Marvell Discovery mv64[345]6x System Controller chips
==========

The Marvell mv64[345]60 series of system controller chips contain many of the peripherals needed to implement a complete computer system. In this section, we define device tree nodes to describe the system controller chip itself and each of the peripherals which it contains. Compatible string values for each node are prefixed with the string "marvell,", for Marvell Technology Group Ltd.

1) The /system-controller node

This node is used to represent the system-controller and must be present when the system uses a system controller chip. The top-level system-controller node contains information that is global to all devices within the system controller chip. The node name begins with "system-controller" followed by the unit address, which is the base address of the memory-mapped register set for the system controller chip.

Required properties:

- ranges : Describes the translation of system controller addresses for memory mapped registers.
- clock-frequency: Contains the main clock frequency for the system controller chip.
- reg : This property defines the address and size of the memory-mapped registers contained within the system controller chip. The address specified in the "reg" property should match the unit address of the system-controller node.
- #address-cells : Address representation for system controller devices. This field represents the number of cells needed to represent the address of the memory-mapped registers of devices within the system controller chip.
- #size-cells : Size representation for the memory-mapped registers within the system controller chip.
- #interrupt-cells : Defines the width of cells used to represent interrupts.

Optional properties:

- model : The specific model of the system controller chip. Such as, "mv64360", "mv64460", or "mv64560".
- compatible : A string identifying the compatibility identifiers of the system controller chip.

The system-controller node contains child nodes for each system controller device that the platform uses. Nodes should not be created for devices which exist on the system controller chip but are not used

Example Marvell Discovery mv64360 system-controller node:

```
system-controller@f1000000 { /* Marvell Discovery mv64360 */
        #address-cells = <1>;
        #size-cells = <1>;
        model = "mv64360";                      /* Default */
```

```
        compatible = "marvell,mv64360";
        clock-frequency = <133333333>;
        reg = <0xf1000000 0x10000>;
        virtual-reg = <0xf1000000>;
        ranges = <0x88000000 0x88000000 0x1000000 /* PCI 0 I/O Space */
                  0x80000000 0x80000000 0x8000000 /* PCI 0 MEM Space */
                  0xa0000000 0xa0000000 0x4000000 /* User FLASH */
                  0x00000000 0xf1000000 0x0010000 /* Bridge's regs */
                  0xf2000000 0xf2000000 0x0040000>;/* Integrated SRAM */

        [ child node definitions... ]
 }
```

2) Child nodes of /system-controller

a) Marvell Discovery MDIO bus

The MDIO is a bus to which the PHY devices are connected. For each device that exists on this bus, a child node should be created. See the definition of the PHY node below for an example of how to define a PHY.

Required properties:
- #address-cells : Should be <1>
- #size-cells : Should be <0>
- device_type : Should be "mdio"
- compatible : Should be "marvell,mv64360-mdio"

Example:

```
  mdio {
          #address-cells = <1>;
          #size-cells = <0>;
          device_type = "mdio";
          compatible = "marvell,mv64360-mdio";

          ethernet-phy@0 {
                  ......
          };
  };
```

b) Marvell Discovery ethernet controller

The Discover ethernet controller is described with two levels of nodes. The first level describes an ethernet silicon block and the second level describes up to 3 ethernet nodes within that block. The reason for the multiple levels is that the registers for the node are interleaved within a single set of registers. The "ethernet-block" level describes the shared register set, and the "ethernet" nodes describe ethernet port-specific properties.

Ethernet block node

Required properties:

```
  - #address-cells : <1>
  - #size-cells : <0>
  - compatible : "marvell,mv64360-eth-block"
  - reg : Offset and length of the register set for this block

Example Discovery Ethernet block node:
  ethernet-block@2000 {
          #address-cells = <1>;
          #size-cells = <0>;
          compatible = "marvell,mv64360-eth-block";
          reg = <0x2000 0x2000>;
          ethernet@0 {
                  .......
          };
  };

Ethernet port node

Required properties:
  - device_type : Should be "network".
  - compatible : Should be "marvell,mv64360-eth".
  - reg : Should be <0>, <1>, or <2>, according to which registers
    within the silicon block the device uses.
  - interrupts : <a> where a is the interrupt number for the port.
  - interrupt-parent : the phandle for the interrupt controller
    that services interrupts for this device.
  - phy : the phandle for the PHY connected to this ethernet
    controller.
  - local-mac-address : 6 bytes, MAC address

Example Discovery Ethernet port node:
  ethernet@0 {
          device_type = "network";
          compatible = "marvell,mv64360-eth";
          reg = <0>;
          interrupts = <32>;
          interrupt-parent = <&PIC>;
          phy = <&PHY0>;
          local-mac-address = [ 00 00 00 00 00 00 ];
  };



c) Marvell Discovery PHY nodes

Required properties:
  - device_type : Should be "ethernet-phy"
  - interrupts : <a> where a is the interrupt number for this phy.
  - interrupt-parent : the phandle for the interrupt controller that
    services interrupts for this device.
  - reg : The ID number for the phy, usually a small integer

Example Discovery PHY node:
  ethernet-phy@1 {
          device_type = "ethernet-phy";
          compatible = "broadcom,bcm5421";
```

```
        interrupts = <76>;      /* GPP 12 */
        interrupt-parent = <&PIC>;
        reg = <1>;
  };
```

d) Marvell Discovery SDMA nodes

Represent DMA hardware associated with the MPSC (multiprotocol serial controllers).

Required properties:
- compatible : "marvell,mv64360-sdma"
- reg : Offset and length of the register set for this device
- interrupts : <a> where a is the interrupt number for the DMA device.
- interrupt-parent : the phandle for the interrupt controller that services interrupts for this device.

Example Discovery SDMA node:

```
  sdma@4000 {
        compatible = "marvell,mv64360-sdma";
        reg = <0x4000 0xc18>;
        virtual-reg = <0xf1004000>;
        interrupts = <36>;
        interrupt-parent = <&PIC>;
  };
```

e) Marvell Discovery BRG nodes

Represent baud rate generator hardware associated with the MPSC (multiprotocol serial controllers).

Required properties:
- compatible : "marvell,mv64360-brg"
- reg : Offset and length of the register set for this device
- clock-src : A value from 0 to 15 which selects the clock source for the baud rate generator. This value corresponds to the CLKS value in the BRGx configuration register. See the mv64x60 User's Manual.
- clock-frequence : The frequency (in Hz) of the baud rate generator's input clock.
- current-speed : The current speed setting (presumably by firmware) of the baud rate generator.

Example Discovery BRG node:

```
  brg@b200 {
        compatible = "marvell,mv64360-brg";
        reg = <0xb200 0x8>;
        clock-src = <8>;
        clock-frequency = <133333333>;
        current-speed = <9600>;
  };
```

f) Marvell Discovery CUNIT nodes

Represent the Serial Communications Unit device hardware.

Required properties:
  - reg : Offset and length of the register set for this device

Example Discovery CUNIT node:
  cunit@f200 {
          reg = <0xf200 0x200>;
  };

g) Marvell Discovery MPSCROUTING nodes

Represent the Discovery's MPSC routing hardware

Required properties:
  - reg : Offset and length of the register set for this device

Example Discovery CUNIT node:
  mpscrouting@b500 {
          reg = <0xb400 0xc>;
  };

h) Marvell Discovery MPSCINTR nodes

Represent the Discovery's MPSC DMA interrupt hardware registers
(SDMA cause and mask registers).

Required properties:
  - reg : Offset and length of the register set for this device

Example Discovery MPSCINTR node:
  mpsintr@b800 {
          reg = <0xb800 0x100>;
  };

i) Marvell Discovery MPSC nodes

Represent the Discovery's MPSC (Multiprotocol Serial Controller)
serial port.

Required properties:
  - device_type : "serial"
  - compatible : "marvell,mv64360-mpsc"
  - reg : Offset and length of the register set for this device
  - sdma : the phandle for the SDMA node used by this port
  - brg : the phandle for the BRG node used by this port
  - cunit : the phandle for the CUNIT node used by this port
  - mpscrouting : the phandle for the MPSCROUTING node used by this port
  - mpscintr : the phandle for the MPSCINTR node used by this port
  - cell-index : the hardware index of this cell in the MPSC core
  - max_idle : value needed for MPSC CHR3 (Maximum Frame Length)

```
    register
  - interrupts : <a> where a is the interrupt number for the MPSC.
  - interrupt-parent : the phandle for the interrupt controller
    that services interrupts for this device.

Example Discovery MPSCINTR node:
  mpsc@8000 {
          device_type = "serial";
          compatible = "marvell,mv64360-mpsc";
          reg = <0x8000 0x38>;
          virtual-reg = <0xf1008000>;
          sdma = <&SDMA0>;
          brg = <&BRG0>;
          cunit = <&CUNIT>;
          mpscrouting = <&MPSCROUTING>;
          mpscintr = <&MPSCINTR>;
          cell-index = <0>;
          max_idle = <40>;
          interrupts = <40>;
          interrupt-parent = <&PIC>;
  };


j) Marvell Discovery Watch Dog Timer nodes

Represent the Discovery's watchdog timer hardware

Required properties:
  - compatible : "marvell,mv64360-wdt"
  - reg : Offset and length of the register set for this device

Example Discovery Watch Dog Timer node:
  wdt@b410 {
          compatible = "marvell,mv64360-wdt";
          reg = <0xb410 0x8>;
  };


k) Marvell Discovery I2C nodes

Represent the Discovery's I2C hardware

Required properties:
  - device_type : "i2c"
  - compatible : "marvell,mv64360-i2c"
  - reg : Offset and length of the register set for this device
  - interrupts : <a> where a is the interrupt number for the I2C.
  - interrupt-parent : the phandle for the interrupt controller
    that services interrupts for this device.

Example Discovery I2C node:
          compatible = "marvell,mv64360-i2c";
          reg = <0xc000 0x20>;
          virtual-reg = <0xf100c000>;
          interrupts = <37>;
          interrupt-parent = <&PIC>;
```

```
  };
```

l) Marvell Discovery PIC (Programmable Interrupt Controller) nodes

Represent the Discovery's PIC hardware

Required properties:
- #interrupt-cells : <1>
- #address-cells : <0>
- compatible : "marvell,mv64360-pic"
- reg : Offset and length of the register set for this device
- interrupt-controller

Example Discovery PIC node:

```
  pic {
          #interrupt-cells = <1>;
          #address-cells = <0>;
          compatible = "marvell,mv64360-pic";
          reg = <0x0 0x88>;
          interrupt-controller;
  };
```

m) Marvell Discovery MPP (Multipurpose Pins) multiplexing nodes

Represent the Discovery's MPP hardware

Required properties:
- compatible : "marvell,mv64360-mpp"
- reg : Offset and length of the register set for this device

Example Discovery MPP node:

```
  mpp@f000 {
          compatible = "marvell,mv64360-mpp";
          reg = <0xf000 0x10>;
  };
```

n) Marvell Discovery GPP (General Purpose Pins) nodes

Represent the Discovery's GPP hardware

Required properties:
- compatible : "marvell,mv64360-gpp"
- reg : Offset and length of the register set for this device

Example Discovery GPP node:

```
  gpp@f000 {
          compatible = "marvell,mv64360-gpp";
          reg = <0xf100 0x20>;
  };
```

o) Marvell Discovery PCI host bridge node

Represents the Discovery's PCI host bridge device. The properties for this node conform to Rev 2.1 of the PCI Bus Binding to IEEE 1275-1994. A typical value for the compatible property is "marvell,mv64360-pci".

Example Discovery PCI host bridge node

```
  pci@80000000 {
          #address-cells = <3>;
          #size-cells = <2>;
          #interrupt-cells = <1>;
          device_type = "pci";
          compatible = "marvell,mv64360-pci";
          reg = <0xcf8 0x8>;
          ranges = <0x01000000 0x0          0x0
                          0x88000000 0x0 0x01000000
                    0x02000000 0x0 0x80000000
                          0x80000000 0x0 0x08000000>;
          bus-range = <0 255>;
          clock-frequency = <66000000>;
          interrupt-parent = <&PIC>;
          interrupt-map-mask = <0xf800 0x0 0x0 0x7>;
          interrupt-map = <
                  /* IDSEL 0x0a */
                  0x5000 0 0 1 &PIC 80
                  0x5000 0 0 2 &PIC 81
                  0x5000 0 0 3 &PIC 91
                  0x5000 0 0 4 &PIC 93

                  /* IDSEL 0x0b */
                  0x5800 0 0 1 &PIC 91
                  0x5800 0 0 2 &PIC 93
                  0x5800 0 0 3 &PIC 80
                  0x5800 0 0 4 &PIC 81

                  /* IDSEL 0x0c */
                  0x6000 0 0 1 &PIC 91
                  0x6000 0 0 2 &PIC 93
                  0x6000 0 0 3 &PIC 80
                  0x6000 0 0 4 &PIC 81

                  /* IDSEL 0x0d */
                  0x6800 0 0 1 &PIC 93
                  0x6800 0 0 2 &PIC 80
                  0x6800 0 0 3 &PIC 81
                  0x6800 0 0 4 &PIC 91
          >;
  };
```

p) Marvell Discovery CPU Error nodes

Represent the Discovery's CPU error handler device.

Required properties:
- compatible : "marvell,mv64360-cpu-error"
- reg : Offset and length of the register set for this device

  - interrupts : the interrupt number for this device
  - interrupt-parent : the phandle for the interrupt controller
    that services interrupts for this device.

Example Discovery CPU Error node:
  cpu-error@0070 {
          compatible = "marvell,mv64360-cpu-error";
          reg = <0x70 0x10 0x128 0x28>;
          interrupts = <3>;
          interrupt-parent = <&PIC>;
  };

q) Marvell Discovery SRAM Controller nodes

Represent the Discovery's SRAM controller device.

Required properties:
  - compatible : "marvell,mv64360-sram-ctrl"
  - reg : Offset and length of the register set for this device
  - interrupts : the interrupt number for this device
  - interrupt-parent : the phandle for the interrupt controller
    that services interrupts for this device.

Example Discovery SRAM Controller node:
  sram-ctrl@0380 {
          compatible = "marvell,mv64360-sram-ctrl";
          reg = <0x380 0x80>;
          interrupts = <13>;
          interrupt-parent = <&PIC>;
  };

r) Marvell Discovery PCI Error Handler nodes

Represent the Discovery's PCI error handler device.

Required properties:
  - compatible : "marvell,mv64360-pci-error"
  - reg : Offset and length of the register set for this device
  - interrupts : the interrupt number for this device
  - interrupt-parent : the phandle for the interrupt controller
    that services interrupts for this device.

Example Discovery PCI Error Handler node:
  pci-error@1d40 {
          compatible = "marvell,mv64360-pci-error";
          reg = <0x1d40 0x40 0xc28 0x4>;
          interrupts = <12>;
          interrupt-parent = <&PIC>;
  };

s) Marvell Discovery Memory Controller nodes

Represent the Discovery's memory controller device.

```
Required properties:
  - compatible : "marvell,mv64360-mem-ctrl"
  - reg : Offset and length of the register set for this device
  - interrupts : the interrupt number for this device
  - interrupt-parent : the phandle for the interrupt controller
    that services interrupts for this device.

Example Discovery Memory Controller node:
  mem-ctrl@1400 {
          compatible = "marvell,mv64360-mem-ctrl";
          reg = <0x1400 0x60>;
          interrupts = <17>;
          interrupt-parent = <&PIC>;
  };
```