_OC#	MAY	–	–	–	–	–	Not connected
AC22	USB_B_D_P	MAY	VDD_USB_3P3	–	–	–	–	Not connected

¹ This pin is connected to the boot mode, so care must be taken during booting.

Table 7: OSM Size-S additional contact assignment

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
A2	CSL_A_DATA1_N	SHOULD	VDD_MIP1_P8	MIPI_CSI1_D1_N	D18	–	–	–
A3	CSL_A_DATA1_P	SHOULD	VDD_MIP1_P8	MIPI_CSI1_D1_P	E17	–	–	–
A4	GND	SHALL	–	–	–	–	–	–
A5	CSL_A_DATA2_N	SHOULD	VDD_MIP1_P8	MIPI_CSI1_D2_N	D14	–	–	–
A6	CSL_A_DATA2_P	SHOULD	VDD_MIP1_P8	MIPI_CSI1_D2_P	E13	–	–	–
A7	GND	SHALL	–	–	–	–	–	–
A8	USB_C_SSTX_P	SHOULD	VDD_USB_0P8	USB1_TX0_P	D26	–	–	–
A9	USB_C_SSTX_N	SHOULD	VDD_USB_0P8	USB1_TX0_N	E25	–	–	–
A10	GND	SHALL	–	–	–	–	–	–
B1	CSL_A_DATA0_P	SHOULD	VDD_MIP1_P8	MIPI_CSI1_D0_P	E19	–	–	–
B2	GND	SHALL	–	–	–	–	–	–
B3	CSL_A_CLOCK_N	SHOULD	VDD_MIP1_P8	MIPI_CSI1_CLK_N	F16	–	–	–
B4	CSL_A_CLOCK_P	SHOULD	VDD_MIP1_P8	MIPI_CSI1_CLK_P	E15	–	–	–
B5	GND	SHALL	–	–	–	–	–	–
B6	CSL_A_DATA3_N	SHOULD	VDD_MIP1_P8	MIPI_CSI1_D3_N	F12	–	–	–
B7	CSL_A_DATA3_P	SHOULD	VDD_MIP1_P8	MIPI_CSI1_D3_P	E11	–	–	–
B8	GND	SHALL	–	–	–	–	–	–
B9	GND	SHALL	–	–	–	–	–	–
B10	USB_C_SSRX_P	SHOULD	VDD_USB_0P8	USB1_RX0_P	A21	–	–	–
B11	USB_C_SSRX_N	SHOULD	VDD_USB_0P8	USB1_RX0_N	B20	–	–	–
C1	CSL_A_DATA0_N	SHOULD	VDD_MIP1_P8	MIPI_CSI1_D0_N	F20	–	–	–
C2	CAM_MCK	SHOULD	NVCC_CCM_DAP	CCM_CLKO1	AH20	–	Output low	–
C3	CAM_A_SDA/ CSL_A_TX_N	SHOULD	NVCC_GPIO	GPIO_IO30	V46	–	Input with PD	with 2.2 kΩ pull-up to +V1.8

Continued on next page

Table 7: OSM Size-S additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
C4	CAM_A_SCL/ CSI_A_TX_P	SHOULD	NVCC_GPIO	GPIO_IO31	V48	–	Input with PD	with 2.2 kΩ pull-up to +V1.8
C5	VCC_6_TEST	MAY	–	–	–	–	–	–
C6	ETH_B_MDC	MAY	–	–	–	–	–	Not connected
C7	ETH_B_MDIO	MAY	–	–	–	–	–	Not connected
C8	USB_C_OC#	SHOULD	–	–	–	GPIO Expander IC16 - P0_3 (4)	Input without PU/PD	with 10 kΩ pull-up to +V1.8
C9	USB_C_VBUS	SHOULD	–	–	–	–	–	–
C10	USB_C_EN	SHOULD	–	–	–	GPIO Expander IC16 - P0_2 (3)	Input without PU/PD	–
C11	GND	SHALL	–	–	–	–	–	–
D1	GND	SHALL	–	–	–	–	–	–
D2	ETH_B_(R)(G)MII_CRS	MAY	–	–	–	–	–	Not connected
D3	GPIO_C_0	SHOULD	–	–	–	GPIO Expander IC6 - P0_0 (1)	Input without PU/PD	–
D4	GPIO_C_1	SHOULD	–	–	–	GPIO Expander IC6 - P0_1 (2)	Input without PU/PD	–
D5	GND	SHALL	–	–	–	–	–	–
D6	Vendor Defined	MAY	NVCC_BB5M	TAMPER0	F36	–	–	–
D7	Vendor Defined	MAY	NVCC_BB5M	TAMPER1	D38	–	–	–
D8	GND	SHALL	–	–	–	–	–	–
D9	USB_C_ID	SHOULD	Internal 1.8 V	USB1_ID	C23	–	–	with 10 kΩ pull-up to +V1.8
D10	USB_C_D_P	SHOULD	–	USB1_D_P	C19	USB 2.0 480-Mbps 2:1 multiplexer	–	Default switch to USB1_D_P
D11	USB_C_D_N	SHOULD	–	USB1_D_N	B18	USB 2.0 480-Mbps 2:1 multiplexer	–	Default switch to USB1_D_N
E1	ETH_B_(R)(G)MII_COL	MAY	–	–	–	–	–	Not connected
E2	GND	SHALL	–	–	–	–	–	–

Continued on next page

Table 7: OSM Size-S additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
E3	GPIO_C_2	SHOULD	–	–	–	GPIO Expander IC6 - P0_2 (3)	Input without PU/PD	–
E4	GPIO_C_3	SHOULD	–	–	–	GPIO Expander IC6 - P0_3 (4)	Input without PU/PD	–
F1	ETH_B_(S)(R)(G)MII_TXD1	SHOULD	NVCC_ENET	ENET2_TD1	AG27	–	Input with PD	–
F2	ETH_B_(S)(R)(G)MII_TXD3	SHOULD	NVCC_ENET	ENET2_TD3	AG29	–	Input with PD	–
F3	DISP_VDD_EN/ GPIO_C_4	SHOULD	–	–	–	GPIO Expander IC6 - P0_4 (5)	Input without PU/PD	–
F4	DISP_BL_EN/ GPIO_C_5	SHOULD	–	–	–	GPIO Expander IC6 - P0_5 (6)	Input without PU/PD	–
G1	ETH_B_(S)(R)(G)MII_TXD0	SHOULD	NVCC_ENET	ENET2_TD0	AG25	–	Input with PD	–
G2	ETH_B_(S)(R)(G)MII_TXD2	SHOULD	NVCC_ENET	ENET2_TD2	AF28	–	Input with PD	–
G3	CAM_A_PWR/ GPIO_C_6	SHOULD	–	–	–	GPIO Expander IC6 - P0_6 (7)	Input without PU/PD	–
G4	CAM_A_RST#/ GPIO_C_7	SHOULD	–	–	–	GPIO Expander IC6 - P0_7 (8)	Input without PU/PD	–
H1	ETH_B_(R)(G)MII_TX_CLK	SHOULD	NVCC_ENET	ENET2_TXC	AG23	–	Input with PD	–
H2	GND	SHALL	–	–	–	–	–	–
H3	RGB_CS#	MAY	–	–	–	–	–	Not connected
H4	GND	SHALL	–	–	–	–	–	–
J1	ETH_B_(S)(R)(G)MII_RXD0	SHOULD	NVCC_ENET	ENET2_RD0	AJ27	–	Input with PD	–
J2	ETH_B_(R)(G)MII_TX_EN(ER)	SHOULD	NVCC_ENET	ENET2_TX_CTL	AF24	–	Input with PD	–
J3	RGB_RESET#	MAY	–	–	–	–	–	Not connected
J4	RGB_DE	MAY	–	–	–	–	–	Not connected
K1	ETH_B_(S)(R)(G)MII_RXD1	SHOULD	NVCC_ENET	ENET2_RD1	AK28	–	Input with PD	–
K2	ETH_B_(R)(G)MII_RX_ER	MAY	–	–	–	–	–	Not connected
K3	RGB_HSYNC	MAY	–	–	–	–	–	Not connected
K4	RGB_DISP	MAY	–	–	–	–	–	Not connected
L1	ETH_B_(R)(G)MII_RX_DV(ER)	SHOULD	NVCC_ENET	ENET2_RX_CTL	AH26	–	Input with PD	–

Continued on next page

Table 7: OSM Size-S additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
L2	GND	SHALL	–	–	–	–	–	–
L3	RGB_VSYNC	MAY	–	–	–	–	–	Not connected
L4	GND	SHALL	–	–	–	–	–	–
M1	ETH_B_(R)(G)MII_RXD2	SHOULD	NVCC_ENET	ENET2_RD2	AJ29	–	Input with PD	–
M2	ETH_B_SDP	MAY	–	–	–	–	–	Not connected
M3	RGB_B5	MAY	–	–	–	–	–	Not connected
M4	RGB_(PIXEL)CLK	MAY	–	–	–	–	–	Not connected
N1	ETH_B_(R)(G)MII_RXD3	SHOULD	NVCC_ENET	ENET2_RD3	AH30	–	Input with PD	–
N2	RESERVED	MAY	–	–	–	–	–	Not connected
N3	RGB_B3	MAY	–	–	–	–	–	Not connected
N4	RGB_B4	MAY	–	–	–	–	–	Not connected
P1	ETH_B_(R)(G)MII_RX_CLK	SHOULD	NVCC_ENET	ENET2_RXC	AJ25	–	Input with PD	–
P2	GND	SHALL	–	–	–	–	–	–
P3	RGB_B2	MAY	–	–	–	–	–	Not connected
P4	GND	SHALL	–	–	–	–	–	–
R1	GND	SHALL	–	–	–	–	–	–
R2	PCIe_SM_ALERT#	MAY	–	–	–	–	–	Not connected
R3	RGB_B1	MAY	–	–	–	–	–	Not connected
R4	RGB_B0	MAY	–	–	–	–	–	Not connected
T1	PCIe_SMCLK	MAY	–	–	–	–	–	Not connected
T2	PCIe_WAKE#	SHOULD	NVCC_ENET	ENET2_MDIO	AJ31	–	Input with PD	with 10 kΩ pull-up to +V1.8
T3	RGB_G4	MAY	–	–	–	–	–	Not connected
T4	RGB_G5	MAY	–	–	–	–	–	Not connected

Continued on next page

Table 7: OSM Size-S additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
U1	PCIe_SMDAT	MAY	–	–	–	–	–	Not connected
U2	GND	SHALL	–	–	–	–	–	–
U3	RGB_G3	MAY	–	–	–	–	–	Not connected
U4	GND	SHALL	–	–	–	–	–	–
V1	GND	SHALL	–	–	–	–	–	–
V2	PCIe_A_PERST#	SHOULD	–	–	–	GPIO Expander IC16 - P1_0 (10)	Input without PU/PD	–
V3	RGB_G1	MAY	–	–	–	–	–	Not connected
V4	RGB_G2	MAY	–	–	–	–	–	Not connected
W1	PCIe_REFCLK_P	SHOULD	VDD_ANA_1P8	PCIE_REF_OUT_CLK_P	B26	–	–	Can be left unconnected
W2	PCIe_CLKREQ#	SHOULD	NVCC_GPIO	GPIO_IO32	V52	–	Input with PD	with 10 kΩ pull-up to +V1.8
W3	GND	SHALL	–	–	–	–	–	–
W4	RGB_G0	MAY	–	–	–	–	–	Not connected
Y1	PCIe_REFCLK_N	SHOULD	VDD_ANA_1P8	PCIE_REF_OUT_CLK_N	C27	–	–	Can be left unconnected
Y2	GND	SHALL	–	–	–	–	–	–
Y3	VCC_5_TEST	MAY	–	–	–	–	–	–
Y4	RGB_R5	MAY	–	–	–	–	–	Not connected
Y5	RGB_R4	MAY	–	–	–	–	–	Not connected
Y6	RGB_R2	MAY	–	–	–	–	–	Not connected
Y7	RGB_R0	MAY	–	–	–	–	–	Not connected
Y8	VCC_IN_5V	MAY	–	–	–	–	–	–
Y9	VCC_IN_5V	MAY	–	–	–	–	–	–
Y10	VCC_IN_5V	MAY	–	–	–	–	–	–
Y11	VCC_IN_5V	MAY	–	–	–	–	–	–

Continued on next page

Table 7: OSM Size-S additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AA1	GND	SHALL	–	–	–	–	–	–
AA2	RESERVED	MAY	–	–	–	–	–	Not connected
AA3	DSI_TE	MAY	–	–	–	–	–	Not connected
AA4	GND	SHALL	–	–	–	–	–	–
AA5	RGB_R3	MAY	–	–	–	–	–	Not connected
AA6	RGB_R1	MAY	–	–	–	–	–	Not connected
AA7	GND	SHALL	–	–	–	–	–	–
AA8	GND	SHALL	–	–	–	–	–	–
AA9	PWR_BTN#	SHALL	NVCC_BBBSM	ONOFF	F40	–	–	with 10 kΩ pull-up to +V1.8_BBBSM
AA10	GND	SHALL	–	–	–	–	–	–
AA11	GND	SHALL	–	–	–	–	–	–
AB1	PCIe_A_HSI0_P	SHOULD	VDD_PCI_1P8	PCIE1_RX0_P	B28	–	–	–
AB2	PCIe_A_HSI0_N	SHOULD	VDD_PCI_1P8	PCIE1_RX0_N	A29	–	–	–
AB3	GND	SHALL	–	–	–	–	–	–
AB4	DSI_DATA3_P/ LVDS_A_LANE3_P	SHOULD	VDD_MIPI_1P8/VDD_LVDS_1P8	MIPI_DSICSI1_D3_P	B6	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AB5	DSI_DATA3_N/ LVDS_A_LANE3_N	SHOULD	VDD_MIPI_1P8/VDD_LVDS_1P8	MIPI_DSICSI1_D3_N	C7	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AB6	GND	SHALL	–	–	–	–	–	–
AB7	DSI_CLOCK_P/ LVDS_A_CLK_P	SHOULD	VDD_MIPI_1P8/VDD_LVDS_1P8	MIPI_DSICSI1_CLK_P	B10	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AB8	DSI_CLOCK_N/ LVDS_A_CLK_N	SHOULD	VDD_MIPI_1P8/VDD_LVDS_1P8	MIPI_DSICSI1_CLK_N	C11	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch

Continued on next page

Table 7: OSM Size-S additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AB9	GND	SHALL	–	–	–	–	–	–
AB10	DSI_DATA0_P/ LVDS_A_LANE0_P	SHOULD	VDD_MIPI_1P8/VDD_LVDS_1P8	MIPI_DSICSI1_D0_P	B14	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AB11	DSI_DATA0_N/ LVDS_A_LANE0_N	SHOULD	VDD_MIPI_1P8/VDD_LVDS_1P8	MIPI_DSICSI1_D0_N	C15	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AC2	PCIE_A_HSO0_P	SHOULD	VDD_PCI_1P8	PCIE1_TX0_P	E29	–	–	–
AC3	PCIE_A_HSO0_N	SHOULD	VDD_PCI_1P8	PCIE1_TX0_N	D30	–	–	–
AC4	GND	SHALL	–	–	–	–	–	–
AC5	DSI_DATA2_P/ LVDS_A_LANE2_P	SHOULD	–	MIPI_DSICSI1_D2_P	B8	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AC6	DSI_DATA2_N/ LVDS_A_LANE2_N	SHOULD	–	MIPI_DSICSI1_D2_N	A9	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AC7	GND	SHALL	–	–	–	–	–	–
AC8	DSI_DATA1_P/ LVDS_A_LANE1_P	SHOULD	–	MIPI_DSICSI1_D1_P	B12	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AC9	DSI_DATA1_N/ LVDS_A_LANE1_N	SHOULD	–	MIPI_DSICSI1_D1_N	A13	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
AC10	GND	SHALL	–	–	–	–	–	–

Table 8: OSM Size-M additional contact assignment

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
A26	GND	SHALL	-	-	-	-	-	-
A27	USB_D_SSTX_P	SHOULD	VDD_USB_0P8	USB1_TX1_P	E21	-	-	-
A28	USB_D_SSTX_N	SHOULD	VDD_USB_0P8	USB1_TX1_N	D22	-	-	-
A29	GND	SHALL	-	-	-	-	-	-
A30	eDP_A_LANE0_P	MAY	-	-	-	-	-	Not connected
A31	eDP_A_LANE0_N	MAY	-	-	-	-	-	Not connected
A32	GND	SHALL	-	-	-	-	-	-
A33	eDP_A_LANE2_P	MAY	-	-	-	-	-	Not connected
A34	eDP_A_LANE2_N	MAY	-	-	-	-	-	Not connected
B25	USB_D_SSRX_P	SHOULD	VDD_USB_0P8	USB1_RX1_P	B16	-	-	-
B26	USB_D_SSRX_N	SHOULD	VDD_USB_0P8	USB1_RX1_N	A17	-	-	-
B27	GND	SHALL	-	-	-	-	-	-
B28	GND	SHALL	-	-	-	-	-	-
B29	VCC_8_TEST	MAY	-	-	-	-	-	-
B30	GND	SHALL	-	-	-	-	-	-
B31	eDP_A_LANE1_P	MAY	-	-	-	-	-	Not connected
B32	eDP_A_LANE1_N	MAY	-	-	-	-	-	Not connected
B33	GND	SHALL	-	-	-	-	-	-
B34	eDP_A_LANE3_P	MAY	-	-	-	-	-	Not connected
B35	eDP_A_LANE3_N	MAY	-	-	-	-	-	Not connected
C25	GND	SHALL	-	-	-	-	-	-
C26	USB_D_EN	MAY	-	-	-	-	-	Not connected
C27	USB_D_VBUS	MAY	-	-	-	-	-	Not connected

Continued on next page

Table 8: OSM Size-M additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
C28	USB_D_OC#	MAY	-	-	-	-	-	Not connected
C29	SPI_C_SDI	MAY	-	-	-	-	-	Not connected
C30	SPI_C_CS0#	MAY	-	-	-	-	-	Not connected
C31	eDP_A_BL_PWM	MAY	-	-	-	-	-	Not connected
C32	GND	SHALL	-	-	-	-	-	-
C33	eDP_A_AUX_P	MAY	-	-	-	-	-	Not connected
C34	eDP_A_AUX_N	MAY	-	-	-	-	-	Not connected
C35	GND	SHALL	-	-	-	-	-	-
D25	USB_D_D_P	MAY	-	-	-	-	-	Not connected
D26	USB_D_D_N	MAY	-	-	-	-	-	Not connected
D27	USB_D_ID	MAY	-	-	-	-	-	Not connected
D28	GND	SHALL	-	-	-	-	-	-
D29	SPI_C_SCK	MAY	-	-	-	-	-	Not connected
D30	SPI_C_SDO	MAY	-	-	-	-	-	Not connected
D31	eDP_A_BL_EN	MAY	-	-	-	-	-	Not connected
D32	eDP_A_AUX_SEL	MAY	-	-	-	-	-	Not connected
D33	eDP_A_HPD	MAY	-	-	-	-	-	Not connected
D34	GND	SHALL	-	-	-	-	-	-
D35	eDP_B_LANE0_P	MAY	-	-	-	-	-	Not connected
E32	eDP_B_BL_EN	MAY	-	-	-	-	-	Not connected
E33	eDP_B_BL_PWM	MAY	-	-	-	-	-	Not connected
E34	eDP_B_LANE1_P	MAY	-	-	-	-	-	Not connected
E35	eDP_B_LANE0_N	MAY	-	-	-	-	-	Not connected

Continued on next page

Table 8: OSM Size-M additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
F32	eDP_B_AUX_SEL	MAY	-	-	-	-	-	Not connected
F33	GND	SHALL	-	-	-	-	-	-
F34	eDP_B_LANE1_N	MAY	-	-	-	-	-	Not connected
F35	GND	SHALL	-	-	-	-	-	-
G32	eDP_B_HPD	MAY	-	-	-	-	-	Not connected
G33	eDP_B_AUX_P	MAY	-	-	-	-	-	Not connected
G34	GND	SHALL	-	-	-	-	-	-
G35	eDP_B_LANE2_P	MAY	-	-	-	-	-	Not connected
H32	GND	SHALL	-	-	-	-	-	-
H33	eDP_B_AUX_N	MAY	-	-	-	-	-	Not connected
H34	eDP_B_LANE3_P	MAY	-	-	-	-	-	Not connected
H35	eDP_B_LANE2_N	MAY	-	-	-	-	-	Not connected
J32	CSI_B_CLOCK_N	SHOULD	-	MIPI_DSICSI1_CLK_N	C11	10-Channel MIPI Switch	-	Configured on the SoC and by the selection pin of the Switch
J33	GND	SHALL	-	-	-	-	-	-
J34	eDP_B_LANE3_N	MAY	-	-	-	-	-	Not connected
J35	GND	SHALL	-	-	-	-	-	-
K32	GND	SHALL	-	-	-	-	-	-
K33	CSI_B_CLOCK_P	SHOULD	-	MIPI_DSICSI1_CLK_P	B10	10-Channel MIPI Switch	-	Configured on the SoC and by the selection pin of the Switch
K34	GND	SHALL	-	-	-	-	-	-
K35	PCIe_B_HSO0_P	SHOULD	VDD_PCI_1P8	PCIE2_TX0_P	E31	-	-	-
L32	CSI_B_DATA0_N	SHOULD	-	MIPI_DSICSI1_D0_N	C15	10-Channel MIPI Switch	-	Configured on the SoC and by the selection pin of the Switch
L33	PCIe_B_PERST#	SHOULD	-	-	-	GPIO Expander IC16 - P1_1 (11)	Input without PU/PD	-

Continued on next page

Table 8: OSM Size-M additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
L34	PCIE_B_HSI0_P	SHOULD	VDD_PCI_1P8	PCIE2_RX0_P	B34	–	–	–
L35	PCIE_B_HSO0_N	SHOULD	VDD_PCI_1P8	PCIE2_TX0_N	F32	–	–	–
M32	GND	SHALL	–	–	–	–	–	–
M33	CSI_B_DATA0_P	SHOULD	–	MIPI_DSICSI1_D0_P	B14	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
M34	PCIE_B_HSI0_N	SHOULD	VDD_PCI_1P8	PCIE2_RX0_N	C35	–	–	–
M35	GND	SHALL	–	–	–	–	–	–
N32	CSI_B_DATA1_N	SHOULD	–	MIPI_DSICSI1_D1_N	C15	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
N33	ETH_CDE_MDC	MAY	–	–	–	–	–	Not connected
N34	GND	SHALL	–	–	–	–	–	–
N35	ETH_C_(R)(G)MII_TX_CLK	MAY	–	–	–	–	–	Not connected
P32	CSI_B_DATA1_P	SHOULD	–	MIPI_DSICSI1_D1_P	B12	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
P33	ETH_CDE_MDIO	MAY	–	–	–	–	–	Not connected
P34	CSI_B_DATA2_N	SHOULD	–	MIPI_DSICSI1_D2_N	A9	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
P35	ETH_C_(R)(G)MII_RXD3	MAY	–	–	–	–	–	Not connected
R32	GND	SHALL	–	–	–	–	–	–
R33	CSI_B_DATA2_P	SHOULD	–	MIPI_DSICSI1_D2_P	B8	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch
R34	ETH_C_SDP	MAY	–	–	–	–	–	Not connected
R35	ETH_C_(R)(G)MII_RXD2	MAY	–	–	–	–	–	Not connected
T32	CSI_B_DATA3_N	SHOULD	–	MIPI_DSICSI1_D3_N	C7	10-Channel MIPI Switch	–	–
T33	CSI_B_DATA3_P	SHOULD	–	MIPI_DSICSI1_D3_P	B6	10-Channel MIPI Switch	–	Configured on the SoC and by the selection pin of the Switch

Continued on next page

Table 8: OSM Size-M additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
T34	GND	SHALL	-	-	-	-	-	-
T35	ETH_C_(R)(G)MII_RX_DV(_ER)	MAY	-	-	-	-	-	Not connected
U32	GPIO_D_0	SHOULD	-	-	-	GPIO Expander IC6 - P1_0 (10)	Input without PU/PD	-
U33	GPIO_D_1	SHOULD	-	-	-	GPIO Expander IC6 - P1_1 (11)	Input without PU/PD	-
U34	ETH_C_(R)(G)MII_RX_ER	MAY	-	-	-	-	-	Not connected
U35	ETH_C_(S)(R)(G)MII_RXD1	SHOULD	NVCC_ENET	ETH_RX0_N	AK12	-	-	-
V32	GPIO_D_2	SHOULD	-	-	-	GPIO Expander IC6 - P1_2 (12)	Input without PU/PD	-
V33	GPIO_D_3	SHOULD	-	-	-	GPIO Expander IC6 - P1_3 (13)	Input without PU/PD	-
V34	ETH_C_(R)(G)MII_TX_EN(_ER)	MAY	-	-	-	-	-	Not connected
V35	ETH_C_(S)(R)(G)MII_RXD0	SHOULD	NVCC_ENET	ETH_RX0_P	AJ13	-	-	-
W32	GPIO_D_4	SHOULD	-	-	-	GPIO Expander IC6 - P1_4 (14)	Input without PU/PD	-
W33	GPIO_D_5	SHOULD	-	-	-	GPIO Expander IC6 - P1_5 (15)	Input without PU/PD	-
W34	GND	SHALL	-	-	-	-	-	-
W35	ETH_C_(R)(G)MII_RX_CLK	MAY	-	-	-	-	-	Not connected
Y25	VCC_IN_5V	SHALL	-	-	-	-	-	-
Y26	VCC_IN_5V	SHALL	-	-	-	-	-	-
Y27	VCC_IN_5V	SHALL	-	-	-	-	-	-
Y28	VCC_IN_5V	SHALL	-	-	-	-	-	-
Y29	Vendor Defined	MAY	AVDD_1P8_ADC	ADC_IN7	A45	-	-	-
Y30	Vendor Defined	MAY	VDD_PCI_1P8	PCIE1_REF_PAD_CLK_N	C31	-	-	with 10 kΩ pull-down
Y31	Vendor Defined	MAY	VDD_PCI_1P8	PCIE1_REF_PAD_CLK_P	B30	-	-	with 10 kΩ pull-down
Y32	GPIO_D_6	SHOULD	-	-	-	GPIO Expander IC6 - P1_6 (16)	Input without PU/PD	-
Y33	SPI_C_CS1#/ GPIO_D_7	SHOULD	-	-	-	GPIO Expander IC6 - P1_7 (17)	Input without PU/PD	-

Continued on next page

Table 8: OSM Size-M additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
Y34	ETH_C_(S)(R)(G)MII_TXD2	MAY	-	-	-	-	-	Not connected
Y35	ETH_C_(S)(R)(G)MII_TXD0	SHOULD	VDD_ETH_0P8	ETH_TX0_P	AJ17	-	-	-
AA25	GND	SHALL	-	-	-	-	-	-
AA26	GND	SHALL	-	-	-	-	-	-
AA27	GND	SHALL	-	-	-	-	-	-
AA28	GND	SHALL	-	-	-	-	-	-
AA29	Vendor Defined	MAY	AVDD_1P8_ADC	ADC_IN4	A41	-	-	-
AA30	Vendor Defined	MAY	AVDD_1P8_ADC	ADC_IN5	B42	-	-	-
AA31	Vendor Defined	MAY	AVDD_1P8_ADC	ADC_IN6	B44	-	-	-
AA32	GND	SHALL	-	-	-	-	-	-
AA33	VCC_7_TEST	MAY	-	-	-	-	-	-
AA34	ETH_C_(S)(R)(G)MII_TXD3	MAY	-	-	-	-	-	Not connected
AA35	ETH_C_(S)(R)(G)MII_TXD1	SHOULD	VDD_ETH_0P8	ETH_TX0_N	AK16	-	-	-
AB25	CAM_B_SCL/ CSI_B_TX_P	SHOULD	NVCC_GPIO	GPIO_IO09	M46	-	Input with PD	with 2.2 kΩ pull-up to +V1.8
AB26	CAM_B_SDA/ CSI_B_TX_N	SHOULD	NVCC_GPIO	GPIO_IO08	M44	-	Input with PD	with 2.2 kΩ pull-up to +V1.8
AB27	UFS_RESET#	MAY	-	-	-	-	-	Not connected
AB28	GND	SHALL	-	-	-	-	-	-
AB29	UFS_TX1_P	MAY	-	-	-	-	-	Not connected
AB30	UFS_TX1_N	MAY	-	-	-	-	-	Not connected
AB31	GND	SHALL	-	-	-	-	-	-
AB32	UFS_RX1_P	MAY	-	-	-	-	-	Not connected
AB33	UFS_RX1_N	MAY	-	-	-	-	-	Not connected
AB34	GND	SHALL	-	-	-	-	-	-

Continued on next page

Table 8: OSM Size-M additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AB35	ETH_C_(R)(G)MII_COL	MAY	-	-	-	-	-	Not connected
AC26	UFS_CLK	MAY	-	-	-	-	-	Not connected
AC27	GND	SHALL	-	-	-	-	-	-
AC28	UFS_TX0_P	MAY	-	-	-	-	-	Not connected
AC29	UFS_TX0_N	MAY	-	-	-	-	-	Not connected
AC30	GND	SHALL	-	-	-	-	-	-
AC31	UFS_RX0_P	MAY	-	-	-	-	-	Not connected
AC32	UFS_RX0_N	MAY	-	-	-	-	-	Not connected
AC33	GND	SHALL	-	-	-	-	-	-
AC34	ETH_C_(R)(G)MII_CRS	MAY	-	-	-	-	-	Not connected

Table 9: OSM Size-L additional contact assignment

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AE2	GND	SHALL	–	–	–	–	–	–
AE3	ETH_D_(R)(G)MII_CRS	MAY	–	–	–	–	–	Not connected
AE4	VCC_IN_5V	MAY	–	–	–	–	–	–
AE32	RESERVED	MAY	–	–	–	–	–	Not connected
AE33	PCIE_C_PERST#	MAY	–	–	–	–	–	Not connected
AE34	GND	SHALL	–	–	–	–	–	–
AF1	ETH_D_(S)(R)(G)MII_TXD0	MAY	–	–	–	–	–	Not connected
AF2	ETH_D_(S)(R)(G)MII_TXD3	MAY	–	–	–	–	–	Not connected
AF3	ETH_D_(R)(G)MII_COL	MAY	–	–	–	–	–	Not connected
AF4	VCC_IN_5V	MAY	–	–	–	–	–	–
AF32	GPIO_E_0	MAY	–	–	–	–	–	Not connected
AF33	GPIO_E_1	MAY	–	–	–	–	–	Not connected
AF34	PCIE_C_HSO3_P	MAY	–	–	–	–	–	Not connected
AF35	GND	SHALL	–	–	–	–	–	–
AG1	ETH_D_(S)(R)(G)MII_TXD1	MAY	–	–	–	–	–	Not connected
AG2	ETH_D_(S)(R)(G)MII_TXD2	MAY	–	–	–	–	–	Not connected
AG3	GND	SHALL	–	–	–	–	–	–
AG4	VCC_IN_5V	MAY	–	–	–	–	–	–
AG32	GPIO_E_2	MAY	–	–	–	–	–	Not connected
AG33	GPIO_E_3	MAY	–	–	–	–	–	Not connected
AG34	PCIE_C_HSO3_N	MAY	–	–	–	–	–	Not connected
AG35	PCIE_C_HSI3_P	MAY	–	–	–	–	–	Not connected
AH1	ETH_D_(R)(G)MII_TX_CLK	MAY	–	–	–	–	–	Not connected

Continued on next page

Table 9: OSM Size-L additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AH2	GND	SHALL	-	-	-	-	-	-
AH3	VCC_IN_5V	MAY	-	-	-	-	-	-
AH4	VCC_IN_5V	MAY	-	-	-	-	-	-
AH32	GPIO_E_4	MAY	-	-	-	-	-	Not connected
AH33	GPIO_E_5	MAY	-	-	-	-	-	Not connected
AH34	GND	SHALL	-	-	-	-	-	-
AH35	PCIe_C_HSI3_N	MAY	-	-	-	-	-	Not connected
AJ1	ETH_D_(S)(R)(G)MII_RXD0	MAY	-	-	-	-	-	Not connected
AJ2	ETH_D_(R)(G)MII_TX_EN(ER)	MAY	-	-	-	-	-	Not connected
AJ3	VCC_IN_5V	MAY	-	-	-	-	-	-
AJ4	VCC_IN_5V	MAY	-	-	-	-	-	-
AJ32	GPIO_E_6	MAY	-	-	-	-	-	Not connected
AJ33	GPIO_E_7	MAY	-	-	-	-	-	Not connected
AJ34	PCIe_C_HSO2_P	MAY	-	-	-	-	-	Not connected
AJ35	GND	SHALL	-	-	-	-	-	-
AK1	ETH_D_(S)(R)(G)MII_RXD1	MAY	-	-	-	-	-	Not connected
AK2	ETH_D_(R)(G)MII_RX_ER	MAY	-	-	-	-	-	Not connected
AK3	GND	SHALL	-	-	-	-	-	-
AK4	VCC_IN_5V	MAY	-	-	-	-	-	-
AK32	Vendor Defined	MAY	VDD_EARC_1P8	EARC_P_UTIL	Y44	-	-	-
AK33	Vendor Defined	MAY	VDD_EARC_1P8	EARC_N_HPD	Y46	-	-	-
AK34	PCIe_C_HSO2_N	MAY	-	-	-	-	-	Not connected
AK35	PCIe_C_HSI2_P	MAY	-	-	-	-	-	Not connected

Continued on next page

Table 9: OSM Size-L additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AL1	ETH_D_(R)(G)MII_RX_DV_(ER)	MAY	–	–	–	–	–	Not connected
AL2	GND	SHALL	–	–	–	–	–	–
AL3	RESERVED	MAY	–	–	–	–	–	Not connected
AL4	RESERVED	MAY	–	–	–	–	–	Not connected
AL32	Vendor Defined	MAY	VDD_EARC_1P8	EARC_AUX	AA45	–	–	–
AL33	Vendor Defined	MAY	–	–	–	–	–	–
AL34	GND	SHALL	–	–	–	–	–	–
AL35	PCIe_C_HSI2_N	MAY	–	–	–	–	–	Not connected
AM1	ETH_D_(R)(G)MII_RXD2	MAY	–	–	–	–	–	Not connected
AM2	ETH_D_SDP	MAY	–	–	–	–	–	Not connected
AM3	RESERVED	MAY	–	–	–	–	–	Not connected
AM4	RESERVED	MAY	–	–	–	–	–	Not connected
AM5	RESERVED	MAY	–	–	–	–	–	Not connected
AM6	RESERVED	MAY	–	–	–	–	–	Not connected
AM7	RESERVED	MAY	–	–	–	–	–	Not connected
AM8	RESERVED	MAY	–	–	–	–	–	Not connected
AM9	RESERVED	MAY	–	–	–	–	–	Not connected
AM10	RESERVED	MAY	–	–	–	–	–	Not connected
AM11	LVDS_I2C_CLK	SHOULD	NVCC_GPIO	GPIO_IO11	M52	–	Input with PD	with 2.2 kΩ pull-up to +V1.8
AM12	LVDS_I2C_DAT	SHOULD	NVCC_GPIO	GPIO_IO10	M48	–	Input with PD	with 2.2 kΩ pull-up to +V1.8
AM13	GND	SHALL	–	–	–	–	–	–
AM14	LVDS_B_LANE3_N	SHOULD	VDD_LVDS_1P8	LVDS1_D3_N	F4	–	–	–
AM15	LVDS_B_LANE3_P	SHOULD	VDD_LVDS_1P8	LVDS1_D3_P	F2	–	–	–

Continued on next page

Table 9: OSM Size-L additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AM16	GND	SHALL	–	–	–	–	–	–
AM17	LVDS_B_LANE2_N	SHOULD	VDD_LVDS_1P8	LVDS1_D2_N	E3	–	–	–
AM18	LVDS_B_LANE2_P	SHOULD	VDD_LVDS_1P8	LVDS1_D2_P	E1	–	–	–
AM19	GND	SHALL	–	–	–	–	–	–
AM20	LVDS_B_LANE0_N	SHOULD	VDD_LVDS_1P8	LVDS1_D0_N	A3	–	–	–
AM21	LVDS_B_LANE0_P	SHOULD	VDD_LVDS_1P8	LVDS1_D0_P	B2	–	–	–
AM22	GND	SHALL	–	–	–	–	–	–
AM23	RESERVED	MAY	–	–	–	–	–	Not connected
AM24	RESERVED	MAY	–	–	–	–	–	Not connected
AM25	RESERVED	MAY	–	–	–	–	–	Not connected
AM26	RESERVED	MAY	–	–	–	–	–	Not connected
AM27	RESERVED	MAY	–	–	–	–	–	Not connected
AM28	RESERVED	MAY	–	–	–	–	–	Not connected
AM29	RESERVED	MAY	–	–	–	–	–	Not connected
AM30	RESERVED	MAY	–	–	–	–	–	Not connected
AM31	RESERVED	MAY	–	–	–	–	–	Not connected
AM32	Vendor Defined	MAY	NVCC_AON	I2C2_SCL	E43	–	Input with PD	with 2.2 kΩ pull-up to +V1.8 ¹
AM33	Vendor Defined	MAY	NVCC_AON	I2C2_SDA	E45	–	Input with PD	with 2.2 kΩ pull-up to +V1.8 ¹
AM34	PCIe_C_HSO1_P	MAY	–	–	–	–	–	Not connected
AM35	GND	SHALL	–	–	–	–	–	–
AN1	ETH_D_(R)(G)MII_RXD3	MAY	–	–	–	–	–	Not connected
AN2	RESERVED	MAY	–	–	–	–	–	Not connected
AN3	GND	SHALL	–	–	–	–	–	–

Continued on next page

Table 9: OSM Size-L additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AN4	ETH_E_(R)(G)MII_CRS	MAY	–	–	–	–	–	Not connected
AN5	RESERVED	MAY	–	–	–	–	–	Not connected
AN6	GND	SHALL	–	–	–	–	–	–
AN7	RESERVED	MAY	–	–	–	–	–	Not connected
AN8	RESERVED	MAY	–	–	–	–	–	Not connected
AN9	GND	SHALL	–	–	–	–	–	–
AN10	ETH_E_(R)(G)MII_RX_CLK	MAY	–	–	–	–	–	Not connected
AN11	GND	SHALL	–	–	–	–	–	–
AN12	LVDS_A_CLK_N	SHOULD	VDD_LVDS_1P8	LVDS0_CLK_N	A5	–	–	–
AN13	LVDS_A_CLK_P	SHOULD	VDD_LVDS_1P8	LVDS0_CLK_P	B4	–	–	–
AN14	LVDS_VDD_EN	SHOULD	–	–	–	GPIO Expander IC16 - P1_3 (13)	Input without PU/PD	–
AN15	GND	SHALL	–	–	–	–	–	–
AN16	LVDS_B_CLK_N	SHOULD	VDD_LVDS_1P8	LVDS1_CLK_N	D4	–	–	–
AN17	LVDS_B_CLK_P	SHOULD	VDD_LVDS_1P8	LVDS1_CLK_P	D2	–	–	–
AN18	GND	SHALL	–	–	–	–	–	–
AN19	LVDS_B_LANE1_N	SHOULD	VDD_LVDS_1P8	LVDS1_D1_N	C3	–	–	–
AN20	LVDS_B_LANE1_P	SHOULD	VDD_LVDS_1P8	LVDS1_D1_P	C1	–	–	–
AN21	GND	SHALL	–	–	–	–	–	–
AN22	LVDS_BL_PWM	SHOULD	NVCC_GPIO	GPIO_IO24	T48	–	Input with PD	–
AN23	LVDS_BL_EN	SHOULD	–	–	–	GPIO Expander IC16 - P1_2 (12)	Input without PU/PD	–
AN24	RESERVED	MAY	–	–	–	–	–	Not connected
AN25	RESERVED	MAY	–	–	–	–	–	Not connected
AN26	RESERVED	MAY	–	–	–	–	–	Not connected

Continued on next page

Table 9: OSM Size-L additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AN27	RESERVED	MAY	–	–	–	–	–	Not connected
AN28	RESERVED	MAY	–	–	–	–	–	Not connected
AN29	RESERVED	MAY	–	–	–	–	–	Not connected
AN30	RESERVED	MAY	–	–	–	–	–	Not connected
AN31	RESERVED	MAY	–	–	–	–	–	Not connected
AN32	PCIe_D_PERST#	MAY	–	–	–	–	–	Not connected
AN33	GND	SHALL	–	–	–	–	–	–
AN34	PCIe_C_HSO1_N	MAY	–	–	–	–	–	Not connected
AN35	PCIe_C_HSI1_P	MAY	–	–	–	–	–	Not connected
AP1	ETH_D_(R)(G)MII_RX_CLK	MAY	–	–	–	–	–	Not connected
AP2	GND	SHALL	–	–	–	–	–	–
AP3	ETH_E_(R)(G)MII_TXD2	MAY	–	–	–	–	–	Not connected
AP4	ETH_E_(R)(G)MII_TXD3	MAY	–	–	–	–	–	Not connected
AP5	GND	SHALL	–	–	–	–	–	–
AP6	ETH_E_(R)(G)MII_TX_EN(_ER)	MAY	–	–	–	–	–	Not connected
AP7	ETH_E_(R)(G)MII_RX_ER	MAY	–	–	–	–	–	Not connected
AP8	GND	SHALL	–	–	–	–	–	–
AP9	ETH_E_SDP	MAY	–	–	–	–	–	Not connected
AP10	RESERVED	MAY	–	–	–	–	–	Not connected
AP11	LVDS_A_LANE3_N	SHOULD	VDD_LVDS_1P8	LVDS0_D3_N	E9	–	–	–
AP12	LVDS_A_LANE3_P	SHOULD	VDD_LVDS_1P8	LVDS0_D3_P	D8	–	–	–
AP13	GND	SHALL	–	–	–	–	–	–
AP14	LVDS_A_LANE2_N	SHOULD	VDD_LVDS_1P8	LVDS0_D2_N	E7	–	–	–

Continued on next page

Table 9: OSM Size-L additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AP15	LVDS_A_LANE2_P	SHOULD	VDD_LVDS_1P8	LVDS0_D2_P	D6	–	–	–
AP16	GND	SHALL	–	–	–	–	–	–
AP17	LVDS_A_LANE0_N	SHOULD	VDD_LVDS_1P8	LVDS0_D0_N	G9	–	–	–
AP18	LVDS_A_LANE0_P	SHOULD	VDD_LVDS_1P8	LVDS0_D0_P	G7	–	–	–
AP19	GND	SHALL	–	–	–	–	–	–
AP20	PCIE_D_HSO0_N	MAY	–	–	–	–	–	Not connected
AP21	PCIE_D_HSO0_P	MAY	–	–	–	–	–	Not connected
AP22	GND	SHALL	–	–	–	–	–	–
AP23	PCIE_D_HSO1_N	MAY	–	–	–	–	–	Not connected
AP24	PCIE_D_HSO1_P	MAY	–	–	–	–	–	Not connected
AP25	GND	SHALL	–	–	–	–	–	–
AP26	PCIE_D_HSI2_N	MAY	–	–	–	–	–	Not connected
AP27	PCIE_D_HSI2_P	MAY	–	–	–	–	–	Not connected
AP28	GND	SHALL	–	–	–	–	–	–
AP29	PCIE_D_HSI3_N	MAY	–	–	–	–	–	Not connected
AP30	PCIE_D_HSI3_P	MAY	–	–	–	–	–	Not connected
AP31	GND	SHALL	–	–	–	–	–	–
AP32	PCIE_C_HSI0_N	MAY	–	–	–	–	–	Not connected
AP33	PCIE_C_HSI0_P	MAY	–	–	–	–	–	Not connected
AP34	GND	SHALL	–	–	–	–	–	–
AP35	PCIE_C_HSI1_N	MAY	–	–	–	–	–	Not connected
AR2	ETH_E_(R)(G)MII_COL	MAY	–	–	–	–	–	Not connected
AR3	ETH_E_(S)(R)(G)MII_TXD0	MAY	–	–	–	–	–	Not connected

Continued on next page

Table 9: OSM Size-L additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AR4	ETH_E_(S)(R)(G)MII_TXD1	MAY	–	–	–	–	–	Not connected
AR5	ETH_E_(R)(G)MII_TX_CLK	MAY	–	–	–	–	–	Not connected
AR6	ETH_E_(S)(R)(G)MII_RXD0	MAY	–	–	–	–	–	Not connected
AR7	ETH_E_(S)(R)(G)MII_RXD1	MAY	–	–	–	–	–	Not connected
AR8	ETH_E_(R)(G)MII_RX_DV(_ER)	MAY	–	–	–	–	–	Not connected
AR9	ETH_E_(R)(G)MII_RXD2	MAY	–	–	–	–	–	Not connected
AR10	ETH_E_(R)(G)MII_RXD3	MAY	–	–	–	–	–	Not connected
AR14	GND	SHALL	–	–	–	–	–	–
AR15	LVDS_A_LANE1_N	SHOULD	VDD_LVDS_1P8	LVDS0_D1_N	F8	–	–	–
AR16	LVDS_A_LANE1_P	SHOULD	VDD_LVDS_1P8	LVDS0_D1_P	F6	–	–	–
AR17	GND	SHALL	–	–	–	–	–	–
AR18	PCIE_D_HSI0_N	MAY	–	–	–	–	–	Not connected
AR19	PCIE_D_HSI0_P	MAY	–	–	–	–	–	Not connected
AR20	GND	SHALL	–	–	–	–	–	–
AR21	PCIE_D_HSI1_N	MAY	–	–	–	–	–	Not connected
AR22	PCIE_D_HSI1_P	MAY	–	–	–	–	–	Not connected
AR26	GND	SHALL	–	–	–	–	–	–
AR27	PCIE_D_HSO2_N	MAY	–	–	–	–	–	Not connected
AR28	PCIE_D_HSO2_P	MAY	–	–	–	–	–	Not connected
AR29	GND	SHALL	–	–	–	–	–	–
AR30	PCIE_D_HSO3_N	MAY	–	–	–	–	–	Not connected
AR31	PCIE_D_HSO3_P	MAY	–	–	–	–	–	Not connected
AR32	GND	SHALL	–	–	–	–	–	–

Continued on next page

Table 9: OSM Size-L additional contact assignment (Continued)

X1 pin	Module specification signal name	Module specification compatibility group	Power group	SoC function name	SoC ball name	Non-SoC function name	Reset status	Remarks
AR33	PCIe_C_HSO0_N	MAY	–	–	–	–	–	Not connected
AR34	PCIe_C_HSO0_P	MAY	–	–	–	–	–	Not connected

¹ For testing purposes only. This interface is not intended for use in production systems because it is used by the PMIC.

4 I/O Pins

4.1 Function Multiplexing

Some of the I/O pins on the NXP i.MX 95 SoC can be configured for up to seven alternate functions (for more information refer to the [Pinout Designer](#) and the module pin multiplexing documentation for the availability and usage constraints of the additional interfaces). Most of these pins can also be used as GPIOs (general-purpose I/O, sometimes referred to as digital I/O), or other specific functions/interfaces. For example, the i.MX 95 signal connected to the OSM LGA interface on pad **E18** exposes the SoC alternate function **tpm3.CH2** (**ALT1**), which corresponds to the OSM standard function **DISP_BL_PWM/PWM_0**. In addition to this PWM function, the pin can also be configured as:

- **GPIO_I012** (general-purpose input/output – **ALT0**)
- **pdm.BIT_STREAM[2]** (Pulse Density Modulation digital microphone data input channel 2 – **ALT2**)
- **flexio1.FLEXIO[12]** (flexible I/O module channel 12 – **ALT3**)
- **spi8.PCS0** (SPI interface 8 peripheral chip-select 0 signal – **ALT4**)
- **uart8.TX** (UART interface 8 transmit data signal – **ALT5**)
- **i2c8.SDA** (I²C interface 8 serial data line – **ALT6**)
- **sai3.RX_SYNC** (SAI interface 3 receive frame synchronization signal – **ALT7**)

Whenever possible, **it is strongly recommended to use functions that are compatible across all OSM modules**. This ensures maximum compatibility with standard software and other modules in the OSM Family.



Multiplexing Conflicts

Some alternate functions are available on more than one pin. **Care must be taken to avoid assigning the same function to multiple pins simultaneously**, as this can result in system instability or unpredictable behavior.

Table 10 lists all pins that support alternate functions, along with the alternate functions available for each pin (sorted by the standard OSM function). Alternate functions highlighted in bold indicate the primary interfaces selected for optimal OSM family compatibility.

Table 10: Alternate functions

OSM ball	OSM function	SoC ball name	Soc ball ID	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
C2	CAM_MCK	CCM_CLKO1	AH20	ccmsrcgpcmix.CLKO1	netc.TMR_1588_TRIG1	–	–	flexio1.FLEXIO[26]	gpio3.IO[26]	–	–
N16	ETH_A_SDP	CCM_CLKO2	AF20	ccmsrcgpcmix.CLKO2	netc.TMR_1588_PP1	–	–	flexio1.FLEXIO[27]	gpio3.IO[27]	–	–
AC19	CAN_B_TX	CCM_CLKO3	AK20	ccmsrcgpcmix.CLKO3	netc.TMR_1588_TRIG2	can3.TX	–	flexio2.FLEXIO[28]	gpio4.IO[28]	–	–
AB19	CAN_B_RX	CCM_CLKO4	AJ21	ccmsrcgpcmix.CLKO4	netc.TMR_1588_PP2	can3.RX	–	flexio2.FLEXIO[29]	gpio4.IO[29]	–	–
N17	JTAG_TCK(SWCLK)	DAP_TCLK_SWCLK	AG21	dap.TCLK_SWCLK	–	can4.RX	–	flexio1.FLEXIO[30]	gpio3.IO[30]	uart5.CTS_B	–
P17	JTAG_TDI	DAP_TDI	AK24	dap.TDI	mqs2.LEFT	netc.TMR_1588_ALARM1	can2.TX	flexio2.FLEXIO[30]	gpio3.IO[28]	uart5.RX	–
R17	JTAG_TDO(SWO)	DAP_TDO_TRACESWO	AJ23	dap.TDO_TRACESWO	mqs2.RIGHT	netc.TMR_1588_ALARM2	can2.RX	flexio1.FLEXIO[31]	gpio3.IO[31]	uart5.TX	–
N19	JTAG_TMS(SWDIO)	DAP_TMS_SWDIO	AH22	dap.TMS_SWDIO	–	can4.TX	–	flexio2.FLEXIO[31]	gpio3.IO[29]	uart5.RTS_B	–
T16	ETH_MDC	ENET1_MDC	AK40	netc.MDC	uart3.DCB_B	i3c2.SCL	usb1.OTG_ID	flexio2.FLEXIO[0]	gpio4.IO[0]	–	–
T15	ETH_MDIO	ENET1_MDIO	AJ39	netc.MDIO	uart3.RIN_B	i3c2.SDA	usb1.OTG_PWR	flexio2.FLEXIO[1]	gpio4.IO[1]	–	–
K15	ETH_A_(S)(R)(G)MII	ENET1_RD0	AJ35	netc.ETH0_RGMII_RXD[0]	uart3.RX	netc.ETH0_RMII_RXD[0]	–	flexio2.FLEXIO[10]	gpio4.IO[10]	–	–
L15	ETH_A_(S)(R)(G)MII	ENET1_RD1	AK36	netc.ETH0_RGMII_RXD[1]	uart3.CTS_B	netc.ETH0_RMII_RXD[1]	lptmr2.ALT1	flexio2.FLEXIO[11]	gpio4.IO[11]	–	–
N15	ETH_A_(R)(G)MII_R	ENET1_RD2	AJ37	netc.ETH0_RGMII_RXD[2]	–	netc.ETH0_RMII_RX_ER	lptmr2.ALT2	flexio2.FLEXIO[12]	gpio4.IO[12]	–	–
P15	ETH_A_(R)(G)MII_R	ENET1_RD3	AH38	netc.ETH0_RGMII_RXD[3]	–	–	lptmr2.ALT3	flexio2.FLEXIO[13]	gpio4.IO[13]	–	–
M15	ETH_A_(R)(G)MII_R	ENET1_RX_CTL	AH34	netc.ETH0_RGMII_RX_CTL	uart3.DSR_B	netc.ETH0_RMII_CR5_DV	usb2.OTG_PWR	flexio2.FLEXIO[8]	gpio4.IO[8]	–	–
R15	ETH_A_(R)(G)MII_R	ENET1_RXC	AJ33	netc.ETH0_RGMII_RX_CLK	netc.ETH0_RMII_RX_ER	–	–	flexio2.FLEXIO[9]	gpio4.IO[9]	–	–
H15	ETH_A_(S)(R)(G)MII	ENET1_TD0	AG33	netc.ETH0_RGMII_TXD[0]	uart3.TX	netc.ETH0_RMII_TXD[0]	–	flexio2.FLEXIO[5]	gpio4.IO[5]	–	–
G15	ETH_A_(S)(R)(G)MII	ENET1_TD1	AG35	netc.ETH0_RGMII_TXD[1]	uart3.RTS_B	i3c2.PUR	usb1.OTG_OC	flexio2.FLEXIO[4]	gpio4.IO[4]	i3c2.PUR_B	netc.ETH0_RMII_TXD[1]
H16	ETH_A_(S)(R)(G)MII	ENET1_TD2	AF36	netc.ETH0_RGMII_TXD[2]	INPUT=netc.ETH0_RMII_REF50_CLK OUTPUT=ccmsrcgpcmix.ENET_REF_CLK_ROOT	can2.RX	usb2.OTG_OC	flexio2.FLEXIO[3]	gpio4.IO[3]	–	–
G16	ETH_A_(S)(R)(G)MII	ENET1_TD3	AG37	netc.ETH0_RGMII_TXD[3]	–	can2.TX	usb2.OTG_ID	flexio2.FLEXIO[2]	gpio4.IO[2]	–	–
K16	ETH_A_(R)(G)MII_T	ENET1_TX_CTL	AF32	netc.ETH0_RGMII_TX_CTL	uart3.DTR_B	netc.ETH0_RMII_TX_EN	–	flexio2.FLEXIO[6]	gpio4.IO[6]	–	–
J15	ETH_A_(R)(G)MII_T	ENET1_TXC	AG31	netc.ETH0_RGMII_TX_CLK	ccmsrcgpcmix.ENET_CLK_ROOT	–	–	flexio2.FLEXIO[7]	gpio4.IO[7]	–	–
–	GPIO expander interrupt	ENET2_MDC	AK32	netc.MDC	uart4.DCB_B	sai2.RX_SYNC	–	flexio2.FLEXIO[14]	gpio4.IO[14]	–	–

Continued on next page

Table 10: Alternate functions (Continued)

OSM ball	OSM function	SoC ball name	Soc ball ID	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
T2	PCIe_WAKE#	ENET2_MDIO	AJ331	netc.MDIO	uart4.RIN_B	sai2.RX_BCLK	–	flexio2.FLEXIO[15]	gpio4.IO[15]	–	–
J1	ETH_B_(S)(R)(G)MII	ENET2_RD0	AJ27	netc.ETH1_RGMII_RXD[0]	uart4.RX	sai2.TX_DATA[2]	sai4.RX_BCLK	flexio2.FLEXIO[24]	gpio4.IO[24]	netc.ETH1_RMII_RXD[0]	–
K1	ETH_B_(S)(R)(G)MII	ENET2_RD1	AK28	netc.ETH1_RGMII_RXD[1]	spdif1.IN	sai2.TX_DATA[3]	sai4.RX_DATA[0]	flexio2.FLEXIO[25]	gpio4.IO[25]	netc.ETH1_RMII_RXD[1]	–
M1	ETH_B_(R)(G)MII_R	ENET2_RD2	AJ29	netc.ETH1_RGMII_RXD[2]	uart4.CTS_B	sai2.MCLK	mqs2.RIGHT	flexio2.FLEXIO[26]	gpio4.IO[26]	netc.ETH1_RMII_RX_ER	–
N1	ETH_B_(R)(G)MII_R	ENET2_RD3	AH30	netc.ETH1_RGMII_RXD[3]	spdif1.OUT	spdif1.IN	mqs2.LEFT	flexio2.FLEXIO[27]	gpio4.IO[27]	–	–
L1	ETH_B_(R)(G)MII_R	ENET2_RX_CTL	AH26	netc.ETH1_RGMII_RX_CTL	uart4.DSR_B	sai2.TX_DATA[0]	–	flexio2.FLEXIO[22]	gpio4.IO[22]	netc.ETH1_RMII_CRD_DV	–
P1	ETH_B_(R)(G)MII_R	ENET2_RXC	AJ25	netc.ETH1_RGMII_RX_CLK	netc.ETH1_RMII_RX_ER	sai2.TX_DATA[1]	sai4.RX_SYNC	flexio2.FLEXIO[23]	gpio4.IO[23]	–	–
G1	ETH_B_(S)(R)(G)MII	ENET2_TD0	AG25	netc.ETH1_RGMII_TXD[0]	uart4.TX	sai2.RX_DATA[3]	sai4.TX_DATA[0]	flexio2.FLEXIO[19]	gpio4.IO[19]	netc.ETH1_RMII_TXD[0]	–
F1	ETH_B_(S)(R)(G)MII	ENET2_TD1	AG27	netc.ETH1_RGMII_TXD[1]	uart4.RTS_B	sai2.RX_DATA[2]	sai4.TX_BCLK	flexio2.FLEXIO[18]	gpio4.IO[18]	netc.ETH1_RMII_TXD[1]	–
G2	ETH_B_(S)(R)(G)MII	ENET2_TD2	AF28	netc.ETH1_RGMII_TXD[2]	INPUT=netc.ETH1_RMII_REF50_CLK OUTPUT=ccmsrcgpcmix.ENET_REF_CLK_ROOT	sai2.RX_DATA[1]	sai4.TX_SYNC	flexio2.FLEXIO[17]	gpio4.IO[17]	–	–
F2	ETH_B_(S)(R)(G)MII	ENET2_TD3	AG29	netc.ETH1_RGMII_TXD[3]	–	sai2.RX_DATA[0]	–	flexio2.FLEXIO[16]	gpio4.IO[16]	–	–
J2	ETH_B_(R)(G)MII_T	ENET2_TX_CTL	AF24	netc.ETH1_RGMII_TX_CTL	uart4.DTR_B	sai2.TX_SYNC	netc.ETH1_RMII_TX_EN	flexio2.FLEXIO[20]	gpio4.IO[20]	–	–
H1	ETH_B_(R)(G)MII_T	ENET2_TXC	AG23	netc.ETH1_RGMII_TX_CLK	ccmsrcgpcmix.ENET_CLK_ROOT	sai2.TX_BCLK	–	flexio2.FLEXIO[21]	gpio4.IO[21]	–	–
D13	UART_B_TX	GPIO_IO00	J49	gpio2.IO[0]	i2c3.SDA	–	–	spi6.PCS0	uart5.TX	i2c5.SDA	flexio1.FLEXIO[0]
D14	UART_B_RX	GPIO_IO01	J51	gpio2.IO[1]	i2c3.SCL	–	–	spi6.SIN	uart5.RX	i2c5.SCL	flexio1.FLEXIO[1]
D16	UART_B_CTS	GPIO_IO02	K48	gpio2.IO[2]	i2c4.SDA	–	–	spi6.SOUT	uart5.CTS_B	i2c6.SDA	flexio1.FLEXIO[2]
D15	UART_B_RTS	GPIO_IO03	K52	gpio2.IO[3]	i2c4.SCL	–	–	spi6.SCK	uart5.RTS_B	i2c6.SCL	flexio1.FLEXIO[3]
B13	UART_A_TX	GPIO_IO04	K46	gpio2.IO[4]	tpm3.CH0	pdm.CLK	can4.TX	spi7.PCS0	uart6.TX	i2c6.SDA	flexio1.FLEXIO[4]
A14	UART_A_RX	GPIO_IO05	L45	gpio2.IO[5]	tpm4.CH0	pdm.BIT_STREAM[0]	can4.RX	spi7.SIN	uart6.RX	i2c6.SCL	flexio1.FLEXIO[5]
C14	UART_A_CTS	GPIO_IO06	L49	gpio2.IO[6]	tpm5.CH0	pdm.BIT_STREAM[1]	–	spi7.SOUT	uart6.CTS_B	i2c7.SDA	flexio1.FLEXIO[6]
C13	UART_A_RTS	GPIO_IO07	L51	gpio2.IO[7]	spi3.PCS1	–	–	spi7.SCK	uart6.RTS_B	i2c7.SCL	flexio1.FLEXIO[7]
AB26	CAM_B_SDA/ CSI_B_TX_N	GPIO_IO08	M44	gpio2.IO[8]	spi3.PCS0	–	–	tpm6.CH0	uart7.TX	i2c7.SDA	flexio1.FLEXIO[8]

Continued on next page

Table 10: Alternate functions (Continued)

OSM ball	OSM function	SoC ball name	Soc ball ID	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
AB25	CAM_B_SCL/ CSI_B_TX_P	GPIO_IO09	M46	gpio2.IO[9]	spi3.SIN	–	–	tpm3.EXTCLK	uart7.RX	i2c7.SCL	flexio1.FLEXIO[9]
AM12	LVDS_I2C_DAT	GPIO_IO10	M48	gpio2.IO[10]	spi3.SOUT	–	–	tpm4.EXTCLK	uart7.CTS_B	i2c8.SDA	flexio1.FLEXIO[10]
AM11	LVDS_I2C_CLK	GPIO_IO11	M52	gpio2.IO[11]	spi3.SCK	–	–	tpm5.EXTCLK	uart7.RTS_B	i2c8.SCL	flexio1.FLEXIO[11]
E18	DISP_BL_PWM/ PWM_0	GPIO_IO12	N45	gpio2.IO[12]	tpm3.CH2	pdm.BIT_STREAM[2]	flexio1.FLEXIO[12]	spi8.PCS0	uart8.TX	i2c8.SDA	sai3.RX_SYNC
F18	PWM_1	GPIO_IO13	N49	gpio2.IO[13]	tpm4.CH2	pdm.BIT_STREAM[3]	–	spi8.SIN	uart8.RX	i2c8.SCL	flexio1.FLEXIO[13]
B23	UART_C_TX	GPIO_IO14	N51	gpio2.IO[14]	uart3.TX	–	–	spi8.SOUT	uart8.CTS_B	uart4.TX	flexio1.FLEXIO[14]
A22	UART_C_RX	GPIO_IO15	P44	gpio2.IO[15]	uart3.RX	–	–	spi8.SCK	uart8.RTS_B	uart4.RX	flexio1.FLEXIO[15]
W20	I2S_A_BITCLK	GPIO_IO16	P46	gpio2.IO[16]	sai3.TX_BCLK	pdm.BIT_STREAM[2]	–	uart3.CTS_B	spi4.PCS2	uart4.CTS_B	flexio1.FLEXIO[16]
V18	I2S_MCLK	GPIO_IO17	P48	gpio2.IO[17]	sai3.MCLK	–	–	uart3.RTS_B	spi4.PCS1	uart4.RTS_B	flexio1.FLEXIO[17]
G18	PWM_2	GPIO_IO18	P52	gpio2.IO[18]	sai3.RX_BCLK	–	–	spi5.PCS0	spi4.PCS0	tpm5.CH2	flexio1.FLEXIO[18]
H18	PWM_3	GPIO_IO19	R45	gpio2.IO[19]	sai3.RX_SYNC	pdm.BIT_STREAM[3]	flexio1.FLEXIO[19]	spi5.SIN	spi4.SIN	tpm6.CH2	sai3.TX_DATA[0]
V21	I2S_A_DATA_IN	GPIO_IO20	R49	gpio2.IO[20]	sai3.RX_DATA[0]	pdm.BIT_STREAM[0]	–	spi5.SOUT	spi4.SOUT	tpm3.CH1	flexio1.FLEXIO[20]
W21	I2S_A_DATA_OUT	GPIO_IO21	R51	gpio2.IO[21]	sai3.TX_DATA[0]	pdm.CLK	flexio1.FLEXIO[21]	spi5.SCK	spi4.SCK	tpm4.CH1	sai3.RX_BCLK
AA21	I2C_B_SDA	GPIO_IO22	T44	gpio2.IO[22]	usdhc3.CLK	spdif1.IN	can5.TX	tpm5.CH1	tpm6.EXTCLK	i2c5.SDA	flexio1.FLEXIO[22]
AA20	I2C_B_SCL	GPIO_IO23	T46	gpio2.IO[23]	usdhc3.CMD	spdif1.OUT	can5.RX	tpm6.CH1	–	i2c5.SCL	flexio1.FLEXIO[23]
AN22	LVDS_BL_PWM	GPIO_IO24	T48	gpio2.IO[24]	usdhc3.DATA0	–	–	tpm3.CH3	dap.TDO_TRACESWO	spi6.PCS1	flexio1.FLEXIO[24]
AC17	CAN_A_TX	GPIO_IO25	T52	gpio2.IO[25]	usdhc3.DATA1	can2.TX	–	tpm4.CH3	dap.TCLK_SWCLK	spi7.PCS1	flexio1.FLEXIO[25]
W18	I2S_A_LRCLK	GPIO_IO26	U45	gpio2.IO[26]	usdhc3.DATA2	pdm.BIT_STREAM[1]	flexio1.FLEXIO[26]	tpm5.CH3	dap.TDI	spi8.PCS1	sai3.TX_SYNC
AB17	CAN_A_RX	GPIO_IO27	U49	gpio2.IO[27]	usdhc3.DATA3	can2.RX	–	tpm6.CH3	dap.TMS_SWIDIO	spi5.PCS1	flexio1.FLEXIO[27]
–	Internal I ² C	GPIO_IO28	U51	gpio2.IO[28]	i2c3.SDA	can3.TX	–	–	–	–	flexio1.FLEXIO[28]
–	Internal I ² C	GPIO_IO29	V44	gpio2.IO[29]	i2c3.SCL	can3.RX	–	–	–	–	flexio1.FLEXIO[29]
C3	CAM_A_SDA/ CSI_A_TX_N	GPIO_IO30	V46	gpio2.IO[30]	i2c4.SDA	can5.TX	–	–	–	–	flexio1.FLEXIO[30]

Continued on next page

Table 10: Alternate functions (Continued)

OSM ball	OSM function	SoC ball name	Soc ball ID	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
C4	CAM_A_SCL/ CSI_A_TX_P	GPIO_IO31	V48	gpio2.IO[31]	i2c4.SCL	can5.RX	–	–	–	–	flexio1.FLEXIO[31]
W2	PCle_CLKREQ#	GPIO_IO32	V52	gpio5.IO[12]	pcie1.CLKREQ_B	uart6.TX	–	spi4.PCS2	–	–	–
K17	SPI_A_CS1#/ GPIO_A_6	GPIO_IO33	W45	gpio5.IO[13]	–	uart6.RX	–	spi4.PCS1	–	–	–
Y15	SPI_A_CS0#	GPIO_IO34	W49	gpio5.IO[14]	–	uart6.CTS_B	–	spi4.PCS0	–	–	–
U15	SPI_A_SDI_1(I01)	GPIO_IO35	W51	gpio5.IO[15]	pcie2.CLKREQ_B	uart6.RTS_B	–	spi4.SIN	–	–	–
V15	SPI_A_SDO_1(I00)	GPIO_IO36	Y48	gpio5.IO[16]	–	uart7.TX	–	spi4.SOUT	–	–	–
U16	SPI_A_SCK	GPIO_IO37	Y52	gpio5.IO[17]	–	uart7.RX	–	spi4.SCK	–	–	–
AA15	I2C_A_SCL	I2C1_SCL	D48	i2c1.SCL	i3c1.SCL	uart1.DCB_B	tpm2.CH0	–	gpio1.IO[0]	–	–
AA16	I2C_A_SDA	I2C1_SDA	D52	i2c1.SDA	i3c1.SDA	uart1.RIN_B	tpm2.CH1	–	gpio1.IO[1]	–	–
–	Internal (PMIC)	I ² C I2C2_SCL	E43	i2c2.SCL	i3c1.PUR	uart2.DCB_B	tpm2.CH2	sai1.RX_SYNC	gpio1.IO[2]	i3c1.PUR_B	–
–	Internal (PMIC)	I ² C I2C2_SDA	E45	i2c2.SDA	–	uart2.RIN_B	tpm2.CH3	sai1.RX_BCLK	gpio1.IO[3]	–	–
AA23	SPI_B_CS0#	PDM_BIT_STREAM0	G45	pdm.BIT_STREAM[0]	mqs1.RIGHT	spi1.PCS1	tpm1.EXTCLK	lptmr1.ALT2	gpio1.IO[9]	can1.RX	–
G19	GPIO_B_3	PDM_BIT_STREAM1	H46	pdm.BIT_STREAM[1]	m33.NMI	spi2.PCS1	tpm2.EXTCLK	lptmr1.ALT3	gpio1.IO[10]	–	–
F19	GPIO_B_2	PDM_CLK	F46	pdm.CLK	mqs1.LEFT	–	–	lptmr1.ALT1	gpio1.IO[8]	can1.TX	–
Y23	SPI_B_SDO	SAI1_RXD0	H52	sai1.RX_DATA[0]	sai1.MCLK	spi1.SOUT	uart2.DSR_B	mqs1.RIGHT	gpio1.IO[14]	–	–
Y22	SPI_B_SDI	SAI1_TXC	G51	sai1.TX_BCLK	uart2.CTS_B	spi1.SIN	uart1.DSR_B	can1.RX	gpio1.IO[12]	–	–
Y21	SPI_B_SCK	SAI1_TXD0	H48	sai1.TX_DATA[0]	uart2.RTS_B	spi1.SCK	uart1.DTR_B	can1.TX	gpio1.IO[13] ccmsrcgpcmix.BOOT_MODE[3]	–	–
L17	SPI_B_CS1#/ GPIO_A_7	SAI1_TXFS	G49	sai1.TX_SYNC	sai1.TX_DATA[1]	spi1.PCS0	uart2.DTR_B	mqs1.LEFT	gpio1.IO[11] ccmsrcgpcmix.BOOT_MODE[2]	–	–
–	eMMC	SD1_CLK	AG39	usdhc1.CLK	–	–	–	flexio1.FLEXIO[8]	gpio3.IO[8]	–	–
–	eMMC	SD1_CMD	AF40	usdhc1.CMD	–	–	–	flexio1.FLEXIO[9]	gpio3.IO[9]	–	–
–	eMMC	SD1_DATA0	AG45	usdhc1.DATA0	–	–	–	flexio1.FLEXIO[10]	gpio3.IO[10]	–	–

Continued on next page

Table 10: Alternate functions (Continued)

OSM ball	OSM function	SoC ball name	Soc ball ID	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
–	eMMC	SD1_DATA1	AE45	usdhc1.DATA1	–	–	–	flexio1.FLEXIO[11]	gpio3.IO[11]	–	–
–	eMMC	SD1_DATA2	AD46	usdhc1.DATA2	–	–	–	flexio1.FLEXIO[12]	gpio3.IO[12]	–	–
–	eMMC	SD1_DATA3	AG43	usdhc1.DATA3	flexspi.A_SS1_B	–	–	flexio1.FLEXIO[13]	gpio3.IO[13]	–	–
–	eMMC	SD1_DATA4	AF44	usdhc1.DATA4	flexspi.A_DATA[4]	–	–	flexio1.FLEXIO[14]	gpio3.IO[14]	xsplv.DATA[4]	–
–	eMMC	SD1_DATA5	AG47	usdhc1.DATA5	flexspi.A_DATA[5]	usdhc1.RESET_B	–	flexio1.FLEXIO[15]	gpio3.IO[15]	xsplv.DATA[5]	–
–	eMMC	SD1_DATA6	AD44	usdhc1.DATA6	flexspi.A_DATA[6]	usdhc1.CD_B	–	flexio1.FLEXIO[16]	gpio3.IO[16]	xsplv.DATA[6]	–
–	eMMC	SD1_DATA7	AC45	usdhc1.DATA7	flexspi.A_DATA[7]	usdhc1.WP	–	flexio1.FLEXIO[17]	gpio3.IO[17]	xsplv.DATA[7]	–
–	eMMC	SD1_STROBE	AG41	usdhc1.STROBE	flexspi.A_DQS	–	–	flexio1.FLEXIO[18]	gpio3.IO[18]	xsplv.DQS	–
J21	SDIO_A_CD#	SD2_CD_B	AD48	usdhc2.CD_B	netc.TMR_1588_TRIG1	i3c2.SCL	–	flexio1.FLEXIO[0]	gpio3.IO[0]	–	–
F21	SDIO_A_CLK	SD2_CLK	AB48	usdhc2.CLK	netc.TMR_1588_PP1	i3c2.SDA	–	flexio1.FLEXIO[1]	gpio3.IO[1]	–	–
E20	SDIO_A_CMD	SD2_CMD	AB52	usdhc2.CMD	netc.TMR_1588_TRIG2	i3c2.PUR	i3c2.PUR_B	flexio1.FLEXIO[2]	gpio3.IO[2]	–	–
G20	SDIO_A_D0	SD2_DATA0	AC51	usdhc2.DATA0	netc.TMR_1588_PP2	can2.TX	–	flexio1.FLEXIO[3]	gpio3.IO[3]	–	–
G21	SDIO_A_D1	SD2_DATA1	AC49	usdhc2.DATA1	netc.TMR_1588_CLK	can2.RX	–	flexio1.FLEXIO[4]	gpio3.IO[4]	–	–
H20	SDIO_A_D2	SD2_DATA2	AA51	usdhc2.DATA2	netc.TMR_1588_PP3	mq2.RIGHT	–	flexio1.FLEXIO[5]	gpio3.IO[5]	–	–
H21	SDIO_A_D3	SD2_DATA3	AA49	usdhc2.DATA3	lptmr2.ALT1	mq2.LEFT	netc.TMR_1588_ALARM1	flexio1.FLEXIO[6]	gpio3.IO[6]	–	–
D21	SDIO_A_PWR_EN	SD2_RESET_B	AD52	usdhc2.RESET_B	lptmr2.ALT2	–	netc.TMR_1588_GCLK	flexio1.FLEXIO[7]	gpio3.IO[7]	–	–
–	PMIC control	SD2_VSELECT	AE51	usdhc2.VSELECT	usdhc2.WP	lptmr2.ALT3	–	flexio1.FLEXIO[19]	gpio3.IO[19]	–	–
K20	SDIO_B_CLK	SD3_CLK	AG51	usdhc3.CLK	flexspi.A_SCLK	sai5.TX_DATA[1]	sai5.RX_DATA[0]	flexio1.FLEXIO[20]	gpio3.IO[20]	xsplv.CLK	–
K21	SDIO_B_CMD	SD3_CMD	AF48	usdhc3.CMD	flexspi.A_SS0_B	sai5.TX_DATA[2]	sai5.RX_SYNC	flexio1.FLEXIO[21]	gpio3.IO[21]	xsplv.CS	–
L20	SDIO_B_D0	SD3_DATA0	AG49	usdhc3.DATA0	flexspi.A_DATA[0]	sai5.TX_DATA[3]	sai5.RX_BCLK	flexio1.FLEXIO[22]	gpio3.IO[22]	xsplv.DATA[0]	–
L21	SDIO_B_D1	SD3_DATA1	AH52	usdhc3.DATA1	flexspi.A_DATA[1]	sai5.RX_DATA[1]	sai5.TX_DATA[0]	flexio1.FLEXIO[23]	gpio3.IO[23]	xsplv.DATA[1]	–
M21	SDIO_B_D2	SD3_DATA2	AE49	usdhc3.DATA2	flexspi.A_DATA[2]	sai5.RX_DATA[2]	sai5.TX_SYNC	flexio1.FLEXIO[24]	gpio3.IO[24]	xsplv.DATA[2]	–
N20	SDIO_B_D3	SD3_DATA3	AF52	usdhc3.DATA3	flexspi.A_DATA[3]	sai5.RX_DATA[3]	sai5.TX_BCLK	flexio1.FLEXIO[25]	gpio3.IO[25]	xsplv.DATA[3]	–
D22	UART_CON_RX	UART1_RXD	E49	uart1.RX	seco.RX	spi2.SIN	tpm1.CH0	vpu.UART_RX	gpio1.IO[4]	–	–

Continued on next page

Table 10: Alternate functions (Continued)

OSM ball	OSM function	SoC ball name	Soc ball ID	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
D23	UART_CON_TX	UART1_TXD	F52	uart1.TX	seco.TX	sai2.PCS0	tpm1.CH1	vpu.UART_TX	gpio1.IO[5] ccmsrcgpcmix.BOOT_MODE[0]	–	–
C22	UART_D_RX	UART2_RXD	E51	uart2.RX	uart1.CTS_B	sai2.SOUT	tpm1.CH2	sai1.MCLK	gpio1.IO[6]	–	–
C23	UART_D_TX	UART2_TXD	F48	uart2.TX	uart1.RTS_B	sai2.SCK	tpm1.CH3	–	gpio1.IO[7] ccmsrcgpcmix.BOOT_MODE[1]	–	–
–	PMIC control	WDOG_ANY	J45	wdog1.WDOG_ANY	fccu.eout1	–	–	–	gpio1.IO[15]	–	–
D17	GPIO_A_0	XSPI1_DATA0	AJ45	flexspi.A_DATA[0]	sai2.TX_DATA[4]	sai4.TX_BCLK	sai4.RX_DATA[1]	xsplvslv.DATA[0]	gpio5.IO[0]	–	–
E17	GPIO_A_1	XSPI1_DATA1	AH46	flexspi.A_DATA[1]	sai2.TX_DATA[5]	sai4.TX_SYNC	sai4.TX_DATA[1]	xsplvslv.DATA[1]	gpio5.IO[1]	–	–
F17	GPIO_A_2	XSPI1_DATA2	AJ47	flexspi.A_DATA[2]	sai2.TX_DATA[6]	sai4.TX_DATA[0]	–	xsplvslv.DATA[2]	gpio5.IO[2]	–	–
G17	GPIO_A_3	XSPI1_DATA3	AK48	flexspi.A_DATA[3]	sai2.TX_DATA[7]	sai4.RX_DATA[0]	–	xsplvslv.DATA[3]	gpio5.IO[3]	–	–
W19	I2S_B_DATA_OUT	XSPI1_DATA4	AJ49	flexspi.A_DATA[4]	sai5.TX_DATA[0]	sai5.RX_DATA[1]	–	xsplvslv.DATA[4]	gpio5.IO[4]	–	–
T18	I2S_B_LRCLK	XSPI1_DATA5	AK50	flexspi.A_DATA[5]	sai5.TX_SYNC	sai5.RX_DATA[2]	sai2.RX_DATA[6]	xsplvslv.DATA[5]	gpio5.IO[5]	–	–
T19	I2S_B_BITCLK	XSPI1_DATA6	AJ51	flexspi.A_DATA[6]	sai5.TX_BCLK	sai5.RX_DATA[3]	sai2.RX_DATA[7]	xsplvslv.DATA[6]	gpio5.IO[6]	–	–
V19	I2S_B_DATA_IN	XSPI1_DATA7	AH50	flexspi.A_DATA[7]	sai5.RX_DATA[0]	sai5.TX_DATA[1]	–	xsplvslv.DATA[7]	gpio5.IO[7]	–	–
E19	GPIO_B_1	XSPI1_DQS	AK44	flexspi.A_DQS	sai5.RX_SYNC	sai5.TX_DATA[2]	sai2.RX_DATA[6]	xsplvslv.DQS	gpio5.IO[8]	–	–
H17	GPIO_A_4	XSPI1_SCLK	AJ43	flexspi.A_SCLK	sai2.RX_DATA[4]	sai4.RX_SYNC	earc.hpd	xsplvslv.CLK	gpio5.IO[9]	–	–
J17	GPIO_A_5	XSPI1_SS0_B	AJ41	flexspi.A_SS0_B	sai2.RX_DATA[5]	sai4.RX_BCLK	earc.cec	xsplvslv.CS	gpio5.IO[10]	–	–
D19	GPIO_B_0	XSPI1_SS1_B	AH42	flexspi.A_SS1_B	sai5.RX_BCLK	sai5.TX_DATA[3]	sai2.RX_DATA[7]	–	gpio5.IO[11]	–	–

Bold text: Alternate function that provides maximum compatibility between modules of the OSM family.

Rows in : Pin used to determine boot mode at power-on reset. Make sure the pins are not driven during the release of the reset, otherwise a 3-state buffer is needed on the carrier board. After reset, it operates according to the selected alternate function.

4.2 Pin Multiplexing and Pad Control

The alternate function of each pin can be configured independently by the `IOMUX` controller. For pins that support alternate functions, the selected function is configured through a dedicated **pad multiplexing control register** named `SW_MUX_CTL_PAD_<x>`, where `<x>` corresponds to the i.MX 95 pad name. The available multiplexing options are pad-dependent and may differ from one pin to another.

In addition, each pin has an associated **pad control register** that allows configuration of electrical characteristics such as **pull-up and pull-down resistors, drive strength, slew rate, hysteresis, and open-drain operation**. This register is named `SW_PAD_CTL_PAD_<x>`, where `<x>` corresponds to the name of the i.MX 95 pad.



Control Fields Availability

Not all control fields are available for every pad.

Input functions that can be routed to more than one physical pin require an additional **input multiplexer**. This multiplexer is configured using a register named `<x>_SELECT_INPUT`, where `<x>` corresponds to the name of the source function.

For a complete description of the available register fields and configuration options, refer to the [NXP i.MX 95 Reference Manual](#).

4.3 Pin Reset Status

After a reset, i.MX 95 pins can be set in different modes. Most are pulled low, while some are in a high-impedance state or pulled high. Refer to [Table 6](#) to [Table 9](#) for a complete list of the reset states of each pin. Once the bootloader is running, pins and their states can be reconfigured.



Reset States

Pin reset states are only guaranteed during the release of the reset signal. During the power-up sequence, pin states may be undefined until the corresponding I/O bank voltage is enabled on the module.

5 Interfaces Description

5.1 ADC – Analog-to-Digital Converter

The OSM iMX95 system-on-module integrates a **12-bit SAR (Successive Approximation Register) Analog-to-Digital Converter (ADC)**, providing support for analog signal acquisition, sensor interfacing, voltage monitoring, and general-purpose measurement applications.

The SAR ADC supports up to **eight external single-ended analog input channels**, in addition to several internal monitoring channels used for system supervision and diagnostics. The converter supports a maximum conversion rate of up to **1 MS/s** and offers flexible clocking and triggering configurations for both low-latency and periodic sampling applications.

The ADC subsystem supports:

- Multiple conversion modes, including One-Shot and Scan (continuous) operation
- Software-initiated and externally triggered conversions
- Independent data registers for each ADC channel
- Programmable DMA enable for each channel
- Interrupt generation for conversion completion, threshold violations, and error conditions
- Alternate analog watchdog thresholds for input monitoring
- Hardware abort support for active conversions
- Runtime self-test and fault indication mechanisms for analog and supply supervision

All external analog inputs are routed through dedicated analog conditioning circuitry, including a **1.3 MHz RC low-pass filter**, before reaching the SoC ADC inputs. This filtering improves signal integrity and reduces high-frequency noise coupling into the analog measurement path.

The ADC inputs operate in **single-ended mode only** and are referenced to the SoC analog ground and filtered 1.8 V rail. For the carrier board, proper PCB layout, grounding, and analog signal routing practices are recommended to achieve optimal conversion accuracy and noise performance.

Table 11 lists the ADC interface signals exposed on the OSM iMX95 module, including the corresponding SoC ADC channels, alternate functions, and signal descriptions.

Table 11: ADC interfaces

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function	I/O	Description/Remarks
M18	ADC_0	ADC_IN0	A37	No muxing	I	Analog Input 1
N18	ADC_1	ADC_IN1	B38	No muxing	I	Analog Input 2
B22	Vendor Defined	ADC_IN2	C39	No muxing	I	Analog Input 3
C16	Vendor Defined	ADC_IN3	B40	No muxing	I	Analog Input 4
AA29	Vendor Defined	ADC_IN4	A41	No muxing	I	Analog Input 5
AA30	Vendor Defined	ADC_IN5	B42	No muxing	I	Analog Input 6
AA31	Vendor Defined	ADC_IN6	B44	No muxing	I	Analog Input 7
Y29	Vendor Defined	ADC_IN7	A45	No muxing	I	Analog Input 8

5.2 Camera Interfaces

The OSM iMX95 System-on-Module provides **high-performance camera connectivity** based on the capabilities of the NXP i.MX 95 SoC. The platform integrates **two MIPI CSI-2 receiver interfaces**, each

supporting up to four data lanes, enabling connection to a wide range of image sensors.

The CSI-2 interfaces are compatible with **single-, dual-, and quad-lane camera configurations**. Supported image formats include **RGB, YUV, and RAW**, making the subsystem suitable for applications ranging from simple vision tasks to advanced imaging and machine vision systems.

The i.MX 95 architecture shares high-speed PHY resources between display and camera subsystems. In particular, one CSI interface shares a **Combo Rx/Tx D-PHY** with the DSI display interface. As a result, **when the DSI interface is active, only one CSI interface can be used simultaneously**.

This shared functionality is managed through a high-speed analog multiplexer **TMUX646NZECR**, which dynamically routes the D-PHY signals between the **CSI camera interface** and the **DSI display interface**. This allows the system to support either a second camera or a display on the shared PHY, depending on the selected configuration.

This flexible architecture enables the OSM iMX95 to support:

- Dual-camera configurations (when DSI is not used)
- Single-camera + display configurations (using shared D-PHY)

Features

- Scalable data lane support, 1 to 4 Data Lanes, 2
- Up to 1x 4Kp60 fps (if one MIPI CSI is enabled), 2x 4Kp30, 4x 1080p60, or 8x 1080p30 @2.5 Gbps/lane)
- Up to 4x cameras with MIPI virtual channels
- Compatible with MIPI Alliance Standard for Camera Serial Interface 2 (CSI-2) Version 2.1
- MIPI D-PHY specification V1.2
- Unidirectional master operation supported
- Supported primary and secondary image format: YUV420 (8-bit and 10-bit), YUV420 CSPS (8-bit and 10-bit), YUV422 (8-bit and 10-bit), RGB888, RGB444, RGB666, RGB565, RGB555, RAW6, RAW7, RAW8, RAW10, RAW12, RAW14, RAW16, RAW20
- User-defined byte-based data packet supported
- Generic 8-bit long packet data types

5.2.1 MIPI CSI – Camera Serial Interface

Table 12 displays the pinout of the primary instance of the MIPI CSI interface.

Table 12: MIPI CSI interface 1

X1 pad	OSM specification signal name	SoC function name	SoC ball number	I/O	Description/Remarks
B3	CSI_A_CLOCK_N	MIPI_CSI1_CLK_N	F16	I	Negative differential CSI interface clock signal
B4	CSI_A_CLOCK_P	MIPI_CSI1_CLK_P	E15	I	Positive differential CSI interface clock signal
C1	CSI_A_DATA0_N	MIPI_CSI1_D0_N	F20	I	Negative differential CSI interface data signal, lane 0
B1	CSI_A_DATA0_P	MIPI_CSI1_D0_P	E19	I	Positive differential CSI interface data signal, lane 0
A2	CSI_A_DATA1_N	MIPI_CSI1_D1_N	D18	I	Negative differential CSI interface data signal, lane 1
A3	CSI_A_DATA1_P	MIPI_CSI1_D1_P	E17	I	Positive differential CSI interface data signal, lane 1

Continued on next page

Table 12: MIPI CSI interface 1 (Continued)

X1 pad	OSM specification signal name	SoC function name	SoC ball number	I/O	Description/Remarks
A5	CSI_A_DATA2_N	MIPI_CSI1_D2_N	D14	I	Negative differential CSI interface data signal, lane 2
A6	CSI_A_DATA2_P	MIPI_CSI1_D2_P	E13	I	Positive differential CSI interface data signal, lane 2
B6	CSI_A_DATA3_N	MIPI_CSI1_D3_N	F12	I	Negative differential CSI interface data signal, lane 3
B7	CSI_A_DATA3_P	MIPI_CSI1_D3_P	E11	I	Positive differential CSI interface data signal, lane 3

5.2.2 MIPI CSI/DSI – Multiplexed Camera Serial Interface

The **TMUX646NZECR** high-speed multiplexer lets the system **select whether the shared D-PHY lanes** are routed to the MIPI CSI-2 receiver or to the MIPI DSI display interface. By switching this path under system control, the platform can be configured for either a dual-camera setup (when DSI is not used) or a single camera and a display configuration on the shared PHY. The device is optimized for low insertion loss, minimal skew, and high bandwidth, helping preserve signal integrity for multi-gigabit CSI-2/DSI operation while maintaining compliance with MIPI D-PHY electrical characteristics.

Table 13 shows the pin assignments for the shared CSI/DSI D-PHY instance.

Table 13: MIPI CSI DSI interface

X1 pad	OSM specification signal name	SoC function	MIPI switch pin	I/O	Description
J32	CSI_B_CLOCK_N	MIPI_DSICSI1_CLK_N	CLKA_N	I	Negative differential CSI interface clock signal
K33	CSI_B_CLOCK_P	MIPI_DSICSI1_CLK_P	CLKA_P	I	Positive differential CSI interface clock signal
L32	CSI_B_DATA0_N	MIPI_DSICSI1_D0_N	DA1_N	I	Negative differential CSI interface data signal, lane 0
M33	CSI_B_DATA0_P	MIPI_DSICSI1_D0_P	DA1_P	I	Positive differential CSI interface data signal, lane 0
N32	CSI_B_DATA1_N	MIPI_DSICSI1_D1_N	DA2_N	I	Negative differential CSI interface data signal, lane 1
P32	CSI_B_DATA1_P	MIPI_DSICSI1_D1_P	DA2_P	I	Positive differential CSI interface data signal, lane 1
P34	CSI_B_DATA2_N	MIPI_DSICSI1_D2_N	DA3_N	I	Negative differential CSI interface data signal, lane 2
R33	CSI_B_DATA2_P	MIPI_DSICSI1_D2_P	DA3_P	I	Positive differential CSI interface data signal, lane 2
T32	CSI_B_DATA3_N	MIPI_DSICSI1_D3_N	DA4_N	I	Negative differential CSI interface data signal, lane 3
T33	CSI_B_DATA3_P	MIPI_DSICSI1_D3_P	DA4_P	I	Positive differential CSI interface data signal, lane 3

¹ Considering the operation as a CSI interface.

5.3 CAN – Controller Area Network

The OSM iMX95 system-on-module provides **Controller Area Network (CAN)** communication interfaces based on the integrated **FlexCAN** controllers of the NXP i.MX 95 system-on-chip. These interfaces are designed for industrial automation, automotive networking, robotics, and real-time distributed control applications requiring deterministic and reliable communication.

The i.MX 95 SoC supports **up to five CAN interfaces**, including support for both **Classical CAN 2.0B** and **CAN FD (Flexible Data-Rate)** operation. The FlexCAN architecture provides high integration, flexible buffering, advanced timestamping, and efficient DMA-based data handling while maintaining compatibility with legacy CAN systems.

The CAN subsystem is compliant with:

- **CAN 2.0B**

- **ISO 11898-1:2015 CAN FD**

and supports:

- Standard and extended CAN frames
- Payload sizes up to **64 bytes** in CAN FD mode
- Bit Rate Switching (BRS) for increased data throughput
- Error state indication and advanced fault handling mechanisms

5.3.1 FlexCAN Features

The integrated FlexCAN controllers include:

- Flexible message buffer allocation with configurable payload sizes
- Legacy RX FIFO support with up to 6 frames
- Enhanced RX FIFO support with up to 20 frames
- DMA support for efficient high-throughput communication
- Individual interrupt flags for message buffers and FIFOs
- Listen-only mode for passive network monitoring
- Programmable loopback mode for self-test and validation

The number of available message buffers depends on the configured payload size, allowing optimization between:

- message storage capacity
- throughput
- memory usage

5.3.2 Timing and Synchronization

The CAN subsystem includes advanced timing and synchronization capabilities suitable for deterministic and time-sensitive applications.

Features include:

- 32-bit free-running timer
- High-resolution timestamping support
- Optional external time base synchronization
- Interrupt-driven and DMA-assisted operation

These capabilities enable accurate event timestamping, synchronization between distributed systems, and improved real-time network diagnostics.

5.3.3 Physical Layer Requirements

The FlexCAN controllers integrated in the i.MX 95 implement only the CAN protocol controller layer. An external CAN transceiver is required to provide the physical CAN bus interface and proper differential signaling to the external network.

Carrier board designers must ensure:

- Proper CAN transceiver selection

- Bus termination
- EMC/ESD protection
- Appropriate CAN FD signal integrity considerations


Permanent Dominant Bus Condition

The FlexCAN implementation does not support handling of a **Permanent Dominant** bus condition. If a Low-Power or Freeze mode request occurs while the CAN bus remains permanently dominant, the corresponding acknowledgment sequence may never complete correctly, potentially preventing proper transition into the requested state.

Table 14 lists the CAN interface signals exposed on the OSM iMX95 module as default. For CAN interfaces available via alternate functions, refer to the Table 15.

Table 14: CAN interfaces available as default

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description
CAN A							
AB17	CAN_A_RX	GPIO_IO27	U49	ALT2	can2.RX	I	CAN OSM port A receive pad
AC17	CAN_A_TX	GPIO_IO25	T52	ALT2	can2.TX	O	CAN OSM port A transmit pad
CAN B							
AB19	CAN_B_RX	CCM_CLK04	Aj21	ALT2	can3.TX	I	CAN OSM port B receive pad
AC19	CAN_B_TX	CCM_CLK03	AK20	ALT2	can3.RX	O	CAN OSM port B transmit pad

Table 15: CAN interfaces available as alternate functions

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function name	SoC alternate function	I/O	Description
CAN1							
AA23	SPI_B_CS0#	PDM_BIT_STREAM0	G45	can1.RX	ALT6	I	CAN port 1 receive pad
F19	GPIO_B_2	PDM_CLK	F46	can1.TX	ALT6	O	CAN port 1 transmit pad
Y22	SPI_B_SDI	SAI1_TXC	G51	can1.RX	ALT4	I	CAN port 1 receive pad
Y21	SPI_B_SCK	SAI1_TXD0	H48	can1.TX	ALT4	O	CAN port 1 transmit pad
CAN2							
G21	SDIO_A_D1	SD2_DATA1	AC49	can2.RX	ALT2	I	CAN port 2 receive pad
G20	SDIO_A_D0	SD2_DATA0	AC51	can2.TX	ALT2	O	CAN port 2 transmit pad
R17	JTAG_TDO(SWO)	DAP_TDO_TRACESWO	AJ23	can2.RX	ALT3	I	CAN port 2 receive pad
P17	JTAG_TDI	DAP_TDI	AK24	can2.TX	ALT3	O	CAN port 2 transmit pad
H16	ETH_A_(S)(R)(G)MII_TXD2	ENET1_TD2	AF36	can2.RX	ALT2	I	CAN port 2 receive pad
G16	ETH_A_(S)(R)(G)MII_TXD3	ENET1_TD3	AG37	can2.TX	ALT2	O	CAN port 2 transmit pad

Continued on next page

Table 15: CAN interfaces available as alternate functions (Continued)

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function name	SoC alternate function	I/O	Description
CAN3							
¹	I2C_SOM_SCL	GPIO_IO29	V44	can3.RX	ALT2	I	CAN port 3 receive pad
¹	I2C_SOM_SDA	GPIO_IO28	U51	can3.TX	ALT2	O	CAN port 3 transmit pad
CAN4							
N17	JTAG_TCK(SWCLK)	DAP_TCLK_SWCLK	AG21	can4.RX	ALT2	I	CAN port 4 receive pad
N19	JTAG_TMS(SWDIO)	DAP_TMS_SWDIO	AH22	can4.TX	ALT2	O	CAN port 4 transmit pad
A14	UART_A_RX	GPIO_IO05	L45	can4.RX	ALT3	I	CAN port 4 receive pad
B13	UART_A_TX	GPIO_IO04	K46	can4.TX	ALT3	O	CAN port 4 transmit pad
CAN5							
AA21	I2C_6_SDA	GPIO_IO22	T44	can5.TX	ALT3	O	CAN port 5 transmit pad
AA20	I2C_6_SCL	GPIO_IO23	T46	can5.RX	ALT3	I	CAN port 5 receive pad
C3	CAM_A_SDA/ CSI_A_TX_N	GPIO_IO30	V46	can5.TX	ALT2	O	CAN port 5 transmit pad
C4	CAM_A_SCL/ CSI_A_TX_P	GPIO_IO31	V48	can5.RX	ALT2	I	CAN port 5 receive pad

¹ Internal only, not available for general use.

5.4 Digital Audio

The OSM iMX95 SoM provides digital audio connectivity through the integrated **Synchronous Audio Interface (SAI)** peripherals of the NXP i.MX 95 SoC. These peripherals support industry-standard digital audio protocols and enable high-quality audio streaming between the SoC and external devices, such as audio codecs, DSPs, amplifiers, and digital microphones.

The SAI peripherals exposed on the module are primarily intended for operation in **I²S-compatible mode**, providing the standard clock, synchronization, and data signals required by most digital audio applications. In addition to the I²S interfaces, the module also exposes a dedicated **Enhanced Audio Return Channel (eARC)** interface for HDMI audio applications.

The following subsections describe the available SAI peripherals, the external I²S interfaces, and the dedicated eARC interface.

5.4.1 SAI – Serial Audio Interface

The i.MX 95 includes multiple SAI interfaces; however, not all are available for general-purpose digital audio use on the module. Some SAI instances and related pins are multiplexed with other interfaces or reserved for dedicated functions within the platform design.

5.4.2 I²S Interfaces – Inter-IC Sound Interfaces

Table 16 lists the digital audio signals exposed on the OSM iMX95 module.

Table 16: I²S interfaces

X1 pad	OSM signal name	SoC function name	SoC ball number	SoC alternate function name	SoC alternate function	I/O	Description/Remarks
Shared Signal							
V18	I2S_MCLK	GPIO_IO17	P48	sai3.MCLK	ALT1	O	I ² S master clock
I²S A							
W20	I2S_A_BITCLK	GPIO_IO16	P46	sai3.TX_BCLK	ALT1	O	Bit clock
V21	I2S_A_DATA_IN	GPIO_IO20	R49	sai3.RX_DATA[0]	ALT1	I	Data in line
W21	I2S_A_DATA_OUT	GPIO_IO21	R51	sai3.TX_DATA[0]	ALT1	O	Data out line
W18	I2S_A_LRCLK	GPIO_IO26	U45	sai3.TX_SYNC	ALT7	I	Left/right clock word select
I²S B							
T19	I2S_B_BITCLK	XSPI1_DATA6	AJ51	sai5.TX_BCLK	ALT1	O	Bit clock
V19	I2S_B_DATA_IN	XSPI1_DATA7	AH50	sai5.RX_DATA[0]	ALT1	I	Data in line
W19	I2S_B_DATA_OUT	XSPI1_DATA4	AJ49	sai5.TX_DATA[0]	ALT1	O	Data out line
T18	I2S_B_LRCLK	XSPI1_DATA5	AK50	sai5.TX_SYNC	ALT1	O	Left/right clock word select
SAI 1 - Available as Alternate Functions							
L17	SPI_B_CS1#/ GPIO_A_7	SAI1_TXFS	G49	sai1.TX_SYNC	ALT0	O	Transmit frame synchronization
L17	SPI_B_CS1#/ GPIO_A_7	SAI1_TXFS	G49	sai1.TX_DATA[1]	ALT1	O	Transmit data line 1
Y22	SPI_B_SDI	SAI1_TXC	G51	sai1.TX_BCLK	ALT0	O	Transmit bit clock
Y21	SPI_B_SCK	SAI1_TXD0	H48	sai1.TX_DATA[0]	ALT0	O	Transmit data line 0
Y23	SPI_B_SDO	SAI1_RXD0	H52	sai1.RX_DATA[0]	ALT0	I	Receive data line 0
Y23	SPI_B_SDO	SAI1_RXD0	H52	sai1.MCLK	ALT1	O	Master clock
C22	UART_D_RX	UART2_RXD	E51	sai1.MCLK	ALT4	O	Master clock

Continued on next page

Table 16: I²S interfaces (Continued)

X1 pad	OSM signal name	SoC function name	SoC ball number	SoC alternate function name	SoC alternate function	I/O	Description/Remarks
SAI 2 - Available as Alternate Functions							
T2	PCIe_WAKE#	ENET2_MDIO	AJ31	sai2.RX_BCLK	ALT2	I	Receive bit clock
F2	ETH_B_TXD3	ENET2_TD3	AG29	sai2.RX_DATA[0]	ALT2	I	Receive data line 0
G2	ETH_B_TXD2	ENET2_TD2	AF28	sai2.RX_DATA[1]	ALT2	I	Receive data line 1
F1	ETH_B_TXD1	ENET2_TD1	AG27	sai2.RX_DATA[2]	ALT2	I	Receive data line 2
G1	ETH_B_TXD0	ENET2_TD0	AG25	sai2.RX_DATA[3]	ALT2	I	Receive data line 3
J2	ETH_B_CTL	ENET2_TX_CTL	AF24	sai2.TX_SYNC	ALT2	O	Transmit frame synchronization
H1	ETH_B_TXC	ENET2_TXC	AG23	sai2.TX_BCLK	ALT2	O	Transmit bit clock
L1	ETH_B_RX_CTL	ENET2_RX_CTL	AH26	sai2.TX_DATA[0]	ALT2	O	Transmit data line 0
P1	ETH_B_RXC	ENET2_RXC	AJ25	sai2.TX_DATA[1]	ALT2	O	Transmit data line 1
J1	ETH_B_RXD0	ENET2_RD0	AJ27	sai2.TX_DATA[2]	ALT2	O	Transmit data line 2
K1	ETH_B_RXD1	ENET2_RD1	AK28	sai2.TX_DATA[3]	ALT2	O	Transmit data line 3
M1	ETH_B_RXD2	ENET2_RD2	AJ29	sai2.MCLK	ALT2	O	Master clock
D17	GPIO_A_0	XSPI1_DATA0	AJ45	sai2.TX_DATA[4]	ALT1	O	Transmit data line 4
E17	GPIO_A_1	XSPI1_DATA1	AH46	sai2.TX_DATA[5]	ALT1	O	Transmit data line 5
F17	GPIO_A_2	XSPI1_DATA2	AJ47	sai2.TX_DATA[6]	ALT1	O	Transmit data line 6
G17	GPIO_A_3	XSPI1_DATA3	AK48	sai2.TX_DATA[7]	ALT1	O	Transmit data line 7
H17	GPIO_A_4	XSPI1_SCLK	AJ43	sai2.RX_DATA[4]	ALT1	I	Receive data line 4
J17	GPIO_A_5	XSPI1_SS0_B	AJ41	sai2.RX_DATA[5]	ALT1	I	Receive data line 5
T18	I2S_B_LRCLK	XSPI1_DATA5	AK50	sai2.RX_DATA[6]	ALT3	I	Receive data line 6
E19	GPIO_B_1	XSPI1_DQS	AK44	sai2.RX_DATA[6]	ALT3	I	Alternative receive data line 6
T19	I2S_B_BITCLK	XSPI1_DATA6	AJ51	sai2.RX_DATA[7]	ALT3	I	Receive data line 7
D19	GPIO_B_0	XSPI1_SS1_B	AH42	sai2.RX_DATA[7]	ALT3	I	Alternative receive data line 7

Continued on next page

Table 16: I²S interfaces (Continued)

X1 pad	OSM signal name	SoC function name	SoC ball number	SoC alternate function name	SoC alternate function	I/O	Description/Remarks
SAI 4 - Available as Alternate Functions							
G2	ETH_B_TXD2	ENET2_TD2	AF28	sai4.TX_SYNC	ALT3	O	Transmit frame synchronization
F1	ETH_B_TXD1	ENET2_TD1	AG27	sai4.TX_BCLK	ALT3	O	Transmit bit clock
G1	ETH_B_TXD0	ENET2_TD0	AG25	sai4.TX_DATA[0]	ALT3	O	Transmit data line 0
P1	ETH_B_RX_CLK	ENET2_RXC	AJ25	sai4.RX_SYNC	ALT3	I	Receive frame synchronization
J1	ETH_B_RXD0	ENET2_RD0	AJ27	sai4.RX_BCLK	ALT3	I	Receive bit clock
K1	ETH_B_RXD1	ENET2_RD1	AK28	sai4.RX_DATA[0]	ALT3	I	Receive data line 0
D17	GPIO_A_0	XSPI1_DATA0	AJ45	sai4.TX_BCLK	ALT2	O	Alternative transmit bit clock
D17	GPIO_A_0	XSPI1_DATA0	AJ45	sai4.RX_DATA[1]	ALT3	I	Receive data line 1
E17	GPIO_A_1	XSPI1_DATA1	AH46	sai4.TX_SYNC	ALT2	O	Alternative transmit synchronization
E17	GPIO_A_1	XSPI1_DATA1	AH46	sai4.TX_DATA[1]	ALT3	O	Transmit data line 1
F17	GPIO_A_2	XSPI1_DATA2	AJ47	sai4.TX_DATA[0]	ALT2	O	Alternative transmit data line 0
G17	GPIO_A_3	XSPI1_DATA3	AK48	sai4.RX_DATA[0]	ALT2	I	Alternative receive data line 0
H17	GPIO_A_4	XSPI1_SCLK	AJ43	sai4.RX_SYNC	ALT2	I	Alternative receive synchronization
J17	GPIO_A_5	XSPI1_SS0_B	AJ41	sai4.RX_BCLK	ALT2	I	Alternative receive bit clock



Clocking

The dedicated `I2S_MCLK` signal can be used as the reference clock for external audio devices, enabling synchronized audio operation and improved *jitter* performance. Audio clock generation and routing are handled internally by the SoC clock subsystem.

5.4.3 eARC – enhanced Audio Return Channel

The eARC interface on the OSM iMX95 is exposed via three SoC pins: `EARC_AUX`, `EARC_N`, and `EARC_P`.

- `EARC_P` and `EARC_N`: form the differential eARC data pair
- `EARC_AUX`: handles auxiliary eARC control and status communication

To use the eARC channel, the respective pads (controlled by registers `SW_MUX_CTL_PAD_XSPI1_SCLK` and `SW_MUX_CTL_PAD_XSPI1_SS0_B`) must be assigned to the eARC functions instead of their default XSPI-related functions. This allows the physical pins to carry the eARC signals rather than the alternate peripheral signals. For detailed registers information, refer to the [i.MX 95 Reference Manual](#).

A complete eARC implementation also requires the software stack to enable the eARC controller, configure the associated clocks, and connect the audio path through the audio/ASoC framework. The eARC interface is intended to operate together with an HDMI transmitter, receiver, or bridge device that supports eARC; it is not normally used as a standalone generic audio interface like I²S or SAI.

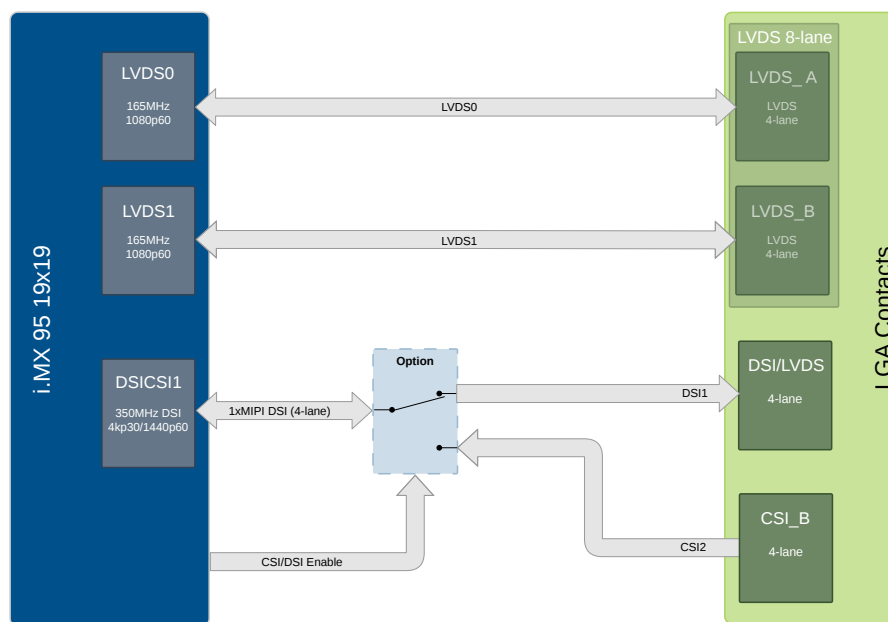
Table 17: eARC interface

X1 pad	OSM signal name	SoC function name	SoC ball number	I/O	Description/Remarks
AL32	Vendor Defined	EARC_AUX	AA45	I/O	eARC auxiliary signals
AK33	Vendor Defined	EARC_N_HPD	Y46	O	eARC data channel negative differential pair
AK32	Vendor Defined	EARC_P_UTIL	Y44	O	eARC data channel positive differential pair

5.5 Displays Interfaces

The OSM iMX95 features a display subsystem with **two display engines** capable of driving up to three simultaneous displays through one MIPI-DSI interface and two LVDS interfaces. The LVDS interfaces can operate either as two independent 4-lane LVDS outputs or combined together as a single dual-link 8-lane LVDS interface for higher-bandwidth display applications. Figure 3 illustrates the display architecture of the OSM iMX95, enabling high-resolution display configurations for industrial, multimedia, and embedded applications.

Figure 3: Displays block diagram



Features

LVDS Interface:

- Two 1080p60 LVDS transmitters, configurable as either two 4-lane channels or one 8-lane LVDS channel
- Native LVDS video output up to 24-bit color depth

MIPI-DSI Interface:

- Single 4-lane interface operating at up to 350 MHz per lane (2.5 Gbps/lane)
- Supports up to 4Kp30 or 3840x1440p60 resolution

5.5.1 LVDS Interface – Low-Voltage Differential Signaling Interface

The i.MX 95 display subsystem supports two independent LVDS transmitter interfaces, each operating as a standard 4-lane LVDS output. These interfaces can be used either independently to drive two separate displays or combined together to form a single high-bandwidth 8-lane LVDS interface. This mode is typically used for high-resolution display panels requiring bandwidth beyond the capability of a single 4-lane LVDS interface.

In dual-channel operation, the two LVDS transmitters are synchronized and operate as a single logical display interface, allowing support for higher display resolutions and increased pixel bandwidth. This configuration is commonly referred to as **Dual-Link LVDS** or **8-lane LVDS** operation.

The supported LVDS operating modes are therefore:

- **2× independent 4-lane LVDS outputs**
- **1× combined 8-lane LVDS output**



Dual-link Mode

When configured in dual-link mode, both LVDS channels share the same timing and video stream and must operate with identical display parameters.

Table 18: LVDS interface signals

X1 pad	OSM standard function	SoC ball name	SoC ball ID	I/O	Description
LVDS_A					
AN12	LVDS_A_CLK_N	LVDS0_CLK_N	A5	O	Negative differential LVDS A Interface clock
AN13	LVDS_A_CLK_P	LVDS0_CLK_P	B4	O	Positive differential LVDS A Interface clock
AP17	LVDS_A_LANE0_N	LVDS0_D0_N	G9	O	Negative differential LVDS A Interface data lane 0
AP18	LVDS_A_LANE0_P	LVDS0_D0_P	G7	O	Positive differential LVDS A Interface data lane 0
AR15	LVDS_A_LANE1_N	LVDS0_D1_N	F8	O	Negative differential LVDS A Interface data lane 1
AR16	LVDS_A_LANE1_P	LVDS0_D1_P	F6	O	Positive differential LVDS A Interface data lane 1
AP14	LVDS_A_LANE2_N	LVDS0_D2_N	E7	O	Negative differential LVDS A Interface data lane 2
AP15	LVDS_A_LANE2_P	LVDS0_D2_P	D6	O	Positive differential LVDS A Interface data lane 2
AP11	LVDS_A_LANE3_N	LVDS0_D3_N	E9	O	Negative differential LVDS A Interface data lane 3
AP12	LVDS_A_LANE3_P	LVDS0_D3_P	D8	O	Positive differential LVDS A Interface data lane 3
LVDS_B					
AN16	LVDS_B_CLK_N	LVDS1_CLK_N	D4	O	Negative differential LVDS B Interface clock
AN17	LVDS_B_CLK_P	LVDS1_CLK_P	D2	O	Positive differential LVDS B Interface clock
AM20	LVDS_B_LANE0_N	LVDS1_D0_N	A3	O	Negative differential LVDS B Interface data lane 0
AM21	LVDS_B_LANE0_P	LVDS1_D0_P	B2	O	Positive differential LVDS B Interface data lane 0
AN19	LVDS_B_LANE1_N	LVDS1_D1_N	C3	O	Negative differential LVDS B Interface data lane 1
AN20	LVDS_B_LANE1_P	LVDS1_D1_P	C1	O	Positive differential LVDS B Interface data lane 1
AM17	LVDS_B_LANE2_N	LVDS1_D2_N	E3	O	Negative differential LVDS B Interface data lane 2
AM18	LVDS_B_LANE2_P	LVDS1_D2_P	E1	O	Positive differential LVDS B Interface data lane 2
AM14	LVDS_B_LANE3_N	LVDS1_D3_N	F4	O	Negative differential LVDS B Interface data lane 3
AM15	LVDS_B_LANE3_P	LVDS1_D3_P	F2	O	Positive differential LVDS B Interface data lane 3

Alternate functions: the SoC has no alternate functions for this interface.

5.5.2 MIPI-DSI – Display Serial Interface

The OSM iMX95 system-on-module provides a high-performance **MIPI Display Serial Interface (MIPI-DSI)** based on the display subsystem integrated in the NXP i.MX 95 system-on-chip. The interface is designed for high-resolution, low-power display connectivity and is compliant with the relevant MIPI Alliance specifications for both the DSI controller and D-PHY physical layer.

The integrated DSI controller supports:

- Up to **4 data lanes**
- MIPI D-PHY v1.2
- High-speed operation up to **2.5 Gbps per lane**
- Pixel clock operation up to **350 MHz**

The interface supports high-resolution display configurations including:

- **4K @ 30 fps**
- **3840 × 1440 @ 60 fps**

The MIPI-DSI subsystem is compliant with:

- **MIPI Alliance Display Serial Interface (DSI) Specification v1.2**
- **MIPI D-PHY Specification v1.2**
- **MIPI CSI-2 Specification v2.1** (shared Combo PHY operation)

Shared DSI / CSI-2 Combo PHY Architecture

The i.MX 95 implements the MIPI-DSI interface using a shared high-speed **MIPI Combo D-PHY**, which is also used by the second MIPI CSI-2 camera interface. This architecture allows the high-speed differential lanes to be dynamically configured either as:

- A MIPI-DSI display output
- A second MIPI CSI-2 camera input



Shared MIPI-CSI/DSI Interface

The shared DSI interface and the second CSI-2 interface are **mutually exclusive** and cannot operate simultaneously.

The interface mode is selected through device-tree configuration in the operating system, setting the `Tx_rxz` bit of the `COMBO_PHY_MODE_CONTROL` register. This register controls Rx/Tx DPHY lane enables, PLL clock output, and CSI/DSI mode. Moreover, the GPIO expander A, `P1_6` pin (`CTRL_DSI_EN/CSI_EN#`), with a 10 kΩ pull-up to 1.8 V, is used to control the switch to control which OSM pads are connected to the SoC. For DSI, `CTRL_DSI_EN/CSI_EN#` must be set to `1` (level high), while for CSI the same signal must be set to `0` (level low).

This shared architecture provides flexibility for application-specific system configurations, allowing the platform to be optimized either for:

- Enhanced display capability using MIPI-DSI¹
- Dual-camera systems using two CSI-2 interfaces

Table 19 lists the MIPI-DSI interface signals exposed on the OSM iMX95 module, including lane assign-

¹Although the OSM standard naming includes LVDS, only DSI is possible. The LVDS interface is only available on the dedicated interface described in [Section 5.5.1](#).

ments, alternate functions, and signal descriptions.

Table 19: MIPI-DSI interface

X1 pad	OSM standard function	SoC ball name	SoC ball	Switch pin	I/O	Description
Clock Signals						
AB8	DSI_CLOCK_N/LVDS_A_CLK_N	MIPI_DSICSI1_CLK_N	C11	CLKB_N	O	Negative differential DSI Interface clock
AB7	DSI_CLOCK_P/LVDS_A_CLK_P	MIPI_DSICSI1_CLK_P	B10	CLKB_P	O	Positive differential DSI Interface clock
Data Signals						
AB11	DSI_DATA0_N/LVDS_LANE0_N	MIPI_DSICSI1_D0_N	C15	DB1_N	O	Negative differential DSI Interface data lane 0
AB10	DSI_DATA0_P/LVDS_LANE0_P	MIPI_DSICSI1_D0_P	B14	DB1_P	O	Positive differential DSI Interface data lane 0
AC9	DSI_DATA1_N/LVDS_LANE1_N	MIPI_DSICSI1_D1_N	A13	DB2_N	O	Negative differential DSI Interface data lane 1
AC8	DSI_DATA1_P/LVDS_LANE1_P	MIPI_DSICSI1_D1_P	B12	DB2_P	O	Positive differential DSI Interface data lane 1
AC6	DSI_DATA2_N/LVDS_LANE2_N	MIPI_DSICSI1_D2_N	A9	DB3_N	O	Negative differential DSI Interface data lane 2
AC5	DSI_DATA2_P/LVDS_LANE2_P	MIPI_DSICSI1_D2_P	B8	DB3_P	O	Positive differential DSI Interface data lane 2
AB5	DSI_DATA3_N/LVDS_LANE3_N	MIPI_DSICSI1_D3_N	C7	DB4_N	O	Negative differential DSI Interface data lane 3
AB4	DSI_DATA3_P/LVDS_LANE3_P	MIPI_DSICSI1_D3_P	B6	DB4_P	O	Positive differential DSI Interface data lane 3

5.6 eMMC – embedded MultiMediaCard Storage Device

The OSM iMX95 system-on-module integrates onboard non-volatile **embedded MultiMediaCard (eMMC)** storage, providing high-performance embedded mass storage for the operating system, bootloader, application software, and user data.

The eMMC device is connected directly to the NXP i.MX 95 SoC through the dedicated **USDHC / SD1 interface**, using an **8-bit parallel data bus** to maximize storage throughput and system performance. The interface includes the standard eMMC signals:

- **eMMC_CLK**: Clock signal
- **eMMC_CMD**: Command line
- **eMMC_DATA0** to **eMMC_DATA7**: 8-bit data bus
- **eMMC_STROBE**: Data strobe signal for HS400 operation

The storage subsystem supports high-speed eMMC operating modes, including:

- Legacy MMC operation
- High-Speed mode
- HS200
- HS400 with data strobe support

The dedicated **eMMC_STROBE** signal enables **HS400 timing operation**, allowing significantly higher transfer performance compared to standard SD/MMC modes.

The eMMC interface operates from the SoC NVCC wakeup power domain and is powered at **1.8 V**, sup-

porting high-speed signaling levels and reduced power consumption.



Onboard eMMC advantages over removable storage solutions

- Improved reliability and vibration resistance
- Faster boot and application loading times
- Lower latency and higher sustained throughput
- Industrial-grade embedded storage solution
- Reduced connector and cabling complexity

The eMMC device is an industrial-temperature component suitable for embedded and industrial applications requiring robust long-term storage reliability and high endurance.

Refer to the [Configure to Options](#) for information about the supported storage capacities.

5.7 Ethernet Interfaces

The OSM iMX95 system-on-module provides **multiple Ethernet interfaces** based on the integrated **NETC (Network Controller)** architecture of the NXP i.MX 95 system-on-chip. The Ethernet subsystem is designed for industrial networking, real-time communication, edge computing, and high-throughput embedded connectivity applications.

The module exposes two independent Gigabit Ethernet interfaces:

- `ETH_A`
- `ETH_B`

Additionally, a 10G interface is also exposed:

- `ETH_C`

Both interfaces are implemented as independent **ENETC Ethernet MAC controllers** and expose digital Ethernet interfaces on the module connectors. Since the OSM iMX95 does not integrate onboard Ethernet PHY devices, external PHYs are required on the carrier board to implement complete Gigabit Ethernet (10/100/1000BASE-T) ports and perform the conversion between the digital R(G)MII interfaces and the analog signaling required for Ethernet cabling and RJ45 connectors with integrated magnetics.

5.7.1 ETH_A Interface

The `ETH_A` interface is the primary Ethernet controller of the platform and supports:

- Gigabit Ethernet (10/100/1000BASE-T)
- RGMII/RMII interface operation
- IEEE 1588 Precision Time Protocol (PTP)
- Time-Sensitive Networking (TSN)

The integrated TSN capabilities enable:

- Deterministic Ethernet communication
- Low-latency packet transport
- Precise network synchronization
- Time-aware traffic scheduling

5.7.2 ETH_B Interface

The `ETH_B` interface provides a second independent Gigabit Ethernet channel through an RGMII interface. This interface supports standard Gigabit Ethernet functionality and enables flexible networking topologies and system expansion.

Supported features include:

- Gigabit Ethernet (10/100/1000BASE-T)
- RGMII interface operation
- IEEE 1588 support
- Energy Efficient Ethernet (EEE)
- Audio Video Bridging (AVB)

Unlike `ETH_A`, the `ETH_B` interface **does not include TSN** functionality.

5.7.3 Shared MDIO Management Interface

Both Ethernet controllers share a common **MDIO management bus** (`MDC` / `MDIO`) used for:

- PHY configuration
- Link monitoring
- PHY register access
- Auto-negotiation management

When multiple PHY devices are connected to the shared MDIO bus, each PHY must be configured with a unique MDIO address to ensure proper operation and software access.

5.7.4 NETC Ethernet Features

The NETC subsystem integrated in the i.MX 95 provides advanced networking capabilities including:

- Hardware packet processing
- DMA-based Ethernet transfers
- Hardware timestamping
- Precision time synchronization
- Traffic prioritization and scheduling
- Energy Efficient Ethernet (EEE)
- Audio Video Bridging (AVB)
- IEEE 1588 PTP support

These features enable efficient high-performance networking with reduced CPU utilization and improved determinism.

5.7.5 ETH_C USXGMII Interface

In addition to the RGMII interfaces, the i.MX 95 also provides an Ethernet-capable high-speed serial interface (`ETH_C`) based on **USXGMII (Universal Serial 10 Gigabit Media Independent Interface)**.

The `ETH_C` interface:

- Provides a high-speed serial Ethernet connection between the SoC and an external Ethernet PHY or switch

- Supports **SGMII** operation at **1 Gbit/s** and **2.5 Gbit/s**
- Supports **10-Gigabit Ethernet** operation through the **XFI/XGMII** interface
- Requires an external PHY, retimer, or Ethernet switch, depending on the selected interface mode and system architecture
- Enables flexible networking designs, including multi-port Ethernet systems, managed switches, and high-bandwidth uplinks
- Supports implementation of Gigabit, 2.5-Gigabit, and 10-Gigabit Ethernet interfaces, subject to the capabilities of the external device and software configuration

This interface is intended for applications requiring: - External Ethernet switches - High-density networking - Backplane connectivity - Advanced industrial communication topologies

5.7.6 Design Considerations

The Ethernet interfaces require:

- External Ethernet PHY devices, fiber-optic transceivers, or other physical-layer (PHY) devices, depending on the selected Ethernet interface and application.
- Proper **RGMII**, **SGMII**, **USXGMII**, **XFI**, or other applicable interface routing between the SoC and the external physical-layer device.
- Controlled-impedance routing for all high-speed differential and single-ended signals, as applicable.
- Magnetics and RJ45 connectors for copper Ethernet implementations, or the appropriate optical or alternative physical-layer components for fiber or other media.
- Proper grounding, power distribution, signal integrity, and EMI/EMC design practices to ensure reliable Ethernet operation.

For reliable Gigabit Ethernet operation, carrier-board layouts should follow:

- Differential impedance control recommendations
- RGMII/USXGMII timing and skew requirements
- PHY vendor layout guidelines
- Ethernet EMI/ESD protection practices

Tables 20 to 22 list the Ethernet interface signals exposed on the OSM iMX95 module, including the RGMII, USXGMII, MDIO, clock, and synchronization signals.

Table 20: Ethernet A interface

X1 pad	OSM aignal name	SoC ball name	SoC ball number	I/O	Description/Remarks
Control Signals					
T15	ETH_MDIO	ENET1_MDIO	AJ39	I/O	Management data input/output ¹
T16	ETH_MDC	ENET1_MDC	AK40	O	Management data clock
R15	ETH_A_(R)(G)MII_RX_CLK	ENET1_RXC	AJ33	I	Receive clock provided by the PHY in RGMII mode
M15	ETH_A_(R)(G)MII_RX_CTL	ENET1_RX_CTL	AH34	I	Receive control ²
J15	ETH_A_(R)(G)MII_TX_CLK	ENET1_TXC	AG31	O	Transmit clock driven by the MAC in RGMII mode
K16	ETH_A_(R)(G)MII_TX_CTL	ENET1_TX_CTL	AF32	O	Transmit control ³

Continued on next page

Table 20: Ethernet A interface (Continued)

X1 pad	OSM signal name	SoC ball name	SoC ball number	I/O	Description/Remarks
Data signals					
K15	ETH_A_(S)(R)(G)MII_RXD0	ENET1_RX0	AJ35	I	Receive Data [3:0] Four-bit data bus carrying received Ethernet frame data from the PHY to the MAC
L15	ETH_A_(S)(R)(G)MII_RXD1	ENET1_RX1	AK36	I	
N15	ETH_A_(S)(R)(G)MII_RXD2	ENET1_RX2	AJ37	I	
P15	ETH_A_(S)(R)(G)MII_RXD3	ENET1_RX3	AH38	I	
H15	ETH_A_(S)(R)(G)MII_TXD0	ENET1_TX0	AG33	O	Transmit Data [3:0] Four-bit data bus carrying transmit Ethernet frame data from the MAC to the PHY
G15	ETH_A_(S)(R)(G)MII_TXD1	ENET1_TX1	AG35	O	
H16	ETH_A_(S)(R)(G)MII_TXD2	ENET1_TX2	AF36	O	
G16	ETH_A_(S)(R)(G)MII_TXD3	ENET1_TX3	AG37	O	

¹ Bidirectional serial data line used to configure and monitor the Ethernet PHY via the MDIO management interface.

² Encodes **RX_DV** (receive data valid) and **RX_ER** (receive error) information for the receive path.

³ Encodes **TX_EN** (transmit enable) and **TX_ER** (transmit error) information for the transmit path.

Table 21: Ethernet B interface

X1 pad	OSM signal name	SoC ball name	SoC ball number	I/O	Description/Remarks
Control Signals					
T15	ETH_MDIO	ENET1_MDIO	AJ39	I/O	Management data input/output ¹
T16	ETH_MDC	ENET1_MDC	AK40	O	Management data clock
P1	ETH_B_(R)(G)MII_RX_CLK	ENET2_RXC	AJ25	I	Receive clock provided by the PHY in RGMII mode
L1	ETH_B_(R)(G)MII_RX_CTL	ENET2_RX_CTL	AH26	I	Receive control ²
H1	ETH_B_(R)(G)MII_TX_CLK	ENET2_TXC	AG23	O	Transmit clock driven by the MAC in RGMII mode
J2	ETH_B_(R)(G)MII_TX_CTL	ENET2_TX_CTL	AF24	O	Transmit control ³
Data signals					
J1	ETH_B_(S)(R)(G)MII_RXD0	ENET2_RX0	AJ27	I	Receive Data [3:0] Four-bit data bus carrying received Ethernet frame data from the PHY to the MAC
K1	ETH_B_(S)(R)(G)MII_RXD1	ENET2_RX1	AK28	I	
M1	ETH_B_(S)(R)(G)MII_RXD2	ENET2_RX2	AJ29	I	
N1	ETH_B_(S)(R)(G)MII_RXD3	ENET2_RX3	AH30	I	
G1	ETH_B_(S)(R)(G)MII_TXD0	ENET2_TX0	AG25	O	Transmit Data [3:0] Four-bit data bus carrying transmit Ethernet frame data from the MAC to the PHY
F1	ETH_B_(S)(R)(G)MII_TXD1	ENET2_TX1	AG27	O	
G2	ETH_B_(S)(R)(G)MII_TXD2	ENET2_TX2	AF28	O	
F2	ETH_B_(S)(R)(G)MII_TXD3	ENET2_TX3	AG29	O	

¹ Bidirectional serial data line used to configure and monitor the Ethernet PHY via the MDIO management interface.

² Encodes **RX_DV** (receive data valid) and **RX_ER** (receive error) information for the receive path.

³ Encodes **TX_EN** (transmit enable) and **TX_ER** (transmit error) information for the transmit path.

Table 22: Ethernet C interface

X1 pad	OSM signal name	SoC ball name	SoC ball number	I/O	Description/Remarks
Control Signals					
T15	ETH_MDIO	ENET1_MDIO	AJ39	I/O	Management data input/output ^{1,2}
T16	ETH_MDC	ENET1_MDC	AK40	O	Management data clock ²
Data signals					
AA35	ETH_C_(S)(R)(G)MII_TXD1	ETH_TX0_P	AJ17	O	Transmit Data differential pair
Y35	ETH_C_(S)(R)(G)MII_TXD0	ETH_TX0_N	AK16	O	
U35	ETH_C_(S)(R)(G)MII_RXD1	ETH_RX0_P	AJ13	I	Receive Data differential pair
V35	ETH_C_(S)(R)(G)MII_RXD0	ETH_RX0_N	AK12	I	

¹ Bidirectional serial data line used to configure and monitor the Ethernet PHY via the MDIO management interface.

² The MDIO and MDC signals are shared between the three Ethernet interfaces.

5.8 GPIO – General-Purpose Input/Output

The OSM iMX95 system-on-module provides **general-purpose digital I/O** functionality both through the native GPIO controllers integrated in the NXP i.MX 95 SoC and through external GPIO expanders based on the NXP [PCAL6416AHF](#) device family.

The GPIOs directly connected to the SoC operate within the native SoC I/O voltage domains and provide high-performance digital signaling. In addition to standard GPIO functionality, many of the SoC pins support alternate multiplexed functions such as PWM generation, communication interfaces, interrupt inputs, timer functions, and peripheral control signals.

Because the native SoC GPIOs are directly controlled by the processor, they provide fast switching performance, low-latency access, interrupt capability, and support for real-time applications. Internal pull-up and pull-down resistors are typically available and configurable through software, although external resistors may still be required depending on the application and signal integrity requirements.

These GPIOs are therefore recommended for:

- Timing-critical signals
- High-frequency switching
- Real-time control
- Peripheral interfaces
- Interrupt-driven applications

To increase the number of available GPIOs without consuming additional SoC pins, the OSM iMX95 also integrates two external GPIO expanders connected through the I²C bus. These expanders provide additional low-speed digital I/O lines intended primarily for control, monitoring, and configuration functions.

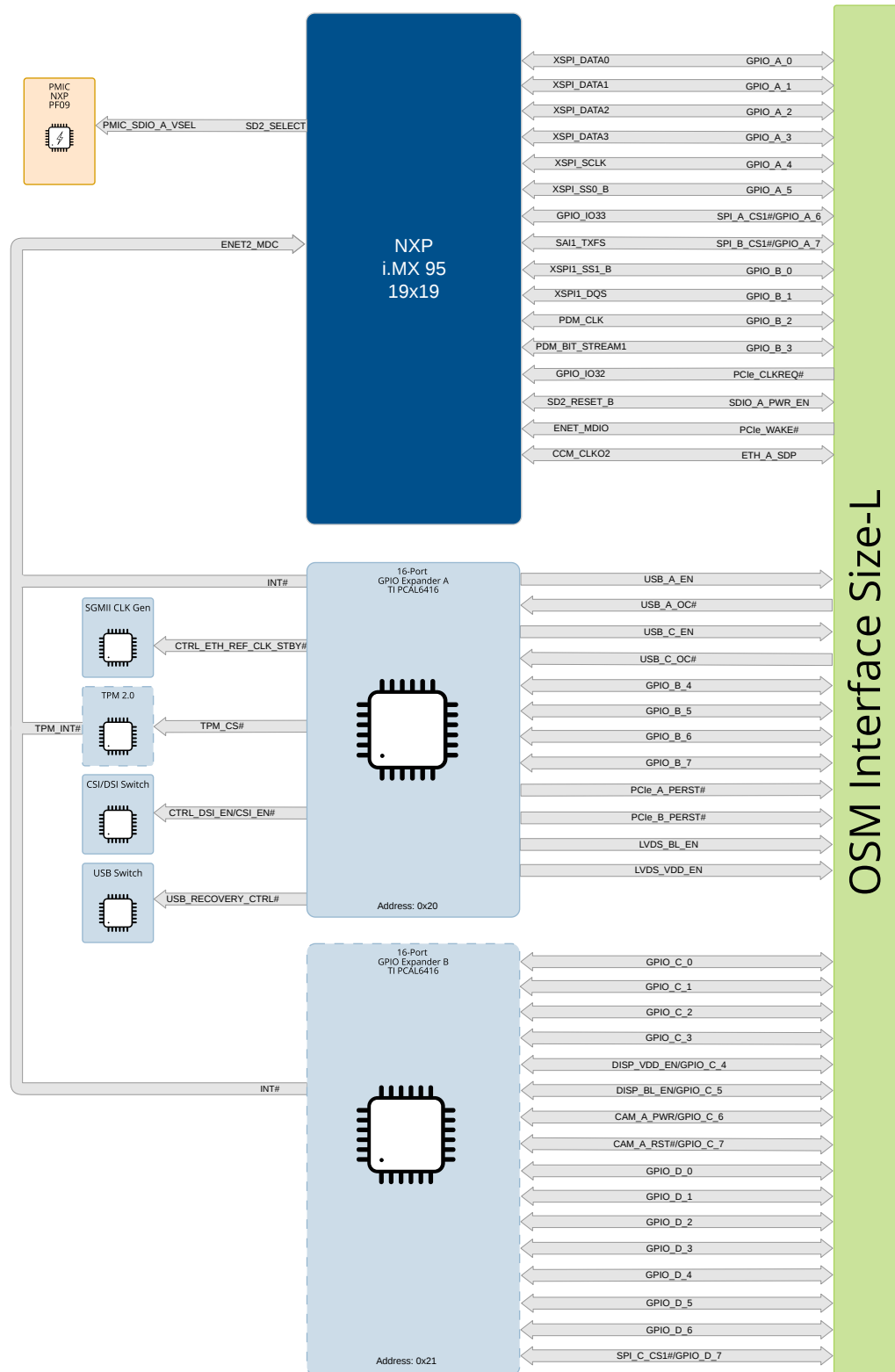
Since the GPIO expanders are accessed through the serial I²C interface, their switching speed and response time are significantly slower than direct SoC GPIO access. Consequently, the expander GPIOs are not suitable for high-frequency switching, precise timing applications, or latency-sensitive signals. The GPIO expander pins are therefore better suited for:

- System reset signals
- Enable controls
- Power management signals
- Status monitoring

- Configuration straps
- Low-speed control functions

Figure 4 shows the block diagram of the GPIOs from the SoC and from the GPIO expanders to the LGA connector of the OSM iMX95 module.

Figure 4: GPIO block diagram



Some of the GPIO expander signals are assigned to critical module-management and system-control functions required for correct operation of the OSM iMX95 platform. For this reason, the primary GPIO expander (IC16) is considered mandatory and must always be assembled. Omitting this device may impact system initialization, peripheral reset handling, power-management sequencing, USB power control, MIPI DSI/CSI interface selection, TPM access, and system recovery mode control.

Tables 23 to 26 list the GPIO interface signals exposed on the OSM iMX95 module, including native SoC GPIOs, GPIOs accessible via alternate functions, and GPIO expander signals.

Table 23: GPIO A interface

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function number	Description/Remarks
D17	GPIO_A_0	XSPI1_DATA0	AJ45	gpio5.IO[0]	ALT5	General-Purpose Input/Output signal
E17	GPIO_A_1	XSPI1_DATA1	AH46	gpio5.IO[1]	ALT5	General-Purpose Input/Output signal
F17	GPIO_A_2	XSPI1_DATA2	AJ47	gpio5.IO[2]	ALT5	General-Purpose Input/Output signal
G17	GPIO_A_3	XSPI1_DATA3	AK48	gpio5.IO[3]	ALT5	General-Purpose Input/Output signal
H17	GPIO_A_4	XSPI1_SCLK	AJ43	gpio5.IO[9]	ALT5	General-Purpose Input/Output signal
J17	GPIO_A_5	XSPI1_SS0_B	AJ41	gpio5.IO[10]	ALT5	General-Purpose Input/Output signal

Table 24: GPIO B interface

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function number	Description/Remarks
D19	GPIO_B_0	XSPI1_SS1_B	AH42	gpio5.IO[11]	ALT5	General-Purpose Input/Output signal
E19	GPIO_B_1	XSPI1_DQS	AK44	gpio5.IO[8]	ALT5	General-Purpose Input/Output signal
F19	GPIO_B_2	PDM_CLK	F46	gpio1.IO[8]	ALT5	General-Purpose Input/Output signal
G19	GPIO_B_3	PDM_BIT_STREAM1	H46	gpio1.IO[10]	ALT5	General-Purpose Input/Output signal
H19	GPIO_B_4	-	-	-	-	General-Purpose Input/Output signal from the GPIO Expander A ¹ pin P0_4
J19	GPIO_B_5	-	-	-	-	General-Purpose Input/Output signal from the GPIO Expander A ¹ pin P0_5
K19	GPIO_B_6	-	-	-	-	General-Purpose Input/Output signal from the GPIO Expander A ¹ pin P0_6
L19	GPIO_B_7	-	-	-	-	General-Purpose Input/Output signal from the GPIO Expander A ¹ pin P0_7

¹ The GPIO Expander A is mandatory, so these GPIO B signals are always available.

Table 25: GPIO C interface

X1 pad	OSM signal name	GPIO expander	Pin	Description/Remarks
D3	GPIO_C_0	B (IC6)	P0_0	General-Purpose Input/Output signal
D4	GPIO_C_1		P0_1	General-Purpose Input/Output signal
E3	GPIO_C_2		P0_2	General-Purpose Input/Output signal
E4	GPIO_C_3		P0_3	General-Purpose Input/Output signal
F3	GPIO_C_4 / DISP_VDD_EN		P0_4	General-Purpose Input/Output signal
F4	GPIO_C_5 / DISP_BL_EN		P0_5	General-Purpose Input/Output signal
G3	GPIO_C_6 / CAM_A_PWR		P0_6	General-Purpose Input/Output signal
G4	GPIO_C_7 / CAM_A_RST#		P0_7	General-Purpose Input/Output signal

Table 26: GPIO D interface

X1 pad	OSM signal name	GPIO expander	Pin	Description/Remarks
U32	GPIO_D_0	B (IC6)	P1_0	General-Purpose Input/Output signal
U33	GPIO_D_1		P1_1	General-Purpose Input/Output signal
V32	GPIO_D_2		P1_2	General-Purpose Input/Output signal
V33	GPIO_D_3		P1_3	General-Purpose Input/Output signal
W32	GPIO_D_4		P1_4	General-Purpose Input/Output signal
W33	GPIO_D_5		P1_5	General-Purpose Input/Output signal
Y32	GPIO_D_6		P1_6	General-Purpose Input/Output signal
Y33	GPIO_D_7 / SPI_C_CS1#		P1_7	General-Purpose Input/Output signal



GPIO E

The GPIO interface E is not provided in the OSM iMX95 SoM.

5.8.1 GPIOs with Defined Functions

The signal on [Table 27](#) are required for the operation of the OSM iMX95 and, therefore, prevent the GPIO expander from not being assembled.

Table 27: Defined functions from mandatory GPIO A (IC16)

X1 pad	OSM function	Pin	I/O	Description/Remarks
AC16	USB_A_EN	P0_0	O	Enable the USB A port
AC15	USB_A_OC#	P0_1	I	Overcurrent signal for the USB A port
C10	USB_C_EN	P0_2	O	Enable the USB C port
C8	USB_C_OC#	P0_3	I	Overcurrent signal for the USB C port
V2	PCIe_A_PERST#	P1_0	O	Reset signal for the PCIe A
L33	PCIe_B_PERST#	P1_1	O	Reset signal for the PCIe B
AN23	LVDS_BL_EN	P1_2	O	Enable the backlight for the LVDS display
AN14	LVDS_VDD_EN	P1_3	O	Enable the LVDS display

Continued on next page

Table 27: Defined functions from mandatory GPIO A (IC16) (Continued)

X1 pad	OSM function	Pin	I/O	Description/Remarks
-	USB_RECOVERY_CTRL#	P1_4	O	Selects the USB for the recovery mode internal signal
-	CTRL_ETH_REF_CLK_STBY#	P1_5	O	Controls the external reference clock for the ethernet internal signal
-	CTRL_DSI_EN/CSI_EN#	P1_6	O	Selects MIPI CSI or MIPI DSI internal signal
-	SPI_B_TPM_CS#	P1_7	O	Chip select for the TPM (active low)

5.9 I²C – Inter-Integrated Circuit - OSM iMX95

The OSM iMX95 system-on-module integrates **eight Inter-Integrated Circuit (I²C)** controllers based on the NXP i.MX 95 system-on-chip, providing low-speed serial communication for peripheral devices, system management, configuration interfaces, and sensor connectivity. The integrated I²C controllers are compliant with the **I²C v2.1 specification** and also support communication with **SMBus (System Management Bus)** compatible devices.



Interface Availability and Usage

Not all I²C controllers are intended for external use.

5.9.1 Exposed I²C Interfaces by Default

The primary external interface, **I2C_A**, is routed to the module connector as the main general-purpose I²C interface and is also the only interface supporting **I³C** operation. This interface is recommended for new peripheral designs requiring either standard I²C compatibility or higher-performance I³C functionality.

A secondary external interface, **I2C_B**, is also routed to the module connector and provides an additional general-purpose I²C communication channel intended for conventional I²C and SMBus-compatible peripherals. Unlike **I2C_A**, this interface does not support I³C operation.

[Table 28](#) lists the external I²C interface signals exposed on the OSM iMX95 module, including the corresponding SoC functions, pin assignments, and signal descriptions.

Table 28: I²C interfaces

X1 pad	OSM standard function	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description/Remarks
I2C_A							
AA15	I2C_A_SCL	I2C1_SCL	D48	ALT0/ALT1 ¹	i2c1.SCL/i3c1.SCL ¹	I/O	Serial clock line of the I2C_A interface; provides the timing reference for synchronous I ² C/I ³ C communication with external devices. 2.2kΩ pull-up to 1.8 V
AA16	I2C_A_SDA	I2C1_SDA	D52	ALT0/ALT1 ¹	i2c1.SDA/i3c1.SDA ¹	I/O	Bidirectional serial data line of the I2C_A interface; used for data transfer between the SoC and connected I ² C/I ³ C peripherals. 2.2kΩ pull-up to 1.8 V
I2C_B							
AA20	I2C_B_SCL	GPIO_IO23	T46	ALT6	i2c5.SCL	I/O	Serial clock line of the I2C_B interface; provides the timing reference for synchronous I ² C communication with external devices. 2.2kΩ pull-up to 1.8 V
AA21	I2C_B_SDA	GPIO_IO22	T44	ALT6	i2c5.SDA	I/O	Bidirectional serial data line of the I2C_B interface; used for data transfer between the SoC and connected I ² C peripherals. 2.2kΩ pull-up to 1.8 V

¹ The **I2C1** may be used as an I³C interface when setting the alternate function to **ALT1** instead of **ALT0**.



I³C Operation

When configured as I³C, the interface must comply with I³C electrical and protocol requirements, and connected devices must support I³C or be compatible with mixed I²C/I³C operation.

5.9.2 Internal I²C Interfaces

Although not all I²C controllers of the i.MX 95 are exposed for external use, documenting the internally used I²C interfaces remains important for system integration, software development, debugging, and carrier-board design. Two I²C controllers are dedicated to critical on-module functions such as PMIC control, GPIO expanders, EEPROM access, and other management functions required for correct operation of the OSM iMX95 platform.

Documenting these internal interfaces allows developers to clearly identify which I²C buses are reserved by the module architecture and therefore unavailable for external peripherals or application use. This helps avoid address conflicts, unintended interference with module-management functions, and incorrect software configuration.

PMIC I²C Interface

- Used internally for **power management** (PMIC control)
- Exposed on the connector **for internal product testing purposes only**
- **Must not be used for external applications**

Table 29 lists the I²C interface used for power control signals exposed on the OSM iMX95 for debugging purposes.

Table 29: PMIC I²C interface

X1 pad	OSM standard function	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description/Remarks
AM32	Vendor Defined	I2C2_SCL	E43	ALT0	i2c2.SCL	I/O	Serial clock line of the PMIC I ² C interface; provides the timing reference for synchronous I ² C communication with external devices 2.2kΩ pull-up to 1.8 V
AM33	Vendor Defined	I2C2_SDA	E45	ALT0	i2c2.SDA	I/O	Bidirectional serial data line of the PMIC I ² C interface; used for data transfer between the SoC and connected I ² C peripherals 2.2kΩ pull-up to 1.8 V

Internal Components I²C Interface

- Reserved for **internal module communication**
- Not intended for external use

Table 30 lists the I²C interface used to control internal components of the OSM iMX95 and are not exposed on the module's connector.

 Table 30: Internal I²C interface

X1 pad	OSM standard function	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description/Remarks
-	Internal I ² C	GPIO_IO29	V44	ALT1	i2c3.SCL	I/O	Serial clock line of the I ² C SoM interface; provides the timing reference for synchronous I ² C communication with on-board devices 2.2kΩ pull-up to 1.8 V
-	Internal I ² C	GPIO_IO28	U51	ALT1	i2c3.SDA	I/O	Bidirectional serial data line of the I ² C SoM interface; used for data transfer between the SoC and on-board devices 2.2kΩ pull-up to 1.8 V

5.9.3 I²C Addresses of Internal Devices

Table 31 lists the addresses of I²C on-board devices.

 Table 31: I²C addresses

Address	Part Number (PN)	Description/Remarks
PMIC I²C		
0x08	MPF0900AVNA1ESR2	Primary PMIC providing the main system and peripheral power rails
0x29	MPF5302AVNAAEPR2	Dedicated dynamic voltage regulator for the SoC core supply domain (VDD_SOC)
0x2A	MPF5301AVNABEPR2	Dedicated dynamic voltage regulator for the ARM processing domain (VDD_ARM)

Continued on next page

Table 31: I²C addresses (Continued)

Address	Part Number (PN)	Description/Remarks
On-Board Internal I²C		
0x20	PCAL6416AHF	Primary GPIO expander used for system-management, reset, and control functions
0x21	PCAL6416AHF	Secondary GPIO expander providing additional low-speed control and status signals
0x32	RX8130CE:B3	Real-time clock (RTC) with backup timekeeping support
0x48	TMP1075	On-module temperature sensor I ² C signals level-shifted to 3.3 V
0x50	M24C02-FMC6TG	On-module EEPROM used for module identification and configuration storage

5.9.4 Additional I²C Interfaces

Additional I²C controllers of the i.MX 95 may be available depending on the selected pin multiplexing configuration. Some interfaces are exposed only through alternate pin functions shared with other peripheral interfaces, which may affect compatibility across different module configurations and carrier-board designs.

Dedicated Camera-Control Interfaces

- **CAM_A:** Dedicated control interface for the CSI A camera connection, exposed through `CAM_A_SCL` and `CAM_A_SDA`.
- **CAM_B:** Dedicated control interface for the CSI B camera connection, exposed through `CAM_B_SCL` and `CAM_B_SDA`.

The camera-control signals are intended for configuring external image sensors, camera modules, serializers, or other devices associated with the corresponding CSI interface.

The `CAM_A_SCL / CAM_A_SDA` and `CAM_B_SCL / CAM_B_SDA` signals are multiplexed with the MIPI CSI-3 transmit signals. When used as I²C control interfaces, they cannot simultaneously operate as the corresponding `CSI_A_TX_P/N` or `CSI_B_TX_P/N` differential signals.

Dedicated LVDS Display-Control Interface

- **LVDS_I2C:** Dedicated display-control interface exposed through `LVDS_I2C_CLK` and `LVDS_I2C_DAT`.

This interface provides the DDC-compatible clock and data signals used for flat-panel detection, identification, configuration, and control. It is intended to be used together with the LVDS display interface.



Additional I²C Interfaces

Refer to the [Pinout Designer](#) and the module pin multiplexing documentation for information about the availability, multiplexing options, and usage constraints of the additional I²C interfaces provided for camera and display control.

5.10 JTAG – Joint Test Action Group

The OSM iMX95 System-on-Module provides a hardware debug interface based on the integrated **JTAG Controller (JTAGC)** of the NXP i.MX 95 SoC. The interface supports both **IEEE 1149.1 JTAG** and **Serial Wire Debug (SWD)** operation, enabling low-level debugging, firmware development, device programming, and boundary scan testing.

The debug interface signals and their pin mapping are described in the corresponding interface table. The interface includes the standard TAP/SWD signals required for external debug probes and ARM-compatible development tools.

The integrated JTAG controller implements the standard IEEE 1149.1-2001 TAP (Test Access Port) architecture, including:

- TAP controller state machine
- Boundary scan register
- Device identification register (IDCODE)
- Bypass and test instruction support

The JTAG/SWD debug interface signals, pin assignments, and corresponding SoC alternate functions are documented in [Table 32](#).

Table 32: JTAG interface

X1 pad	OSM signal name	SoC ball name	SoC alternate function	SoC alternate function name	I/O	Description
P17	JTAG_1_TDI	DAP_TDI	ALT0	dap.TDI	I	Test data in
R17	JTAG_1_TDO	DAP_TDO_TRACESWO	ALT0	dap.TDO_TRACESWO	O	Test data out
N17	JTAG_1_TCK	DAP_TCLK_SWCLK	ALT0	dap.TCLK_SWCLK	I	Test Clock
N19	JTAG_1_TMS	DAP_TMS_SWDIO	ALT0	dap.TMS_SWDIO	I	Test Mode Select

Supported capabilities include:

- Firmware programming
- Real-time debugging
- Boundary scan validation
- Low-level processor inspection and tracing



Recovery Mode

Instead of using the JTAG port, it is strongly recommended to reprogram the module using the Recovery Mode over USB.

5.11 PWM – Pulse Width Modulation

The OSM iMX95 system-on-module integrates multiple **timer and Pulse Width Modulation (PWM)** resources based on the TPM (Timer/PWM Module) architecture of the NXP i.MX 95 SoC. The i.MX 95 incorporates six TPM peripherals within the Always-On (AON) power domain. All six peripherals share the module's 1.8 V I/O power rail with the remaining I/O interfaces:

- AON_TPM_1 to AON_TPM_6

Each TPM module includes:

- **One 32-bit timer/counter**
- **Six independently configurable channels**
- Programmable prescaler division: 1, 2, 4, 8, 16, 32, 64, or 128
- Support for asynchronous clocking
- DMA request generation
- Interrupt generation per channel and overflow event
- Configurable low-power and debug operation behavior

Together, the TPM peripherals provide multiple **PWM-capable channels** within the SoC. Each channel can be independently configured as:

- PWM output
- Input capture input
- Output compare channel



Operating Modes

The TPM architecture supports multiple operating modes, including:

- Edge-Aligned PWM (EPWM)
- Center-Aligned PWM (CPWM)
- Combined PWM mode with dead-time insertion
- Input Capture mode
- Output Compare mode
- Quadrature Decoder mode



Naming Convention

In this section, **TPM** refers to the **Timer/PWM Module** peripheral of the NXP i.MX 95 SoC. In other sections of this document, the acronym **TPM** may instead refer to a **Trusted Platform Module** security device.

5.11.1 PWM Signal Availability

The OSM iMX95 module provides several PWM-capable signals through dedicated and multiplexed SoC pins. Some PWM outputs are available as alternate pin functions and may share pins with GPIOs or other peripheral interfaces, depending on the selected pin multiplexing configuration.

Two PWM interfaces are reserved for display backlight control:

- **MIPI DSI Backlight PWM:** Dedicated to controlling the backlight inverter of MIPI DSI displays.
- **LVDS Backlight PWM:** Dedicated to controlling the backlight of LVDS displays.

The PWM signals are assigned to module-specific interfaces or are available only through alternate pin functions. Availability and carrier-board compatibility may differ between OSM modules and carrier boards. Designers should verify the selected pin assignments and multiplexing options to ensure compatibility with the target hardware platform.

Table 33 lists the PWM-capable signals exposed on the OSM iMX95 module connector in the default pin multiplexing configuration.

Table 33: PWM interfaces

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function name	Soc alternate function	I/O	Description/Remarks
Application-Specific							
E18	DISP_BL_PWM / PWM_0	GPIO_IO12	N45	tpm3.CH2	ALT1	O	Backlight control or general-purpose PWM ¹
AN22	LVDS_BL_PWM	GPIO_IO24	T48	tpm3.CH3	ALT4	O	Backlight control of the LVDS display(s) ¹

Continued on next page

Table 33: PWM interfaces (Continued)

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function name	Soc alternate function	I/O	Description/Remarks
General-Purpose							
F18	PWM_1	GPIO_IO13	N49	tpm4.CH2	ALT1	O	General Purpose PWM
G18	PWM_2	GPIO_IO18	P52	tpm5.CH2	ALT6	O	General Purpose PWM
H18	PWM_3	GPIO_IO19	R45	tpm6.CH2	ALT6	O	General Purpose PWM

¹ The **PWM_0** (**TPM3_CH0**) and **LVDS_BL_PWM** (**TPM3_CH3**) outputs are provided by the same **TPM3** timer module. Since all channels within a TPM module share a common timer counter, these outputs must use the same PWM period (frequency). The duty cycle of each channel can be configured independently.

5.11.2 Additional PWM Interfaces

Additional PWM-capable signals of the i.MX 95 may be available depending on the selected pin multiplexing configuration. Some PWM functions are exposed only through alternate pin functions shared with other peripheral interfaces, which may affect compatibility across different module configurations and carrier-board designs.

Table 34 lists the PWM-capable signals available through pin multiplexing (alternate functions) and their corresponding multiplexed interfaces.

Table 34: PWM signals available as function multiplexing

X1 pad	OSM signal name	SoC ball name	SoC ball number	Soc alternate function	SoC alternate function name	I/O	Description/Remarks
TPM 1							
AA23	SPI_B_CS0#	PDM_BIT_STREAM0	G45	ALT3	tpm1.EXTCLK	I	External clock input for TPM 1
D22	UART_CON_RX	UART1_RXD	E49	ALT3	tpm1.CH0	I/O	Channel 0 of Timer/PWM Module (TPM) 1
D20	UART_CON_TX	UART1_TXD	F52	ALT3	tpm1.CH1	O	Channel 1 of Timer/PWM Module (TPM) 1 Boot mode 0, with 10kΩ pull-down.
C22	UART_D_RX	UART2_RXD	E51	ALT3	tpm1.CH2	I/O	Channel 2 of Timer/PWM Module (TPM) 1
C23	UART_D_TX	UART2_TXD	F48	ALT3	tpm1.CH3	O	Channel 3 of Timer/PWM Module (TPM) 1 Boot mode 1, with 10kΩ pull-down.
TPM 2							
G19	GPIO_B_3	PDM_BIT_STREAM1	H46	ALT3	tpm2.EXTCLK	I	External clock input for TPM 2
AA15	I2C_A_SCL	I2C1_SCL	D48	ALT3	tpm2.CH0	I/O	Channel 0 of Timer/PWM Module (TPM) 2
AA16	I2C_A_SDA	I2C1_SDA	D52	ALT3	tpm2.CH1	I/O	Channel 1 of Timer/PWM Module (TPM) 2
AM32	Vendor Defined PMIC_I2C_SCL	I2C2_SCL	F43	ALT3	tpm2.CH2	I/O	Channel 2 of Timer/PWM Module (TPM) 2 for debugging purposes only
AM33	Vendor Defined PMIC_I2C_SDA	I2C2_SDA	E45	ALT3	tpm2.CH3	I/O	Channel 3 of Timer/PWM Module (TPM) 2 for debugging purposes only

Continued on next page

Table 34: PWM signals available as function multiplexing (Continued)

X1 pad	OSM signal name	SoC ball name	SoC ball number	Soc alternate function	SoC alternate function name	I/O	Description/Remarks
TPM 3							
AB25	CAM_B_SCL/ CSI_B_TX_P	GPIO_IO09	M46	ALT4	tpm3.EXTCLK	I	External clock input for TPM 3
B13	UART_A_TX	GPIO_IO04	K46	ALT1	tpm3.CH0	I/O	Channel 0 of Timer/PWM Module (TPM) 3
V21	I2S_A_DATA_IN	GPIO_IO20	R49	ALT6	tpm3.CH1	I/O	Channel 1 of Timer/PWM Module (TPM) 3
TPM 4							
AM12	LVDS_I2C_DAT	GPIO_IO10	M48	ALT4	tpm4.EXTCLK	I	External clock input for TPM 4
A14	UART_A_RX	GPIO_IO05	L45	ALT1	tpm4.CH0	I/O	Channel 0 of Timer/PWM Module (TPM) 4
W21	I2S_A_DATA_OUT	GPIO_IO21	R51	ALT6	tpm4.CH1	I/O	Channel 1 of Timer/PWM Module (TPM) 4
AC17	CAN_A_TX	GPIO_IO25	T52	ALT4	tpm4.CH3	I/O	Channel 3 of Timer/PWM Module (TPM) 4
TPM 5							
AM11	LVDS_I2C_CLK	GPIO_IO11	M52	ALT4	tpm5.EXTCLK	I	External clock input for TPM 5
C14	UART_A_CTS	GPIO_IO06	L49	ALT1	tpm5.CH0	I/O	Channel 0 of Timer/PWM Module (TPM) 5
AA21	I2C_B_SDA	GPIO_IO22	T44	ALT4	tpm5.CH1	I/O	Channel 1 of Timer/PWM Module (TPM) 5
W18	I2S_A_LRCLK	GPIO_IO26	U45	ALT4	tpm5.CH3	I/O	Channel 3 of Timer/PWM Module (TPM) 5
TPM 6							
AA21	I2C_B_SDA	GPIO_IO22	T44	ALT5	tpm6.EXTCLK	I	External clock input for TPM 6 2.2kΩ pull-up
AA26	CAM_B_SDA/ CSI_B_TX_N	GPIO_IO08	M44	ALT4	tpm6.CH0	I/O	Channel 0 of Timer/PWM Module (TPM) 6 2.2kΩ pull-up
AA20	I2C_B_SCL	GPIO_IO23	T46	ALT4	tpm6.CH1	I/O	Channel 1 of Timer/PWM Module (TPM) 6 2.2kΩ pull-up
AB17	CAN_A_RX	GPIO_IO27	U49	ALT4	tpm6.CH3	I/O	Channel 3 of Timer/PWM Module (TPM) 6

Rows in : The I²C or alternate functions associated with the PMIC shall not be used, otherwise the system would not work.

Refer to the [Pinout Designer](#) and the module pin multiplexing documentation for the availability and usage constraints of the additional PWM interfaces.

5.12 SPI – Serial Peripheral Interface

The OSM iMX95 system-on-module provides multiple **Serial Peripheral Interface (SPI)** controllers based on the Low Power Serial Peripheral Interface (**LPSPi**) peripherals integrated in the NXP i.MX 95 SoC. These interfaces enable high-speed, full-duplex synchronous serial communication with a wide range of external peripherals such as sensors, ADCs, DACs, displays, memory devices, FPGAs, and communication controllers.

The i.MX 95 SoC integrates a total of **eight LPSPi controllers**, offering flexible connectivity and pin multiplexing options for embedded applications. Depending on the selected pin multiplexing configuration, additional ECSPI instances may be enabled for application-specific use.

The SPI interfaces provide the standard SPI signals:

- **SCK**: Serial clock
- **SDI**: MISO: Serial data input
- **SDO**: MOSI: Serial data output
- **CS#**: Chip select²

The ECSPi peripherals support:

- Full-duplex synchronous communication
- Master and slave operation
- Configurable clock polarity and phase (SPI modes 0–3)
- Programmable word length
- Multiple transfer formats
- Interrupt and DMA-based data transfers

Performance

The SPI interfaces support high-speed operation, capable of:

- **Up to 60 Mbps in master transmit mode**
- **Up to 30 Mbps in master receive mode**



Maximum Operating Frequency

The maximum operating frequency of other ECSPi instances may differ depending on the specific peripheral instance, clock configuration, routing, and selected pin multiplexing. Refer to the NXP i.MX 95 documentation for detailed timing and performance limitations.

Table 35 lists the SPI interface signals exposed on the OSM iMX95 module, including the corresponding ECSPi controller assignments, pin multiplexing options, and signal descriptions.

Table 35: SPI interfaces

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description/Remarks
SPI_A							
Y15	SPI_A_CS0#	GPIO_IO34	W49	ALT4	spi4.PCS0	O	Peripheral select 0
K17	SPI_A_CS1#/GPIO_A_6	GPIO_IO33	W45	ALT4	spi4.PCS1	I/O ¹	Peripheral select 1 or general-purpose pin
U15	SPI_A_SDI_(IO1)	GPIO_IO35	W51	ALT4	spi4.SIN	I	Controller input, peripheral output
V15	SPI_A_SDO_(IO0)	GPIO_IO36	Y48	ALT4	spi4.SOUT	O	Controller output, peripheral input
U16	SPI_A_SCK	GPIO_IO36	Y52	ALT4	spi4.SCK	O	Serial clock

Continued on next page

²Some of the interfaces have more than one chip select.

Table 35: SPI interfaces (Continued)

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description/Remarks
SPI_B							
AA23	SPI_B_CS0#	PDM_BIT_STREAM0	G45	ALT2	spi1.PCS1	O	Peripheral select 0
L17	SPI_B_CS1#/GPIO_A_7	SAI1_TXFS	G49	ALT2	spi1.PCS0	I/O ¹	Peripheral select 1 Boot mode 2, with 10 kΩ pull-down ²
Y22	SPI_B_SDI	SAI1_TXC	G51	ALT2	spi1.SIN	I	Controller input, peripheral output
Y23	SPI_B_SDO	SAI1_RXD0	H52	ALT2	spi1.SOUT	O	Controller output, peripheral input
Y21	SPI_B_SCK	SAI1_TXD0	H48	ALT2	spi1.SCK	O	Serial clock Boot mode 2, with 10 kΩ pull-up to 1.8 V
SPI2 – Alternate Functions							
D23	UART_CON_TX	UART1_TXD	F52	ALT2	spi2.PCS0	O	Peripheral select 0
G19	GPIO_B_3	PDM_BIT_STREAM1	H46	ALT2	spi2.PCS1	I/O ¹	Peripheral select 1
D22	UART_CON_RX	UART1_RXD	E49	ALT2	spi2.SIN	I	Controller input, peripheral output
C22	UART_D_RX	UART2_RXD	E51	ALT2	spi2.SOUT	O	Controller output, peripheral input
C23	UART_D_TX	UART2_TXD	F48	ALT2	spi2.SCK	O	Serial clock
SPI3 – Alternate Functions							
AB26	CAM_B_SDA/CSI_B_TX_N	GPIO_IO08	M44	ALT1	spi3.PCS0	O	Peripheral select 0
C13	UART_A_RTS	GPIO_IO07	L51	ALT1	spi3.PCS1	I/O ¹	Peripheral select 1
AB25	CAM_B_SCL/CSI_B_TX_P	GPIO_IO09	M46	ALT1	spi3.SIN	I	Controller input, peripheral output
AM12	LVDS_I2C_DAT	GPIO_IO10	M48	ALT1	spi3.SOUT	O	Controller output, peripheral input
AM11	LVDS_I2C_CLK	GPIO_IO11	M52	ALT1	spi3.SCK	O	Serial clock
SPI5 – Alternate Functions							
G18	PWM_2	GPIO_IO18	P52	ALT4	spi5.PCS0	O	Peripheral select 0
AB17	CAN_A_RX	GPIO_IO27	U49	ALT6	spi5.PCS1	I/O ¹	Peripheral select 1
H18	PWM_3	GPIO_IO19	R45	ALT4	spi5.SIN	I	Controller input, peripheral output
V21	I2S_A_DATA_IN	GPIO_IO20	R49	ALT4	spi5.SOUT	O	Controller output, peripheral input
W21	I2S_A_DATA_OUT	GPIO_IO21	R51	ALT4	spi5.SCK	O	Serial clock
SPI6 – Alternate Functions							
D13	UART_B_TX	GPIO_IO00	J49	ALT4	spi6.PCS0	O	Peripheral select 0
AN22	LVDS_BL_PWM	GPIO_IO24	T48	ALT6	spi6.PCS1	I/O ¹	Peripheral select 1
D14	UART_B_RX	GPIO_IO01	J51	ALT4	spi6.SIN	I	Controller input, peripheral output
D16	UART_B_CTS	GPIO_IO02	K48	ALT4	spi6.SOUT	O	Controller output, peripheral input
D15	UART_B_RTS	GPIO_IO03	K52	ALT4	spi6.SCK	O	Serial clock

Continued on next page

Table 35: SPI interfaces (Continued)

X1 pad	OSM signal name	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description/Remarks
SPI7 – Alternate Functions							
B13	UART_A_TX	GPIO_IO04	K46	ALT4	spi7.PCS0	O	Peripheral select 0
AC17	CAN_A_TX	GPIO_IO25	T52	ALT6	spi7.PCS1	I/O ¹	Peripheral select 1
A14	UART_A_RX	GPIO_IO05	L45	ALT4	spi7.SIN	I	Controller input, peripheral output
C14	UART_A_CTS	GPIO_IO06	L49	ALT4	spi7.SOUT	O	Controller output, peripheral input
C13	UART_A_RTS	GPIO_IO07	L51	ALT4	spi7.SCK	O	Serial clock
SPI8 – Alternate Functions							
E18	DISP_BL_PWM/PWM_0	GPIO_IO12	N45	ALT4	spi8.PCS0	O	Peripheral select 0
W18	I2S_A_LRCLK	GPIO_IO26	U45	ALT6	spi8.PCS1	I/O ¹	Peripheral select 1
F18	PWM_1	GPIO_IO13	N49	ALT4	spi8.SIN	I	Controller input, peripheral output
B23	UART_C_TX	GPIO_IO14	N51	ALT4	spi8.SOUT	O	Controller output, peripheral input
A22	UART_C_RX	GPIO_IO15	P44	ALT4	spi8.SCK	O	Serial clock

¹ May be an input when used as a general-purpose input/output pin.

² Care must be taken during reset because the boot mode is selected by the state of the `SPI_B_CS1#` / `GPIO_A_7` pin.



TPM SPI Interface

When the TPM device is populated, the `SPI_B` interface is dedicated to communication with the TPM. Consequently, it is not available for general-purpose use, and the alternate functions multiplexed on the corresponding pins must not be used.

In particular, the SoC SPI interface connected to the TPM module reuses the following signals through alternate pin multiplexing:

- `SPI_B_CS1#`: care must be taken when re-purposing this signal at start-up because it acts as the boot mode 2 pin.
- `SPI_B_SCK`: care must be taken when re-purposing this signal at start-up because it acts as the boot mode 3 pin.

5.13 Tamper Detection

The OSM iMX95 system-on-module provides **support for tamper detection** through the tamper pins integrated in the NXP i.MX 95 system-on-chip. These pins are part of the SoC secure subsystem and are intended to detect physical intrusion or unauthorized access attempts, making them suitable for security-sensitive and anti-tamper applications.

The tamper pins are connected internally to the i.MX 95 security and low-power domains, allowing tamper monitoring to remain active independently of the main application processors. The pins can be configured to monitor external signals for predefined tamper conditions, such as voltage level changes, edge transitions, or active tamper patterns, and may trigger security events even while the system operates in low-power states, provided the secure power domain remains powered.

Tamper events can be configured to generate interrupts, status indications, or programmable security responses, including:

- Event logging and timestamping
- System reset generation

- Secure state transitions
- Cryptographic key invalidation or protection actions, depending on the software configuration

The i.MX 95 supports both passive and active tamper mechanisms, enabling implementation of advanced physical security monitoring and intrusion-detection schemes for industrial, secure-edge, and embedded applications.

Table 36 lists the tamper interface signals exposed on the OSM iMX95, including their pin assignments, signal direction, and functional descriptions.

Table 36: Tamper detection

OSM ball	OSM signal name	SoC ball name	SoC ball number	I/O	Description
D6	TAMPER0 Vendor Defined	TAMPER0	E36	I/O	Tamper detection pin
D7	TAMPER1 Vendor Defined	TAMPER1	D38	I/O	Tamper detection pin

5.14 UART – Universal Asynchronous Receiver/Transmitter

The OSM iMX95 system-on-module provides multiple serial communication interfaces based on the **Universal Asynchronous Receiver/Transmitter (UART)** interfaces integrated in the NXP i.MX 95 SoC. These interfaces enable reliable, low-latency, point-to-point asynchronous communication and are commonly used for system debugging, console access, firmware development, boot monitoring, and communication with external serial peripherals.

The UART controllers of the i.MX 95 support full-duplex asynchronous communication and configurable baud rates and frame formats. Depending on the selected interface and pin multiplexing configuration, optional hardware flow control signals (**RTS/CTS**) are also available. The UART subsystem integrates with the SoC interrupt and DMA infrastructure, allowing efficient high-speed serial communication with minimal CPU overhead.

5.14.1 Key Features and Supported Capabilities

Interface standard

- Asynchronous serial UART

Data formats

- Configurable word lengths (typically 7 or 8 bits)
- Optional parity:
 - None
 - Even
 - Odd
- One or two stop bits

Baud rates

- Supports standard and high-speed baud rates
- Maximum baud rate depends on:
 - Peripheral clock configuration
 - Oversampling configuration

- Signal integrity and carrier-board design

Flow control

- Optional hardware flow control (RTS / CTS)
- Software flow control supported by software stack

System integration

- Interrupt-driven operation
- DMA-assisted transfers
- Suitable for debug consoles and high-throughput serial communication

Table 37 lists the UART interface signals exposed on the OSM iMX95, including their pin assignments, alternate functions, signal directions, and functional descriptions.

Table 37: UART interfaces

OSM pad	OSM function name	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description/Remarks
UART_A							
B13	UART_A_TX	GPIO_IO04	K46	ALT5	uart6.TX	O	Transmitted Data of general-purpose UART_A
A14	UART_A_RX	GPIO_IO05	L45	ALT5	uart6.RX	I	Received Data of general-purpose UART_A
C14	UART_A_CTS	GPIO_IO06	L49	ALT5	uart6.CTS_B	O	Clear to Send of general-purpose UART_A
C13	UART_A_RTS	GPIO_IO07	L51	ALT5	uart6.RTS_B	I	Request to Send of general-purpose UART_A
UART_B							
D13	UART_B_TX	GPIO_IO00	J49	ALT5	uart5.TX	O	Transmitted Data of general-purpose UART_2B
D14	UART_B_RX	GPIO_IO01	J51	ALT5	uart5.RX	I	Received Data of general-purpose UART_B
D16	UART_B_CTS	GPIO_IO02	K48	ALT5	uart5.CTS_B	I	Clear to Send of general-purpose UART_B
D15	UART_B_RTS	GPIO_IO03	K52	ALT5	uart5.RTS_B	O	Request to Send of general-purpose UART_B
UART_C							
B23	UART_C_TXD	GPIO_IO14	N51	ALT1	uart3.TX	O	Transmitted Data of UART_3
A22	UART_C_RXD	GPIO_IO15	P44	ALT1	uart3.RX	I	Received Data of UART_3
UART_CON							
D23	UART_CON_TX	UART1_TXD	F52	ALT0	uart1.TX	O	Transmitted Data of A55 debug UART_CON Boot mode 0 with 10kΩ pull-down controlled by control recovery during boot ¹
D22	UART_CON_RX	UART1_RXD	E49	ALT0	uart1.RX	I	Received Data of A55 debug UART_CON
UART_D							
C23	UART_D_RXD	UART2_RXD	E51	ALT0	uart2.RX	I	Received Data of A55 debug UART_D
C22	UART_D_TXD	UART2_TXD	F48	ALT0	uart2.TX	O	Transmitted Data of A55 debug UART_D Boot mode 1 with 10kΩ pull-down

Continued on next page

Table 37: UART interfaces (Continued)

OSM pad	OSM function name	SoC ball name	SoC ball number	SoC alternate function	SoC alternate function name	I/O	Description/Remarks
UART4 – Alternate Functions							
B23	UART_C_TXD	GPIO_IO14	N51	ALT6	uart4.TX	O	Transmitted data of UART4; multiplexed with <code>UART_C_TXD</code>
A22	UART_C_RXD	GPIO_IO15	P44	ALT6	uart4.RX	I	Received data of UART4; multiplexed with <code>UART_C_RXD</code>
W20	I2S_A_BITCLK	GPIO_IO16	P48	ALT6	uart4.CTS_B	I	Clear to Send input of UART4; multiplexed with <code>I2S_A_BITCLK</code>
V18	I2S_MCLK	GPIO_IO17	P52	ALT6	uart4.RTS_B	O	Request to Send output of UART4; multiplexed with <code>I2S_MCLK</code>
UART7 – Alternate Functions							
AB26	CAM_B_SDA/ CSI_B_TX_N	GPIO_IO08	M44	ALT5	uart7.TX	O	Transmitted data of UART7; multiplexed with the CAM_B data or CSI_B transmit signal
AB25	CAM_B_SCL/ CSI_B_TX_P	GPIO_IO09	M46	ALT5	uart7.RX	I	Received data of UART7; multiplexed with the CAM_B clock or CSI_B transmit signal
AM12	LVDS_I2C_DAT	GPIO_IO10	M48	ALT5	uart7.CTS_B	I	Clear to Send input of UART7; multiplexed with <code>LVDS_I2C_DAT</code>
AM11	LVDS_I2C_CLK	GPIO_IO11	M52	ALT5	uart7.RTS_B	O	Request to Send output of UART7; multiplexed with <code>LVDS_I2C_CLK</code>
UART8 – Alternate Functions							
E18	DISP_BL_PWM/ PWM_0	GPIO_IO12	N45	ALT5	uart8.TX	O	Transmitted data of UART8; multiplexed with the display backlight PWM signal
F18	PWM_1	GPIO_IO13	N49	ALT5	uart8.RX	I	Received data of UART8; multiplexed with <code>PWM_1</code>
B23	UART_C_TXD	GPIO_IO14	N51	ALT5	uart8.CTS_B	I	Clear to Send input of UART8; multiplexed with <code>UART_C_TXD</code>
A22	UART_C_RXD	GPIO_IO15	P44	ALT5	uart8.RTS_B	O	Request to Send output of UART8; multiplexed with <code>UART_C_RXD</code>

¹ Care must be taken during reset because the boot mode is selected by the state of this pin.

5.14.2 UART Recommended Usage

The OSM iMX95 module provides five UART interfaces intended for different purposes. While all interfaces are fully functional and may be reassigned through software and pin multiplexing, using them according to the recommendations below maximizes compatibility with other OSM iMX95 modules and simplifies software portability across carrier board designs.

UART_A – General-Purpose UART

`UART_A` is connected to the `uart6` peripheral of the i.MX 95 SoC and provides a general-purpose serial communication interface. Although it can be assigned to either the Cortex®-A55 or Cortex®-M33 cores, it has no dedicated purpose within the OSM specification.

Typical applications include:

- External serial peripherals
- Industrial communication interfaces
- Sensor and controller connectivity

- Application-specific serial protocols

UART_B – Auxiliary General-Purpose UART

UART_B is connected to the `uart5` peripheral of the i.MX 95 SoC and provides an additional general-purpose serial communication interface. It may be assigned to either the Cortex®-A55 or Cortex®-M33 cores depending on the software architecture.

Typical applications include:

- External serial peripherals
- Industrial communication interfaces
- Auxiliary communication channels
- Application-specific serial protocols

UART_C – General-Purpose UART

UART_C is connected to the `uart3` peripheral of the i.MX 95 SoC and provides another flexible serial communication interface for application-specific use. It may be used by either the Cortex®-A55 or Cortex®-M33 cores as required by the application.

Typical applications include:

- External controllers
- Sensor interfaces
- Industrial communication adapters
- Auxiliary serial channels

UART_CON – Primary Cortex®-A55 Debug Console

UART_CON is connected to the `uart1` peripheral of the i.MX 95 SoC and serves as the primary debug and console interface for the Cortex®-A55 application processors. This interface is active throughout the complete boot process, including:

- ROM bootloader
- Early boot stages
- U-Boot
- Linux kernel
- Linux user space

It is the primary interface used for:

- Boot messages
- System logs
- Board bring-up
- Software development
- System recovery and diagnostics

For development, manufacturing, and recovery, it is strongly recommended that the `UART_CON` interface remains accessible on the carrier board.

UART_D – Recommended Cortex®-M33 Debug Console

`UART_D` is connected to the `uart2` peripheral of the i.MX 95 SoC and is the recommended debug and console interface for firmware running on the Cortex®-M33 real-time core. Using `UART_D` for real-time software development maintains compatibility with other OSM iMX95 modules and allows the Cortex®-M33 debug console to operate independently of the primary Cortex®-A55 debug console provided by `UART_CON`.

Typical applications include:

- Cortex®-M33 firmware debugging
- RTOS console output
- Bare-metal software development
- Real-time diagnostic logging
- Independent debug console during heterogeneous software development

5.15 USB – Universal Serial Bus

The OSM iMX95 system-on-module provides **USB connectivity based on the native USB controllers** integrated in the NXP i.MX 95 system-on-chip, enabling high-speed host, peripheral, and dual-role USB operation for a wide range of embedded and industrial applications. The i.MX 95 integrates multiple USB interfaces supporting both **USB 2.0 High-Speed (HS)** and **USB 3.0 SuperSpeed operation**.

Features

- **USB 2.0 High-Speed** operation up to **480 Mbps**
- **USB 3.0 SuperSpeed** differential interfaces up to **5 Gbps**
- **Host, Device**, and **OTG dual-role** operation

Each USB interface uses standard USB signaling, including:

- Differential data pairs (`DP` / `DN`)
- SuperSpeed transmit and receive differential pairs (`SSTX` / `SSRX`)
- VBUS sensing
- ID-based OTG role detection, with 10 kΩ pull-up biased to the **1.8 V I/O domain**

The `USBx_VBUS` pins are used for VBUS voltage sensing and **can be directly connected to the 5 V USB VBUS rail**.

The on-SoC PHY calibration circuitry is used to achieve the nominal USB output impedance required for compliant high-speed operation while maintaining reliable signal integrity across process, voltage, and temperature variations.

[Table 38](#) summarizes the role capabilities and supported operating modes of the USB interfaces available on the OSM iMX95 system-on-module.

Table 38: USB overview

Interface	Host	Device/Client	OTG
USB_A	Yes	Yes	Yes
USB_C	Yes	Yes	Yes

The `USB_D` interface exposes only the **USB 3.0 SuperSpeed** differential pairs and does not include the USB 2.0 signaling, VBUS sensing, or USB ID detection signals required for standalone USB host, device, or OTG

operation. As a result, **USB_D** cannot operate independently as a complete USB interface. Instead, it is intended to function as a companion SuperSpeed interface associated with another USB port, typically within USB Type-C or USB multiplexing architectures where the USB 2.0 and USB 3.x paths are routed separately. This arrangement enables high-speed USB connectivity while maintaining compatibility with Type-C orientation switching, USB hubs, or external SuperSpeed routing logic.

Table 39 shows the mapping between USB signals and the LGA connector pads of the OSM iMX95 module.

Table 39: USB interfaces

OSM pad	OSM standard function	SoC ball name	SoC ball number	I/O	Description / Remarks
USB_A					
AC14	USB_A_D_P ¹	_2	–	I	Positive USB 2.0 differential pair signal
AB13	USB_A_D_N ¹	_2	–	I	Negative USB 2.0 differential pair signal
AB14	USB_A_ID	USB2_ID	E24	I	USB OTG identification input
AC16	USB_A_EN	_3	–	O	Enable the USB A port
AC15	USB_A_OC#	_3	–	I	Overcurrent signal for the USB A port
USB_C					
D10	USB_C_D_P ⁴	_2	–	I	Positive USB 2.0 differential pair signal
D11	USB_C_D_N ⁴	_2	–	I	Negative USB 2.0 differential pair signal
A8	USB_C_SSTX_P	USB1_TX0_P	D26	O	Positive SuperSpeed transmit differential pair
A9	USB_C_SSTX_N	USB1_TX0_N	E25	O	Negative SuperSpeed transmit differential pair
B10	USB_C_SSRX_P	USB1_RX0_P	A21	I	Positive SuperSpeed receive differential pair
B11	USB_C_SSRX_N	USB1_RX0_N	B20	I	Negative SuperSpeed receive differential pair
D9	USB_C_ID	USB1_ID	C23	I	USB OTG identification input
C10	USB_C_EN	_3	–	O	Enable the USB C port
C8	USB_C_OC#	_3	–	I	Overcurrent signal for the USB C port
USB_D⁵					
A27	USB_D_SSTX_P	USB1_TX1_P	E21	O	Positive SuperSpeed transmit differential pair
A28	USB_D_SSTX_N	USB1_TX1_N	D22	O	Negative SuperSpeed transmit differential pair
B25	USB_D_SSRX_P	USB1_RX1_P	B16	I	Positive SuperSpeed receive differential pair
B26	USB_D_SSRX_N	USB1_RX1_N	A17	I	Negative SuperSpeed receive differential pair

¹ The USB_C_D_P and USB_C_D_N signals are connected to the USB 1 or USB 2 through a USB 2.0 multiplexer. For the recovery mode, the USB_A_D_P and USB_A_D_N signals must be connected to the USB 1 of the SoC.

² Not connected directly to the SoC.

³ Connected to the mandatory GPIO A (IC16) of the OSM.

⁴ Available when not in recovery mode. The USB_C_D_P and USB_C_D_N signals are connected to the USB 1 of the SoC via an USB 2.0 multiplexer.

⁵ The USB_D interface is a companion port of the USB_C interface and not an independent USB interface. It is only available when the USB_C interface is in host mode.

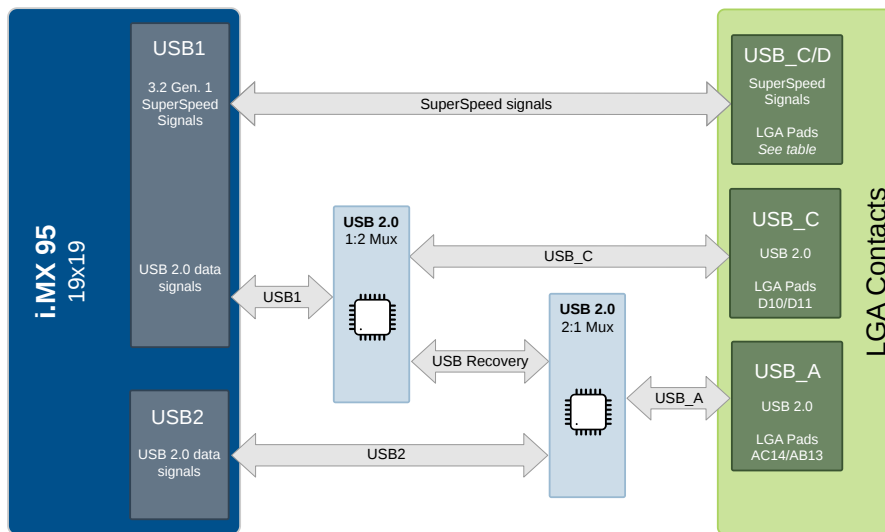
5.15.1 Recovery Mode

Only the **USB1** interface of the i.MX 95 SoC supports Serial Download Mode (recovery mode). However, the OSM specification defines the **USB_A** interface as the recovery interface exposed by the module. To satisfy both requirements, the module temporarily routes the SoC **USB1** interface to the **USB_A** pins during reset, as illustrated in Figure 5.

Immediately after reset is released, a GPIO expander controls the analog switches to restore the normal USB routing, connecting **USB1** to the **USB_C** interface and **USB2** to the **USB_A** interface. This automatic switching ensures OSM-compliant recovery operation while allowing the USB interfaces to operate in

their intended configuration during normal system operation.

Figure 5: USB block diagram



For more information about the Recovery Mode, refer to [Section 7](#).

6 Low Power Modes



TBA

More information about low power modes is planned for a future release of the datasheet.

7 Recovery Mode

The **Recovery Mode (USB Serial Loader)** can be used to download new software to the OSM iMX95 even when the bootloader is no longer capable of booting the module. In the normal development process, this mode is not needed. When the module is in recovery mode (for more details, refer to [Section 5.15.1](#)), the **USB1** interface of the SoC is used to connect it to a host computer. Additional information is available at the Toradex Developer Center:

<https://developer.toradex.com/hardware/hardware-resources/recovery-mode/imx-ti-recovery-mode>

7.1 Boot Mode Selection and Pin Mapping

On the OSM iMX95 module, the boot mode is defined by the **BOOT_MODE[3:0]** strapping pins of the i.MX 95 SoC. These pins are multiplexed with **UART** and **SPI** transmit signals and are available on the module.

7.1.1 Pin Mapping

- **BOOT_MODE0** is connected to **UART_CON_TX** (LGA pad **D23**) and includes a 10 kΩ pull-down resistor
- **BOOT_MODE1** is connected to **UART_D_TX** (LGA pad **C23**) and includes a 10 kΩ pull-down resistor
- **BOOT_MODE2** is connected to **SPI_B_CS1#** (LGA pad **L17**) and includes a 10 kΩ pull-down resistor
- **BOOT_MODE3** is connected to **SPI_B_SCK** (LGA pad **Y21**) and includes a 10 kΩ pull-up resistor to 1.8 V

The **BOOT_MODE[3:0]** pins are sampled during Power-On Reset (POR) or warm reset and latched internally for the duration of the boot sequence. The default boot configuration is defined by external pull-up or pull-down resistors on the carrier board. On the OSM iMX95 module, the default boot mode is **1000** (Boot from Fuses). Additional supported boot mode configurations are summarized in [Table 40](#).

Table 40: Boot modes

BOOT_MODE[3:0]	Core	Boot Device
Core A55		
0000	A55	Boot from Fuses
0001	A55	USB Serial Downloader
0010	A55	USDHC1 (eMMC)
0011	A55	USDHC2 (SD 4-bit)
0100	A55	FlexSPI Serial NOR
0101	A55	FlexSPI Serial NAND 2K page
0110	A55	FlexSPI Serial NAND 4K page
0111	A55	Reserved
Core M33		
1000	M33	Boot from Fuses (default)
1001	M33	USB Serial Downloader
1010	M33	USDHC1 (eMMC)
1011	M33	USDHC2 (SD 4-bit)
1100	M33	FlexSPI Serial NOR
1101	M33	FlexSPI Serial NAND 2K page
1110	M33	FlexSPI Serial NAND 4K page
1111	M33	Reserved

To ensure correct boot mode detection, external circuitry connected to the `BOOT_MODE[3:0]` pins shall not actively drive these signals until the boot mode has been sampled and latched by the SoC. After the boot mode has been latched, the pins assume their multiplexed UART or SPI transmit functions and may be used normally by the application.

When `BOOT_MODE[3:0]` is set to `1001`, or alternatively when the `FORCE_RECOVERY#` signal is asserted during reset, the internal ROM bootloader bypasses all external boot devices and enters **USB Serial Downloader (Serial Loader) mode**, enabling firmware download and system recovery over USB.

7.1.2 Recovery Mode Control

The dedicated recovery input `FORCE_RECOVERY#` (LGA pad `T17`) is standardized by the OSM specification and provides an alternative mechanism to enter USB Serial Downloader mode. When asserted low during initial power-on (cold boot) with a resistance of $\leq 1\text{ k}\Omega$, this signal internally overrides `BOOT_MODE0` from `0` to `1`. The remaining boot mode pins (`BOOT_MODE[3:1]`) shall remain in their default state (`100`), as established by the module's internal pull resistors, and shall not be driven externally. It is recommended to provide at least a test point on the carrier board connected to pad `T17` to allow manual access to recovery mode. No external pull-up resistor is required on the carrier board.



Boot from SD Card

When this mode is selected, the ROM bootloader does not attempt to boot from an external storage device such as the SD Card and instead initializes the USB interface for firmware download.

7.2 Recovery Procedure

1. Power off the system.
2. Connect the USB OTG/device port of the carrier board to a host PC.
3. Assert the recovery signal by pulling `FORCE_RECOVERY#` (OSM LGA pad `T17`) **low** (active-low) during the initial power-on sequence. Ensure the signal is held low throughout reset and early boot.
4. Power on the system. The module enumerates as a USB device and is controlled by the internal ROM bootloader.
5. Download and execute the required firmware image using the appropriate host tools (refer to the [Toradex Developer Website³](https://developer.toradex.com/)).
6. Power off the system and deassert `FORCE_RECOVERY#`.
7. Power on the system to resume normal boot operation.

³<https://developer.toradex.com/>

8 Known Issues



TBA

More information about known issues is planned for a future release of the datasheet.

9 Technical Specification

9.1 Electrical Characteristics

This section describes the **electrical characteristics and power-related specifications** of the OSM iMX95 System-on-Module. It covers the integrated power management architecture, absolute maximum ratings, recommended operating conditions, and power sequencing requirements necessary to ensure reliable operation, compliance with device limits, and long-term system stability. The information provided is intended to support system design, power budgeting, and validation of carrier boards and end applications using the module.

9.1.1 Absolute Maximum Ratings

Table 41 shows the absolute maximum ratings of the module. Functional operation at or beyond these limits is not guaranteed and may result in permanent damage.

Table 41: Absolute maximum ratings

Symbol	Description	Minimum	Maximum	Unit
VCC	Main power supply	-0.3	6.0	V
VCC_BACKUP	RTC power supply	-0.3	6.0	V
IO_1.8V	SoC IO pins with 1.8 V logic level	-0.3	2.1	V
ADC	ADC analog input	-0.3	3.6	V
USBx_VBUS	USB VBUS input detected	-0.3	3.96	V

Notes

- Table 41 shows stress limits only.
- ADC inputs must not exceed the analog supply ($V_{DD_ANA_1P8}$) + 0.3 V.
- USB_VBUS is sense-only on the i.MX 95 SoC.

9.1.2 Recommended Operating Conditions

The recommended operating conditions for normal operation and long-term reliability are listed in Table 42. Functional operation outside recommended conditions is not guaranteed.

Table 42: Recommended operating conditions

Symbol	Description	Minimum	Typical	Maximum	Unit
VCC	Main power supply	4.75	5.00	5.25	V
VCC_BACKUP	RTC power supply	1.10	3.00	5.50	V

9.1.3 PMIC – Power Management IC

The OSM iMX95 system-on-module integrates a multi-device power management architecture based on three dedicated PMICs: the MPF0900, MPF5301, and MPF5302. Together, these devices provide all power rails required by the NXP i.MX 95 system-on-chip and the on-module peripherals while ensuring proper power sequencing, voltage monitoring, fault handling, and low-power operation.

The MPF0900 serves as the primary system PMIC and generates the majority of the module supply rails, including the I/O, memory, analog, and auxiliary power domains. It integrates multiple buck converters,

low-dropout regulators (LDOs), GPIO-controlled power management functions, and supervisory features required for the i.MX 95 power architecture. The device also manages the system reset signals and coordinates the overall power-up and power-down sequence.

To support the high-current dynamic core supplies required by the i.MX 95, two additional dedicated regulators are used. The `MPF5302` generates the `VDD_SOC` rail powering the SoC core domain, while the `MPF5301` generates the `VDD_ARM` rail supplying the Arm Cortex-A55 processing subsystem. These regulators support dynamic voltage scaling and fast transient response, enabling efficient operation across the various performance and low-power modes of the processor (for more information about the power modes, refer to the Power Modes section of the i.MX 95 datasheet).

The three PMIC devices are interconnected and controlled through a dedicated internal I²C bus (`PMIC_I2C`). The primary PMIC supervises the enable signals of the auxiliary regulators and coordinates the complete power sequencing process. This architecture allows software-controlled voltage management, power-state transitions, brown-out detection, fault reporting, and system reset generation.

The PMIC devices are accessible through the internal `PMIC_I2C` interface using the following I²C addresses:

- `0x08`: `MPF0900` (Primary PMIC)
- `0x29`: `MPF5302` (`VDD_SOC` regulator)
- `0x2A`: `MPF5301` (`VDD_ARM` regulator)

9.1.4 Power Sequence

The power-up and power-down behavior of the i.MX 95 is defined by a strict sequencing of supply rails and control signals to ensure correct initialization, stable operation, and reliable shutdown of the device. This sequence is managed in coordination with the PMIC and the SoC's Battery-Backed State Machine (BBSM), governing transitions between the secure always-on state, power-up, run, and power-down modes.

For more information about the power-up and power-down sequences, refer to the [datasheet of the PMIC `MPF0900AVNA1ESR2`](#).

9.1.5 Watchdog Signal

The OSM iMX95 system-on-module implements a hardware watchdog supervision mechanism based on the interaction between the NXP i.MX 95 SoC and the on-module PMIC. The watchdog function utilizes the dedicated `PMIC_WDOG` signal, connected to ball `J45` of the i.MX 95 and to the `FCCU1` input (pin 47) of the `MPF0900` PMIC. In addition, the PMIC monitors the `RESET_IN#` signal available on OSM pad `U17`, and forms part of the system supervision and recovery infrastructure.

`PMIC_WDOG` is an **active-low watchdog output** from the SoC to the PMIC. When asserted low, the PMIC initiates a cold reset of the system.

If the PMIC detects that `PMIC_WDOG` is not asserted or toggled within the configured timeout window, it interprets this as a watchdog fault and initiates a corrective action, typically forcing a system reset by asserting the appropriate reset signals to the SoC. This mechanism enables recovery from software lockups or abnormal operating conditions.

9.2 Mechanical Characteristics

9.2.1 SoM Outline Dimensions

- 45 mm x 45 mm



TBA

Mechanical drawings are planned for a future release of the datasheet.

9.2.2 Thermal Specifications



TBA

Thermal specifications are planned for a future release of the datasheet.

10 Soldering and Assembly



TBA

More information about soldering and assembling the module is planned for a future release of the datasheet.

11 OSM Size-L Adapter

The OSM iMX95 System-on-Module is compliant with the **OSM Size-L specification** and exposes its functionality through the standardized OSM pad array (**X1**). Samples purchased for evaluation are delivered pre-soldered onto a dedicated adapter board, which is intended for evaluation and translates the OSM pad matrix to two high-density 400-contact board-to-board connectors. This setup ensures out-of-the-box compatibility with Toradex OSM evaluation board, enabling immediate evaluation without additional hardware.

The adapter mates with two **Samtec APF6-100-03.5-L-04-2-A-TR** connectors mounted on the carrier board. These connectors are the recommended mating connectors for custom carrier board designs and interface with the module's onboard **Samtec APM6-100-06.5-L-04-0-A-TR** connectors.

The connector system provides:

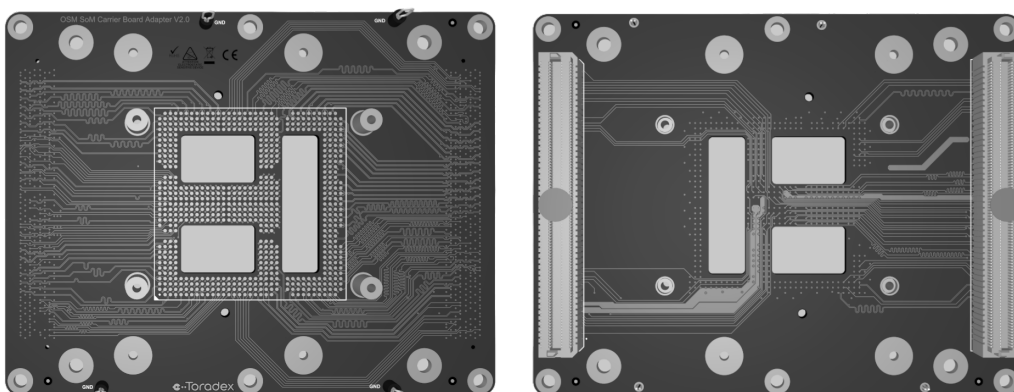
- Recommended carrier board connector: **Samtec APF6-100-03.5-L-04-2-A-TR**
- Module connector: **Samtec APM6-100-06.5-L-04-0-A-TR**
- AcceleRate® HP high-performance array interconnect
- 0.635 mm contact pitch
- 10 mm board-to-board stacking height
- Surface-mount termination
- Gold-plated contacts

These connectors establish a robust and electrically optimized interface between the OSM module and the development board. Their architecture supports the routing of all module power domains, high-speed serial interfaces, control signals, and general-purpose I/O, while maintaining controlled impedance and minimizing signal discontinuities across the interconnect boundary.

The breakout architecture maintains **one-to-one traceability** between each OSM pad (**X1**) and its corresponding adapter connector position (**M1 / M2**). This structured mapping simplifies schematic review, layout validation, and debugging. The connector topology and signal grouping preserve signal-integrity constraints for high-speed and impedance-controlled interfaces such as PCIe, USB, SDIO, RGMII, and MIPI, ensuring compliance with high-speed design requirements.

A mechanical reference view of the adapter board with the mounted OSM iMX95 module is provided in [Figure 6](#).

Figure 6: Mechanical reference view (left: top view, right: bottom view)



For clarity and maintainability, the complete OSM pad breakout is organized into **eight connector sec-**

tions corresponding to the two 400-pin connectors, as shown in Tables 43 through 50.

Each section is documented in a dedicated table specifying:

- Adapter connector pad (M1 / M2)
- Board-to-board connector reference
- OSM pad designation (X1)
- OSM signal function

11.1 Ground and Power Net Mapping Considerations

11.1.1 Ground (GND)

Ground connections on the adapter do not exhibit a strict one-to-one mapping between individual M1 / M2 pins and single OSM pads. Instead, multiple connector pins are tied to the common ground plane, which in turn connects to multiple OSM pads assigned to the GND net.

These pads are electrically shorted through the module's ground plane and distributed across both high-density connectors to ensure low impedance return paths, uniform current distribution, and compliance with high-speed signal integrity requirements.

11.1.2 Power Rails

Similar to ground, power rails are distributed across multiple connector pins and multiple OSM pads per rail. Individual connector pins do not correspond uniquely to a single OSM pad; rather, they are tied to shared power nets within the module.

The distribution of supply rails across multiple OSM pads improves current-carrying capability, and enhances thermal performance. This architecture also supports proper decoupling and stable operation of high-speed and high-current subsystems.

11.2 OSM to Development Board Adapter Pin Assignment

Table 43: OSM adapter connector M1 row A

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
A1	GND	– ¹	GND	–
A2	RGB_R4/LVDS_CLK_P	Y5	RGB_R4/LVDS_CLK_P	–
A3	RGB_R2/LVDS_CLK_N	Y6	RGB_R2/LVDS_CLK_N	–
A4	GND	– ¹	GND	–
A5	RGB_G4	T3	RGB_G4	–
A6	RGB_B1	R3	RGB_B1	–
A7	GND	– ¹	GND	–
A8	RGB_B2	P3	RGB_B2	–
A9	RGB_B3	N3	RGB_B3	–
A10	GND	– ¹	GND	–
A11	RGB_B5	M3	RGB_B5	–
A12	RGB_VSYNC	L3	RGB_VSYNC	–
A13	RGB_RESET#	J3	RGB_RESET#	–

Continued on next page

Table 43: OSM adapter connector M1 row A (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
A14	GND	– ¹	GND	–
A15	RGB_CS#	H3	RGB_CS#	–
A16	ETH_B_(R)(G)(M)II_CRS	D2	ETH_B_(R)(G)(M)II_CRS	–
A17	GND	– ¹	GND	–
A18	CSI_A_DATA0_N	C1	CSI_A_DATA0_N	–
A19	CSI_A_DATA0_P	B1	CSI_A_DATA0_P	–
A20	GND	– ¹	GND	–
A21	CSI_A_CLOCK_N	B3	CSI_A_CLOCK_N	–
A22	CSI_A_CLOCK_P	B4	CSI_A_CLOCK_P	–
A23	GND	– ¹	GND	–
A24	CSI_A_DATA3_N	B6	CSI_A_DATA3_N	–
A25	CSI_A_DATA3_P	B7	CSI_A_DATA3_P	–
A26	GND	– ¹	GND	–
A27	ETH_A_(S)(R)(G)(M)II_RXD0	K15	ETH_A_(S)(R)(G)(M)II_RXD0	–
A28	ETH_A_(S)(R)(G)(M)II_RXD1	L15	ETH_A_(S)(R)(G)(M)II_RXD1	–
A29	GND	– ¹	GND	–
A30	USB_C_SSRX_P	B10	USB_C_SSRX_P	–
A31	USB_C_SSRX_N	B11	USB_C_SSRX_N	–
A32	GND	– ¹	GND	–
A33	USB_C_D_P	D10	USB_C_D_P	–
A34	USB_C_D_N	D11	USB_C_D_N	–
A35	GND	– ¹	GND	–
A36	UART_B_RX	D14	UART_B_RX	–
A37	UART_A_RTS	C13	UART_A_RTS	–
A38	GND	– ¹	GND	–
A39	UART_A_TX	B13	UART_A_TX	–
A40	UART_A_CTS	C14	UART_A_CTS	–
A41	GND	– ¹	GND	–
A42	CH1_RXC	C15	CH1_RXC	Not connected ²
A43	UART_A_RX	A14	UART_A_RX	–
A44	GND	– ¹	GND	–
A45	I2S_B_LRCLK	T18	I2S_B_LRCLK	–
A46	I2S_B_DATA_OUT	W19	I2S_B_DATA_OUT	–
A47	I2S_B_DATA_IN	V19	I2S_B_DATA_IN	–
A48	GND	– ¹	GND	–
A49	UART_B_CTS	D16	UART_B_CTS	–
A50	GPIO_A_1	E17	GPIO_A_1	–
A51	GPIO_A_0	D17	GPIO_A_0	–
A52	GND	– ¹	GND	–

Continued on next page

Table 43: OSM adapter connector M1 row A (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
A53	CH1_TXC	C17	CH1_TXC	Not connected ²
A54	GPIO_A_5	– ¹	GND	–
A55	GND	– ¹	GND	–
A56	CH1_SYNC_TRIGGER	B18	CH1_SYNC_TRIGGER	Not connected ²
A57	CH0_SYNC_TRIGGER	A18	CH0_SYNC_TRIGGER	Not connected ²
A58	GND	– ¹	GND	–
A59	CH0_RX_N	A19	CH0_RX_N	Not connected ²
A60	CH0_RX_P	B19	CH0_RX_P	Not connected ²
A61	GND	– ¹	GND	–
A62	CH0_LINK	A20	CH0_LINK	Not connected ²
A63	CH0_ACT	B20	CH0_ACT	Not connected ²
A64	GND	– ¹	GND	–
A65	CH0_TX_N	A21	CH0_TX_N	Not connected ²
A66	CH0_TX_P	B21	CH0_TX_P	Not connected ²
A67	GND	– ¹	GND	–
A68	UART_C_RX	A22	UART_C_RX	–
A69	UART_D_RX	C22	UART_D_RX	–
A70	GND	– ¹	GND	–
A71	UART_C_TX	B23	UART_C_TX	–
A72	UART_D_TX	C23	UART_D_TX	–
A73	GND	– ¹	GND	–
A74	UART_CON_RX	D22	UART_CON_RX	–
A75	UART_CON_TX	D23	UART_CON_TX	–
A76	GND	– ¹	GND	–
A77	eDP_A_LANE0_P	A30	eDP_A_LANE0_P	Not connected ²
A78	eDP_A_LANE0_N	A31	eDP_A_LANE0_N	Not connected ²
A79	GND	– ¹	GND	–
A80	eDP_A_LANE2_P	A33	eDP_A_LANE2_P	Not connected ²
A81	eDP_A_LANE2_N	A34	eDP_A_LANE2_N	Not connected ²
A82	GND	– ¹	GND	–
A83	eDP_A_LANE3_P	B34	eDP_A_LANE3_P	Not connected ²
A84	eDP_A_LANE3_N	B35	eDP_A_LANE3_N	Not connected ²
A85	GND	– ¹	GND	–
A86	eDP_B_LANE0_P	D35	eDP_B_LANE0_P	Not connected ²
A87	eDP_B_LANE0_N	E35	eDP_B_LANE0_N	Not connected ²
A88	GND	– ¹	GND	–
A89	eDP_B_LANE2_P	G35	eDP_B_LANE2_P	Not connected ²
A90	eDP_B_LANE2_N	H35	eDP_B_LANE2_N	Not connected ²
A91	GND	– ¹	GND	–

Continued on next page

Table 43: OSM adapter connector M1 row A (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
A92	PCIe_B_HSO0_P	K35	PCIe_B_HSO0_P	–
A93	PCIe_B_HSO0_N	L35	PCIe_B_HSO0_N	–
A94	GND	– ¹	GND	–
A95	PCIe_B_HSI0_P	L34	PCIe_B_HSI0_P	–
A96	PCIe_B_HSI0_N	M34	PCIe_B_HSI0_N	–
A97	GND	– ¹	GND	–
A98	eDP_B_AUX_P	G33	eDP_B_AUX_P	Not connected ²
A99	eDP_B_AUX_N	H33	eDP_B_AUX_N	Not connected ²
A100	GND	– ¹	GND	–

¹ Connected to ground net and not associated with any specific pin.

² Not connected on the module side.

Table 44: OSM adapter connector M1 row B

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
B1	GND	– ¹	GND	–
B2	RGB_G0/LVDS_D3_N	W4	RGB_G0	–
B3	RGB_G2/LVDS_D3_P	V4	RGB_G2	–
B4	GND	– ¹	GND	–
B5	RGB_G5/LVDS_D2_N	T4	RGB_G5	–
B6	RGB_B0/LVDS_D2_P	R4	RGB_B0	–
B7	GND	– ¹	GND	–
B8	RGB_B4/LVDS_D1_N	N4	RGB_B4	–
B9	RGB_(PIXEL)CLK/LVDS_D1_P	M4	RGB_(PIXEL)CLK	–
B10	GND	– ¹	GND	–
B11	RGB_HSYNC	K3	RGB_HSYNC	–
B12	GND	– ¹	GND	–
B13	RGB_DISP/LVDS_D0_N	K4	RGB_DISP	–
B14	RGB_DE/LVDS_D0_P	J4	RGB_DE	–
B15	GND	– ¹	GND	–
B16	CAM_A_PWR/ GPIO_C_6	G3	CAM_A_PWR/ GPIO_C_6	–
B17	CAM_MCK	C2	CAM_MCK	–
B18	GND	– ¹	GND	–
B19	CSI_A_DATA1_N	A2	CSI_A_DATA1_N	–
B20	CSI_A_DATA1_P	A3	CSI_A_DATA1_P	–
B21	GND	– ¹	GND	–
B22	CSI_A_DATA2_N	A5	CSI_A_DATA2_N	–
B23	CSI_A_DATA2_P	A6	CSI_A_DATA2_P	–
B24	GND	– ¹	GND	–
B25	USB_C_SSTX_N	A9	USB_C_SSTX_N	–
B26	USB_C_SSTX_P	A8	USB_C_SSTX_P	–
B27	GND	– ¹	GND	–
B28	USB_C_EN	C10	USB_C_EN	–
B29	GND	– ¹	GND	–
B30	ETH_A_(S)(R)(G)MII_TXD3	G16	ETH_A_(S)(R)(G)MII_TXD3	–
B31	ETH_A_(S)(R)(G)MII_TXD2	H16	ETH_A_(S)(R)(G)MII_TXD2	–
B32	GND	– ¹	GND	–
B33	ETH_A_(R)(G)MII_RXD3	P15	ETH_A_(R)(G)MII_RXD3	–
B34	UART_B_TX	D13	UART_B_TX	–
B35	GND	– ¹	GND	–
B36	USB_C_ID	D9	USB_C_ID	–
B37	Vendor Defined	P16	Vendor Defined	–
B38	JTAG_TDO(SWO)	R17	JTAG_TDO(SWO)	–
B39	GND	– ¹	GND	–

Continued on next page

Table 44: OSM adapter connector M1 row B (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
B40	SPI_B_CS1#/ GPIO_A_7	L17	SPI_B_CS1#/ GPIO_A_7	–
B41	FORCE_RECOVERY#	T17	FORCE_RECOVERY#	–
B42	ETH_IOPWR	M17	ETH_IOPWR	–
B43	GND	– ¹	GND	–
B44	CH1_RX_P	B15	CH1_RX_P	Not connected ³
B45	CH1_RX_N	A15	CH1_RX_N	Not connected ³
B46	GND	– ¹	GND	–
B47	NC	–	–	Not connected ³
B48	NC	–	–	Not connected ³
B49	CH1_LINK	A16	CH1_LINK	Not connected ³
B50	GND	– ¹	GND	–
B51	CH1_TX_P	B17	CH1_TX_P	Not connected ³
B52	CH1_TX_N	A17	CH1_TX_N	Not connected ³
B53	GND	– ¹	GND	–
B54	NC	–	–	Not connected ³
B55	NC	–	–	Not connected ³
B56	NC	–	–	Not connected ³
B57	GND	– ¹	GND	–
B58	PWM_2	G18	PWM_2	–
B59	PWM_1	F18	PWM_1	–
B60	GND	– ¹	GND	–
B61	DISP_BL_PWM/PWM_0	E18	DISP_BL_PWM/ PWM_0	–
B62	PWM_3	H18	PWM_3	–
B63	PWM_4	J18	PWM_4	–
B64	GND	– ¹	GND	–
B65	PWM_5	K18	PWM_5	–
B66	GPIO_B_5	J19	GPIO_B_5	–
B67	GND	– ¹	GND	–
B68	GPIO_B_4	H19	GPIO_B_4	–
B69	GPIO_B_0	D19	GPIO_B_0	–
B70	GND	– ¹	GND	–
B71	Vendor Defined	B22	Vendor Defined	–
B72	CH0_RXC	C19	CH0_RXC	Not connected ³
B73	GPIO_B_1	E19	GPIO_B_1	–
B74	GPIO_B_2	F19	GPIO_B_2	–
B75	GPIO_B_3	G19	GPIO_B_3	–
B76	GND	– ¹	GND	–
B77	NC	–	–	Not connected ³
B78	NC	–	–	Not connected ³

Continued on next page

Table 44: OSM adapter connector M1 row B (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
B79	NC	–	–	Not connected ³
B80	CH0_TXC	C21	CH0_TXC	Not connected ³
B81	VCC_8_TEST	– ²	VCC_8_TEST	0.5 V DDR test point
B82	GND	– ¹	GND	–
B83	eDP_A_LANE1_P	B31	eDP_A_LANE1_P	Not connected ³
B84	eDP_A_LANE1_N	B32	eDP_A_LANE1_N	Not connected ³
B85	GND	– ¹	GND	–
B86	eDP_B_LANE1_P	E34	eDP_B_LANE1_P	Not connected ³
B87	eDP_B_LANE1_N	F34	eDP_B_LANE1_N	Not connected ³
B88	GND	– ¹	GND	–
B89	NC	–	–	Not connected ³
B90	eDP_B_LANE3_P	H34	eDP_B_LANE3_P	Not connected ³
B91	eDP_B_LANE3_N	J34	eDP_B_LANE3_N	Not connected ³
B92	GND	– ¹	GND	–
B93	eDP_A_BL_PWM	C31	eDP_A_BL_PWM	Not connected ³
B94	eDP_A_AUX_SEL	D32	eDP_A_AUX_SEL	Not connected ³
B95	GND	– ¹	GND	–
B96	eDP_A_HPD	D33	eDP_A_HPD	Not connected ³
B97	eDP_B_BL_EN	E32	eDP_B_BL_EN	Not connected ³
B98	GND	– ¹	GND	–
B99	VCC_IN_5V	– ²	VCC_IN_5V	–
B100	VCC_IN_5V	– ²	VCC_IN_5V	–

¹ Connected to ground net and not associated with any specific pin.

² Connected to power net and not associated with any specific pin.

³ Not connected on the module side.

Table 45: OSM adapter connector M1 row C

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
C1	GND	– ¹	GND	–
C2	RGB_G1	V3	RGB_G1	–
C3	RGB_G3	U3	RGB_G3	–
C4	GND	– ¹	GND	–
C5	NC	–	–	Not connected ³
C6	NC	–	–	Not connected ³
C7	NC	–	–	Not connected ³
C8	NC	–	–	Not connected ³
C9	NC	–	–	Not connected ³
C10	GND	– ¹	GND	–
C11	ETH_B_(R)(G)MII_RX_ER	K2	ETH_B_(R)(G)MII_RX_ER	–
C12	ETH_B_(R)(G)MII_TX_CLK	H1	ETH_B_(R)(G)MII_TX_CLK	–
C13	GND	– ¹	GND	–
C14	ETH_B_(R)(G)MII_TX_EN(ER)	J2	ETH_B_(R)(G)MII_TX_EN(ER)	–
C15	ETH_B_(R)(G)MII_COL	E1	ETH_B_(R)(G)MII_COL	–
C16	GND	– ¹	GND	–
C17	NC	–	–	Not connected ³
C18	GND	– ¹	GND	–
C19	CAM_A_RST#/ GPIO_C_7	G4	CAM_A_RST#/ GPIO_C_7	–
C20	GPIO_C_0	D3	GPIO_C_0	–
C21	GPIO_C_3	E4	GPIO_C_3	–
C22	GND	– ¹	GND	–
C23	NC	–	–	Not connected ³
C24	NC	–	–	Not connected ³
C25	GND	– ¹	GND	–
C26	VCC_6_TEST	– ²	VCC_6_TEST	0.8 V test point
C27	NC	–	–	Not connected ³
C28	Vendor Defined	D6	Vendor Defined	–
C29	GND	– ¹	GND	–
C30	Vendor Defined	D7	Vendor Defined	–
C31	NC	–	–	Not connected ³
C32	GND	– ¹	GND	–
C33	USB_C_OC#	C8	USB_C_OC#	–
C34	USB_C_VBUS	C9	USB_C_VBUS	–
C35	GND	– ¹	GND	–
C36	NC	–	–	Not connected ³
C37	SPI_A_CS1#/ GPIO_A_6	K17	SPI_A_CS1#/ GPIO_A_6	–
C38	ETH_A_(R)(G)MII_RX_DV(ER)	M15	ETH_A_(R)(G)MII_RX_DV(ER)	–
C39	GND	– ¹	GND	–

Continued on next page

Table 45: OSM adapter connector M1 row C (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
C40	ETH_MDC	T16	ETH_MDC	-
C41	ETH_A_(R)(G)MII_CRS	E16	ETH_A_(R)(G)MII_CRS	-
C42	GND	- ¹	GND	-
C43	GPIO_A_4	H17	GPIO_A_4	-
C44	GPIO_A_3	G17	GPIO_A_3	-
C45	UART_B_RTS	D15	UART_B_RTS	-
C46	JTAG_TCK(SWCLK)	N17	JTAG_TCK(SWCLK)	-
C47	CH1_ACT	B16	CH1_ACT	Not connected ³
C48	GND	- ¹	GND	-
C49	JTAG_TDI	P17	JTAG_TDI	-
C50	Vendor Defined	C16	Vendor Defined	-
C51	JTAG_nTRST	R19	JTAG_nTRST	-
C52	JTAG_RTCK	P19	JTAG_RTCK	-
C53	GND	- ¹	GND	-
C54	JTAG_TMS(SWDIO)	N19	JTAG_TMS(SWDIO)	-
C55	NC	-	-	Not connected ³
C56	NC	-	-	Not connected ³
C57	VCC_2_TEST2	- ²	VCC_2_TEST2	PMIC test point
C58	NC	-	-	Not connected ³
C59	TEST_GENERIC	C18	TEST_GENERIC	PMIC test point
C60	GND	- ¹	GND	-
C61	GPIO_B_7	L19	GPIO_B_7	-
C62	GPIO_B_6	K19	GPIO_B_6	-
C63	GND	- ¹	GND	-
C64	NC	-	-	Not connected ³
C65	NC	-	-	Not connected ³
C66	NC	-	-	Not connected ³
C67	GND	- ¹	GND	-
C68	SDIO_A_D2	H20	SDIO_A_D2	-
C69	GND	- ¹	GND	-
C70	NC	-	-	Not connected ³
C71	NC	-	-	Not connected ³
C72	NC	-	-	Not connected ³
C73	NC	-	-	Not connected ³
C74	NC	-	-	Not connected ³
C75	SDIO_A_IOPWR	C20	SDIO_A_IOPWR	-
C76	SDIO_A_WP	D20	SDIO_A_WP	-
C77	GND	- ¹	GND	-
C78	SDIO_A_CMD	E20	SDIO_A_CMD	-

Continued on next page

Table 45: OSM adapter connector M1 row C (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
C79	SDIO_A_D0	G20	SDIO_A_D0	–
C80	GND	– ¹	GND	–
C81	SDIO_A_PWR_EN	D21	SDIO_A_PWR_EN	–
C82	SDIO_A_D3	H21	SDIO_A_D3	–
C83	GND	– ¹	GND	–
C84	USB_D_EN	C26	USB_D_EN	Not connected ³
C85	PCIe_B_PERST#	L33	PCIe_B_PERST#	Not connected ³
C86	NC	–	–	Not connected ³
C87	GND	– ¹	GND	–
C88	SPI_C_SCK	D29	SPI_C_SCK	Not connected ³
C89	SPI_C_SDO	D30	SPI_C_SDO	Not connected ³
C90	eDP_A_BL_EN	D31	eDP_A_BL_EN	Not connected ³
C91	GND	– ¹	GND	–
C92	VCC_OUT_IO	– ²	VCC_OUT_IO	–
C93	VCC_IN_5V	– ²	VCC_IN_5V	–
C94	VCC_IN_5V	– ²	VCC_IN_5V	–
C95	VCC_IN_5V	– ²	VCC_IN_5V	–
C96	VCC_IN_5V	– ²	VCC_IN_5V	–
C97	VCC_IN_5V	– ²	VCC_IN_5V	–
C98	VCC_IN_5V	– ²	VCC_IN_5V	–
C99	VCC_IN_5V	– ²	VCC_IN_5V	–
C100	VCC_IN_5V	– ²	VCC_IN_5V	–

¹ Connected to ground net and not associated with any specific pin.

² Connected to power net and not associated with any specific pin.

³ Not connected on the module side.

Table 46: OSM adapter connector M1 row D

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
D1	GND	– ¹	GND	–
D2	ETH_B_(R)(G)MII_RX_CLK	P1	ETH_B_(R)(G)MII_RX_CLK	–
D3	ETH_B_(R)(G)MII_RXD3	N1	ETH_B_(R)(G)MII_RXD3	–
D4	RESERVED	N2	RESERVED	–
D5	ETH_B_(R)(G)MII_RXD2	M1	ETH_B_(R)(G)MII_RXD2	–
D6	GND	– ¹	GND	–
D7	ETH_B_SDP	M2	ETH_B_SDP	–
D8	ETH_B_(R)(G)MII_RX_DV(ER)	L1	ETH_B_(R)(G)MII_RX_DV(ER)	–
D9	GND	– ¹	GND	–
D10	ETH_B_(S)(R)(G)MII_RXD1	K1	ETH_B_(S)(R)(G)MII_RXD1	–
D11	ETH_B_(S)(R)(G)MII_RXD0	J1	ETH_B_(S)(R)(G)MII_RXD0	–
D12	GND	– ¹	GND	–
D13	ETH_B_(S)(R)(G)MII_TXD0	G1	ETH_B_(S)(R)(G)MII_TXD0	–
D14	ETH_B_(S)(R)(G)MII_TXD1	F1	ETH_B_(S)(R)(G)MII_TXD1	–
D15	GND	– ¹	GND	–
D16	ETH_B_(S)(R)(G)MII_TXD2	G2	ETH_B_(S)(R)(G)MII_TXD2	–
D17	ETH_B_(S)(R)(G)MII_TXD3	F2	ETH_B_(S)(R)(G)MII_TXD3	–
D18	GND	– ¹	GND	–
D19	DISP_VDD_EN/ GPIO_C_4	F3	DISP_VDD_EN/ GPIO_C_4	–
D20	CAM_A_SDA/ CSI_A_TX_N	C3	CAM_A_SDA/ CSI_A_TX_N	–
D21	GPIO_C_2	E3	GPIO_C_2	–
D22	CAM_A_SCL/ CSI_A_TX_P	C4	CAM_A_SCL/ CSI_A_TX_P	–
D23	GND	– ¹	GND	–
D24	GPIO_C_1	D4	GPIO_C_1	–
D25	ETH_A_(R)(G)MII_TX_CLK	J15	ETH_A_(R)(G)MII_TX_CLK	–
D26	DISP_BL_EN/ GPIO_C_5	F4	DISP_BL_EN/ GPIO_C_5	–
D27	ETH_B_MDC	C6	ETH_B_MDC	–
D28	GND	– ¹	GND	–
D29	ETH_A_(R)(G)MII_TX_EN(ER)	K16	ETH_A_(R)(G)MII_TX_EN(ER)	–
D30	ETH_B_MDIO	C7	ETH_B_MDIO	–
D31	GND	– ¹	GND	–
D32	ETH_A_(R)(G)MII_RXD2	N15	ETH_A_(R)(G)MII_RXD2	–
D33	ETH_A_(R)(G)MII_RX_CLK	R15	ETH_A_(R)(G)MII_RX_CLK	–
D34	GND	– ¹	GND	–
D35	ETH_A_(R)(G)MII_RX_ER	L16	ETH_A_(R)(G)MII_RX_ER	–
D36	ETH_MDIO	T15	ETH_MDIO	–
D37	ETH_A_SDP	N16	ETH_A_SDP	–
D38	ETH_A_(R)(G)MII_COL	F15	ETH_A_(R)(G)MII_COL	–
D39	GND	– ¹	GND	–

Continued on next page

Table 46: OSM adapter connector M1 row D (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
D40	ETH_A_(S)(R)(G)MII_TXD1	G15	ETH_A_(S)(R)(G)MII_TXD1	-
D41	ETH_A_(S)(R)(G)MII_TXD0	H15	ETH_A_(S)(R)(G)MII_TXD0	-
D42	GND	- ¹	GND	-
D43	ADC_1	N18	ADC_1	-
D44	BOOT_SEL1#	R18	BOOT_SEL1#	-
D45	ADC_0	M18	ADC_0	-
D46	GND	- ¹	GND	-
D47	GPIO_A_2	F17	GPIO_A_2	-
D48	BOOT_SEL0#	U19	BOOT_SEL0#	-
D49	SDIO_B_WP	U20	SDIO_B_WP	-
D50	GND	- ¹	GND	-
D51	SDIO_B_PWR_EN	U21	SDIO_B_PWR_EN	-
D52	I2S_B_BITCLK	T19	I2S_B_BITCLK	-
D53	SDIO_B_IOPWR	T20	SDIO_B_IOPWR	-
D54	GND	- ¹	GND	-
D55	SDIO_B_D5	P20	SDIO_B_D5	-
D56	SDIO_B_D3	N20	SDIO_B_D3	-
D57	GND	- ¹	GND	-
D58	SDIO_B_CD#	T21	SDIO_B_CD#	-
D59	SDIO_B_D7	R21	SDIO_B_D7	-
D60	GND	- ¹	GND	-
D61	SDIO_B_D6	P21	SDIO_B_D6	-
D62	SDIO_B_D4	N21	SDIO_B_D4	-
D63	GND	- ¹	GND	-
D64	SDIO_B_D0	L20	SDIO_B_D0	-
D65	SDIO_B_CLK	K20	SDIO_B_CLK	-
D66	GND	- ¹	GND	-
D67	SDIO_B_D2	M21	SDIO_B_D2	-
D68	SDIO_B_D1	L21	SDIO_B_D1	-
D69	GND	- ¹	GND	-
D70	SDIO_B_CMD	K21	SDIO_B_CMD	-
D71	SDIO_A_CD#	J21	SDIO_A_CD#	-
D72	GND	- ¹	GND	-
D73	USB_D_D_P	D25	USB_D_D_P	Not connected ³
D74	USB_D_D_N	D26	USB_D_D_N	Not connected ³
D75	GND	- ¹	GND	-
D76	USB_D_SSTX_P	A27	USB_D_SSTX_P	-
D77	USB_D_SSTX_N	A28	USB_D_SSTX_N	-
D78	GND	- ¹	GND	-

Continued on next page

Table 46: OSM adapter connector M1 row D (Continued)

M1 pin	B2B function name	OSM pad	OSM function name	Remarks
D79	USB_D_SSRX_P	B25	USB_D_SSRX_P	Not connected ³
D80	USB_D_SSRX_N	B26	USB_D_SSRX_N	Not connected ³
D81	GND	– ¹	GND	–
D82	SDIO_A_CLK	F21	SDIO_A_CLK	–
D83	SDIO_A_D1	G21	SDIO_A_D1	–
D84	GND	– ¹	GND	–
D85	USB_D_ID	D27	USB_D_ID	Not connected ³
D86	USB_D_VBUS	C27	USB_D_VBUS	Not connected ³
D87	USB_D_OC#	C28	USB_D_OC#	Not connected ³
D88	GND	– ¹	GND	–
D89	SPI_C_SDI	C29	SPI_C_SDI	Not connected ³
D90	SPI_C_CS0#	C30	SPI_C_CS0#	Not connected ³
D91	GND	– ¹	GND	–
D92	eDP_A_AUX_P	C33	eDP_A_AUX_P	Not connected ³
D93	eDP_A_AUX_N	C34	eDP_A_AUX_N	Not connected ³
D94	GND	– ¹	GND	–
D95	eDP_B_BL_PWM	E33	eDP_B_BL_PWM	Not connected ³
D96	eDP_B_AUX_SEL	F32	eDP_B_AUX_SEL	Not connected ³
D97	eDP_B_HPD	G32	eDP_B_HPD	Not connected ³
D98	GND	– ¹	GND	–
D99	VCC_IN_5V	– ²	VCC_IN_5V	–
D100	VCC_IN_5V	– ²	VCC_IN_5V	–

¹ Connected to ground net and not associated with any specific pin.

² Connected to power net and not associated with any specific pin.

³ Not connected on the module side.

Table 47: OSM adapter connector M2 row A

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
A1	PWR_BTN#	AA9	PWR_BTN#	–
A2	PCIe_SM_ALERT#	R2	PCIe_SM_ALERT#	–
A3	RESET_IN#	U17	RESET_IN#	–
A4	GND	– ¹	GND	–
A5	USB_A_D_N	AB13	USB_A_D_N	–
A6	USB_A_D_P	AC14	USB_A_D_P	–
A7	GND	– ¹	GND	–
A8	CARRIER_PWR_EN	V17	CARRIER_PWR_EN	–
A9	RESET_OUT#	Y14	RESET_OUT#	–
A10	CARRIER_STBY#	Y13	CARRIER_STBY#	–
A11	RTC_PWR	W17	RTC_PWR	–
A12	VCC_IN_3V3	– ²	VCC_IN_3V3	–
A13	GND	– ¹	GND	–
A14	V_BAT	– ²	V_BAT	–
A15	GND	– ¹	GND	–
A16	PCIe_A_HSI0_P	AB1	PCIe_A_HSI0_P	–
A17	PCIe_A_HSI0_N	AB2	PCIe_A_HSI0_N	–
A18	GND	– ¹	GND	–
A19	PCIe_A_HSO0_P	AC2	PCIe_A_HSO0_P	–
A20	PCIe_A_HSO0_N	AC3	PCIe_A_HSO0_N	–
A21	GND	– ¹	GND	–
A22	CAN_B_RX	AB19	CAN_B_RX	–
A23	CAN_B_TX	AC19	CAN_B_TX	–
A24	CAN_A_TX	AC17	CAN_A_TX	–
A25	GND	– ¹	GND	–
A26	DEBUG_EN	AC18	DEBUG_EN	–
A27	USB_A_OC#	AC15	USB_A_OC#	–
A28	GND	– ¹	GND	–
A29	ETH_D_(R)(G)MII_RX_ER	AK2	ETH_D_(R)(G)MII_RX_ER	Not connected ³
A30	ETH_D_(R)(G)MII_RX_DV(ER)	AL1	ETH_D_(R)(G)MII_RX_DV(ER)	Not connected ³
A31	ETH_D_(R)(G)MII_RXD2	AM1	ETH_D_(R)(G)MII_RXD2	Not connected ³
A32	GND	– ¹	GND	–
A33	ETH_D_(R)(G)MII_RXD3	AN1	ETH_D_(R)(G)MII_RXD3	Not connected ³
A34	ETH_D_(R)(G)MII_RX_CLK	AP1	ETH_D_(R)(G)MII_RX_CLK	Not connected ³
A35	GND	– ¹	GND	–
A36	ETH_E_(S)(R)(G)MII_TXD0	AR3	ETH_E_(S)(R)(G)MII_TXD0	Not connected ³
A37	ETH_E_(S)(R)(G)MII_TXD1	AR4	ETH_E_(S)(R)(G)MII_TXD1	Not connected ³
A38	GND	– ¹	GND	–
A39	ETH_E_(S)(R)(G)MII_RXD0	AR6	ETH_E_(S)(R)(G)MII_RXD0	Not connected ³

Continued on next page

Table 47: OSM adapter connector M2 row A (Continued)

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
A40	ETH_E_(S)(R)(G)MII_RXD1	AR7	ETH_E_(S)(R)(G)MII_RXD1	Not connected ³
A41	GND	– ¹	GND	–
A42	ETH_E_(R)(G)MII_RX_ER	AP7	ETH_E_(R)(G)MII_RX_ER	Not connected ³
A43	ETH_E_(R)(G)MII_RX_DV(_ER)	AR8	ETH_E_(R)(G)MII_RX_DV(_ER)	Not connected ³
A44	GND	– ¹	GND	–
A45	ETH_E_(R)(G)MII_RXD2	AR9	ETH_E_(R)(G)MII_RXD2	Not connected ³
A46	ETH_E_(R)(G)MII_RXD3	AR10	ETH_E_(R)(G)MII_RXD3	Not connected ³
A47	GND	– ¹	GND	–
A48	LVDS_A_LANE2_N	AB5	LVDS_A_LANE2_N	Not connected ³
A49	LVDS_A_LANE2_P	AC5	LVDS_A_LANE2_P	Not connected ³
A50	GND	– ¹	GND	–
A51	LVDS_A_LANE0_N	AB11	LVDS_A_LANE0_N	Not connected ³
A52	LVDS_A_LANE0_P	AB10	LVDS_A_LANE0_P	Not connected ³
A53	GND	– ¹	GND	–
A54	PCIE_D_HSO0_N	AP20	PCIE_D_HSO0_N	Not connected ³
A55	PCIE_D_HSO0_P	AP21	PCIE_D_HSO0_P	Not connected ³
A56	GND	– ¹	GND	–
A57	PCIE_D_HSO1_N	AP23	PCIE_D_HSO1_N	Not connected ³
A58	PCIE_D_HSO1_P	AP24	PCIE_D_HSO1_P	Not connected ³
A59	GND	– ¹	GND	–
A60	PCIE_D_HSO2_N	AR27	PCIE_D_HSO2_N	Not connected ³
A61	PCIE_D_HSO2_P	AR28	PCIE_D_HSO2_P	Not connected ³
A62	GND	– ¹	GND	–
A63	PCIE_D_HSO3_N	AR30	PCIE_D_HSO3_N	Not connected ³
A64	PCIE_D_HSO3_P	AR31	PCIE_D_HSO3_P	Not connected ³
A65	GND	– ¹	GND	–
A66	PCIE_C_HSO0_N	AR33	PCIE_C_HSO0_N	Not connected ³
A67	PCIE_C_HSO0_P	AR34	PCIE_C_HSO0_P	Not connected ³
A68	GND	– ¹	GND	–
A69	PCIE_C_HSO1_N	AN34	PCIE_C_HSO1_N	Not connected ³
A70	PCIE_C_HSO1_P	AM34	PCIE_C_HSO1_P	Not connected ³
A71	GND	– ¹	GND	–
A72	UFS_TX1_N	AB30	UFS_TX1_N	Not connected ³
A73	UFS_TX1_P	AB29	UFS_TX1_P	Not connected ³
A74	GND	– ¹	GND	–
A75	UFS_RX1_P	AB32	UFS_RX1_P	Not connected ³
A76	UFS_RX1_N	AB33	UFS_RX1_N	Not connected ³
A77	GND	– ¹	GND	–
A78	USB_B_EN	AC20	USB_B_EN	–

Continued on next page

Table 47: OSM adapter connector M2 row A (Continued)

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
A79	USB_B_OC#	AC21	USB_B_OC#	–
A80	USB_B_VBUS	AB20	USB_B_VBUS	–
A81	USB_B_ID	AB22	USB_B_ID	–
A82	GND	– ¹	GND	–
A83	VCC_4_TEST	– ²	VCC_4_TEST	3.3 V test point
A84	VCC_7_TEST	– ²	VCC_7_TEST	Not connected ³
A85	GND	– ¹	GND	–
A86	GPIO_D_2	V32	GPIO_D_2	–
A87	GPIO_D_3	V33	GPIO_D_3	–
A88	GND	– ¹	GND	–
A89	SPI_B_CS0#	AA23	SPI_B_CS0#	–
A90	SPI_B_SCK	Y21	SPI_B_SCK	–
A91	GND	– ¹	GND	–
A92	SPI_B_SDI	Y22	SPI_B_SDI	–
A93	SPI_B_SDO	Y23	SPI_B_SDO	–
A94	I2S_A_LRCLK	W18	I2S_A_LRCLK	–
A95	I2S_A_BITCLK	W20	I2S_A_BITCLK	–
A96	I2S_A_DATA_OUT	W21	I2S_A_DATA_OUT	–
A97	GND	– ¹	GND	–
A98	CSI_B_CLOCK_P	K33	CSI_B_CLOCK_P	–
A99	CSI_B_CLOCK_N	J32	CSI_B_CLOCK_N	–
A100	GND	– ¹	GND	–

¹ Connected to ground net and not associated with any specific pin.

² Connected to power net and not associated with any specific pin.

³ Not connected on the module side.

Table 48: OSM adapter connector M2 row B

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
B1	GND	– ¹	GND	–
B2	RGB_R0	Y7	RGB_R0	–
B3	RGB_R3	AA5	RGB_R3	–
B4	GND	– ¹	GND	–
B5	PCIE_SMCLK	T1	PCIE_SMCLK	–
B6	PCIE_SMDAT	U1	PCIE_SMDAT	–
B7	PCIE_A_PERST#	V2	PCIE_A_PERST#	–
B8	PCIE_CLKREQ#	W2	PCIE_CLKREQ#	–
B9	GND	– ¹	GND	–
B10	PCIE_REFCLK_P	W1	PCIE_REFCLK_P	–
B11	PCIE_REFCLK_N	Y1	PCIE_REFCLK_N	–
B12	GND	– ¹	GND	–
B13	DSI_TE	AA3	DSI_TE	–
B14	RESERVED	AA2	RESERVED	–
B15	GND	– ¹	GND	–
B16	RESERVED	AA13	RESERVED	–
B17	I2C_B_SCL	AA20	I2C_B_SCL	–
B18	I2C_B_SDA	AA21	I2C_B_SDA	–
B19	GND	– ¹	GND	–
B20	DSI_DATA2_P/ LVDS_A_LANE2_P	AC5	DSI_DATA2_P/ LVDS_A_LANE2_P	–
B21	DSI_DATA2_N/ LVDS_A_LANE2_N	AC6	DSI_DATA2_N/ LVDS_A_LANE2_N	–
B22	GND	– ¹	GND	–
B23	ETH_D_(S)(R)(G)MII_TXD3	AF2	ETH_D_(S)(R)(G)MII_TXD3	Not connected ³
B24	ETH_D_(S)(R)(G)MII_TXD2	AG2	ETH_D_(S)(R)(G)MII_TXD2	Not connected ³
B25	GND	– ¹	GND	–
B26	ETH_D_(S)(R)(G)MII_TXD0	AF1	ETH_D_(S)(R)(G)MII_TXD0	Not connected ³
B27	ETH_D_(S)(R)(G)MII_TXD1	AG1	ETH_D_(S)(R)(G)MII_TXD1	Not connected ³
B28	GND	– ¹	GND	–
B29	RESERVED	AM6	RESERVED	Not connected ³
B30	RESERVED	AN7	RESERVED	Not connected ³
B31	RESERVED	AN8	RESERVED	Not connected ³
B32	RESERVED	AM10	RESERVED	Not connected ³
B33	GND	– ¹	GND	–
B34	RESERVED	AN2	RESERVED	Not connected ³
B35	LVDS_VDD_EN	AN14	LVDS_VDD_EN	–
B36	GND	– ¹	GND	–
B37	ETH_E_(R)(G)MII_TX_CLK	AR5	ETH_E_(R)(G)MII_TX_CLK	Not connected ³
B38	ETH_E_(R)(G)MII_TX_EN(ER)	AP6	ETH_E_(R)(G)MII_TX_EN(ER)	Not connected ³
B39	VCC_IN_5V	– ²	VCC_IN_5V	–

Continued on next page

Table 48: OSM adapter connector M2 row B (Continued)

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
B40	VCC_IN_5V	– ²	VCC_IN_5V	–
B41	VCC_IN_5V	– ²	VCC_IN_5V	–
B42	VCC_IN_5V	– ²	VCC_IN_5V	–
B43	GND	– ¹	GND	–
B44	ETH_E_SDP	AP9	ETH_E_SDP	Not connected ³
B45	LVDS_BL_PWM	AN22	LVDS_BL_PWM	Not connected ³
B46	LVDS_BL_EN	AN23	LVDS_BL_EN	Not connected ³
B47	GND	– ¹	GND	–
B48	RESERVED	AM25	RESERVED	Not connected ³
B49	RESERVED	AN25	RESERVED	Not connected ³
B50	GND	– ¹	GND	–
B51	NC	–	–	Not connected ³
B52	GND	– ¹	GND	–
B53	RESERVED	AM28	RESERVED	Not connected ³
B54	RESERVED	AN28	RESERVED	Not connected ³
B55	RESERVED	AM29	RESERVED	Not connected ³
B56	GND	– ¹	GND	–
B57	RESERVED	AM31	RESERVED	Not connected ³
B58	RESERVED	AN31	RESERVED	Not connected ³
B59	NC	–	–	Not connected ³
B60	GND	– ¹	GND	–
B61	PCIe_D_PERST#	AN32	PCIe_D_PERST#	Not connected ³
B62	Vendor Defined	AM32	Vendor Defined	–
B63	Vendor Defined	AL32	Vendor Defined	–
B64	Vendor Defined	AL33	Vendor Defined	–
B65	GND	– ¹	GND	–
B66	GPIO_E_6	AJ32	GPIO_E_6	Not connected ³
B67	GPIO_E_7	AJ33	GPIO_E_7	Not connected ³
B68	GPIO_E_4	AH32	GPIO_E_4	Not connected ³
B69	GND	– ¹	GND	–
B70	GPIO_E_0	AF32	GPIO_E_0	Not connected ³
B71	GPIO_E_1	AF33	GPIO_E_1	Not connected ³
B72	RESERVED	AE32	RESERVED	Not connected ³
B73	GND	– ¹	GND	–
B74	CAM_B_SCL/CSI_B_TX_P	AB25	CAM_B_SCL/CSI_B_TX_P	–
B75	CAM_B_SDA/CSI_B_TX_N	AB26	CAM_B_SDA/CSI_B_TX_N	–
B76	GND	– ¹	GND	–
B77	USB_B_D_P	AC22	USB_B_D_P	–
B78	USB_B_D_N	AB23	USB_B_D_N	–

Continued on next page

Table 48: OSM adapter connector M2 row B (Continued)

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
B79	GND	– ¹	GND	–
B80	UFS_RX0_P	AC31	UFS_RX0_P	Not connected ³
B81	UFS_RX0_N	AC32	UFS_RX0_N	Not connected ³
B82	GND	– ¹	GND	–
B83	ETH_C_(R)(G)MII_CRSS	AC34	ETH_C_(R)(G)MII_CRSS	Not connected ³
B84	ETH_C_(R)(G)MII_COL	AB35	ETH_C_(R)(G)MII_COL	Not connected ³
B85	NC	–	–	Not connected ³
B86	GND	– ¹	GND	–
B87	ETH_C_(R)(G)MII_RX_CLK	W35	ETH_C_(R)(G)MII_RX_CLK	Not connected ³
B88	ETH_C_(R)(G)MII_TX_EN(ER)	V34	ETH_C_(R)(G)MII_TX_EN(ER)	Not connected ³
B89	GND	– ¹	GND	–
B90	Vendor Defined	Y29	Vendor Defined	Not connected ³
B91	Vendor Defined	Y30	Vendor Defined	Not connected ³
B92	GND	– ¹	GND	–
B93	Vendor Defined	Y31	Vendor Defined	Not connected ³
B94	ETH_CDE_MDIO	P33	ETH_CDE_MDIO	Not connected ³
B95	ETH_C_SDP	R34	ETH_C_SDP	Not connected ³
B96	GND	– ¹	GND	–
B97	ETH_CDE_MDC	N33	ETH_CDE_MDC	Not connected ³
B98	ETH_C_(R)(G)MII_TX_CLK	N35	ETH_C_(R)(G)MII_TX_CLK	Not connected ³
B99	GND	– ¹	GND	–
B100	I2S_A_DATA_IN	V21	I2S_A_DATA_IN	–

¹ Connected to ground net and not associated with any specific pin.

² Connected to power net and not associated with any specific pin.

³ Not connected on the module side.

Table 49: OSM adapter connector M2 row C

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
C1	GND	– ¹	GND	–
C2	VCC_5_TEST	Y3	VCC_5_TEST	1.8 V BBSM
C3	SPI_A_SDI_(IO1)	U15	SPI_A_SDI_(IO1)	–
C4	SPI_A_SDO_(IO0)	V15	SPI_A_SDO_(IO0)	–
C5	SPI_A_HOLD_(IO3)	W15	SPI_A_HOLD_(IO3)	–
C6	GND	– ¹	GND	–
C7	SPI_A_SCK	U16	SPI_A_SCK	–
C8	I2S_MCLK	V18	I2S_MCLK	–
C9	NC	–	–	Not connected ²
C10	NC	–	–	Not connected ²
C11	GND	– ¹	GND	–
C12	SPI_A_CS0#	Y15	SPI_A_CS0#	–
C13	SPI_A_WP_(IO2)	W16	SPI_A_WP_(IO2)	–
C14	VCC_3_TEST	Y16	VCC_3_TEST	PMIC test point
C15	GND	– ¹	GND	–
C16	ETH_D_(R)(G)MII_CRS	AE3	ETH_D_(R)(G)MII_CRS	Not connected ²
C17	ETH_D_(R)(G)MII_COL	AF3	ETH_D_(R)(G)MII_COL	Not connected ²
C18	RESERVED	AL3	RESERVED	Not connected ²
C19	GND	– ¹	GND	–
C20	RESERVED	AL4	RESERVED	Not connected ²
C21	RESERVED	AM3	RESERVED	Not connected ²
C22	RESERVED	AM4	RESERVED	Not connected ²
C23	GND	– ¹	GND	–
C24	ETH_E_(S)(R)(G)MII_TXD2	AP3	ETH_E_(S)(R)(G)MII_TXD2	Not connected ²
C25	ETH_E_(S)(R)(G)MII_TXD3	AP4	ETH_E_(S)(R)(G)MII_TXD3	Not connected ²
C26	GND	– ¹	GND	–
C27	ETH_E_(R)(G)MII_CRS	AN4	ETH_E_(R)(G)MII_CRS	Not connected ²
C28	RESERVED	AM5	RESERVED	Not connected ²
C29	RESERVED	AN5	RESERVED	Not connected ²
C30	RESERVED	AM7	RESERVED	Not connected ²
C31	RESERVED	AM8	RESERVED	Not connected ²
C32	RESERVED	AM9	RESERVED	Not connected ²
C33	GND	– ¹	GND	–
C34	LVDS_I2C_CLK	AM11	LVDS_I2C_CLK	Not connected ²
C35	LVDS_I2C_DAT	AM12	LVDS_I2C_DAT	Not connected ²
C36	ETH_E_(R)(G)MII_COL	AR2	ETH_E_(R)(G)MII_COL	Not connected ²
C37	NC	–	–	Not connected ²
C38	GND	– ¹	GND	–
C39	LVDS_B_CLK_N	AN16	LVDS_B_CLK_N	Not connected ²

Continued on next page

Table 49: OSM adapter connector M2 row C (Continued)

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
C40	LVDS_B_CLK_P	AN17	LVDS_B_CLK_P	Not connected ²
C41	GND	– ¹	GND	–
C42	LVDS_B_LANE1_N	AN19	LVDS_B_LANE1_N	Not connected ²
C43	LVDS_B_LANE1_P	AN20	LVDS_B_LANE1_P	Not connected ²
C44	GND	– ¹	GND	–
C45	RESERVED	AM23	RESERVED	Not connected ²
C46	RESERVED	AP10	RESERVED	Not connected ²
C47	RESERVED	AM24	RESERVED	Not connected ²
C48	RESERVED	AN24	RESERVED	Not connected ²
C49	GND	– ¹	GND	–
C50	RESERVED	AM26	RESERVED	Not connected ²
C51	RESERVED	AN26	RESERVED	Not connected ²
C52	RESERVED	AM27	RESERVED	Not connected ²
C53	RESERVED	AN27	RESERVED	Not connected ²
C54	GND	– ¹	GND	–
C55	RESERVED	AN29	RESERVED	Not connected ²
C56	RESERVED	AM30	RESERVED	Not connected ²
C57	RESERVED	AN30	RESERVED	Not connected ²
C58	GND	– ¹	GND	–
C59	UFS_TX0_N	AC29	UFS_TX0_N	Not connected ²
C60	UFS_TX0_P	AC28	UFS_TX0_P	Not connected ²
C61	GND	– ¹	GND	–
C62	NC	–	–	Not connected ²
C63	Vendor Defined	AK32	Vendor Defined	Not connected ²
C64	Vendor Defined	AM33	Vendor Defined	Not connected ²
C65	Vendor Defined	AK33	Vendor Defined	Not connected ²
C66	GND	– ¹	GND	–
C67	NC	–	–	Not connected ²
C68	GPIO_E_5	AH33	GPIO_E_5	Not connected ²
C69	GPIO_E_2	AG32	GPIO_E_2	Not connected ²
C70	GPIO_E_3	AG33	GPIO_E_3	Not connected ²
C71	GND	– ¹	GND	–
C72	PCIe_C_PERST#	AE33	PCIe_C_PERST#	Not connected ²
C73	UFS_CLK	AC26	UFS_CLK	Not connected ²
C74	Vendor Defined	AA31	Vendor Defined	Not connected ²
C75	GND	– ¹	GND	–
C76	NC	–	–	Not connected ²
C77	NC	–	–	Not connected ²
C78	UFS_RESET#	AB27	UFS_RESET#	Not connected ²

Continued on next page

Table 49: OSM adapter connector M2 row C (Continued)

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
C79	GND	– ¹	GND	–
C80	GPIO_D_6	Y32	GPIO_D_6	Not connected ²
C81	SPI_C_CS1#/ GPIO_D_7	Y33	SPI_C_CS1#/ GPIO_D_7	Not connected ²
C82	Vendor Defined	AA29	Vendor Defined	Not connected ²
C83	Vendor Defined	AA30	Vendor Defined	Not connected ²
C84	GND	– ¹	GND	–
C85	GPIO_D_4	W32	GPIO_D_4	Not connected ²
C86	GPIO_D_5	W33	GPIO_D_5	Not connected ²
C87	GND	– ¹	GND	–
C88	GPIO_D_0	U32	GPIO_D_0	Not connected ²
C89	GPIO_D_1	U33	GPIO_D_1	Not connected ²
C90	GND	– ¹	GND	–
C91	CSI_B_DATA3_N	T32	CSI_B_DATA3_N	Not connected ²
C92	CSI_B_DATA3_P	T33	CSI_B_DATA3_P	Not connected ²
C93	GND	– ¹	GND	–
C94	NC	–	–	Not connected ²
C95	ETH_C_(R)(G)MII_RXD2	R35	ETH_C_(R)(G)MII_RXD2	Not connected ²
C96	ETH_C_(R)(G)MII_RXD3	P35	ETH_C_(R)(G)MII_RXD3	Not connected ²
C97	GND	– ¹	GND	–
C98	CSI_B_DATA0_P	M33	CSI_B_DATA0_P	Not connected ²
C99	CSI_B_DATA0_N	L32	CSI_B_DATA0_N	Not connected ²
C100	GND	– ¹	GND	–

¹ Connected to ground net and not associated with any specific pin.

² Not connected on the module side.

Table 50: OSM adapter connector M2 row D

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
D1	GND	– ¹	GND	–
D2	RGB_R5	Y4	RGB_R5	–
D3	RGB_R1	AA6	RGB_R1	–
D4	PCIe_WAKE#	T2	PCIe_WAKE#	–
D5	GND	– ¹	GND	–
D6	DSI_DATA3_P/ LVDS_A_LANE3_P	AB4	DSI_DATA3_P/ LVDS_A_LANE3_P	–
D7	DSI_DATA3_N/ LVDS_A_LANE3_N	AB5	DSI_DATA3_N/ LVDS_A_LANE3_N	–
D8	GND	– ¹	GND	–
D9	DSI_CLOCK_P/ LVDS_A_CLK_P	AB7	DSI_CLOCK_P/ LVDS_A_CLK_P	–
D10	DSI_CLOCK_N/ LVDS_A_CLK_N	AB8	DSI_CLOCK_N/ LVDS_A_CLK_N	–
D11	GND	– ¹	GND	–
D12	DSI_DATA1_P/ LVDS_A_LANE1_P	AC8	DSI_DATA1_P/ LVDS_A_LANE1_P	–
D13	DSI_DATA1_N/ LVDS_A_LANE1_N	AC9	DSI_DATA1_N/ LVDS_A_LANE1_N	–
D14	GND	– ¹	GND	–
D15	DSI_DATA0_P/ LVDS_A_LANE0_P	AB10	DSI_DATA0_P/ LVDS_A_LANE0_P	–
D16	DSI_DATA0_N/ LVDS_A_LANE0_N	AB11	DSI_DATA0_N/ LVDS_A_LANE0_N	–
D17	GND	– ¹	GND	–
D18	I2C_A_SCL	AA15	I2C_A_SCL	–
D19	I2C_A_SDA	AA16	I2C_A_SDA	–
D20	GND	– ¹	GND	–
D21	CAN_A_RX	AB17	CAN_A_RX	–
D22	USB_A_ID	AB14	USB_A_ID	–
D23	USB_A_VBUS	AB16	USB_A_VBUS	–
D24	USB_A_EN	AC16	USB_A_EN	–
D25	GND	– ¹	GND	–
D26	ETH_D_(R)(G)MII_TX_CLK	AH1	ETH_D_(R)(G)MII_TX_CLK	Not connected ²
D27	ETH_D_(R)(G)MII_TX_EN(ER)	AJ2	ETH_D_(R)(G)MII_TX_EN(ER)	Not connected ²
D28	GND	– ¹	GND	–
D29	ETH_D_(S)(R)(G)MII_RXD0	AJ1	ETH_D_(S)(R)(G)MII_RXD0	Not connected ²
D30	ETH_D_(S)(R)(G)MII_RXD1	AK1	ETH_D_(S)(R)(G)MII_RXD1	Not connected ²
D31	GND	– ¹	GND	–
D32	ETH_D_SDP	AM2	ETH_D_SDP	Not connected ²
D33	ETH_E_(R)(G)MII_RX_CLK	AN10	ETH_E_(R)(G)MII_RX_CLK	Not connected ²
D34	GND	– ¹	GND	–
D35	LVDS_A_CLK_N	AN12	LVDS_A_CLK_N	Not connected ²
D36	LVDS_A_CLK_P	AN13	LVDS_A_CLK_P	Not connected ²
D37	GND	– ¹	GND	–
D38	LVDS_B_LANE3_N	AM14	LVDS_B_LANE3_N	Not connected ²
D39	LVDS_B_LANE3_P	AM15	LVDS_B_LANE3_P	Not connected ²

Continued on next page

Table 50: OSM adapter connector M2 row D (Continued)

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
D40	GND	– ¹	GND	–
D41	LVDS_B_LANE2_N	AM17	LVDS_B_LANE2_N	Not connected ²
D42	LVDS_B_LANE2_P	AM18	LVDS_B_LANE2_P	Not connected ²
D43	GND	– ¹	GND	–
D44	LVDS_B_LANE0_N	AM20	LVDS_B_LANE0_N	Not connected ²
D45	LVDS_B_LANE0_P	AM21	LVDS_B_LANE0_P	Not connected ²
D46	GND	– ¹	GND	–
D47	LVDS_A_LANE3_N	AP11	LVDS_A_LANE3_N	Not connected ²
D48	LVDS_A_LANE3_P	AP12	LVDS_A_LANE3_P	Not connected ²
D49	GND	– ¹	GND	–
D50	LVDS_A_LANE1_N	AR15	LVDS_A_LANE1_N	Not connected ²
D51	LVDS_A_LANE1_P	AR16	LVDS_A_LANE1_P	Not connected ²
D52	GND	– ¹	GND	–
D53	PCIE_D_HSI0_N	AR18	PCIE_D_HSI0_N	Not connected ²
D54	PCIE_D_HSI0_P	AR19	PCIE_D_HSI0_P	Not connected ²
D55	GND	– ¹	GND	–
D56	PCIE_D_HSI1_N	AR21	PCIE_D_HSI1_N	Not connected ²
D57	PCIE_D_HSI1_P	AR22	PCIE_D_HSI1_P	Not connected ²
D58	GND	– ¹	GND	–
D59	PCIE_D_HSI2_N	AP26	PCIE_D_HSI2_N	Not connected ²
D60	PCIE_D_HSI2_P	AP27	PCIE_D_HSI2_P	Not connected ²
D61	GND	– ¹	GND	–
D62	PCIE_D_HSI3_N	AP29	PCIE_D_HSI3_N	Not connected ²
D63	PCIE_D_HSI3_P	AP30	PCIE_D_HSI3_P	Not connected ²
D64	GND	– ¹	GND	–
D65	PCIE_C_HSI0_N	AP32	PCIE_C_HSI0_N	Not connected ²
D66	PCIE_C_HSI0_P	AP33	PCIE_C_HSI0_P	Not connected ²
D67	GND	– ¹	GND	–
D68	PCIE_C_HSI1_N	AP35	PCIE_C_HSI1_N	Not connected ²
D69	PCIE_C_HSI1_P	AN35	PCIE_C_HSI1_P	Not connected ²
D70	GND	– ¹	GND	–
D71	PCIE_C_HSI2_N	AL35	PCIE_C_HSI2_N	Not connected ²
D72	PCIE_C_HSI2_P	AK35	PCIE_C_HSI2_P	Not connected ²
D73	GND	– ¹	GND	–
D74	PCIE_C_HSO2_N	AK34	PCIE_C_HSO2_N	Not connected ²
D75	PCIE_C_HSO2_P	AJ34	PCIE_C_HSO2_P	Not connected ²
D76	GND	– ¹	GND	–
D77	PCIE_C_HSI3_N	AH35	PCIE_C_HSI3_N	Not connected ²
D78	PCIE_C_HSI3_P	AG35	PCIE_C_HSI3_P	Not connected ²

Continued on next page

Table 50: OSM adapter connector M2 row D (Continued)

M2 pin	B2B function name	OSM pad	OSM function name	Remarks
D79	GND	– ¹	GND	–
D80	PCIe_C_HSO3_N	AG34	PCIe_C_HSO3_N	Not connected ²
D81	PCIe_C_HSO3_P	AF34	PCIe_C_HSO3_P	Not connected ²
D82	GND	– ¹	GND	–
D83	ETH_C_(S)(R)(G)MII_TXD3	AA34	ETH_C_(S)(R)(G)MII_TXD3	Not connected ²
D84	ETH_C_(S)(R)(G)MII_TXD2	Y34	ETH_C_(S)(R)(G)MII_TXD2	Not connected ²
D85	GND	– ¹	GND	–
D86	ETH_C_(S)(R)(G)MII_TXD1	AA35	ETH_C_(S)(R)(G)MII_TXD1	Not connected ²
D87	ETH_C_(S)(R)(G)MII_TXD0	Y35	ETH_C_(S)(R)(G)MII_TXD0	Not connected ²
D88	GND	– ¹	GND	–
D89	ETH_C_(S)(R)(G)MII_RXD0	V35	ETH_C_(S)(R)(G)MII_RXD0	Not connected ²
D90	ETH_C_(S)(R)(G)MII_RXD1	U35	ETH_C_(S)(R)(G)MII_RXD1	Not connected ²
D91	GND	– ¹	GND	–
D92	ETH_C_(R)(G)MII_RX_ER	U34	ETH_C_(R)(G)MII_RX_ER	Not connected ²
D93	ETH_C_(R)(G)MII_RX_DV(_ER)	T35	ETH_C_(R)(G)MII_RX_DV(_ER)	Not connected ²
D94	GND	– ¹	GND	–
D95	CSI_B_DATA2_P	R33	CSI_B_DATA2_P	Not connected ²
D96	CSI_B_DATA2_N	P34	CSI_B_DATA2_N	Not connected ²
D97	GND	– ¹	GND	–
D98	CSI_B_DATA1_P	P32	CSI_B_DATA1_P	Not connected ²
D99	CSI_B_DATA1_N	N32	CSI_B_DATA1_N	Not connected ²
D100	GND	– ¹	GND	–

¹ Connected to ground net and not associated with any specific pin.

² Not connected on the module side.

12 Product Compliance

Up-to-date information about product compliance, such as RoHS, CE, UL 94, Conflict Minerals, REACH, and others, can be found on [Toradex website](https://www.toradex.com/support/product-compliance)⁴.

⁴<https://www.toradex.com/support/product-compliance>

13 Device and Documentation Support

13.1 Trademarks

NXP is a trademark of NXP B.V. All other product or service names are the property of their respective owners. © 2023 NXP B.V.

Arm®, Cortex®, TrustZone® are registered trademarks or trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. Neon™ is a trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

MIPI® is a registered trademark of MIPI Alliance.

Open Standard Module™ (OSM™) is a trademark of the SGET (Standardization Group for Embedded Technologies) e.V.

The Bluetooth® word mark is a registered trademark owned by Bluetooth SIG, Inc.

Secure Digital® and SD® are registered trademarks of SD Card Association.

DISCLAIMER

Copyright © Toradex AG. All rights are reserved. The information and content in this document are provided “as-is” with no warranties of any kind and are for informational purposes only. Data and information have been carefully checked and are believed to be accurate; however, no liability or responsibility for any errors, omissions or inaccuracies is assumed.

Brand and product names are trademarks or registered trademarks of their respective owners. Specifications are subject to change without notice.