T264-BPMPFW-7.0: ODM Calibration Data (BPMP-DTB) Specification for Functional Safety

Version: Aug 26, 2024

SWE-BPMPFW-231-ICD | August 26, 2024
NVIDIA CONFIDENTIAL | Prepared and provided under NDA

All information is subject to change.

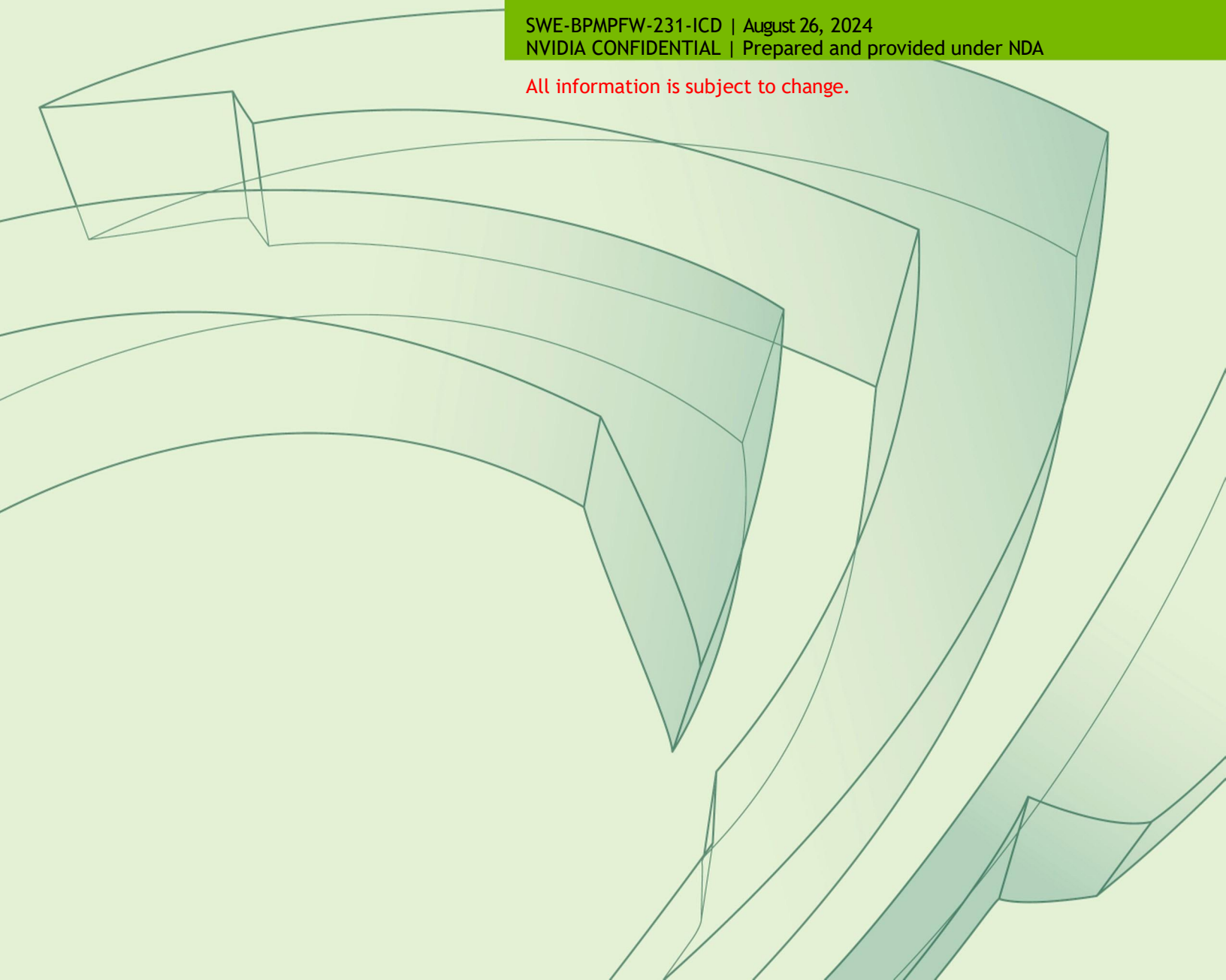

Table of Contents

Document ID	SWE-BPMPFW-231-ICD
Document Revision	Sep 20, 2024

Revision History

Date	Summary of Change
Jan 26, 2023	Initial draft
May 09, 2023	Add Sensor-Fwk binding definitions
Jun 30, 2023	Add osp-controller binding definitions
Sep 21, 2023	Add PCIe controller binding definitions
Apr 11, 2024	Add uphy binding definitions
Apr 19, 2024	Add nvtherm binding definitions
Apr 23, 2024	Separate BPMP-DTB Functional Safety specification to own document
Aug 26, 2024	Remove IO-DPD binding definitions
Sep 18, 2024	Update VRMON properties
Sep 20, 2024	Make pwm-rate a required property for VRS-11 regulators

1 Introduction

The Boot and Power Management Processor Firmware (BPMP-FW) is a platform- and os-agnostic binary executable. At boot, it accepts configuration data in the form of a device tree blob (DTB). A device tree blob describes a tree of named nodes. Each named node may have one or more named properties each of which may or may not have an associated value. To create a DTB file, compile a device tree source (DTS) file using a utility called device tree compiler (DTC).

The 2 Device Tree Bindings section describes how the BPMP-FW interprets the nodes and properties in a DTB. Each binding documentation describes how to convey some set of configuration data to the BPMP-FW. The binding documentation describes:

- The set of nodes that BPMP-FW interprets.
- The set of mandatory properties for those nodes.
- The set of optional properties for those nodes.
- The format and range of possible values for the properties.
- How the BPMP-FW interprets the node names, property names, and values.

1.1 Forwards and Backwards Compatibility

The bindings that the BPMP-FW understands may evolve from version to version. However, there are rules governing that evolution. The purpose of the rules is to ensure backward and forward compatibility between the versions of DTB and BPMP-FW as much as possible. A DTB created according to older binding documentation should work appropriately with a newer BPMP-FW (or vice versa).

1.2 Important Terms

This section defines important acronyms, abbreviations, and terms that are required to understand this document.

Term or Abbreviation	Definition
ACL	Access control list

2 Device Tree Bindings

This section describes the schema used for the DTS files, often referred to as the device tree bindings.

2.1 /clocks

The `clocks` node initializes clocks to a known state and allows overriding selected hard-coded clock configuration properties.

The node can contain following optional sub-nodes:

- `/clocks/init`
- `/clocks/clock@xxx`

Optional Properties

default-acl
: A 32-bit unsigned bitmask specifying the default Access Control List (ACL). This is the default value for `acl` property in each `/clock/clock@xxx` node.

2.1.1 /clocks/init

Contains one property for each clock that is to be initialized to non-default values. The value of each property is a list of four 32-bit unsigned integer elements. The property name is ignored, but for clarity it should be the same as the clock name.

The elements of the properties are:

- element 0: *Clock ID*
- element 1: *Parent clock ID*
- element 2: *Rate* (desired initial rate in Hz)
- element 3: *Enable mask*

The enable mask can be used to make a clock initially enabled on behalf of a given initiator. The bitmask is a bitwise OR of Doorbell IDs. The list of valid IDs is defined in section 2.19.3 Doorbell Identifiers

Enabling a clock on behalf of BPMP, that is, *Enable mask* set as TEGRA264_DB_BPMP, will keep the clock always ON.

If the value of element 1, 2, or 3 is 0, the corresponding clock setting is not changed. When a clock ID used more than once in the list of properties, the last use takes precedence.

Valid clock IDs are defined in section 2.19.2 Clock Identifiers.

2.1.2 /clocks/clock@xxx

Each `/clocks/clock@xxx` node contains properties to limit a single clock's possible parents and to limit its configurability (from the perspective of software on the CPU).

Required properties

clk-id
: A clock ID of the clock to be updated. If a clock ID does not refer to a valid clock, the node is ignored. When a clock ID appears more than once, the last use takes precedence.

Optional properties

acl
: 32-bit unsigned integer. Specifies ACL of this clock. This value is composed as a bitwise OR of Doorbell IDs. The valid values for IDs are defined in 2.19.3 Doorbell Identifiers. Each set bit indicates that the corresponding MRQ initiator is authorized to set this clock's rate or parent. If both this property and `/clocks/default-acl` are absent, only those peers preconfigured in the firmware are authorized.

disable-spread
: A 32-bit unsigned integer. Setting this property to non-zero value, disables spread spectrum of the target PLL clock. When the property is not present or if it is set to zero, spread on target PLL is enabled with spread settings preconfigured in the firmware.

  This property is applicable only to `TEGRA264_CLK_PLLC4` and `TEGRA264_CLK_SPPLL1`. When specified on other clocks, this property has no effect.

max-rate
: A 64-bit unsigned integer. A nonzero value indicates maximum limit BPMP-FW should clamp the clock's rate to. Multiple entries with "max-rate" root, and different arbitrary suffixes are combined to apply minimum nonzero clamp to the clock rate.

2.1.3 /clocks Example

Example of /clocks binding

```
/ {
    clocks {
        default-acl = <(TEGRA264_DB_CPU_NS | TEGRA264_DB_CPU_S>;
        init {
            i2c1 = <TEGRA264_CLK_SPI1 TEGRA264_CLK_PLLP_OUT0 0 0>;
            i2c2 = <TEGRA264_CLK_SPI2 TEGRA264_CLK_CLK_M 0 0>;
        };

        clock@emc {
            clk-id = <TEGRA264_CLK_EMC>;
            acl = <0>;
        };

        clock@pllc4 {
            clk-id = <TEGRA264_CLK_PLLC4>;
            acl = <TEGRA264_DB_CPU_NS>;
            disable-spread = <1>;
        };
    };
};
```

2.2 /diagnostics

The optional `/diagnostics` node controls the availability of BPMP-FW diagnostic features.

Optional Properties

level
: A 32-bit unsigned integer value that represents a maximum allowed level of diagnostics features to be enabled.

  This integer value controls the level of available BPMP-FW diagnostics functionality together with the PRODUCTION_MODE (also known as "NV production") and the SECURITY_MODE (also known as "ODM production") fuses.

  The value may be:

- 0: Diagnostics are always disabled and eliminated from the firmware.
- 1: Unprivileged diagnostics are available, but only if SECURITY_MODE fuse is not burned.
- 2: Privileged diagnostics are available, but only if PRODUCTION_MODE or SECURITY_MODE fuses are not burned.

If property is absent, the default diagnostic level is 2.

Example of /diagnostics binding

```
/ {
    diagnostics {
        level = <0>;
    };
};
```

2.3 /fmon

The optional `/fmon` node controls actions of frequency monitors (FMON). There should be an `fmon@xxx` subnode for each controlled FMON instance.

2.3.1 /fmon/fmon@xxx

Required properties

clk-id
: ID of the clock monitored by FMON. Valid IDs of the monitored clocks are specified by the 2.19.4 Monitored Clocks Identifiers list. Subnode with invalid ID is ignored.

Optional properties

fault-actions-disabled
: When this property is specified, faults detected by the FMON attached to the specified clock are not propagated to HSM. If this property is absent, sku-specific control of faults actions is followed.

2.3.2 /fmon Example

Example of /fmon binding

```
/ {
    fmon {
        fmon@bpmp_cpu {
            clk-id = <TEGRA264_CLK_BPMP_CPU>;
            fault-actions-disabled;
        };
        fmon@i2c5 {
            clk-id = <TEGRA264_CLK_I2C5>;
            fault-actions-disabled;
        };
        fmon@vic {
            clk-id = <TEGRA264_CLK_VIC>;
            fault-actions-disabled;
        };
    };
};
```

2.4 /fps

The `/fps` node enables platform specific I2C writes, GPIO toggles, and delays at specific moments during system power transitions. Its name is an acronym for “flexible power sequence.”

The node supports a set of optional `-sequence` properties. The value of such a property specifies a sequence of operations that the firmware is to take at an occurrence of the event indicated by the property name.

Optional properties

sc7-sequence
: Sequence run during SC7 entry just after the BPMP-FW has stopped accepting and processing service requests.

sc7-exit-sequence
: Sequence run during SC7 exit just before BPMP-FW starts accepting service requests.

shutdown-sequence
: Sequence run to finalize system shutdown.

2.4.1 Syntax for Defining a Sequence

Each sequence is a list of zero or more operations. Each operation is defined by a three-byte value. The following table describes the binary encoding of the operations.

Operation	Binary Encoding	Notes
I2C byte write	0aaaaaaa rrrrrrrr vvvvvvvv	a: 7-bit I2C bus address
		r: register address
		v: data byte
Delay	11111111 MMMMMMMM LLLLLLLL	M: MSB of microseconds to delay
		L: LSB of microseconds to delay
GPIO tristate	11111110 MMMMMMMM LLLLLLLL	M: MSB of GPIO identifier
		L: LSB of GPIO identifier
GPIO drive high	11111101 MMMMMMMM LLLLLLLL	M: MSB of GPIO identifier
		L: LSB of GPIO identifier
GPIO drive low	11111100 MMMMMMMM LLLLLLLL	M: MSB of GPIO identifier
		L: LSB of GPIO identifier
I2C multibyte write	11111011 000000ss vvvvvvvv	s: 0b01 = start
		s: 0b00 = continue
		s: 0b10 = stop
		v: I2C bus address / data

I2C writes are only done on the PWR_I2C bus. There is no support for other buses.

I2C multibyte writes have the following constraints.

- A transfer must begin with a *start* opcode. The data associated with the start condition contains the I2C bus address.
- A transfer must end with a *stop* opcode.
- Transfers must be at most eight bytes (excluding the I2C bus address).

- All operations between the *start* and *stop* conditions must be of the *I2C multibyte write* type.

If any of the preceding conditions are violated, boot will be halted.

Valid values for the GPIO identifiers are defined in section 2.19.1 GPIO Port Identifiers.

Example of /fps binding

```
/ {
    fps {
        sc7-sequence = [3c 22 44 ff 01 ff fc 00 3c];
        sc7-exit-sequence = [3c 3d 08];
        shutdown-sequence = [3c 42 87 3c 41 c0];
    };
};
```

Note:

The data in the above example will do the following.

- On SC7 entry
  - Write 0x44 to register 0x22 of slave with address 0x3c
  - Delay 511 microseconds
  - Drive GPIO 60 low
- On SC7 exit
  - Write 0x08 to register 0x3d of slave with address 0x3c
- On system shutdown request
  - Write 0x87 to register 0x42 of slave with address 0x3c
  - Write 0xc0 to register 0x41 of slave with address 0x3c

2.5 /i2c-busses/xxx

The `i2c-busses` node must contain a single freely named subnode which configures PWR_I2C controller managed by the BPMP.

Required properties

bus-clock
: Clock rate of the target clock for the I2C bus in hertz. For proper operation, the value must be less or equal to 1000000 (1,000,000).

Optional properties

tlow
: A timing parameter which influences the width of the low phase of a bus clock cycle. If omitted, defaults to HW recommended value based on bus-clock rate.

thigh
: A timing parameter which influences the width of the high phase of a bus clock cycle. If omitted, defaults to HW recommended value based on bus-clock rate.

Each controller subnode may have a subnode `i2c_firewall_rules`, see section 2.6 /i2c-busses/xxx/i2c_firewall_rules.

Example of /i2c-busses binding

The following DTS yields an I2C bus speed approaching (but not matching) 400kHz

```
/ {
    i2c-busses {
        i2c5 {
            bus-clock = <400000>;
            tlow = <2>;
            thigh = <4>;
            i2c_firewall_rules {
                ...
            };
        };
    };
};
```

2.6 /i2c-busses/xxx/i2c_firewall_rules

The firmware owns the I2C controllers that it accesses. To give the CPU access to the attached I2C buses, the firmware provides #MRQ_I2C as a proxy interface. The proxy interface includes a configurable firewall to allow or deny a I2C write transactions based on rules. This section describes the firewall and its device tree binding.

> **Note:**
>
> Platform designers should configure the I2C firewall to prevent the CPU from interfering with the I2C devices owned by the firmware.

The `i2c_firewall_rules` node can contain arbitrarily named and arbitrary number of sub-nodes referred in this document as `i2c_firewall_rule@xxx`. Each of `i2c_firewall_rule@xxx` sub-node contain properties that are used to configure a single firewall rule to filter out unwanted transactions. The node contains following properties.

Required properties

addr_range_low
: Start of address range to apply the firewall. 32-bit unsigned integer. Valid range 0-1023

addr_range_high
: End of address range to apply the firewall. 32-bit unsigned integer. Valid range 0-1023

Optional properties

allow_reads
: When set to nonzero value, the firewall rule is relaxed to allow simple I2C read transactions.

The I2C firewall rules are of the following type:

- Deny transaction if any of the operations within the transaction target I2C slave device where address is in range [`addr_range_low`, `addr_range_high`].

If the optional property `allow_reads` is set to nonzero value, then the preceding rule is relaxed to allow simple read transactions targeting single slave device. The transactions in following format are allowed:

S	<addr> <R/W>	<data 0>	Sr	<addr> <R>	<data 1>	…	<data N>	P

(S = Start, Sr = Repeated Start, P = Stop)

That is:

- Transaction must consist of exactly two operations.
- The second operation of the transaction must be read.
- The length of the first operation must be one byte.
- The slave address in both operations must be equal.

Whenever any rule that denies the transaction is found, the rest of the rules are not evaluated at all. If no rule matches, then transaction is allowed.

Example of i2c_firewall_rules binding

```
/ {
    i2c-busses {
        i2c5 {
            i2c_firewall_rules {
                i2c_firewall_rule@1 {
                    addr_range_low = <0x48>;
                    addr_range_high = <0x48>;
                };
                i2c_firewall_rule@2 {
                    addr_range_low = <0x52>;
                    addr_range_high = <0x54>;
                    allow_reads = <1>;
                };
            };
        };
    };
};
```

2.7 /mail

The `/mail` node controls the permissions of clients of BPMP-FW to access the services via the MRQ interface. The node is optional. When absent, BPMP-FW accepts messages from the CPU (in TrustZone normal world).

Required properties

edition
: A 32-bit unsigned integer indicating the edition of this binding. Must be set to 1. Other values are reserved for future use.

Optional properties

dbs
: A 32-bit bitmask representing the peers from which the firmware accepts MRQ messages. The mask is a bitwise OR of Doorbell IDs, see 2.19.3 Doorbell Identifiers. If this property is missing, the default mask of TEGRA264_DB_CPU_NS will be used.

The `/mail` node can contain optional subnode `/mail/acl`

2.7.1 /mail/acl

Contains one property for each ACL mask to be updated. If no properties are present, the built-in firmware permission table will

be used as is.

The elements of each property are:

- element 0: MRQ ID
- element 1: An updated ACL mask. A 32-bit mask representing the peers from which the firmware accepts messages for a given MRQ ID The mask is a bitwise OR of Doorbell IDs, defined in 2.19.3 Doorbell Identifiers.

The updated ACL mask can only restrict the current permissions as determined by the built-in firmware table and the doorbell mask (see the `dbs` property).

2.7.2 /mail Example

Example of /mail binding

```
/{
    mail {
        edition = <1>;
        dbs = <(TEGRA264_DB_CPU_NS | TEGRA264_DB_RCE)>;
        acl {
            mrq_ping = <MRQ_PING (TEGRA264_DB_CPU_NS)>;
        };
    };
};
```

2.8 /pcie/xxx

The `pcie` node contains per PCIe controller freely named unique subnodes which configures PCIE controllers.

Required properties

status
: This is a string type property that indicates whether or not to enable Tegra264 PCIe driver. If this property is missing or `status` does not match “okay”, PCIe driver is disabled.

pcie-id
: A 32-bit unsigned integer defining the PCIe controller number. If the property is missing or nonspecified value is provided, BPMP-FW will raise a fatal error and prevent the system from starting. Following are the valid options for Tegra264.

Option	pcie-id
0	PCIe controller C0
1	PCIe controller C1
2	PCIe controller C2
3	PCIe controller C3
4	PCIe controller C4
5	PCIe controller C5

pcie-mode
: A 32-bit unsigned integer defining the PCIe modes supported by the controller. If nonspecified value is provided, BPMP-FW will raise a fatal error and prevent the system from starting. Following are the valid options for Tegra264.

Option	pcie-mode
1	PCIe Root port mode
2	PCIe Endpoint mode

Optional properties

max-link-speed

A 32-bit unsigned integer defining the maximum link speed supported by the controller. If nonspecified value is provided, BPMP-FW will raise a fatal error and prevent the system from starting. This feature is not supported in BPMP-FW releases targeted for functional safety. If property is not present, PCIe Gen5 speed is used. Following are the valid options for Tegra264.

Option	max-link-speed
1	PCIe Gen1 speed
2	PCIe Gen2 speed
3	PCIe Gen3 speed
4	PCIe Gen4 speed
5	PCIe Gen5 speed

clk-scheme

A 32-bit unsigned integer defining the PCIe clocke scheme supported by the controller. If nonspecified value is provided, BPMP-FW will raise a fatal error and prevent the system from starting. If property is not present, PCIe common clock mode is used. Following are the valid options for Tegra264.

Option	clk-scheme
0	PCIe Common Clock mode
1	PCIe SRIS mode
2	PCIe SRNS mode

aspm-capability

A 32-bit unsigned integer defining the PCIe ASPM capabilities supported by the controller. If nonspecified value is provided, BPMP-FW will raise a fatal error and prevent the system from starting. If property is not present, all ASPM states are marked as not-supported. Following are the valid options for Tegra264. Please note that ASPM L1.1 and ASPM L1.2 depend on ASPM L1 as per PCIe spec, hence either or both of them can't be selected without selecting ASPM L1. This feature is not supported in BPMP-FW releases targeted for functional safety. This DT property won't come into effect when *hot-plug-capable* property is set to 1.

Option	aspm-capability
0	No ASPM states are suported
bit-0 (0x01)	Reserved
bit-1 (0x02)	ASPM L1 state is supported
bit-2 (0x04)	ASPM L1.1 state is supported
bit-3 (0x08)	ASPM L1.2 state is supported
bit-4 (0x10)	ASPM L1 PLL PD state is supported
bit-5 (0x10)	ASPM L1 CPM state is supported

hot-plug-capable:

A 32-bit unsigned integer defining the PCIe hot plug capable supported by the controller. If nonspecified value is provided,

BPMP-FW will raise a fatal error and prevent the system from starting. If property is not present, PCIe hot plug capable feature is not used. Following are the valid options for Tegra264.

Option	hot-plug-capable
0	Not a hot Plug capable controller
1	Hot Plug capable controller

Example of /pcie binding

For example, assuming a PCIe controller C2 with Gen5 x8 configuration, the following DTS yields a PCIe C2 Rootport controller with Gen5 x8 max capability.

```
/ {
    pcie {
        pcie@2 {
            status = "okay";
            pcie-id = <2>;
            max-link-speed = <5>;
            pcie-mode = <1>;
            clk-scheme = <2>;
        };
    };
};
```

2.9 /pdomains

The `/pdomains` node configures the access permissions to control Tegra's power partitions and allows specifying a power partition to be permanently enabled or disabled. The node is optional.

There should be a `/pdomains/domain@xxx` sub-node for each power domain that must be configured.

2.9.1 /pdomains/domain@xxx

Each `/pdomains/domain@xxx` node contains properties to configure the specified power partition.

Required properties

id
: ID (32-bit unsigned) of the power domain to be updated. Valid power domain IDs are defined in 2.19.5 Power Domain Identifiers.

Optional properties

always-off
: A 32-bit unsigned integer. A nonzero value indicates that the power domain must be kept always OFF. If the property is absent, a value of zero is assumed.

always-on
: A 32-bit unsigned integer. A nonzero value indicates that the power domain must be kept always ON. If the property is absent, a value of zero is assumed.

on-by
: A 32-bit unsigned integer. Specifies initiators for initial power on. This value is composed as a bitwise OR of Doorbell IDs. The list of valid IDs is defined in section 2.19.3 Doorbell Identifiers. When present the power domain initially powered on

behalf of the given initiators. The property is restricted for use with TEGRA264_POWER_DOMAIN_GPU only. Having the property defined together with a nonzero always-on or always-off property is considered an invalid configuration. The initiators specified by the property must be present in the acl list, either explicitly by the acl property or implicitly when absent.

acl
: 32-bit unsigned integer. Specifies the ACL of this power partition. The value is composed as a bitwise OR of Doorbell IDs, defined in 2.19.3 Doorbell Identifiers. Each set bit indicates that the corresponding MRQ initiator is authorized to control this domain with `MRQ_PG.CMD_PG_SET_STATE` request of BPMP-FW runtime interface. If this property is absent, all initiators are allowed control.

A nonzero always-off property takes precedence over a nonzero always-on property.

Example of /pdomains binding

```
/ {
    pdomains {
        domain@gpu {
            id = <TEGRA264_POWER_DOMAIN_GPU>;
            on-by = <TEGRA264\_DB_CCPLEX>;
        };
        domain@disp {
            id = <TEGRA264_POWER_DOMAIN_DISP>;
            always-on = <1>;
        };
        domain@vic {
            id = <TEGRA264_POWER_DOMAIN_VIC>;
            acl = <(TEGRA264_DB_CPU_NS | TEGRA264_DB_CPU_S)>;
        };
    };
};
```

2.10 /regulators

The `/regulators` node describes the power-tree topology of the Tegra's core power rails for BPMP-FW to scale voltages and control regulator states for power state transitions of Tegra SOC or its internal power partitions. It must contain one subnode for each of the regulators driving Tegra's core power rails.

Child nodes of `/regulators` have the following properties and nodes.

Required properties

rail-id
: A list of 32-bit unsigned rail IDs. Valid rail IDs are defined in 2.19.6 Voltage Rail Identifiers. Multiple `rail-id` are treated as aliases for the regulator and can be used when two or more rails are driven by a single regulator.

dev
: A phandle to the node that describes the actual regulator device.

Optional properties

regulator-min-microvolt
: A 32-bit unsigned integer. Specifies the smallest voltage in microvolts that is permissible in the system. If omitted, defaults to the underlying hardware limit of the regulator device.

regulator-max-microvolt
: A 32-bit unsigned integer. Specifies the largest voltage in microvolts that is permissible in the system. If omitted, defaults to

the underlying hardware limit of the regulator device.

regulator-ramp-delay-const
: A 32-bit unsigned integer. Specifies a wait time in microseconds that is respected when voltage changes. This time is added to wait time value computed based on `regulator-ramp-delay-linear`.

regulator-ramp-delay-linear
: A 32-bit unsigned integer. Specifies a scaling factor in microvolt/microsecond. For a voltage change of N microvolts, the total wait time is (`regulator-ramp-delay-const` + N / `regulator-ramp-delay-linear`).

regulator-enable-ramp-delay
: A 32-bit unsigned integer. Specifies a wait time in microseconds to be observed after a regulator is enabled. The wait ensures that the target voltage has been reached. `regulator-enable-ramp-delay` is the only wait time applied when a regulator has been enabled, that is the wait time defined by `regulator-ramp-delay-*` is not applied in this case.

regulator-disable-ramp-delay
: A 32-bit unsigned integer. Specifies a wait time in microseconds to be observed after a regulator has been disabled, to ensure that the voltage has been fully ramped down. The regulator-disable-ramp-delay is used exclusively when a regulator is disabled. The wait time defined by `regulator-ramp-delay-*` is not used in this case.

Example of /regulators binding

```
/ {
        regulators {
                vdd_cpu {
                        rail-id = <TEGRA264_REGULATOR_RAIL_VDD_CPU>;
                        dev = <&cpu_vreg>;
                        regulator-ramp-delay = <12000>;
                        regulator-enable-ramp-delay = <210>;
                        regulator-min-microvolt = <550000>;
                        regulator-max-microvolt = <1120000>;
                };

                vdd_gpu {
                        rail-id = <TEGRA264_REGULATOR_RAIL_VDD_GPU>;
                        dev = <&gpu_vreg>;
                        regulator-ramp-delay = <22000>;
                        regulator-enable-ramp-delay = <210>;
                        regulator-min-microvolt = <550000>;
                        regulator-max-microvolt = <1120000>;
                };

                vdd_core {
                        rail-id = <TEGRA264_REGULATOR_RAIL_VDD_CORE>;
                        dev = <&core_vreg>;
                        regulator-ramp-delay = <12000>;
                        regulator-enable-ramp-delay = <210>;
                        regulator-max-microvolt = <1120000>;
                        regulator-min-microvolt = <550000>;
                };
        };
};
```

2.11 Regulator Device Bindings

Each regulator device driver defines its own set of device tree bindings. the bindings below are for the currently supported devices.

2.11.1 VRS-11

VRS-11 is a dual-channel high-current voltage regulator. BPMP-FW uses the I2C interface of the regulator for initial configuration and PWM interface for controlling the output voltage.

Required Properties

compatible
: Must be set to string “vrs-11”.

pwm-id
: A 32-bit unsigned integer. Specifies the ID of Tegra PWM controller associated with this regulator. Valid values are 1-8.

slave-address
: A 32-bit unsigned integer. Specifies the I2C slave address of the device. Valid range is 0-127.

channel
: A 32-bit unsigned integer. Specifies the channel. Valid values are 0 and 1, where 0 indicates channel A and 1 indicates channel B.

pwm-rate
: A 32-bit unsigned integer. Specifies the target frequency in Hz for the PWM controller output. Valid range 400000-600000.

Optional properties

enable-gpio
: A 32-bit unsigned integer. Specifies the ID of a GPIO that controls the regulator’s enable pin (see 2.19.1 GPIO Port Identifiers). If absent, the regulator is assumed to be always on.

Example of VRS-11 binding

```
/ {
        core_vreg: vrs-11@pwm_core {
                compatible = "vrs-11";
                pwm-id = <4>;
                pwm-rate = <531250>;
                slave-address = <0x22>;
                channel = <0>;
        };
};
```

2.11.2 Fixed Voltage Regulator

Some of the Tegra clocks under BPMP-FW control may be supplied by a fixed voltage regulator that is not controlled by BPMP-FW. For BPMP-FW to restrict the maximum frequencies of such clocks, the firmware needs information of the configured voltage. The “fixed” voltage regulator configuration is used for this purpose.

For Tegra264 SOC, a fixed voltage rail is typically VDD_AON voltage rail, identified with ID

- TEGRA264_REGULATOR_RAIL_VDD_AON

Required Properties

compatible
: Must be set to string “fixed”.

voltage-uv
: A 32-bit unsigned integer. Specifies the fixed voltage value (in microvolt) that the regulator is supplying. Valid range [0, 1250000].

Example of Fixed Voltage Regulator binding

```
/ {
        aon_vreg: fixed@aon {
                compatible = "fixed";
                voltage-uv = <850000>;
        };
};
```

2.12 /reset

The optional `/reset` node configures the access permissions of clients of BPMP-FW to access the service to control state of individual resets for IP blocks within Tegra SOC.

There should be a `/reset/reset@xxx` subnode for each reset that needs explicit configuration data.

2.12.1 /reset/reset@xxx

Each `/reset/reset@xxx` node contains configuration of which initiators (clients) are permitted to control the given reset.

Required properties

reset-id
: A 32-bit unsigned integer. Specifies a reset ID of an IP block. Valid IDs are defined in 2.19.7 Reset Identifiers. If a reset ID appears in more than one `/reset/reset@xxx` subnode the last instance takes precedence.

Optional properties

acl
: A 32-bit unsigned integer. Specifies the ACL of this reset. This value is composed as a bitwise OR of Doorbell IDs. Valid IDs are defined in 2.19.3 Doorbell Identifiers. Each set bit indicates that the corresponding MRQ initiator is authorized to control this reset. If this property is absent, only those initiators preconfigured in the firmware are authorized. The allowed values for `acl` depends on the current initiator setting of the reset. If the current initiator is TEGRA264_DB_CPU_S, then only TEGRA264_DB_CPU_S, 0 or TEGRA264_DB_BPMP are allowed. If the current initiator is 0 or TEGRA264_DB_BPMP, then only 0 or TEGRA264_DB_BPMP are allowed. For any other set of initiators, all values are allowed.

Example of /reset binding

```
/{
    reset {
        reset@host1x {
            reset-id = <TEGRA264_RESET_VIC>;
            acl = <(TEGRA264_DB_CPU_NS | TEGRA264_DB_CPU_S)>;
        };
    };
};
```

2.13 /serial

The `/serial` node configures BPMP-FW's serial I/O interface. The interface can be made available only when BPMP-FW's diagnostic capabilities are enabled. Refer to `/diagnostics` for more information.

The `/serial` node is optional. If it is absent, BPMP-FW's serial I/O interface is disabled.

Optional properties

port
: A 32-bit unsigned integer. Specifies the serial I/O interface to use. Valid values are:

  - 0: UART0
  - 5: UART5
  - 127: UART0 through Tegra's UART Trace Controller (UTC) hardware
  - Other: Disable UART console

  BPMP-FW requires that the selected serial I/O hardware in Tegra has been configured and enabled by previous boot stages.

log-level
: An integer value representing verbosity of the BPMP-FW logs. Messages with a log level less than or equal to the specified log level are shown. A higher value may cause more output and slow down firmware execution. Allowed values are:

- 1: Show errors.
- 2: Show warnings.
- 3: Show info messages.
- 4: Show debug messages.

has_input
: When this property is specified, BPMP-FW console will accept input from its serial I/O interface when BPMP-FW's diagnostic level is 2, see document section `/diagnostics` for further details. When this property is absent, BPMP-FW console remains in output-only mode.

Example of /serial binding

```
/ {
    serial {
        port = <2>;
        log-level = <3>;
        has_input;
    };
};
```

2.14 /nvtherm

The `/nvtherm` node specifies configuration data for the NV_THERM hardware and driver.

> **Note:**
>
> Temperature format: Unless specified otherwise, all temperature values specified in this document must be represented as 32-bit signed integers. The temperatures values must be specified in millidegree Celsius (0.001C) and must be multiples of 0.125C. Example: a temperature of 20.5C must be provided as 20500, 70.125C as 70125 and so on. The firmware will trigger a fatal error and halt boot if any nonconforming values such as 70200 are provided.

Optional properties

> **Note:**
>
> A fatal error is triggered and boot is halted if any of the properties listed in this section is malformed.

thermtrip_hot
: A 32-bit signed integer that specifies the hot temperature threshold, exceeding which, chip shutdown is triggered. The firmware triggers a fatal error and halts boot if the value specified exceeds the shutdown temperature specified by NVIDIA for the chip. NVIDIA specified shutdown limit is used if the property is omitted.

thermtrip_cold
: A 32-bit signed integer that specifies the cold temperature threshold, falling below which, chip shutdown is triggered. The firmware triggers a fatal error and halts boot if the value specified is less than the cold shutdown temperature specified by NVIDIA for the chip. NVIDIA specified cold shutdown limit is used if the property is omitted.

hsmtrip_hot
: A 32-bit signed integer that specifies the hot temperature threshold, exceeding which, the Hardware Safety Manager (HSM) is notified. Hot trip notifications to HSM are disabled if the property is omitted.

hsmtrip_cold
: A 32-bit signed integer that specifies the cold temperature threshold, falling below which, the HSM is notified. Cold trip notifications to the HSM are disabled if the property is omitted.

consistency_thresh
: A 32-bit signed integer that specifies the threshold used by Hardware Consistency Checker (to verify all sensors are reporting temperatures within an acceptable range). A notification to the HSM is sent if the temperature delta between the hottest and the coldest sensor exceeds the specified threshold. An NVIDIA specified threshold will be used if the property is omitted. The firmware triggers a fatal error and halts boot if the value specified exceeds NVIDIA specified threshold.

bjt_offsets
: An array of 32-bit integers specifying per BJT hotspot offset. See `/nvtherm/bjt_offsets` section.

disable_thermtrip_reset
: A 32-bit unsigned integer that controls the action on violation of thermtrip hot or cold thresholds. If set to non-zero value, BPMP-FW disables the hardware configuration that resets the chip and asserts SHUTDOWN_N external signal when a *thermtrip_hot* or *thermtrip_cold* threshold is breached, and thus only notification to HSM is generated. If set to zero, or the property is omitted, BPMP-FW enables the chip reset and SHUTDOWN_N assertion in hardware.

2.14.1 /nvtherm/bjt_offsets

Placement of BJTs on the die can often be away from the actual hotspot. NV_THERM hardware allows specifying a hotspot offset that can be added to the temperature reported by the BJT.

The `bjt_offsets` property is an array of 32-bit signed integers where each value is a temperature offset and must be within the valid [min, max] range specified by NVIDIA for the chip. The length of the array must be 63.

> **Note:**
>
> Firmware expects the BJT offsets in the array to appear in the following order:

Global BJT#	TSENSE Group	Group BJT#
0	GPC TSENSE 0	0
1	GPC TSENSE 0	1

Global BJT#	TSENSE Group	Group BJT#
2	GPC TSENSE 0	2
3	GPC TSENSE 0	3
4	GPC TSENSE 0	4
5	GPC TSENSE 0	5
6	GPC TSENSE 0	6
7	GPC TSENSE 0	7
8	GPC TSENSE 1	0
9	GPC TSENSE 1	1
10	GPC TSENSE 1	2
11	GPC TSENSE 1	3
12	GPC TSENSE 1	4
13	GPC TSENSE 1	5
14	GPC TSENSE 1	6
15	GPC TSENSE 1	7
16	GPC TSENSE 2	0
17	GPC TSENSE 2	1
18	GPC TSENSE 2	2
19	GPC TSENSE 2	3
20	GPC TSENSE 2	4
21	GPC TSENSE 2	5
22	GPC TSENSE 2	6
23	GPC TSENSE 2	7
24	CPU TSENSE 0	0
25	CPU TSENSE 0	1
26	CPU TSENSE 0	2
27	CPU TSENSE 0	3
28	CPU TSENSE 0	4
29	CPU TSENSE 0	5
30	CPU TSENSE 1	0
31	CPU TSENSE 1	1
32	CPU TSENSE 1	2
33	CPU TSENSE 1	3
34	CPU TSENSE 1	4
35	CPU TSENSE 1	5
36	SOC TSENSE 0	0
37	SOC TSENSE 0	1
38	SOC TSENSE 0	2
39	SOC TSENSE 0	3

Global BJT#	TSENSE Group	Group BJT#
40	SOC TSENSE 0	4
41	SOC TSENSE 0	5
42	SOC TSENSE 1	0
43	SOC TSENSE 1	1
44	SOC TSENSE 1	2
45	SOC TSENSE 2	0
46	SOC TSENSE 2	1
47	SOC TSENSE 2	2
48	SOC TSENSE 2	3
49	SOC TSENSE 2	4
50	SOC TSENSE 2	5
51	SOC TSENSE 2	6
52	SOC TSENSE 3	0
53	SOC TSENSE 3	1
54	SOC TSENSE 3	2
55	SOC TSENSE 3	3
56	SOC TSENSE 3	4
57	SOC TSENSE 4	0
58	SOC TSENSE 4	1
59	SOC TSENSE 4	2
60	SOC TSENSE 5	0
61	SOC TSENSE 5	1
62	SOC TSENSE 5	2

The firmware triggers a fatal error and halts boot if

- Offset value is out of bounds.
- The array is not of required size.

2.14.2 /nvtherm Example

Example of /nvtherm binding

```
/ {
    nvtherm {
        thermtrip_hot = <105000>;
        thermtrip_cold = <(-40000)>;
        hsmtrip_hot = <100000>;
        hsmtrip_cold = <(-35000)>;
        consistency_thresh = <30000>;

        bjt_offsets = < 0 0 0 0 0 0 0 0
                        0 0 0 0 0 0 0 0
                        0 0 0 0 0 0 0 0
```

```
                        0 0 0 0 0 0 0 0
                        0 0 0 0 0 0 0 0
                        0 0 0 0 0 0 0 0
                        0 0 0 0 0 0 0 0
                        0 0 0 0 0 0 0 0 >;
    };
};
```

2.15 /system-cfg

Optional properties

soc-reset-en

A byte array of length 86. The array entries specify the configuration of the reset sources controlled by BPMP-FW. The mapping of array entries to the corresponding reset sources, as well as valid entry values and its interpretation by BPMP-FW is specified in the table below. If array size is not 86, or array entry value is other than possible values specified in the table, BPMP-FW raises Fatal Error and aborts boot. If property is not present, boot configuration of all reset sources is retained.

Entry Index	Reset Source	Possible Values
0		128 = Reserved
1	CSDC_RTC_XTAL	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
2	VREFRO_POWER_BAD	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
3	SCPM_SOC_XTAL	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
4	SCPM_RTC_XTAL	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
5	FMON_32K	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
6	FMON_OSC	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
7		128 = Reserved
8	POD_IO	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
9	POD_PLUS_IO_SPLL	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
10	POD_PLUS_SOC	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
11	VMON_PLUS_UV	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
12	VMON_PLUS_OV	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
13		128 = Reserved
14		128 = Reserved
15	BPMP_BOOT_FAULT	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
16	SCPM_BPMP_CORE_CLK	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
17	SCPM_PSC_SE_CLK	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
18	VMON_SOC_MIN	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
19	VMON_SOC_MAX	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
20	VMON_MSS_MIN	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
21	VMON_MSS_MAX	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
22	POD_PLUS_IO_VMON	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
23		128 = Reserved
24	NV_THERM_FAULT	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration

Entry Index	Reset Source	Possible Values
25	FSI_THERM_FAULT	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
26		128 = Reserved
27	SCPM_OESP_SE_CLK	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
28	SCPM_SB_SE_CLK	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
29	POD_CPU	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
30	POD_GPU	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
31	DCLS_GPU	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
32	POD_MSS	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
33	FMON_FSI	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
34		128 = Reserved
35	VMON_FSI_MIN	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
36	VMON_FSI_MAX	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
37	VMON_CPU0_MIN	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
38	VMON_CPU0_MAX	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
39	BPMP_FMON	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
40	AO_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
41	BPMP_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
42	AO_TKE_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
43	RCE0_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
44	RCE1_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
45	DCE_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
46	FSI_R5_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
47	FSI_R52_0_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
48	FSI_R52_1_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
49	FSI_R52_2_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
50	FSI_R52_3_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
51	TOP_0_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
52	TOP_1_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
53	TOP_2_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
54	APE_C0_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
55	APE_C1_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
56	GPU_TKE_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
57		128 = Reserved
58	OESP_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
59	SB_WDT_POR	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
60		128 = Reserved
61	L0L1_RST_OUT_N	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
62		128 = Reserved

Entry Index	Reset Source	Possible Values
63	CSITE_SW	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
64	AO_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
65	BPMP_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
66	AO_TKE_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
67	RCE0_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
68	RCE1_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
69	DCE_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
70	FSI_R5_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
71	FSI_R52_0_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
72	FSI_R52_1_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
73	FSI_R52_2_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
74	FSI_R52_3_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
75	TOP_0_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
76	TOP_1_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
77	TOP_2_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
78	APE_C0_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
79	APE_C1_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
80		128 = Reserved
81	OESP_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
82	SB_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
83	TSC_0_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
84	TSC_1_WDT_DBG	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration
85	L2_RST_OUT_N	0 = DISABLE, 1 = ENABLE, 255 = Preserve boot configuration

2.15.1 /system-cfg/sc7

Optional node that contains properties related to SC7.

Optional properties

pwrgood-timer
: A 32-bit unsigned integer. Specifies the value for SoC rail power-up stabilization timer. Timer value is the configured value + 1 32.768kHz clock cycles, that is, (pwrgood-timer + 1) * 30.518 us. Valid range is 0-65535. When this property is omitted, value 127 is used.

sc7-res-timer
: A 32-bit unsigned integer that specifies the number of 32.768 KHz clock cycles the SC7 HW sequencer waits before declaring SC7 residency after power off. Valid range is 0-255. When this property is omitted, value 4 is used.

Example of /system-cfg/sc7 binding

```
/ {
    system-cfg {
```

```
        sc7 {
            pwrgood-timer = <0x70>;
        };
    };
};
```

2.15.2 /system-cfg/sc7/debug

Optional node that contains properties related to SC7 debug.

Optional properties

enabled
: A 32-bit unsigned integer. SC7 debug mode is enabled if the property is present and has a non-zero value, otherwise disabled.

Example of /system-cfg/sc7/debug binding

```
/ {
    system-cfg {
        sc7 {
            debug {
                enabled = <1>;
            };
        };
    };
};
```

2.16 /vrmon

The `/vrmon` node configures BPMP-FW support for onboard voltage rail monitors. It must contain one `vrmon@xxx` subnode for each voltage rail that is controlled by BPMP-FW (referred to as controlled rail in the following sections) and observed by the onboard voltage monitor (vrmon).

2.16.1 /vrmon/vrmon@xxx

Required properties

rail-id
: A 32-bit unsigned controlled rail ID. Valid rail IDs are defined in 2.19.6 Voltage Rail Identifiers. When multiple rails are driven by the same regulator, connected to a single monitor, either one rail id can be specified.

mon-dev
: A phandle to the node that describes the vrmon device rail is connected to

channel
: A 32-bit unsigned vrmon device channel that monitors the rail. Accepted range 0-64. Valid channels are also limited by device capabilities.

underv-uv-min
: A 32-bit unsigned minimum under-voltage threshold specified in microvolts. Accepted range 0-1500000.

underv-uv-per-v
: A 32-bit unsigned under-voltage slope specified in microvolts per volt. Accepted range 0-500000.

overv-uv-max
A 32-bit unsigned maximum over-voltage threshold specified in microvolts. Accepted range 0-1500000.

overv-uv-per-v
A 32-bit unsigned over-voltage slope specified in microvolts per volt. Accepted range 0-500000.

Optional properties

underv-uv-offs
A 32-bit unsigned under-voltage offset specified in microvolts. Accepted range 0-100000. If omitted, set to “0” by default.

overv-uv-offs
A 32-bit unsigned over-voltage offset specified in microvolts. Accepted range 0-100000. If omitted, set to “0” by default.

underv-hf-uv-offs
A 32-bit unsigned under-voltage HF offset specified in microvolts. Accepted range 0-100000. If omitted, set to *underv-uv-offs* by default.

overv-hf-uv-offs
A 32-bit unsigned over-voltage HF offset specified in microvolts. Accepted range 0-100000. If omitted, set to *overv-uv-offs* by default.

underv-hf-uv-per-v
A 32-bit unsigned under-voltage HF slope specified in microvolts per volt. Accepted range 0-500000. If omitted, set to underv-uv-per-v by default.

overv-hf-uv-per-v
A 32-bit unsigned over-voltage HF slope specified in microvolts per volt. Accepted range 0-500000. If omitted, set to overv-uv-per-v by default.

uvlf-signal-en
A 32-bit unsigned integer. Valid values are 0 and 1. When set to value 1, BPMP-FW will enable or disable the LF fault interrupt signaling when the monitored voltage regulator device is enabled or disabled, respectively. When omitted, or set to 0, BPMP-FW will not alter LF fault interrupt signaling configured previously.

hf-signal-en
A 32-bit unsigned integer. Valid values are 0 and 1. When set to value 1, BPMP-FW will enable or disable the HF fault interrupt signaling when the monitored voltage regulator device is enabled or disabled, respectively. When omitted, or set to 0, BPMP-FW will not alter HF fault interrupt signaling configured previously.

scale-hf
A 32-bit unsigned integer. Valid values are 0 and 1. When set to value 1, BPMP-FW will configure the high frequency monitoring thresholds when the monitored voltage regulator device is enabled, disabled or changing voltage. When omitted, or set to 0, BPMP-FW will not set high frequency monitoring thresholds.

Note:

Rail monitoring thresholds Vmin and Vmax are calculated according to the following functions of voltage V in microvolts.

V_min(V) = V - (“underv-uv-offs” + “underv-uv-per-v” * V / 1000000)

V_max(V) = V + (“overv-uv-offs” + “overv-uv-per-v” * V / 1000000)

V_min_hf(V) = V - (“underv-hf-uv-offs” + “underv-hf-uv-per-v” * V / 1000000)

V_max_hf(V) = V + (“overv-hf-uv-offs” + “overv-hf-uv-per-v” * V / 1000000)

2.17 Vrmon Device

Each vrmon device has its own set of device tree bindings. The following bindings are for the currently supported devices.

2.17.1 VRS12

VRS12 is a vrmon device with 6 channels out of which only channels 1 and 2 are valid for monitoring of the controlled rails. For each VRS12 device on the board connected to a controlled rail a separate `vrs12@xxx` node must be defined.

2.17.1.1 …/vrs12@xxx

Required properties

compatible
: Must be set to “vrs12”.

i2c-addr
: 32-bit unsigned I2C slave address to access the device. Accepted range 0-127.

Optional properties

boot-config
: A 32-bit unsigned integer that represents bitmask to enable configuration options. If omitted, set to “0” by default.

- Bit 0 - If set, enables reading and printing device status during initialization.
- Bit 1 - If set, enables initial configuration of fault interrupts by BPMP-FW: all interrupts are enabled for VDD_CORE rail, and only OVLF interrupt for other rails. If cleared, interrupt enable configuration set by MCU is initially used.
- bit 2 … 31 - must be 0.

soc-rst-channel
: A 32-bit unsigned integer that represents VRS-12 device channel that is connected to SoC reset signal. Accepted range 3-6.

Example of VRS12 Vrmon Device binding

```
/ {
    vrs12_0: vrs12@30 {
        compatible = "vrs12";
        i2c-addr = <0x30>;
        soc-rst-channel = <3>;
    };
    vrs12_1: vrs12@31 {
        compatible = "vrs12";
        i2c-addr = <0x31>;
    };

    vrmon {
        vdd_core_vrmon {
            rail-id = <TEGRA264_REGULATOR_RAIL_VDD_CORE>;
            mon-dev = <&vrs12_1>;
            channel = <1>;
            scale-hf = <1>;
            underv-uv-per-v = <35000>;     /* -3.5% */
            overv-uv-per-v =  <35000>;     /* +3.5% */
            underv-uv-min = <600000>;
            overv-uv-max =  <1080000>;
        };
```

```
            vdd_cpu_vrmon {
                rail-id = <TEGRA264_REGULATOR_RAIL_VDD_CPU>;
                mon-dev = <&vrs12_0>;
                channel = <1>;
                hf-signal-en = <0x1>;
                uvlf-signal-en = <0x1>;
                underv-uv-per-v = <35000>;     /* -3.5% */
                overv-uv-per-v =  <35000>;     /* +3.5% */
                underv-uv-min = <600000>;
                overv-uv-max =  <1080000>;
            };

            vdd_gpu_vrmon {
                rail-id = <TEGRA264_REGULATOR_RAIL_VDD_GPU>;
                mon-dev = <&vrs12_0>;
                channel = <2>;
                hf-signal-en = <0x1>;
                uvlf-signal-en = <0x1>;
                underv-uv-per-v = <35000>;     /* -3.5% */
                overv-uv-per-v =  <35000>;     /* +3.5% */
                underv-uv-min = <600000>;
                overv-uv-max =  <1080000>;
            };

            vdd_cv_vrmon {
                rail-id = <TEGRA264_REGULATOR_RAIL_VDD_CV>;
                mon-dev = <&vrs12_1>;
                channel = <2>;
                hf-signal-en = <0x1>;
                uvlf-signal-en = <0x1>;
                underv-uv-per-v = <35000>;     /* -3.5% */
                overv-uv-per-v =  <35000>;     /* +3.5% */
                underv-uv-min = <600000>;
                overv-uv-max =  <1080000>;
            };
        };
    };
```

2.18 /uphy

The `/uphy` node contains properties that describe UPHY0/UPHY1 configuration of a Tegra264 platform. All properties are optional.

/uphy/status

This is a string type property that indicates whether or not to enable Tegra264 UPHY driver. If this property is missing or `status` does not match “okay”, UPHY driver is disabled.

/uphy/uphy0-config

A 32-bit unsigned integer that identifies UPHY0 lanes mapping. If `uphy0-config` property is absent, UPHY driver will keep UPHY0 in disabled(/reset) state. Valid values are listed in following table. If a nonspecified value is provided, BPMP-FW will raise Fatal Error and halt its execution. “DM” means the PCIe controller can be enabled with either RP (rootport) role or EP (endpoint) role.

Option	Lane 0	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7
0	USB3 x1 P0	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C2 x4 (DM)			

Option	Lane 0	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7
1	USB3 x1 P0	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C2 x2 (DM)		PCIe C3 x2 (RP)	
2	USB3 x1 P0	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C1 x1 (RP)	PCIe C2 x1 (RP)	PCIe C3 x2 (RP)	
3	USB3 x1 P0	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C2 x2 (DM)		UFS x2	
4	USB3 x1 P0	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C1 x1 (RP)	PCIe C2 x1 (RP)	UFS x2	
5	USB3 x1 P0	USB3 x1 P1	USB3 x1 P2	PCIe C1 x1 (RP)	PCIe C2 x4 (DM)			
6	USB3 x1 P0	USB3 x1 P1	USB3 x1 P2	PCIe C1 x1 (RP)	PCIe C2 x2 (DM)		PCIe C3 x2 (RP)	
7	USB3 x1 P0	USB3 x1 P1	USB3 x1 P2	PCIe C1 x1 (RP)	PCIe C2 x2 (DM)		UFS x2	
8	USB3 x1 P0	USB3 x1 P1	PCIe C1 x2 (RP)		PCIe C2 x4 (DM)			
9	USB3 x1 P0	USB3 x1 P1	PCIe C1 x2 (RP)		PCIe C2 x2 (DM)		PCIe C3 x2 (RP)	
10	USB3 x1 P0	USB3 x1 P1	PCIe C1 x4 (RP)				UFS x2	
11	USB3 x1 P0	USB3 x1 P1	PCIe C1 x2 (RP)		PCIe C2 x2 (DM)		UFS x2	
12	EQOS SOC	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C2 x4 (DM)			
13	EQOS SOC	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C1 x2 (RP)		PCIe C3 x2 (RP)	
14	EQOS SOC	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C2 x2 (DM)		PCIe C3 x2 (RP)	
15	EQOS SOC	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C1 x1 (RP)	PCIe C2 x1 (RP)	PCIe C3 x2 (RP)	
16	EQOS SOC	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C2 x2 (DM)		UFS x2	
17	EQOS SOC	USB3 x1 P1	USB3 x1 P2	USB3 x1 P3	PCIe C1 x1 (RP)	PCIe C2 x1 (RP)	UFS x2	
18	EQOS SOC	USB3 x1 P1	USB3 x1 P2	PCIe C1 x1 (RP)	PCIe C2 x4 (DM)			
19	EQOS SOC	USB3 x1 P1	USB3 x1 P2	PCIe C1 x1 (RP)	PCIe C2 x2 (DM)		PCIe C3 x2 (RP)	
20	EQOS SOC	USB3 x1 P1	USB3 x1 P2	PCIe C1 x1 (RP)	PCIe C2 x2 (DM)		UFS x2	
21	EQOS SOC	USB3 x1 P1	PCIe C1 x2 (RP)		PCIe C2 x4 (DM)			

Option	Lane 0	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7
22	EQOS SOC	USB3 x1 P1	PCIe C1 x2 (RP)		PCIe C2 x2 (DM)		PCIe C3 x2 (RP)	
23	EQOS SOC	USB3 x1 P1	PCIe C1 x4 (RP)				UFS x2	
24	EQOS SOC	USB3 x1 P1	PCIe C1 x2 (RP)		PCIe C2 x2 (DM)		UFS x2	

/uphy/uphy1-config

A 32-bit unsigned integer that identifies UPHY1 lanes mapping. If `uphy1-config` property is absent, UPHY driver will keep UPHY1 in disabled(/reset) state. Valid values are listed in following table. If a nonspecified value is provided, BPMP-FW will raise Fatal Error and halt its execution. “DM” means the PCIe controller can be enabled with either RP (rootport) role or EP (endpoint) role.

Option	Lane 0	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7
0	PCIe C4 x8 (DM)							
1	PCIe C5 x4 (DM)				PCIe C4 x4 (DM)			
2	PCIe C5 x4 (DM)				PCIe C4 x2 (DM)		MGBE C2 2.5/5/10G	MGBE C3 2.5/5/10G
3	PCIe C5 x4 (DM)				PCIe C4 x2 (DM)		MGBE C2 25G	MGBE C3 25G
4	PCIe C5 x4 (DM)				PCIe C4 x1 (RP)	MGBE C1 2.5/5/10G	MGBE C2 2.5/5/10G	MGBE C3 2.5/5/10G
5	PCIe C5 x4 (DM)				PCIe C4 x1 (RP)	MGBE C1 25G	MGBE C2 25G	MGBE C3 25G
6	PCIe C5 x4 (DM)				MGBE C0 2.5/5/10/25G	MGBE C1 2.5/5/10/25G	MGBE C2 2.5/5/10/25G	MGBE C3 2.5/5/10/25G
7	PCIe C5 x4 (DM)				MGBE C0 2.5/5/10G	MGBE C1 2.5/5/10G	MGBE C2 2.5/5/10G	MGBE C3 2.5/5/10G
8	PCIe C5 x4 (DM)				MGBE C0 25G	MGBE C1 25G	MGBE C2 25G	MGBE C3 25G
9	PCIe C4 x2 (DM)		PCIe C5 x2 (DM)		MGBE C0 2.5/5/10/25G	MGBE C1 2.5/5/10/25G	MGBE C2 2.5/5/10/25G	MGBE C3 2.5/5/10/25G
10	PCIe C4 x2 (DM)		PCIe C5 x2 (DM)		MGBE C0 2.5/5/10G	MGBE C1 2.5/5/10G	MGBE C2 2.5/5/10G	MGBE C3 2.5/5/10G
11	PCIe C4 x2 (DM)		PCIe C5 x2 (DM)		MGBE C0 25G	MGBE C1 25G	MGBE C2 25G	MGBE C3 25G

/uphy/pcie-c2-endpoint-enable

When this property is present, it indicates PCIE controller C2 is enabled at Endpoint mode.

/uphy/pcie-c2-endpoint-use-int-refclk

When this property is present, it indicates PCIe controller C2 PLL is using internal refclk. By default, an endpoint controller uses external refclk. If this property is specified but PCIe controller C2 endpoint mode is not enabled, BPMP-FW will raise Fatal Error and halt its execution.

/uphy/pcie-c4-endpoint-enable

When this property is present, it indicates PCIE controller C4 is enabled at Endpoint mode.

/uphy/pcie-c4-endpoint-use-int-refclk

When this property is present, it indicates PCIe controller C4 PLL is using internal refclk. By default, an endpoint controller uses external refclk. If this property is specified but PCIe controller C4 endpoint mode is not enabled, BPMP-FW will raise Fatal Error and halt its execution.

/uphy/pcie-c5-endpoint-enable
When this property is present, it indicates PCIE controller C5 is enabled at Endpoint mode.

/uphy/pcie-c5-endpoint-use-int-refclk
When this property is present, it indicates PCIe controller C5 PLL is using internal refclk. By default, an endpoint controller uses external refclk. If this property is specified but PCIe controller C5 endpoint mode is not enabled, BPMP-FW will raise Fatal Error and halt its execution.

/uphy/pcie-c5-dc-couple-enable
When this property is present, it enables DC couple support for PCIE controller C5.

/uphy/force-ufs-init
When this property is present, BPMP-FW will perform unconditional re-initialization of all UFS PLLs and lanes during startup. When the property is omitted, BPMP-FW will detect the status of UFS lanes and will skip initialization of lanes that are already initialized by previous boot stages.

/uphy/mgbe0-speed
This property configures MGBE0 controller speed. Valid options are: 0 for 2.5 Gbps, 1 for 5Gbps, 2 for 10Gbps, and 3 for 25Gbps. If this property is not present, UPHY1 lane 4 is disabled. If the specified speed isn't supported by the selected `uphy1-config`, BPMP-FW will raise Fatal Error and halt its execution. For example, `uphy1-config` indicates MGBE0 supports 2/5/10G but `mgbe0-speed` property is "3" (=25Gbps).

/uphy/mgbe1-speed
This property configures MGBE1 controller speed. Valid options are: 0 for 2.5 Gbps, 1 for 5Gbps, 2 for 10Gbps, and 3 for 25Gbps. If this property is not present, UPHY1 lane 5 is disabled. If the specified speed isn't supported by the selected `uphy1-config`, BPMP-FW will raise Fatal Error and halt its execution. For example, `uphy1-config` indicates MGBE1 supports 2/5/10G but `mgbe1-speed` property is "3" (=25Gbps).

/uphy/mgbe2-speed
This property configures MGBE2 controller speed. Valid options are: 0 for 2.5 Gbps, 1 for 5Gbps, 2 for 10Gbps, and 3 for 25Gbps. If this property is not present, UPHY1 lane 6 is disabled. If the specified speed isn't supported by the selected `uphy1-config`, BPMP-FW will raise Fatal Error and halt its execution. For example, `uphy1-config` indicates MGBE2 supports 2/5/10G but `mgbe2-speed` property is "3" (=25Gbps).

/uphy/mgbe3-speed
This property configures MGBE3 controller speed. Valid options are: 0 for 2.5 Gbps, 1 for 5Gbps, 2 for 10Gbps, and 3 for 25Gbps. If this property is not present, UPHY1 lane 7 is disabled. If the specified speed isn't supported by the selected `uphy1-config`, BPMP-FW will raise Fatal Error and halt its execution. For example, `uphy1-config` indicates MGBE3 supports 2/5/10G but `mgbe3-speed` property is "3" (=25Gbps).

/uphy/uphy0-eqos-speed
This property configures UPHY0 EQOS lane speed. Valid options are: 0 for 1 Gbps, 1 for 2.5Gbps. If this property is not present, UPHY0 EQOS lane is configured with default 1 Gbps. If this property is assigned with an invalid value, BPMP-FW will raise Fatal Error and halt its execution.

/uphy/hsstp-lane-map
A 32-bit unsigned integer that identifies the lanes could be enabled for HSSTP. If `hsstp-lane-map` property is absent, HSSTP will not be enabled. Valid values are listed in following table. If a nonspecified value is provided, BPMP-FW will raise Fatal Error and halt its execution.

Option	Lane Mapping
1	UPHY0 Lane-0 mapped to HSSTP Lane-0
2	UPHY0 Lane-1 mapped to HSSTP Lane-0
3	UPHY0 Lane-1 mapped to HSSTP Lane-0 UPHY0 Lane-2 mapped to HSSTP Lane-1
4	UPHY0 Lane-2 mapped to HSSTP Lane-0
5	UPHY0 Lane-3 mapped to HSSTP Lane-0

/uphy/mgbe0-tx_drv_fir_cpre1

A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE1 for MGBE0 lane. Valid range is from 0x0 to 0x1F. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe0-tx_drv_fir_cpre2

A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE2 for MGBE0 lane. Valid range is from 0x0 to 0xB. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe0-tx_drv_fir_cpre3

A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE3 for MGBE0 lane. Valid range is from 0x0 to 0x7. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe0-tx_drv_fir_cpost1

A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPOST1 for MGBE0 lane. Valid range is from 0x0 to 0x1C. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe0-tx_drv_amp

A 32-bit unsigned integer that indicates the value to override TX_DRV_AMP for MGBE0 lane. Valid range is from 0x0 to 0x3F. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x3F is used.

/uphy/mgbe1-tx_drv_fir_cpre1

A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE1 for MGBE1 lane. Valid range is from 0x0 to 0x1F. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe1-tx_drv_fir_cpre2

A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE2 for MGBE1 lane. Valid range is from 0x0 to 0xB. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe1-tx_drv_fir_cpre3

A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE3 for MGBE1 lane. Valid range is from 0x0 to 0x7. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe1-tx_drv_fir_cpost1

A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPOST1 for MGBE1 lane. Valid range is from 0x0 to 0x1C. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe1-tx_drv_amp

A 32-bit unsigned integer that indicates the value to override TX_DRV_AMP for MGBE1 lane. Valid range is from 0x0 to 0x3F. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x3F is used.

/uphy/mgbe2-tx_drv_fir_cpre1
A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE1 for MGBE2 lane. Valid range is from 0x0 to 0x1F. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe2-tx_drv_fir_cpre2
A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE2 for MGBE2 lane. Valid range is from 0x0 to 0xB. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe2-tx_drv_fir_cpre3
A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE3 for MGBE2 lane. Valid range is from 0x0 to 0x7. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe2-tx_drv_fir_cpost1
A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPOST1 for MGBE2 lane. Valid range is from 0x0 to 0x1C. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe2-tx_drv_amp
A 32-bit unsigned integer that indicates the value to override TX_DRV_AMP for MGBE2 lane. Valid range is from 0x0 to 0x3F. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x3F is used.

/uphy/mgbe3-tx_drv_fir_cpre1
A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE1 for MGBE3 lane. Valid range is from 0x0 to 0x1F. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe3-tx_drv_fir_cpre2
A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE2 for MGBE3 lane. Valid range is from 0x0 to 0xB. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe3-tx_drv_fir_cpre3
A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPRE3 for MGBE3 lane. Valid range is from 0x0 to 0x7. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe3-tx_drv_fir_cpost1
A 32-bit unsigned integer that indicates the value to override TX_DRV_FIR_CPOST1 for MGBE3 lane. Valid range is from 0x0 to 0x1C. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x0 is used.

/uphy/mgbe3-tx_drv_amp
A 32-bit unsigned integer that indicates the value to override TX_DRV_AMP for MGBE3 lane. Valid range is from 0x0 to 0x3F. If the integer is out of range, BPMP-FW will raise Fatal Error and halt its execution. When this property is absent, default value 0x3F is used.

2.19 Appendix: Resource Identifiers

2.19.1 GPIO Port Identifiers

The BPMP-FW's GPIO framework is capable of controlling GPIOs on NVIDIA Tegra itself (but not those on external chips). The framework identifies each GPIO by a small integer ID. This binding doc describes how to map from a named NVIDIA Tegra GPIO to a GPIO ID understood by the firmware.

Tegra GPIOs are organized into ports of one to eight pins. A typical GPIO name looks like GPIO_PH.06. That indicates pin 6 of GPIO port H. The ports are named alphabetically (that is, A, B, C,…Z, AA, BB, …). However, the firmware numbers the ports: port A = port 0, port B = port 1, port Z = port 25, and so on.

The ID for a GPIO is 8 times its port number plus the number of the GPIO within the port. So, the ID for GPIO_PH.06 is 7*8 + 6 = 62. Similarly, the ID for GPIO_PB.00 is 1*8 + 0 = 8.

2.19.2 Clock Identifiers

```
/*
 * SPDX-FileCopyrightText: Copyright (c) 2022-2024 NVIDIA CORPORATION & AFFILIATES. All rights
reserved.
 * SPDX-License-Identifier: LicenseRef-NvidiaProprietary
 *
 * NVIDIA CORPORATION, its affiliates and licensors retain all intellectual
 * property and proprietary rights in and to this material, related
 * documentation and any modifications thereto. Any use, reproduction,
 * disclosure or distribution of this material and related documentation
 * without an express license agreement from NVIDIA CORPORATION or
 * its affiliates is strictly prohibited.
 */

#ifndef BPMP_ABI_MACH_T264_CLOCK_H
#define BPMP_ABI_MACH_T264_CLOCK_H

/**
 * @file
 * @defgroup bpmp_clock_ids Clock identifiers
 * @{
 */

#define TEGRA264_CLK_OSC                        1U
#define TEGRA264_CLK_CLK_S                      2U
#define TEGRA264_CLK_JTAG_REG                   3U
#define TEGRA264_CLK_SPLL                       4U
#define TEGRA264_CLK_SPLL_OUT0                  5U
#define TEGRA264_CLK_SPLL_OUT1                  6U
#define TEGRA264_CLK_SPLL_OUT2                  7U
#define TEGRA264_CLK_SPLL_OUT3                  8U
#define TEGRA264_CLK_SPLL_OUT4                  9U
#define TEGRA264_CLK_SPLL_OUT5                  10U
#define TEGRA264_CLK_SPLL_OUT6                  11U
#define TEGRA264_CLK_SPLL_OUT7                  12U
#define TEGRA264_CLK_AON_I2C                    13U
#define TEGRA264_CLK_HOST1X                     14U
#define TEGRA264_CLK_ISP                        15U
#define TEGRA264_CLK_ISP1                       16U
#define TEGRA264_CLK_ISP_ROOT                   17U
#define TEGRA264_CLK_NAFLL_PVA0_CORE            18U
#define TEGRA264_CLK_NAFLL_PVA0_VPS             19U
#define TEGRA264_CLK_NVCSI                      20U
#define TEGRA264_CLK_NVCSILP                    21U
#define TEGRA264_CLK_PLLP_OUT0                  22U
#define TEGRA264_CLK_PVA0_CPU_AXI               23U
#define TEGRA264_CLK_PVA0_VPS                   24U
```

```
#define TEGRA264_CLK_PWM10                      25U
#define TEGRA264_CLK_PWM2                       26U
#define TEGRA264_CLK_PWM3                       27U
#define TEGRA264_CLK_PWM4                       28U
#define TEGRA264_CLK_PWM5                       29U
#define TEGRA264_CLK_PWM9                       30U
#define TEGRA264_CLK_QSPI0                      31U
#define TEGRA264_CLK_QSPI0_2X_PM                32U
#define TEGRA264_CLK_RCE1_CPU                   33U
#define TEGRA264_CLK_RCE1_NIC                   34U
#define TEGRA264_CLK_RCE_CPU                    35U
#define TEGRA264_CLK_RCE_NIC                    36U
#define TEGRA264_CLK_SE                         37U
#define TEGRA264_CLK_SEU1                       38U
#define TEGRA264_CLK_SEU2                       39U
#define TEGRA264_CLK_SEU3                       40U
#define TEGRA264_CLK_SE_ROOT                    41U
#define TEGRA264_CLK_SPI1                       42U
#define TEGRA264_CLK_SPI2                       43U
#define TEGRA264_CLK_SPI3                       44U
#define TEGRA264_CLK_SPI4                       45U
#define TEGRA264_CLK_SPI5                       46U
#define TEGRA264_CLK_TOP_I2C                    47U
#define TEGRA264_CLK_TSEC                       48U
#define TEGRA264_CLK_TSEC_PKA                   49U
#define TEGRA264_CLK_UART0                      50U
#define TEGRA264_CLK_UART10                     51U
#define TEGRA264_CLK_UART11                     52U
#define TEGRA264_CLK_UART4                      53U
#define TEGRA264_CLK_UART5                      54U
#define TEGRA264_CLK_UART8                      55U
#define TEGRA264_CLK_UART9                      56U
#define TEGRA264_CLK_VI                         57U
#define TEGRA264_CLK_VI1                        58U
#define TEGRA264_CLK_VIC                        59U
#define TEGRA264_CLK_VI_ROOT                    60U
#define TEGRA264_CLK_DISPPLL                    61U
#define TEGRA264_CLK_SPPLL0                     62U
#define TEGRA264_CLK_SPPLL0_CLKOUT1A            63U
#define TEGRA264_CLK_SPPLL0_CLKOUT2A            64U
#define TEGRA264_CLK_SPPLL1                     65U
#define TEGRA264_CLK_VPLL0                      66U
#define TEGRA264_CLK_VPLL1                      67U
#define TEGRA264_CLK_VPLL2                      68U
#define TEGRA264_CLK_VPLL3                      69U
#define TEGRA264_CLK_VPLL4                      70U
#define TEGRA264_CLK_VPLL5                      71U
#define TEGRA264_CLK_VPLL6                      72U
#define TEGRA264_CLK_VPLL7                      73U
#define TEGRA264_CLK_RG0_DIV                    74U
#define TEGRA264_CLK_RG1_DIV                    75U
#define TEGRA264_CLK_RG2_DIV                    76U
#define TEGRA264_CLK_RG3_DIV                    77U
#define TEGRA264_CLK_RG4_DIV                    78U
#define TEGRA264_CLK_RG5_DIV                    79U
#define TEGRA264_CLK_RG6_DIV                    80U
#define TEGRA264_CLK_RG7_DIV                    81U
#define TEGRA264_CLK_RG0                        82U
#define TEGRA264_CLK_RG1                        83U
#define TEGRA264_CLK_RG2                        84U
#define TEGRA264_CLK_RG3                        85U
#define TEGRA264_CLK_RG4                        86U
#define TEGRA264_CLK_RG5                        87U
#define TEGRA264_CLK_RG6                        88U
```

```
#define TEGRA264_CLK_RG7                        89U
#define TEGRA264_CLK_DISP                       90U
#define TEGRA264_CLK_DSC                        91U
#define TEGRA264_CLK_DSC_ROOT                   92U  /* TODO: remove */
#define TEGRA264_CLK_HUB                        93U
#define TEGRA264_CLK_VPLLX_SOR0_MUXED           94U
#define TEGRA264_CLK_VPLLX_SOR1_MUXED           95U
#define TEGRA264_CLK_VPLLX_SOR2_MUXED           96U
#define TEGRA264_CLK_VPLLX_SOR3_MUXED           97U
#define TEGRA264_CLK_LINKA_SYM                  98U
#define TEGRA264_CLK_LINKB_SYM                  99U
#define TEGRA264_CLK_LINKC_SYM                  100U
#define TEGRA264_CLK_LINKD_SYM                  101U
#define TEGRA264_CLK_PRE_SOR0                   102U
#define TEGRA264_CLK_PRE_SOR1                   103U
#define TEGRA264_CLK_PRE_SOR2                   104U
#define TEGRA264_CLK_PRE_SOR3                   105U
#define TEGRA264_CLK_SOR0_PLL_REF               106U
#define TEGRA264_CLK_SOR1_PLL_REF               107U
#define TEGRA264_CLK_SOR2_PLL_REF               108U
#define TEGRA264_CLK_SOR3_PLL_REF               109U
#define TEGRA264_CLK_SOR0_PAD                   110U
#define TEGRA264_CLK_SOR1_PAD                   111U
#define TEGRA264_CLK_SOR2_PAD                   112U
#define TEGRA264_CLK_SOR3_PAD                   113U
#define TEGRA264_CLK_SOR0_REF                   114U
#define TEGRA264_CLK_SOR1_REF                   115U
#define TEGRA264_CLK_SOR2_REF                   116U
#define TEGRA264_CLK_SOR3_REF                   117U
#define TEGRA264_CLK_SOR0_DIV                   118U
#define TEGRA264_CLK_SOR1_DIV                   119U
#define TEGRA264_CLK_SOR2_DIV                   120U
#define TEGRA264_CLK_SOR3_DIV                   121U
#define TEGRA264_CLK_SOR0                       122U
#define TEGRA264_CLK_SOR1                       123U
#define TEGRA264_CLK_SOR2                       124U
#define TEGRA264_CLK_SOR3                       125U
#define TEGRA264_CLK_SF0_SOR                    126U
#define TEGRA264_CLK_SF1_SOR                    127U
#define TEGRA264_CLK_SF2_SOR                    128U
#define TEGRA264_CLK_SF3_SOR                    129U
#define TEGRA264_CLK_SF4_SOR                    130U
#define TEGRA264_CLK_SF5_SOR                    131U
#define TEGRA264_CLK_SF6_SOR                    132U
#define TEGRA264_CLK_SF7_SOR                    133U
#define TEGRA264_CLK_SF0                        134U
#define TEGRA264_CLK_SF1                        135U
#define TEGRA264_CLK_SF2                        136U
#define TEGRA264_CLK_SF3                        137U
#define TEGRA264_CLK_SF4                        138U
#define TEGRA264_CLK_SF5                        139U
#define TEGRA264_CLK_SF6                        140U
#define TEGRA264_CLK_SF7                        141U
#define TEGRA264_CLK_MAUD                       142U
#define TEGRA264_CLK_AZA_2XBIT                  143U
#define TEGRA264_CLK_DCE_CPU                    144U
#define TEGRA264_CLK_DCE_NIC                    145U
#define TEGRA264_CLK_PLLC4                      146U
#define TEGRA264_CLK_PLLC4_OUT0                 147U
#define TEGRA264_CLK_PLLC4_OUT1                 148U
#define TEGRA264_CLK_PLLC4_MUXED                149U
#define TEGRA264_CLK_SDMMC1                     150U
#define TEGRA264_CLK_SDMMC_LEGACY_TM            151U
#define TEGRA264_CLK_PLLC0                      152U
```

```
#define TEGRA264_CLK_NAFLL_BPMP                 153U
#define TEGRA264_CLK_PLLP_OUT_PDIV              154U
#define TEGRA264_CLK_DISP_ROOT                  155U
#define TEGRA264_CLK_ADSP                       156U
#define TEGRA264_CLK_PLLA                       157U
#define TEGRA264_CLK_PLLA1                      158U
#define TEGRA264_CLK_PLLA1_OUT1                 159U
#define TEGRA264_CLK_PLLAON                     160U
#define TEGRA264_CLK_PLLAON_APE                 161U
#define TEGRA264_CLK_PLLA_OUT0                  162U
#define TEGRA264_CLK_AHUB                       163U
#define TEGRA264_CLK_APE                        164U
#define TEGRA264_CLK_I2S1_SCLK_IN               165U
#define TEGRA264_CLK_I2S2_SCLK_IN               166U
#define TEGRA264_CLK_I2S3_SCLK_IN               167U
#define TEGRA264_CLK_I2S4_SCLK_IN               168U
#define TEGRA264_CLK_I2S5_SCLK_IN               169U
#define TEGRA264_CLK_I2S6_SCLK_IN               170U
#define TEGRA264_CLK_I2S7_SCLK_IN               171U
#define TEGRA264_CLK_I2S8_SCLK_IN               172U
#define TEGRA264_CLK_I2S9_SCLK_IN               173U
#define TEGRA264_CLK_I2S1_AUDIO_SYNC            174U
#define TEGRA264_CLK_I2S2_AUDIO_SYNC            175U
#define TEGRA264_CLK_I2S3_AUDIO_SYNC            176U
#define TEGRA264_CLK_I2S4_AUDIO_SYNC            177U
#define TEGRA264_CLK_I2S5_AUDIO_SYNC            178U
#define TEGRA264_CLK_I2S6_AUDIO_SYNC            179U
#define TEGRA264_CLK_I2S7_AUDIO_SYNC            180U
#define TEGRA264_CLK_I2S8_AUDIO_SYNC            181U
#define TEGRA264_CLK_DMIC1_AUDIO_SYNC           182U
#define TEGRA264_CLK_DSPK1_AUDIO_SYNC           183U
#define TEGRA264_CLK_I2S1                       184U
#define TEGRA264_CLK_I2S2                       185U
#define TEGRA264_CLK_I2S3                       186U
#define TEGRA264_CLK_I2S4                       187U
#define TEGRA264_CLK_I2S5                       188U
#define TEGRA264_CLK_I2S6                       189U
#define TEGRA264_CLK_I2S7                       190U
#define TEGRA264_CLK_I2S8                       191U
#define TEGRA264_CLK_I2S9                       192U
#define TEGRA264_CLK_DMIC1                      193U
#define TEGRA264_CLK_DMIC5                      194U
#define TEGRA264_CLK_DSPK1                      195U
#define TEGRA264_CLK_AON_CPU                    196U
#define TEGRA264_CLK_AON_NIC                    197U
#define TEGRA264_CLK_BPMP                       198U
#define TEGRA264_CLK_AXI_CBB                    199U
#define TEGRA264_CLK_FUSE                       200U
#define TEGRA264_CLK_TSENSE                     201U
#define TEGRA264_CLK_CSITE                      202U
#define TEGRA264_CLK_HCSITE                     203U
#define TEGRA264_CLK_DBGAPB                     204U
#define TEGRA264_CLK_LA                         205U
#define TEGRA264_CLK_PLLREFGP                   206U
#define TEGRA264_CLK_PLLE0                      207U
#define TEGRA264_CLK_UPHY0_PLL0_XDIG            208U
#define TEGRA264_CLK_EQOS_APP                   209U
#define TEGRA264_CLK_EQOS_MAC                   210U
#define TEGRA264_CLK_EQOS_MACSEC                211U
#define TEGRA264_CLK_EQOS_TX_PCS                212U
#define TEGRA264_CLK_MGBES_PTP_REF              213U
#define TEGRA264_CLK_MGBE0_UPHY1_PLL_XDIG       214U
#define TEGRA264_CLK_MGBE0_TX_PCS               215U
#define TEGRA264_CLK_MGBE0_MAC                  216U
```

```
#define TEGRA264_CLK_MGBE0_MACSEC                   217U
#define TEGRA264_CLK_MGBE0_APP                      218U
#define TEGRA264_CLK_MGBE1_UPHY1_PLL_XDIG           219U
#define TEGRA264_CLK_MGBE1_TX_PCS                   220U
#define TEGRA264_CLK_MGBE1_MAC                      221U
#define TEGRA264_CLK_MGBE1_MACSEC                   222U
#define TEGRA264_CLK_MGBE1_APP                      223U
#define TEGRA264_CLK_MGBE2_UPHY1_PLL_XDIG           224U
#define TEGRA264_CLK_MGBE2_TX_PCS                   225U
#define TEGRA264_CLK_MGBE2_MAC                      226U
#define TEGRA264_CLK_MGBE2_MACSEC                   227U
#define TEGRA264_CLK_MGBE2_APP                      228U
#define TEGRA264_CLK_MGBE3_UPHY1_PLL_XDIG           229U
#define TEGRA264_CLK_MGBE3_TX_PCS                   230U
#define TEGRA264_CLK_MGBE3_MAC                      231U
#define TEGRA264_CLK_MGBE3_MACSEC                   232U
#define TEGRA264_CLK_MGBE3_APP                      233U
#define TEGRA264_CLK_PLLREFUFS                      234U
#define TEGRA264_CLK_PLLREFUFS_CLKOUT624            235U
#define TEGRA264_CLK_PLLREFUFS_REFCLKOUT            236U
#define TEGRA264_CLK_PLLREFUFS_UFSDEV_REFCLKOUT     237U
#define TEGRA264_CLK_UFSHC_CG_SYS                   238U
#define TEGRA264_CLK_MPHY_L0_RX_LS_BIT_DIV          239U
#define TEGRA264_CLK_MPHY_L0_RX_LS_BIT              240U
#define TEGRA264_CLK_MPHY_L0_RX_LS_SYMB_DIV         241U
#define TEGRA264_CLK_MPHY_L0_RX_HS_SYMB_DIV         242U
#define TEGRA264_CLK_MPHY_L0_RX_SYMB                243U
#define TEGRA264_CLK_MPHY_L0_UPHY_TX_FIFO           244U
#define TEGRA264_CLK_MPHY_L0_TX_LS_3XBIT_DIV        245U
#define TEGRA264_CLK_MPHY_L0_TX_LS_SYMB_DIV         246U
#define TEGRA264_CLK_UPHY0_PLL4_XDIG                247U
#define TEGRA264_CLK_MPHY_L0_TX_HS_SYMB_DIV         248U
#define TEGRA264_CLK_MPHY_L0_TX_SYMB                249U
#define TEGRA264_CLK_MPHY_L0_TX_LS_3XBIT            250U
#define TEGRA264_CLK_MPHY_L0_RX_ANA                 251U
#define TEGRA264_CLK_MPHY_L1_RX_ANA                 252U
#define TEGRA264_CLK_MPHY_TX_1MHZ_REF               253U
#define TEGRA264_CLK_MPHY_CORE_PLL_FIXED            254U
#define TEGRA264_CLK_MPHY_IOBIST                    255U
#define TEGRA264_CLK_UFSHC_CG_SYS_DIV               256U
#define TEGRA264_CLK_XUSB1_CORE                     257U
#define TEGRA264_CLK_XUSB1_FALCON                   258U
#define TEGRA264_CLK_XUSB1_FS                       259U
#define TEGRA264_CLK_XUSB1_SS                       260U
#define TEGRA264_CLK_UPHY0_USB_P0_RX_CORE           261U
#define TEGRA264_CLK_UPHY0_USB_P1_RX_CORE           262U
#define TEGRA264_CLK_UPHY0_USB_P2_RX_CORE           263U
#define TEGRA264_CLK_UPHY0_USB_P3_RX_CORE           264U
#define TEGRA264_CLK_XUSB1_CLK480M_NVWRAP_CORE      265U
#define TEGRA264_CLK_XUSB1_CORE_HOST                266U
#define TEGRA264_CLK_XUSB1_CORE_DEV                 267U
#define TEGRA264_CLK_XUSB1_CORE_SUPERSPEED          268U
#define TEGRA264_CLK_XUSB1_FALCON_HOST              269U
#define TEGRA264_CLK_XUSB1_FALCON_SUPERSPEED        270U
#define TEGRA264_CLK_XUSB1_FS_HOST                  271U
#define TEGRA264_CLK_XUSB1_FS_DEV                   272U
#define TEGRA264_CLK_XUSB1_HS_HSICP                 273U
#define TEGRA264_CLK_XUSB1_SS_DEV                   274U
#define TEGRA264_CLK_XUSB1_SS_SUPERSPEED            275U
#define TEGRA264_CLK_AON_TOUCH                      276U
#define TEGRA264_CLK_AUD_MCLK                       277U
#define TEGRA264_CLK_EXTPERIPH1                     278U
#define TEGRA264_CLK_EXTPERIPH2                     279U
#define TEGRA264_CLK_EXTPERIPH3                     280U
```

```
#define TEGRA264_CLK_EXTPERIPH4                 281U
#define TEGRA264_CLK_JTAG_REG_UNGATED           282U
#define TEGRA264_CLK_IST_BUS                    283U
#define TEGRA264_CLK_IST_BUS_RIST_MCC           284U
#define TEGRA264_CLK_MATHS_SEC_RIST             285U
#define TEGRA264_CLK_NAFLL_IST                  286U
#define TEGRA264_CLK_RIST_ROOT                  287U
#define TEGRA264_CLK_IST_CONTROLLER_RIST        288U
#define TEGRA264_CLK_MSS_ENCRYPT                289U
#define TEGRA264_CLK_EMC                        290U
#define TEGRA264_CLK_SPPLL0_CLKOUT100           291U
#define TEGRA264_CLK_SPPLL0_CLKOUT270           292U
#define TEGRA264_CLK_SPPLL1_CLKOUT100           293U
#define TEGRA264_CLK_SPPLL1_CLKOUT270           294U
#define TEGRA264_CLK_DP_LINKA_REF               295U
#define TEGRA264_CLK_DP_LINKB_REF               296U
#define TEGRA264_CLK_DP_LINKC_REF               297U
#define TEGRA264_CLK_DP_LINKD_REF               298U
#define TEGRA264_CLK_PLLNVCSI                   299U
#define TEGRA264_CLK_PLLBPMPCAM                 300U
#define TEGRA264_CLK_UTMI_PLL1                  301U
#define TEGRA264_CLK_UTMI_PLL1_CLKOUT48         302U
#define TEGRA264_CLK_UTMI_PLL1_CLKOUT60         303U
#define TEGRA264_CLK_UTMI_PLL1_CLKOUT480        304U
#define TEGRA264_CLK_NAFLL_ISP                  305U
#define TEGRA264_CLK_NAFLL_RCE                  306U
#define TEGRA264_CLK_NAFLL_RCE1                 307U
#define TEGRA264_CLK_NAFLL_SE                   308U
#define TEGRA264_CLK_NAFLL_VI                   309U
#define TEGRA264_CLK_NAFLL_VIC                  310U
#define TEGRA264_CLK_NAFLL_DCE                  311U
#define TEGRA264_CLK_NAFLL_TSEC                 312U
#define TEGRA264_CLK_NAFLL_CPAIR0               313U
#define TEGRA264_CLK_NAFLL_CPAIR1               314U
#define TEGRA264_CLK_NAFLL_CPAIR2               315U
#define TEGRA264_CLK_NAFLL_CPAIR3               316U
#define TEGRA264_CLK_NAFLL_CPAIR4               317U
#define TEGRA264_CLK_NAFLL_CPAIR5               318U
#define TEGRA264_CLK_NAFLL_CPAIR6               319U
#define TEGRA264_CLK_NAFLL_GPU_SYS              320U
#define TEGRA264_CLK_NAFLL_GPU_NVD              321U
#define TEGRA264_CLK_NAFLL_GPU_UPROC            322U
#define TEGRA264_CLK_NAFLL_GPU_GPC0             323U
#define TEGRA264_CLK_NAFLL_GPU_GPC1             324U
#define TEGRA264_CLK_NAFLL_GPU_GPC2             325U
#define TEGRA264_CLK_SOR_LINKA_INPUT            326U
#define TEGRA264_CLK_SOR_LINKB_INPUT            327U
#define TEGRA264_CLK_SOR_LINKC_INPUT            328U
#define TEGRA264_CLK_SOR_LINKD_INPUT            329U
#define TEGRA264_CLK_SOR_LINKA_AFIFO            330U
#define TEGRA264_CLK_SOR_LINKB_AFIFO            331U
#define TEGRA264_CLK_SOR_LINKC_AFIFO            332U
#define TEGRA264_CLK_SOR_LINKD_AFIFO            333U
#define TEGRA264_CLK_I2S1_PAD_M                 334U
#define TEGRA264_CLK_I2S2_PAD_M                 335U
#define TEGRA264_CLK_I2S3_PAD_M                 336U
#define TEGRA264_CLK_I2S4_PAD_M                 337U
#define TEGRA264_CLK_I2S5_PAD_M                 338U
#define TEGRA264_CLK_I2S6_PAD_M                 339U
#define TEGRA264_CLK_I2S7_PAD_M                 340U
#define TEGRA264_CLK_I2S8_PAD_M                 341U
#define TEGRA264_CLK_I2S9_PAD_M                 342U
#define TEGRA264_CLK_BPMP_NIC                   343U
#define TEGRA264_CLK_CLK1M                      344U
```

```
#define TEGRA264_CLK_RDET                       345U
#define TEGRA264_CLK_ADC_SOC_REF                346U
#define TEGRA264_CLK_UPHY0_PLL0_TXREF           347U
#define TEGRA264_CLK_EQOS_TX                    348U
#define TEGRA264_CLK_EQOS_TX_M                  349U
#define TEGRA264_CLK_EQOS_RX_PCS_IN             350U
#define TEGRA264_CLK_EQOS_RX_PCS_M              351U
#define TEGRA264_CLK_EQOS_RX_IN                 352U
#define TEGRA264_CLK_EQOS_RX                    353U
#define TEGRA264_CLK_EQOS_RX_M                  354U
#define TEGRA264_CLK_MGBE0_UPHY1_PLL_TXREF      355U
#define TEGRA264_CLK_MGBE0_TX                   356U
#define TEGRA264_CLK_MGBE0_TX_M                 357U
#define TEGRA264_CLK_MGBE0_RX_PCS_IN            358U
#define TEGRA264_CLK_MGBE0_RX_PCS_M             359U
#define TEGRA264_CLK_MGBE0_RX_IN                360U
#define TEGRA264_CLK_MGBE0_RX_M                 361U
#define TEGRA264_CLK_MGBE1_UPHY1_PLL_TXREF      362U
#define TEGRA264_CLK_MGBE1_TX                   363U
#define TEGRA264_CLK_MGBE1_TX_M                 364U
#define TEGRA264_CLK_MGBE1_RX_PCS_IN            365U
#define TEGRA264_CLK_MGBE1_RX_PCS_M             366U
#define TEGRA264_CLK_MGBE1_RX_IN                367U
#define TEGRA264_CLK_MGBE1_RX_M                 368U
#define TEGRA264_CLK_MGBE2_UPHY1_PLL_TXREF      369U
#define TEGRA264_CLK_MGBE2_TX                   370U
#define TEGRA264_CLK_MGBE2_TX_M                 371U
#define TEGRA264_CLK_MGBE2_RX_PCS_IN            372U
#define TEGRA264_CLK_MGBE2_RX_PCS_M             373U
#define TEGRA264_CLK_MGBE2_RX_IN                374U
#define TEGRA264_CLK_MGBE2_RX_M                 375U
#define TEGRA264_CLK_MGBE3_UPHY1_PLL_TXREF      376U
#define TEGRA264_CLK_MGBE3_TX                   377U
#define TEGRA264_CLK_MGBE3_TX_M                 378U
#define TEGRA264_CLK_MGBE3_RX_PCS_IN            379U
#define TEGRA264_CLK_MGBE3_RX_PCS_M             380U
#define TEGRA264_CLK_MGBE3_RX_IN                381U
#define TEGRA264_CLK_MGBE3_RX_M                 382U
#define TEGRA264_CLK_UPHY0_USB_P0_TX_CORE       383U
#define TEGRA264_CLK_UPHY0_USB_P1_TX_CORE       384U
#define TEGRA264_CLK_UPHY0_USB_P2_TX_CORE       385U
#define TEGRA264_CLK_UPHY0_USB_P3_TX_CORE       386U
#define TEGRA264_CLK_UPHY0_USB_P0_TX            387U
#define TEGRA264_CLK_UPHY0_USB_P1_TX            388U
#define TEGRA264_CLK_UPHY0_USB_P2_TX            389U
#define TEGRA264_CLK_UPHY0_USB_P3_TX            390U
#define TEGRA264_CLK_UPHY0_USB_P0_RX_IN         391U
#define TEGRA264_CLK_UPHY0_USB_P1_RX_IN         392U
#define TEGRA264_CLK_UPHY0_USB_P2_RX_IN         393U
#define TEGRA264_CLK_UPHY0_USB_P3_RX_IN         394U
#define TEGRA264_CLK_UPHY0_USB_P0_RX_M          395U
#define TEGRA264_CLK_UPHY0_USB_P1_RX_M          396U
#define TEGRA264_CLK_UPHY0_USB_P2_RX_M          397U
#define TEGRA264_CLK_UPHY0_USB_P3_RX_M          398U
#define TEGRA264_CLK_UPHY0_LANE0_TX_M           399U
#define TEGRA264_CLK_PCIE_C1_XCLK_NOBG_M        400U
#define TEGRA264_CLK_PCIE_C2_XCLK_NOBG_M        401U
#define TEGRA264_CLK_PCIE_C3_XCLK_NOBG_M        402U
#define TEGRA264_CLK_PCIE_C4_XCLK_NOBG_M        403U
#define TEGRA264_CLK_PCIE_C5_XCLK_NOBG_M        404U
#define TEGRA264_CLK_PCIE_C1_L0_RX_M            405U
#define TEGRA264_CLK_PCIE_C1_L1_RX_M            406U
#define TEGRA264_CLK_PCIE_C1_L2_RX_M            407U
#define TEGRA264_CLK_PCIE_C1_L3_RX_M            408U
```

```
#define TEGRA264_CLK_PCIE_C2_L0_RX_M              409U
#define TEGRA264_CLK_PCIE_C2_L1_RX_M              410U
#define TEGRA264_CLK_PCIE_C2_L2_RX_M              411U
#define TEGRA264_CLK_PCIE_C2_L3_RX_M              412U
#define TEGRA264_CLK_PCIE_C3_L0_RX_M              413U
#define TEGRA264_CLK_PCIE_C3_L1_RX_M              414U
#define TEGRA264_CLK_PCIE_C4_L0_RX_M              415U
#define TEGRA264_CLK_PCIE_C4_L1_RX_M              416U
#define TEGRA264_CLK_PCIE_C4_L2_RX_M              417U
#define TEGRA264_CLK_PCIE_C4_L3_RX_M              418U
#define TEGRA264_CLK_PCIE_C4_L4_RX_M              419U
#define TEGRA264_CLK_PCIE_C4_L5_RX_M              420U
#define TEGRA264_CLK_PCIE_C4_L6_RX_M              421U
#define TEGRA264_CLK_PCIE_C4_L7_RX_M              422U
#define TEGRA264_CLK_PCIE_C5_L0_RX_M              423U
#define TEGRA264_CLK_PCIE_C5_L1_RX_M              424U
#define TEGRA264_CLK_PCIE_C5_L2_RX_M              425U
#define TEGRA264_CLK_PCIE_C5_L3_RX_M              426U
#define TEGRA264_CLK_MPHY_L0_RX_PWM_BIT_M         427U
#define TEGRA264_CLK_MPHY_L1_RX_PWM_BIT_M         428U
#define TEGRA264_CLK_DBB_UPHY0                    429U
#define TEGRA264_CLK_UPHY0_UXL_CORE               430U
#define TEGRA264_CLK_ISC_CPU_ROOT                 431U
#define TEGRA264_CLK_ISC_NIC                      432U
#define TEGRA264_CLK_CTC_TXCLK0_M                 433U
#define TEGRA264_CLK_CTC_TXCLK1_M                 434U
#define TEGRA264_CLK_CTC_RXCLK0_M                 435U
#define TEGRA264_CLK_CTC_RXCLK1_M                 436U
#define TEGRA264_CLK_PLLREFGP_OUT                 437U
#define TEGRA264_CLK_PLLREFGP_OUT1                438U
#define TEGRA264_CLK_GPU_SYS                      439U
#define TEGRA264_CLK_GPU_NVD                      440U
#define TEGRA264_CLK_GPU_UPROC                    441U
#define TEGRA264_CLK_GPU_GPC0                     442U
#define TEGRA264_CLK_GPU_GPC1                     443U
#define TEGRA264_CLK_GPU_GPC2                     444U
#define TEGRA264_CLK_PLLX                         445U
#define TEGRA264_CLK_APE_SOUNDWIRE_MSRC0          446U
#define TEGRA264_CLK_APE_SOUNDWIRE_DATA_EN_SHAPER 447U
#define TEGRA264_CLK_AO_SOUNDWIRE_MSRC0           448U
#define TEGRA264_CLK_AO_SOUNDWIRE_DATA_EN_SHAPER  449U
#define TEGRA264_CLK_MUX_CPU_SLC_JTAG_REG         450U
#define TEGRA264_CLK_PLLX_RIST                    451U
#define TEGRA264_CLK_NAFLL_AON                    452U
#define TEGRA264_CLK_SMMU                         453U
#define TEGRA264_CLK_PLLHUB1                      454U
#define TEGRA264_CLK_MCF                          455U
#define TEGRA264_CLK_SCF                          456U
#define TEGRA264_CLK_MPHY_L0_TX_SYMB_M            457U
#define TEGRA264_CLK_MPHY_L0_RX_SYMB_M            458U
#define TEGRA264_CLK_MGBE0_TX_SER                 459U
#define TEGRA264_CLK_MGBE1_TX_SER                 460U
#define TEGRA264_CLK_MGBE2_TX_SER                 461U
#define TEGRA264_CLK_MGBE3_TX_SER                 462U
#define TEGRA264_CLK_MGBE0_RX_SER                 463U
#define TEGRA264_CLK_MGBE1_RX_SER                 464U
#define TEGRA264_CLK_MGBE2_RX_SER                 465U
#define TEGRA264_CLK_MGBE3_RX_SER                 466U
#define TEGRA264_CLK_DPAUX                        467U
#define TEGRA264_CLK_NAFLL_MCF                    468U
#define TEGRA264_CLK_NAFLL_SCF                    469U

/** @brief Largest supported public clock identifier for this platform */
#define TEGRA264_MAX_PUBLIC_CLK_ID                469U
```

```
/** @} */

#endif
```

2.19.3 Doorbell Identifiers

```
/*
 * Copyright (c) 2016-2023, NVIDIA CORPORATION. All rights reserved.
 *
 * NVIDIA CORPORATION and its licensors retain all intellectual property
 * and proprietary rights in and to this software, related documentation
 * and any modifications thereto.  Any use, reproduction, disclosure or
 * distribution of this software and related documentation without an express
 * license agreement from NVIDIA CORPORATION is strictly prohibited.
 */

#ifndef ABI_MACH_T264_MAIL_H
#define ABI_MACH_T264_MAIL_H

/**
 * @file
 * @defgroup mail_ids Doorbell Identifiers
 * @brief Doorbell Identifiers
 * @{
 * @def TEGRA264_DB_GSP
 * @def TEGRA264_DB_DCE
 * @def TEGRA264_DB_RCE
 * @def TEGRA264_DB_BPMP
 * @def TEGRA264_DB_CPU_NS
 * @def TEGRA264_DB_CPU_S
 * @}
 */

/** @brief Doorbell ID for non secure GSP */
#define TEGRA264_DB_GSP  0x20U

/** @brief Doorbell ID for non secure DCE */
#define TEGRA264_DB_DCE  0x10U

/** @brief Doorbell ID for non secure RCE */
#define TEGRA264_DB_RCE  0x08U

/** @brief Doorbell ID for non secure BPMP */
#define TEGRA264_DB_BPMP 0x04U

/** @brief Doorbell ID for non secure CCPLEX */
#define TEGRA264_DB_CPU_NS 0x02U

/** @brief Doorbell ID for secure CCPLEX */
#define TEGRA264_DB_CPU_S 0x01U

#endif
```

2.19.4 Monitored Clocks Identifiers

2.19.5 Power Domain Identifiers

```
/*
```

```
 * Copyright (c) 2022-2024, NVIDIA CORPORATION.  All rights reserved.
 *
 * NVIDIA CORPORATION and its licensors retain all intellectual property
 * and proprietary rights in and to this software, related documentation
 * and any modifications thereto.  Any use, reproduction, disclosure or
 * distribution of this software and related documentation without an express
 * license agreement from NVIDIA CORPORATION is strictly prohibited.
 */

#ifndef BPMP_ABI_MACH_T264_POWERGATE_T264_H
#define BPMP_ABI_MACH_T264_POWERGATE_T264_H

/**
 * @file
 * @defgroup bpmp_pdomain_ids Power Domain ID's
 * This is a list of power domain IDs provided by the firmware.
 * @{
 */
#define TEGRA264_POWER_DOMAIN_DISP 1U
#define TEGRA264_POWER_DOMAIN_AUD 2U
/* RESERVED 3:9 */
#define TEGRA264_POWER_DOMAIN_XUSB_SS 10U
#define TEGRA264_POWER_DOMAIN_XUSB_DEV 11U
#define TEGRA264_POWER_DOMAIN_XUSB_HOST 12U
#define TEGRA264_POWER_DOMAIN_MGBE0 13U
#define TEGRA264_POWER_DOMAIN_MGBE1 14U
#define TEGRA264_POWER_DOMAIN_MGBE2 15U
#define TEGRA264_POWER_DOMAIN_MGBE3 16U
#define TEGRA264_POWER_DOMAIN_VI 17U
#define TEGRA264_POWER_DOMAIN_VIC 18U
#define TEGRA264_POWER_DOMAIN_ISP0 19U
#define TEGRA264_POWER_DOMAIN_ISP1 20U
#define TEGRA264_POWER_DOMAIN_PVA0 21U
#define TEGRA264_POWER_DOMAIN_GPU 22U

#define TEGRA264_POWER_DOMAIN_MAX 22U
/** @} */

#endif
```

2.19.6 Voltage Rail Identifiers

```
/*
 * Copyright (c) 2022-2023, NVIDIA CORPORATION
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
```

```
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

#ifndef BPMP_ABI_MACH_T264_REGULATOR_T264_H
#define BPMP_ABI_MACH_T264_REGULATOR_T264_H

/**
 * @file
 * @defgroup bpmp_rail_ids Voltage Rail ID's
 * @{
 */
#define TEGRA264_REGULATOR_RAIL_VDD_CPU  0U
#define TEGRA264_REGULATOR_RAIL_VDD_CORE 1U
#define TEGRA264_REGULATOR_RAIL_VDD_GPU  2U
#define TEGRA264_REGULATOR_RAIL_VDD_MSS  3U
#define TEGRA264_REGULATOR_RAIL_VDD_AON  4U
#define TEGRA264_REGULATOR_RAIL_MAX  5U
/** @} */

#endif
```

2.19.7 Reset Identifiers

```
/*
 * Copyright (c) 2022-2024, NVIDIA CORPORATION. All rights reserved.
 *
 * NVIDIA CORPORATION and its licensors retain all intellectual property
 * and proprietary rights in and to this software, related documentation
 * and any modifications thereto.  Any use, reproduction, disclosure or
 * distribution of this software and related documentation without an express
 * license agreement from NVIDIA CORPORATION is strictly prohibited.
 */

#ifndef BPMP_ABI_MACH_T264_RESET_H
#define BPMP_ABI_MACH_T264_RESET_H

/**
 * @file
 * @defgroup bpmp_reset_ids Reset ID's
 * @brief Identifiers for Resets controllable by firmware
 * @{
 */

#define TEGRA264_RESET_APE_TKE   1U
#define TEGRA264_RESET_CEC   2U
#define TEGRA264_RESET_ADSP_ALL   3U
#define TEGRA264_RESET_RCE_ALL   4U
#define TEGRA264_RESET_UFSHC   5U
#define TEGRA264_RESET_UFSHC_AXI_M  6U
#define TEGRA264_RESET_UFSHC_LP_SEQ  7U
#define TEGRA264_RESET_DPAUX   8U
#define TEGRA264_RESET_EQOS_PCS   9U
#define TEGRA264_RESET_HWPM   10U
#define TEGRA264_RESET_I2C1   11U
#define TEGRA264_RESET_I2C2   12U
#define TEGRA264_RESET_I2C3   13U
#define TEGRA264_RESET_I2C4   14U
#define TEGRA264_RESET_I2C6   15U
#define TEGRA264_RESET_I2C7   16U
#define TEGRA264_RESET_I2C8   17U
#define TEGRA264_RESET_I2C9   18U
#define TEGRA264_RESET_ISP   19U
```

```
#define TEGRA264_RESET_LA   20U
#define TEGRA264_RESET_NVCSI   21U
#define TEGRA264_RESET_EQOS_MAC   22U
#define TEGRA264_RESET_PWM10   23U
#define TEGRA264_RESET_PWM2   24U
#define TEGRA264_RESET_PWM3   25U
#define TEGRA264_RESET_PWM4   26U
#define TEGRA264_RESET_PWM5   27U
#define TEGRA264_RESET_PWM9   28U
#define TEGRA264_RESET_QSPI0   29U
#define TEGRA264_RESET_HDA   30U
#define TEGRA264_RESET_HDACODEC   31U
#define TEGRA264_RESET_I2C0   32U
#define TEGRA264_RESET_I2C10   33U
#define TEGRA264_RESET_SDMMC1   34U
#define TEGRA264_RESET_MIPI_CAL   35U
#define TEGRA264_RESET_SPI1   36U
#define TEGRA264_RESET_SPI2   37U
#define TEGRA264_RESET_SPI3   38U
#define TEGRA264_RESET_SPI4   39U
#define TEGRA264_RESET_SPI5   40U
/* RESERVED 41:43 */
#define TEGRA264_RESET_TACH0   44U
#define TEGRA264_RESET_TSEC   45U
#define TEGRA264_RESET_VI   46U
#define TEGRA264_RESET_VI1   47U
#define TEGRA264_RESET_PVA0_ALL   48U
#define TEGRA264_RESET_VIC   49U
#define TEGRA264_RESET_MPHY_CLK_CTL  50U
#define TEGRA264_RESET_MPHY_L0_RX  51U
#define TEGRA264_RESET_MPHY_L0_TX  52U
#define TEGRA264_RESET_MPHY_L1_RX  53U
#define TEGRA264_RESET_MPHY_L1_TX  54U
#define TEGRA264_RESET_ISP1   55U
#define TEGRA264_RESET_I2C11   56U
#define TEGRA264_RESET_I2C12   57U
#define TEGRA264_RESET_I2C14   58U
#define TEGRA264_RESET_I2C15   59U
#define TEGRA264_RESET_I2C16   60U
#define TEGRA264_RESET_EQOS_MACSEC  61U
#define TEGRA264_RESET_MGBE0_PCS  62U
#define TEGRA264_RESET_MGBE0_MAC  63U
#define TEGRA264_RESET_MGBE0_MACSEC  64U
#define TEGRA264_RESET_MGBE1_PCS  65U
#define TEGRA264_RESET_MGBE1_MAC  66U
#define TEGRA264_RESET_MGBE1_MACSEC  67U
#define TEGRA264_RESET_MGBE2_PCS  68U
#define TEGRA264_RESET_MGBE2_MAC  69U
#define TEGRA264_RESET_MGBE2_MACSEC  70U
#define TEGRA264_RESET_MGBE3_PCS  71U
#define TEGRA264_RESET_MGBE3_MAC  72U
#define TEGRA264_RESET_MGBE3_MACSEC  73U
#define TEGRA264_RESET_ADSP_CORE0  74U
#define TEGRA264_RESET_ADSP_CORE1  75U
#define TEGRA264_RESET_APE   76U
#define TEGRA264_RESET_XUSB1_PADCTL  77U
#define TEGRA264_RESET_AON_CPU_ALL  78U
/* RESERVED 79 */
#define TEGRA264_RESET_UART4   80U
#define TEGRA264_RESET_UART5   81U
#define TEGRA264_RESET_UART9   82U
#define TEGRA264_RESET_UART10   83U
#define TEGRA264_RESET_UART8   84U
#define TEGRA264_RESET_ADSPCSITEPTM  85U
```

```
#define TEGRA264_MAX_PUBLIC_RESET_ID  85U

/** @} */

#endif
```

2.19.8 Thermal Sensor Identifiers

```
/*
 * SPDX-FileCopyrightText: Copyright (c) 2023-2024 NVIDIA CORPORATION & AFFILIATES. All rights
reserved.
 * SPDX-License-Identifier: MIT
 *
 * Permission is hereby granted, free of charge, to any person obtaining a
 * copy of this software and associated documentation files (the "Software"),
 * to deal in the Software without restriction, including without limitation
 * the rights to use, copy, modify, merge, publish, distribute, sublicense,
 * and/or sell copies of the Software, and to permit persons to whom the
 * Software is furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
 * THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
 * FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
 * DEALINGS IN THE SOFTWARE.
 */

#ifndef BPMP_ABI_MACH_T264_THERMAL_H
#define BPMP_ABI_MACH_T264_THERMAL_H

/**
 * @file
 * @defgroup bpmp_thermal_ids Thermal Zone ID's
 * @{
 */

#define TEGRA264_THERMAL_ZONE_TJ_MAX  0U
#define TEGRA264_THERMAL_ZONE_TJ_MIN  1U
#define TEGRA264_THERMAL_ZONE_GPU_AVG  2U
#define TEGRA264_THERMAL_ZONE_CPU_AVG  3U
#define TEGRA264_THERMAL_ZONE_SOC_012_AVG 4U /* Powered by Vdd_SoC */
#define TEGRA264_THERMAL_ZONE_SOC_45_AVG 5U /* Powered by Vdd_MSS */
#define TEGRA264_THERMAL_ZONE_SOC_3_AVG  6U /* Powered by Uphy_Vdd */
#define TEGRA264_THERMAL_ZONE_GPU_MAX  7U
#define TEGRA264_THERMAL_ZONE_CPU_MAX  8U
#define TEGRA264_THERMAL_ZONE_SOC_012_MAX 9U /* Powered by Vdd_SoC */
#define TEGRA264_THERMAL_ZONE_SOC_345_MAX 10U /* Powered by Uphy_Vdd and Vdd_MSS */
#define TEGRA264_THERMAL_ZONE_TJ_AVG  11U

#define TEGRA264_THERMAL_ZONE_COUNT  12U

/** @} */

/* Deprecated */

#define TEGRA264_THERMAL_ZONE_SECURE0  0U
```

```
#define TEGRA264_THERMAL_ZONE_SECURE1  1U
#define TEGRA264_THERMAL_ZONE_SECURE2  2U
#define TEGRA264_THERMAL_ZONE_SECURE3  3U
#define TEGRA264_THERMAL_ZONE_SECURE4  4U
#define TEGRA264_THERMAL_ZONE_SECURE5  5U
#define TEGRA264_THERMAL_ZONE_GENERAL0  6U
#define TEGRA264_THERMAL_ZONE_GENERAL1  7U
#define TEGRA264_THERMAL_ZONE_GENERAL2  8U
#define TEGRA264_THERMAL_ZONE_GENERAL3  9U
#define TEGRA264_THERMAL_ZONE_GENERAL4  10U
#define TEGRA264_THERMAL_ZONE_GENERAL5  11U

#endif /* BPMP_ABI_MACH_T264_THERMAL_H */
```