

* Freescale QUICC Engine module (QE)

This represents qe module that is installed on PowerQUICC II Pro.

NOTE: This is an interim binding; it should be updated to fit in with the CPM binding later in this document.

Basically, it is a bus of devices, that could act more or less as a complete entity (UCC, USB etc). All of them should be siblings on the "root" qe node, using the common properties from there. The description below applies to the qe of MPC8360 and more nodes and properties would be extended in the future.

i) Root QE device

Required properties:

- compatible : should be "fsl,qe";
- model : precise model of the QE, Can be "QE", "CPM", or "CPM2"
- reg : offset and length of the device registers.
- bus-frequency : the clock frequency for QUICC Engine.
- fsl,qe-num-riscs: define how many RISC engines the QE has.
- fsl,qe-num-snums: define how many serial number(SNUM) the QE can use for the threads.

Optional properties:

- fsl,firmware-phandle:
 - Usage: required only if there is no fsl,qe-firmware child node
 - Value type: <phandle>
 - Definition: Points to a firmware node (see "QE Firmware Node" below) that contains the firmware that should be uploaded for this QE. The compatible property for the firmware node should say, "fsl,qe-firmware".

Recommended properties

- brg-frequency : the internal clock source frequency for baud-rate generators in Hz.

Example:

```
qe@e0100000 {
    #address-cells = <1>;
    #size-cells = <1>;
    #interrupt-cells = <2>;
    compatible = "fsl,qe";
    ranges = <0 e0100000 00100000>;
    reg = <e0100000 480>;
    brg-frequency = <0>;
    bus-frequency = <179A7B00>;
}
```

* Multi-User RAM (MURAM)

Required properties:

- compatible : should be "fsl,qe-muram", "fsl,cpm-muram".
- mode : the could be "host" or "slave".
- ranges : Should be defined as specified in 1) to describe the translation of MURAM addresses.
- data-only : sub-node which defines the address area under MURAM

qe.txt

bus that can be allocated as data/parameter

Example:

```
muram@10000 {
    compatible = "fsl,qe-muram", "fsl,cpm-muram";
    ranges = <0 00010000 0000c000>;

    data-only@0{
        compatible = "fsl,qe-muram-data",
                    "fsl,cpm-muram-data";
        reg = <0 c000>;
    };
};
```

* QE Firmware Node

This node defines a firmware binary that is embedded in the device tree, for the purpose of passing the firmware from bootloader to the kernel, or from the hypervisor to the guest.

The firmware node itself contains the firmware binary contents, a compatible property, and any firmware-specific properties. The node should be placed inside a QE node that needs it. Doing so eliminates the need for a `fsl,firmware-phandle` property. Other QE nodes that need the same firmware should define an `fsl,firmware-phandle` property that points to the firmware node in the first QE node.

The `fsl,firmware` property can be specified in the DTS (possibly using `incbin`) or can be inserted by the boot loader at boot time.

Required properties:

- compatible
 - Usage: required
 - Value type: <string>
 - Definition: A standard property. Specify a string that indicates what kind of firmware it is. For QE, this should be "fsl,qe-firmware".
- fsl,firmware
 - Usage: required
 - Value type: <prop-encoded-array>, encoded as an array of bytes
 - Definition: A standard property. This property contains the firmware binary "blob".

Example:

```
qe1@e0080000 {
    compatible = "fsl,qe";
    qe_firmware:qe-firmware {
        compatible = "fsl,qe-firmware";
        fsl,firmware = [0x70 0xcd 0x00 0x00 0x01 0x46 0x45 ...];
    };
    ...
};

qe2@e0090000 {
    compatible = "fsl,qe";
```

```
                                qe.txt
fsl,firmware-phandle = <&qe_firmware>;
...
};
```