Freescale Semiconductor
Application Note

Document Number: AN4199
Rev. 0, 09/2010

Using the i.MX28 Power Management Unit and Battery Charger

Getting Started with Power Supply

by *Multimedia Applications Division*
Freescale Semiconductor, Inc.
Austin, TX

Contents

1 Introduction

This application note describes the low-level blocks of the Power Management Unit (PMU). This includes the basic power-supply and power-rail configurations and how to use these building blocks to perform useful functions in the i.MX28 processor, such as handling power source transitions and raising and lowering power rails. The application note also describes the battery charger block and provides instruction to charge a battery. The power savings features of the i.MX28 processor are also described in this application note.

2 System Overview

The i.MX28 application processor integrates a highly-efficient and comprehensive power supply. This power supply is powered by a Li-ion battery or 5 V power source and generates five internal power rails through two separate supplies.

The hardware in the PMU is configured through firmware. The i.MX28 processor requires direct access to the registers in the power block. Though this increases the system

freescale™ semiconductor

complexity, it provides firmware the ability to optimize the chip power beyond that of a PMU.

NOTE

The information provided in this application note is intended to supplement the information in the *i.MX28 Applications Processor Reference Manual* (MC1MX28RM).

The abbreviated form of the register names and bit fields that are frequently referred in this application note are given in Table 1.

Table 1. Glossary

Term	Definitions
DC-DC	Is the only internal DC-DC switching converter in the i.MX233 processor. This is sometimes referred as DCDC (without the hyphen) in the register names and bit fields.
LinReg	Common abbreviation used for the linear regulator.
4P2	Alternate name for the 4.2 V rail. Here, P is the abbreviation used for point while referring to rail in the data sheet or software.

The PMU consists of the following parts:

- Power sources—provide power to the internal power supplies
- Power supplies—convert the input sources to the voltage levels that the i.MX28 processor can use
- Power rails—fed with the supply output
- Battery charger—activated by a 5 V supply that charges the attached Li-ion battery. The architecture of the i.MX28 processor allows the battery to be fully charged while powering devices through a separate power supply.

Figure 1 shows the logical diagram of the PMU.

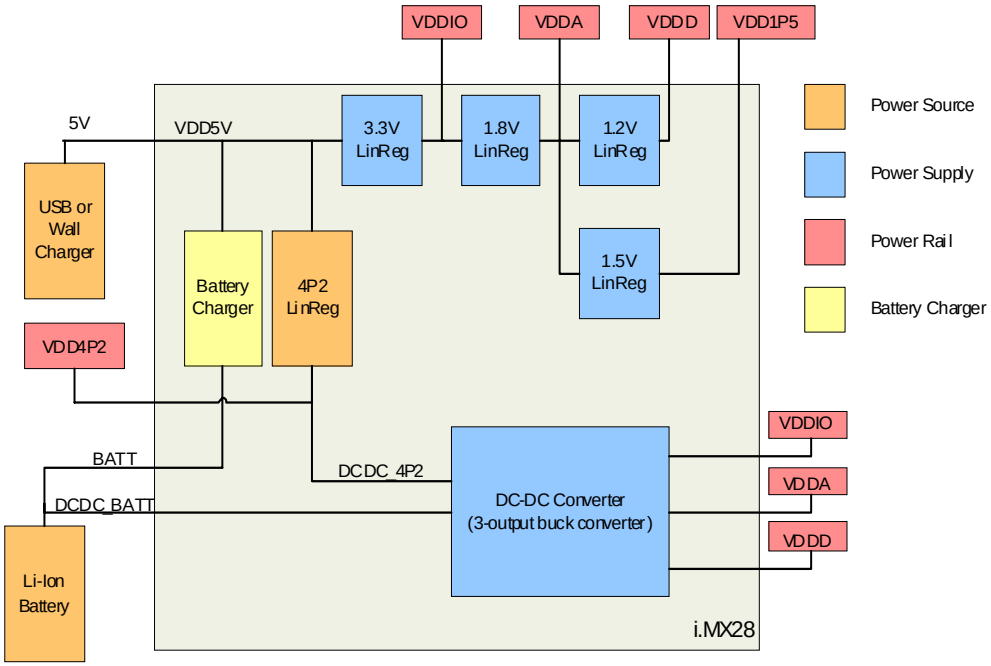

Figure 1. i.MX28 PMU

2.1 Power Sources

The i.MX28 processor can be powered from the following sources:

- Li-ion battery—output voltage should be in the range of 3.0–4.2 V when the device is powered.
- 5 V power source—taken from a Universal Serial Bus (USB) connection or wall charger that converts the wall power to 5 V. The voltage should not drop below 4.4 V as this can trigger a 5 V loss event in the device. When the power source provides a 5 V output, the i.MX28 processor turns ON an internal 4.2 V LinReg which serves as an input to the DC-DC converter.

The battery charger can also be considered as a power source. However, it provides power only to the battery and not directly to the system.

2.2 Power Supplies

The power supplies in the i.MX28 processor are as follows:

- DC-DC switching converter—uses the input from the Li-ion battery or internally generated 4P2 rail to supply power to three output rails.
- Multiple LinRegs—contains a group of LinRegs where each takes a single input and regulates the voltage to their target output level. The LinRegs for the power rails have a daisy-chained architecture where the output voltage is available from higher to lower values. In cases where the input is an internal power rail, the LinReg can alternatively be powered from the DC-DC output of the rail.

The 4.2 V LinReg uses the 5 V supply to provide 4.2 V to the DC-DC converter. This is called the 4P2 rail and it takes input only from the DC-DC converter. This allows the i.MX28 processor to use the more efficient DC-DC converters instead of internal LinRegs when the 5 V supply is powered-ON. This is because the most power efficient system uses the DC-DC converter regardless of the power source. The LinRegs should only be used to power the rails during boot up when the 4.2 V LinReg is not initialized.

NOTE

Generally, the 4P2 LinReg is referred as an internal power source for the i.MX28 processor and also referred as the 4P2 power rail as it creates an output rail. In this application note, the 4P2 power source refers to the actual LinReg and the 4P2 rail refers to the LinReg output.

2.3 Power Rails

The i.MX28 processor has five output power rails and are as follows:

- VDDD—provides power to the digital components of the i.MX28 processor. The system clocks use the VDDD power rail.
- VDDA—provides power specifically to the audio systems, headphone amp. The VDDA power rail also provides power to regulate the VDDMEM rail.
- VDDIO—provides power to the I/O peripherals and also to the NAND Flash and external Secure Digital/MultiMedia Cards (SD/MMC).
- VDDMEM—provides power to the external memories such as Low-Power Double Data Rate 1 (LP-DDR1) and Low-Power Double Data Rate 2 (LV-DDR2).
- VDD4P2—takes power from an external 5 V supply to act as an alternative DC-DC power source.

2.4 Battery Charger

The integrated battery charger can fully charge a Li-ion battery with a 5 V supply. The PMU should be configured to use the 4P2 source for the DC-DC converter when a load is removed from a battery that is being charged. This allows the battery to reach its fully charged state even if the battery is loaded normally.

The capacity of a battery is expressed in milli ampere hour (mAh). A 600 mAh battery can deliver an output current of 600 mA for 1 hour or 100 mA for 6 hours. The typical charging current of a Li-ion battery is 1C and 1C charging current for a 600 mAh battery is 600 mA. The maximum available charging current from the i.MX28 processor is 785 mA and the Li-ion battery should be charged at the rate of at least 1 C or 785 mA or with a VDD5V current limit.

2.5 Hardware Requirements

The i.MX28 processor is a highly integrated chip. However, the processor requires a few external components. The reference schematics for the processor is available at the link, https://www.freescale.com/imx28.

2.5.1 Inductor

An inductor is required for the DC-DC switching converter. The size of the inductor depends on the application and product specifications. For more information, refer to the *i.MX28 Reference Schematics* available at the link, https://www.freescale.com/imx28.

2.5.2 Capacitor

Capacitors are used for the PMU. Each power rail requires a capacitor of at least 33 µF for decoupling in addition to the capacitors of 0.01 µF, 0.1 µF, and 1 µF. Additionally, the 4P2 rail, battery, and VDD5V rail require capacitors ranging from 0.01 µF to 33 µF. For the exact capacitor locations and sizes, refer to the *i.MX28 Reference Schematics*.

2.5.3 Li-Ion Battery

The i.MX28 processor can operate without a battery if the 5 V supply is powered-ON. However, this is applicable only for permanent power situations. In other cases, the device requires a Li-ion battery that provides an output voltage of 3.0–4.2V.

2.5.4 External Power Supply

The internal DC-DC converter can deliver an output power up to 1.27 W from a battery of 3.0 V. For applications that require high powers, an external DC-DC converter is required to off load some of the power from the internal DC-DC converter. For more information, refer to the *i.MX28 Reference Schematics*.

3 Getting Started with Power Subsystem

This section provides a detailed description of the power sources and power supplies.

3.1 Power Sources

The external power sources used by the i.MX28 processor are as follows:

- 5 V supply
- Battery

These power sources are not required to be operated simultaneously. The PMU uses the 5 V supply to power the internal 4P2 power source to generate the 4P2 rail. This 4P2 rail is used as an input to the DC-DC converter.

3.1.1 Battery Source

The Li-ion battery provides power to the i.MX28 processor. The battery voltage should be in the range of 3.0–4.2 V to provide a reliable power to the processor. The DC-DC converter starts to operate when the battery power is detected and PSWITCH button is pressed. The rails are then raised to their default values and the Read Only Memory (ROM) begins to execute. The battery voltage monitor, which is managed by the firmware, should alert the system when the voltage reaches a critically low level. This prevents brownouts and unexpected device shut down.

3.1.2 5 V Supply

The 5 V power source can be taken from a USB connection or wall-powered source. The wall-powered source converts the AC wall power to a 5 V DC power before the i.MX28 processor is powered. The i.MX28 processor can be programmed to distinguish between the USB and wall-power sources by sensing the status of the D+/D- USB signals.

When a 5 V supply is detected, the internal LinRegs powers ON the i.MX28 processor. The power rails are raised to their default values and the ROM begins to process. When the i.MX28 processor is booted, the device enables the 4.2 V LinReg to supply power to the DC-DC converter instead of using the internal LinRegs to power the rails.

The 5 V-removal detection should be set up to alert the system to switch to the battery power. However, hardware can be configured to switch automatically to the battery power when the 5 V input supply is powered-OFF.

3.1.2.1 5 V-Detection Methods

There are two 5 V-detection methods available in the i.MX28 processor—VBUSVALID and VDD5V_GT_VDDIO. Each method detects the 5 V voltage value differently. Additionally, the 5 V-detection method must be configured once for the hardware (the DC-DC control logic) and software. The hardware configuration for the DC-DC control logic is set by software through the register settings.

VBUSVALID

The VBUSVALID detection method is the most accurate 5 V-detection method in the i.MX28 processor. It is recommended to use this method during the normal operation. The VBUSVALID method compares the VDD5V voltage level with an internal bias voltage. The threshold value for a valid 5 V status is configured in the software. Additionally, the VBUSVALID method can be configured to generate an interrupt signal when the 5 V supply is removed or inserted.

VDD5V_GT_VDDIO

The VDD5V_GT_VDDIO detection method is used at the device startup. This allows the DC-DC converter to determine the status of the 5 V source without enabling extra comparators. When the software gains control over the processor, the VBUSVALID method is initialized and used as the 5 V-detection method.

The VDD5V_GT_VDDIO method compares VDD5V with the VDDIO voltage plus an offset. The offset is approximately 600 mV. This detection method is not robust for detection when the 5 V supply is

removed. This is because the VDDIO LinReg is powered from the VDD5V rail and during heavy loads on the VDDIO rail, the VDDIO voltage drops at the same rate as that of VDD5V. This delays the detection when the 5 V source is removed. However, this method works for the device startup and is not recommended during the normal operation. Therefore, it is recommended to use the VBUSVALID 5 V-detection method for applications that require accurate 5 V removal detection.

3.1.2.2 Selecting 5 V-Detection Method

The DC-DC control logic requires a 5 V supply for the operation. By default, the DC-DC control logic uses the VDD5V_GT_VDDIO detection method. However, this is changed to the VBUSVALID method during the device initialization for the proper operation (for more information, refer to VBUSVALID).

Generally, the 5 V-detection threshold is set to 4 V. However, the threshold value is configurable and can be changed based on the required application. The steps to change the 5 V-detection method from VDD5V_GT_VDDIO to VBUSVALID are as follows:

1. Turn ON the VBUS comparators:
   `HW_POWER_5VCTRL[PWRUP_VBUS_CMPS] = 1`
2. Set the VBUS threshold value to 4.0 V (register setting—`0x1`):
   `HW_POWER_5VCTRL[VBUSVALID_TRSH] = 0x1`
3. Change the 5 V-detection method to VBUSVALID:
   `HW_POWER_5VCTRL[VBUSVALID_5VDETECT] = 1`

3.1.2.3 Enabling 5 V-Detection Interrupts

The i.MX28 processor generates interrupts when the 5 V supply is inserted or removed. The device is detected with an interrupt when the threshold value is crossed with the correct polarity. The 5 V interrupt has a standard Interrupt Request (IRQ) status and enables the corresponding bit set. The bit set includes a polarity bit that triggers when the voltage value exceeds or falls behind 5 V. This triggers an interrupt in the Interrupt Collector (ICOLL) block which is handled by the software.

The steps to configure the VBUSVALID detection method to generate interrupt when the 5 V supply is removed are as follows:

1. Set the threshold voltage for the VBUSVALID detection method to 4.0 V (register setting—`0x1`):
   `HW_POWER_5VCTRL[VBUSVALID_TRSH] = 0x1`
2. Clear the 5 V-interrupt status bit by writing 1 to the SCT clear address space of the IRQ status bit:
   `HW_POWER_CTRL_CLR[VBUSVALID_IRQ] = 1`
3. Set the polarity field to detect when the 5 V supply is removed:
   `HW_POWER_CTRL[POLARITY_VBUSVALID] = 0`
4. Initialize the interrupt handler in the ICOLL block.
5. Enable the VBUSVALID interrupt:
   `HW_POWER_CTRL[ENIRQ_VBUS_VALID]`

3.1.3 4P2 Power Source

The 4.2 V power source powers the 4P2 rail from the 5 V power supply. The 4.2 V output is used as an input to the DC-DC converter. The 4.2 V power source should be used when the 5 V supply is powered-ON to achieve the best power efficiency.

Figure 2 shows how a 5 V power is converted to the three power rails using the DC-DC converter.

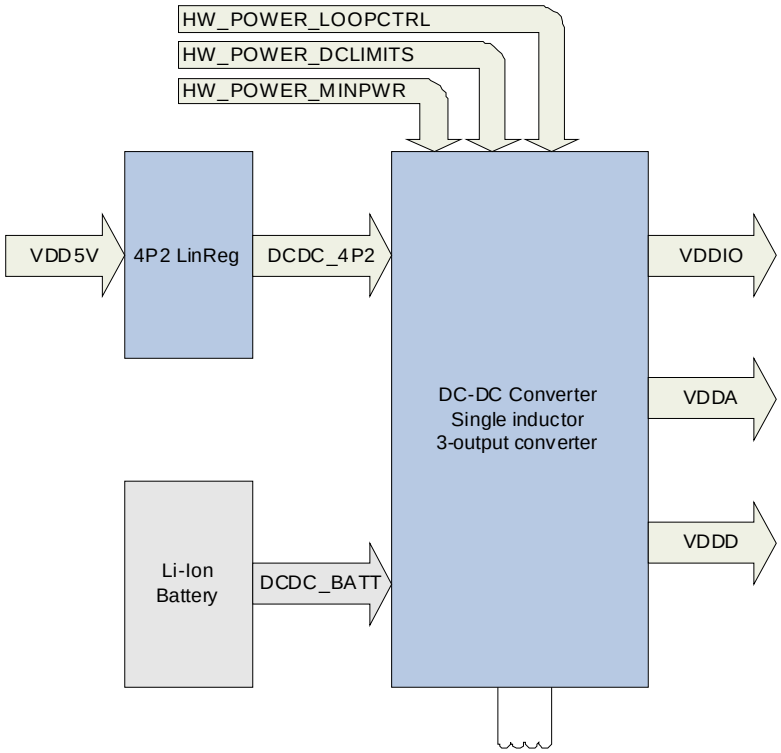

Figure 2. DC-DC Input Power Sources and Output Power Rails

The steps to enable the 4P2 power source, which serves as an input to the DC-DC converter are as follows:

1. Enable 4P2 LinReg
2. Charge the capacitor to prevent large current spikes in the 5 V source
3. Configure the DC-DC converter to accept the 4P2 LinReg output as an input

During these steps, the power rails receive power from their LinRegs and the DC-DC converter is not active.

3.1.3.1 Enabling 4P2 LinReg

The steps to initialize 4P2 LinReg before it is activated are as follows:

1. Set the 4P2 target to 4.2 V:
   ```
   HW_POWER_DCDC4P2[TRG] = 0
   ```
2. Enable the 4P2 circuitry to control the LinReg:
   ```
   HW_POWER_DCDC4P2[ENABLE_4P2] = 1
   ```

3. The 4P2 LinReg requires a static load to operate properly. As the DC-DC converter does not load the LinReg at this point, another load should be used:
   `HW_POWER_CHARGE[ENABLE_LOAD] = 1`
4. Provide an initial current limit for the 4P2 LinReg with the smallest possible value:
   `HW_POWER_5VCTRL[CHARGE_4P2_ILIMIT] = 1`
5. Power ON the 4P2 LinReg:
   `HW_POWER_5VCTRL[PWD_CHARGE_4P2] = 0`
6. Ungate the path from the 4P2 LinReg to DC-DC converter:
   `HW_POWER_DCDC4P2[ENABLE_DCDC] = 1`

NOTE

The `ENABLE_DCDC` bit field is different from the `ENABLE_DCDC` bit field that is located in the `HW_POWER_5VCTRL` register.

3.1.3.2 Charging 4P2 Capacitor

The capacitor in the 4P2 output and the parasitic capacitance should be charged before they attempt to power ON the 4P2 LinReg. Failure in slowly ramping up the current results in brownouts in the 5 V source.

To charge the 4P2 capacitor (including the parasitic capacitance), slowly increment the current limit of the 4P2 rail with the smallest resolution. The steps to charge the capacitor are as follows:

1. Set `HW_POWER_5VCTRL[CHARGE_4P2_ILIMIT] += 1`
2. Wait at least for 10 µs and repeat Step 1 until the current reaches the maximum limit of 780 mA

When the current reaches the maximum limit of 780 mA, the voltage across the VDD4P2 pin should read 4.2 V. Now, the LinReg is fully powered.

3.1.3.3 Enabling 4P2 Input for DC-DC

In this section, the 4P2 LinReg is configured as the input to the DC-DC converter so that the converter can supply power to the rails. The DC-DC converter is configured to choose the power source based on the voltage comparisons. The converter can also arbitrate between the 5 V source and battery charger. After configuring the input, the automatic DC-DC activation should be disabled as the converter should not be activated when turned ON. The DC-DC converter is then enabled and supply power to the rails.

To arbitrate between the 5 V source and battery charger, the comparator trip point for the DC-DC controller block is configured to select the 4P2 LinReg input. This is accomplished by setting the trip point to compare the 4P2 voltage with 85% of the battery voltage. The larger between these two is selected to be used as the source. Though only 85% of the battery voltage is compared, 100% of the battery voltage is used if it is selected as the source.

The battery charger and 4P2 LinReg share the 5 V power source. As the 4P2 rail powers the devices, the rail is given priority over the 5 V source in case a conflict should arise. The dropout control logic specifies the tolerable output voltage drop before stealing the 5 V current from the charger.

The steps to configure the allowable voltage drop are as follows:

1. Adjust the comparison between the battery and 4P2 LinReg to 85% of the rated voltage value:
   `HW_POWER_DCDC4P2[CMPTRIP] = 0x0`
2. Configure the DC-DC control logic to select the greater among the 4P2 LinReg voltage and 85% of the battery voltage. Set the dropout voltage threshold to 200 mV before stealing current from the battery charger. As both these settings are held in one bit field, write the settings to the bit field simultaneously.
   The steps to set the bit field are as follows:
   a) Bit field for the 200 mV threshold = `0b11XX`
   b) Bit field for the DC-DC converter to select the greater between 4P2 and battery = `0bXX1X`
   c) Combining the setting in Step a and Step b, the bit field setting should be 0b111X or 0b1110
   d) Set `HW_POWER_DCDC4P2[DROPOUT_CTRL] = 0b1110`
3. Disable the automatic DC-DC startup when the 5 V supply is powered-OFF:
   `HW_POWER_5VCTRL[DCDC_XFER] = 0`
4. Enable the DC-DC converter. This should be done by following the steps for enabling the DC-DC converter when the 5 V source is supplied:
   `HW_POWER_5VCTRL[ENABLE_DCDC] = 1`

3.2 Power Supplies

The power supplies used by the i.MX28 processor are as follows:

- Integrated single-inductor three-output DC-DC switching converter—is the most power efficient supply on the i.MX28 processor.
- Collection of LinRegs—supplies power to specific power rails. The LinRegs are reliable and stable. However, they are extremely inefficient.

NOTE

The 4P2 LinReg is not discussed in this section as the logic enabling and set up are more complex than the other supplies. For more information about the 4P2 LinReg, refer to Section 3.1.3, “4P2 Power Source.”

3.2.1 LinReg Power Supply

The LinReg power supply follows a linear-regulator daisy-chained architecture. Here, the VDDIO LinReg receives power from the 5 V source and VDDA and VDDD receive power from another power rail. The VDDMEM rail receives power from VDDA. As the DC-DC converter powers the VDDIO, VDDA, and VDDD rails, VDDIO rail powers VDDA, and VDDA rail powers VDDD and VDDMEM, the DC-DC converter is the power source for the LinRegs.

Figure 3 shows the daisy-chain architecture of the LinRegs.

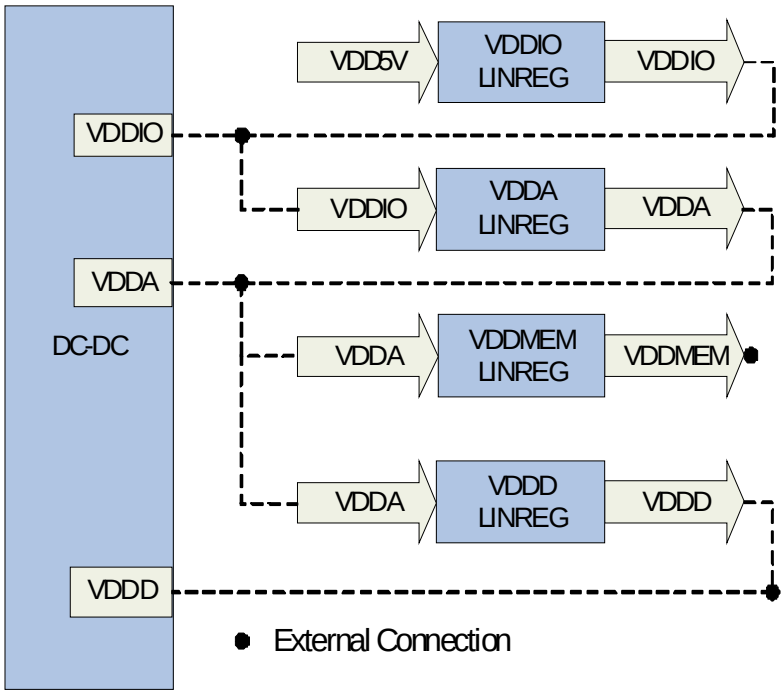

Figure 3. Linear-Regulator Daisy-Chain Architecture

3.2.1.1 LinReg Supply Enable Logic

The VDDD, VDDA, VDDIO, and VDDMEM rails are powered by the LinRegs. To operate the VDDIO rail, the LinReg is enabled only when the 5 V supply is powered-ON. Additionally, the 5 V current limiter should be disabled for the VDDIO to operate. To operate the VDDD and VDDA rails, the corresponding LinRegs are enabled with particular combinations of the 5 V supply, ENABLE_DCDC, and LinReg enable bit. The VDDMEM rail requires only the ENABLE_LINREG bit to be set for the operation.

Figure 4 shows the power source and enable logic for the LinReg.

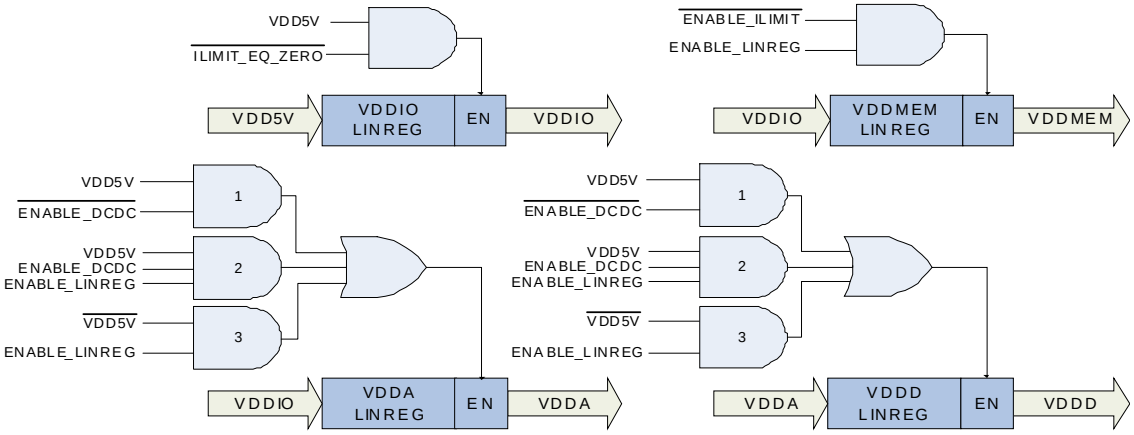

Figure 4. Enable Logic for Linear Regulators

As shown in Figure 4, the logic to enable the VDDIO and VDDMEM LinRegs is straightforward. In this case, only a single combination of bits is required to enable the LinReg. For the VDDD and VDDA power rails, three different combinations of bits are considered to enable the LinReg output—two combinations when the 5 V supply is powered-ON and the other when the 5 V supply is powered-off.

The three different combinations of the bits to enable the VDDD and VDDA LinRegs are described as follows:

- The combination with the AND gate 1 in Figure 4 is the default configuration. In this case, the AND gate is triggered only when the ENABLE_DCDC is false and 5 V supply is powered-ON. This means that the availabilty of the 5 V supply can activate or deactivate the corresponding LinReg without any software interaction.
- In the combination with the AND gate 2 in Figure 4, the LinReg is activated when the 5 V supply is powered-ON, ENABLE_DCDC bit is true, and ENABLE_LINREG bit is true. When the 5 V supply is powered-ON and ENABLE_DCDC bit is true, the ENABLE_LINREG bit of the rail should be set to activate the LinReg. Note that the DC-DC converter is active because the ENABLE_DCDC bit is true.
- The combination with the AND gate 3 in Figure 4 occurs only when the 5 V supply is powered-OFF. Here, the VDDD and VDDA LinRegs are enabled by setting the ENABLE_LINREG bit.

3.2.1.2 Enabling LinReg Supply

The steps to enable the LinReg supply are as follows:

1. From Figure 4, determine the logic that enables the LinReg

NOTE

The 5 V supply is powered-ON in this example.

2. Set `HW_POWER_VDDIOCTRL[ILIMIT_EQ_ZERO] = 0`
3. Set `HW_POWER_VDDIOCTRL[ENABLE_LINREG_ILIMIT] = 0`
4. Set `HW_POWER_VDDACTRL[ENABLE_LINREG] = 1`
5. Set `HW_POWER_VDDDCTRL[ENABLE_LINREG] = 1`
6. Set `HW_POWER_VDDDMEM[ENABLE_LINREG] = 1`

3.2.1.3 LinReg Voltage Offset

The LinReg target voltage can be configured to have an offset, which is –50 mV –25 mV, 0 mV, or +25 mV from the DC-DC target voltage. To prevent the LinReg and DC-DC contention, the LinReg voltage should always be configured as 25 mV below the DC-DC target voltage.

Figure 5 shows the LinReg offset logic.

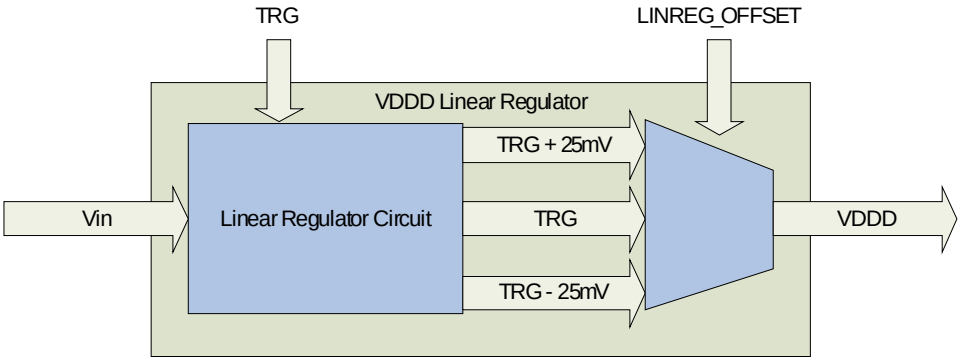

Figure 5. VDDD LinReg Offset Logic

3.2.1.4 Setting LinReg Voltage Offset

The steps to set the LinReg voltage offset are as follows:

1. Set `HW_POWER_VDDIOCTRL[LINREG_OFFSET] = 0x2`
2. Set `HW_POWER_VDDACTRL[LINREG_OFFSET] = 0x2`
3. Set `HW_POWER_VDDDCTRL[LINREG_OFFSET] = 0x2`

3.2.1.5 5 V Current and VDDIO LinReg

During the device boot up, a large amount of current is drawn from the 5 V supply to charge the chip capacitance. This violates the USB specifications for the inrush current. The software should control the amount of current that passes through the VDDIO LinReg.

As shown in Figure 3, the VDDIO LinReg receives power from the 5 V supply and then the VDDIO LinReg provides power to the other LinRegs. Therefore, limiting the current through the VDDIO LinReg limits the current through all the LinRegs.

The software controls the current through the VDDIO LinReg in the following ways:

- To meet the USB specification for the inrush current limits, the `HW_POWER_5VCTRL[ENABLE_LINREG_ILIMIT]` bit limits the VDDIO current to 100 mA. The bit is enabled during the device boot up and is cleared before the ROM begins execution.
- The `HW_POWER_5VCTRL[ILIMIT_EQ_ZERO]` bit minimizes the 5 V current drawn by the device through the LinRegs.

Figure 6 shows the 5 V current limiting logic in the VDDIO LinReg.

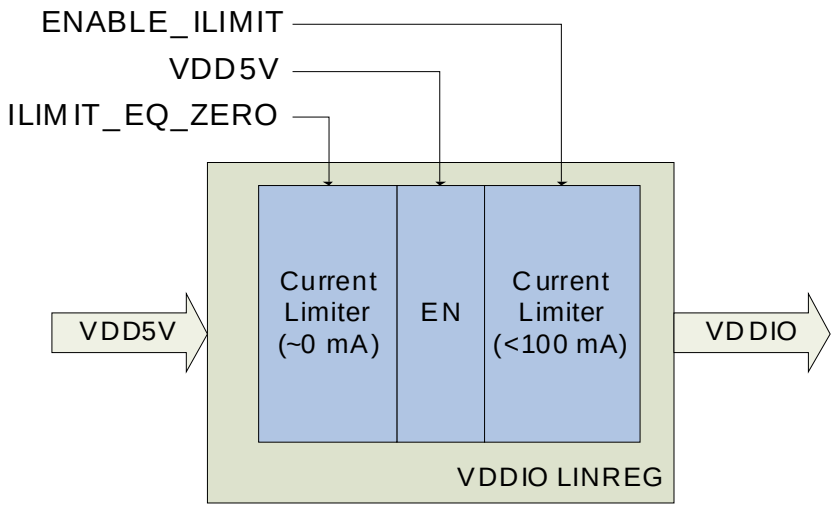

Figure 6. 5 V Current Limiting Logic

3.2.1.6 Setting VDDIO LinReg 5 V Current Limits

The steps to set the VDDIO LinReg 5 V current limits are as follows:

1. Limit the VDDIO LinReg current to 0 mA:
   - Set `HW_POWER_5VCTRL[ILIMIT_EQ_ZERO] = 1`
   - Set `HW_POWER_5VCTRL[ENABLE_LINREG_ILIMIT] = X`
2. Limit the VDDIO LinReg current to 100 mA:
   - Set `HW_POWER_5VCTRL[ILIMIT_EQ_ZERO] = 0`
   - Set `HW_POWER_5VCTRL[ENABLE_LINREG_ILIMIT] = 1`
3. Remove the VDDIO LinReg current limit:
   - Set `HW_POWER_5VCTRL[ILIMIT_EQ_ZERO] = 0`
   - Set `HW_POWER_5VCTRL[ENABLE_LINREG_ILIMIT] = 0`

3.2.2 DC-DC Power Supply

The single-inductor three-output DC-DC switching converter uses a 3.0–4.2 V input to output three rails with the voltage levels of 3.3 V, 1.8 V, and 1.2 V respectively. This is the most efficient power supply that is available in the i.MX28 processor. An application can achieve maximum power efficiency by using the DC-DC converter in the 5 V and battery power source configurations. The DC-DC converter has additional features that allow the converter to reduce the power consumption even when the overall system power requirement is low. Refer to Figure 2, for the DC-DC switching converter block diagram.

3.2.2.1 Automatic Battery Voltage Input

The DC-DC control logic requires the battery voltage as the input data. The `HW_POWER_BATTMONITOR[BATT_VAL]` field should contain an accurate battery voltage for the feedback circuit

to operate properly. The field is updated automatically by using a special channel in the Low-Resolution Analog to Digital Converter (LRADC) block.

The battery voltage is monitored by channel 7 in the LRADC block. The channel 7 is configured to accept voltages in the range of the Li-ion battery output and has the ability to automatically update the `HW_POWER_BATTMONITOR[BATT_VAL]` field. The channel is initialized and configured by the software. The automatic reading and conversion requires an LRADC delay channel. The delay channel is used to restart the conversion after a certain time period. Refer to the LRADC section in the *i.MX28 Applications Processor Reference Manual* (MC1MX28RM) for more information.

After the set up is complete, further modifications are not required for the feedback circuit to operate normally. Figure 7 shows a block diagram of the automatic battery update for the DC-DC converter.

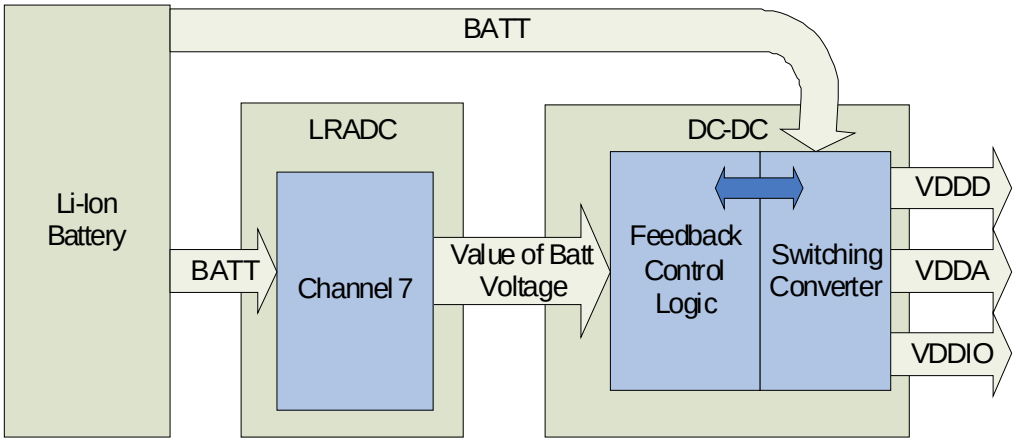

Figure 7. Automatic Battery Update

3.2.2.2 Enabling Automatic Battery Voltage Input

The steps to enable the automatic battery voltage input are as follows:

1. Select the Li-ion conversion factor and configure the automatic update in the power register:
   - Set `HW_LRADC_CONVERSION[SCALE_FACTOR] = 0x2`
   - Set `HW_LRADC_CONVERSION[AUTOMATIC] = 0x1`
2. Configure channel 7 in the LRADC block to use an undivided reading:
   `HW_LRADC_CTRL2[DIVIDE_BY_TWO] |= 0x80`

NOTE

Bit 7 of the `DIVIDE_BY_TWO` field corresponds to channel 7.

3. Configure channel 7 to sample the battery voltage without accumulating its sampled values. Disable accumulation and set the number of samples to sum up as zero:
   - Set `HW_LRADC_CH7[ACCUMULATE] = 0`
   - Set `HW_LRADC_CH7[NUM_SAMPLES] = 0`

4. Schedule the channel 7 conversion:
   `HW_LRADC_CTRL0[SCHEDULE] |= 0x80`

NOTE

Bit 7 of the SCHEDULE field corresponds to channel 7.

5. Configure the delay channel 3 so that the LRADC channel 7 takes readings and performs conversion every 50 ms:
   - Set `HW_LRADC_DELAY3[TRIGGER_LRADCS] = 0x80`
   - Set `HW_LRADC_DELAY3[TRIGGER_DELAYS] = 0x8`
   - Set `HW_LRADC_DELAY3[LOOP_COUNT] = 0`
   - Set `HW_LRADC_DELAY3[DELAY] = 0x64`
   - Set `HW_LRADC_DELAY3[KICK] = 1`
6. Enable the battery voltage as an input to the DC-DC switching logic:
   `HW_POWER_BATTMONITOR[EN_BATADJ] = 1`

Now, the battery voltage with an 8 mV resolution measurement, appears in the `HW_POWER_BATTMONITOR[BATT_VAL]` field.

3.2.2.3 5 V Detection for DC-DC Control Logic

The DC-DC control logic requires the 5 V supply to determine when to activate and deactivate the DC-DC output. There are two methods—VDD5V_GT_VDDIO and VBUSVALID—to detect the 5 V supply. Between these two methods, VBUSVALID is the recommended configuration.

NOTE

The 5 V detection for the DC-DC control logic is different from the 5 V detection used by the software. The 5 V detection for the DC-DC converter is performed internally by the control block, whereas the software should set up the 5 V detection for the software and interrupt usages.

3.2.2.4 Enabling 5 V Detection for DC-DC Control Logic

1. Turn ON the VBUS comparator:
   `HW_POWER_5VCTRL[PWRUP_VBUS_CMPS] = 1`
2. Select the threshold voltage for the 5 V detection. A voltage reading above the threshold value is considered as a valid 5 V:
   `HW_POWER_5VCTRL[VBUSVALID_TRSH] = 0x1`
3. Configure the DC-DC converter to use the VBUSVALID 5 V-detection method:
   `HW_POWER_5VCTRL[VBUSVALID_5VDETECT] = 1`

3.2.3 DC-DC Converter when 5 V Supply Powered-ON

By default, the DC-DC converter is disabled when the 5 V supply is powered-ON. However, the converter can be enabled by using the ENABLE_DCDC bit. The default configuration of the DC-DC converter is intended to prevent the LinReg and DC-DC contention. When the 5 V supply is powered-ON, the DC-DC converter

should be powered from the 4P2 rail for the maximum power efficiency. Therefore, special care should be taken to prevent the LinReg and DC-DC contention.

NOTE

Contention occurs when the LINREG_OFFSET fields are not configured correctly.

3.2.3.1 DC-DC and LinReg Contention

Contention between the DC-DC converter and LinReg occurs when both the power sources are activated and attempts to output simultaneously to a rail. If the target output voltage is not configured correctly, both the sources attempt to regulate the target voltage to different voltage levels. This situation can be avoided by configuring the DC-DC converter and LinReg with mutually exclusive voltage goals so that extra attention is required only when both the sources are enabled simultaneously.

Contention Causes

The main cause of contention is that the LinRegs are enabled when the 5 V supply is powered-ON, unless they are disabled by the software. By nature, the LinRegs try to pull its output up while the DC-DC converter tries to pull its output up or down to reach their targets. When the LinReg target is higher than the DC-DC converter target, the LinRegs attempt to pull the rail high while the DC-DC converter pulls the rail down.

Contention Effects

The effects of contention are as follows:

- DC-DC converter pulls a high amount of current that leads to dips in the power source
- 5 V current exceeds the USB specifications
- High current flow causes the processor to heat up that raises the device temperature

Contention Prevention

The DC-DC and LinReg targets should be configured so that the conditions for both are satisfied. This means that the LinReg target output should be less than the DC-DC target output.Therefore, as the LinReg is satisfied when the output voltage is at the target voltage, the DC-DC converter output should be configured to be higher than the LinReg output voltage. This configuration is set in the hardware through the LINREG_OFFSET bit. For each main power rails—VDDD, VDDA, and VDDIO—set the LINREG_OFFSET bit to 25 mV below the target voltage.

3.2.3.2 Enabling DC-DC when 5 V Supply Powered-ON

The steps to enable the DC-DC converter when the 5 V supply is powered-ON are as follows:

1. Set the LINREG_OFFSET to 25 mV below the target voltage for each power rail. This is done to prevent contention between the DC-DC converter and LinReg:
   - — Set `HW_POWER_VDDDCTRL[LINREG_OFFSET] = 0x2`
   - — Set `HW_POWER_VDDACTRL[LINREG_OFFSET] = 0x2`

— Set `HW_POWER_VDDIOCTRL[LINREG_OFFSET] = 0x2`

2. By default, the DC-DC converter is OFF when the 5 V supply is powered-ON. Set the `ENABLE_DCDC` bit to override this default setting:
   `HW_POWER_5VCTRL[ENABLE_DCDC] = 1`

3.3 Power Rails

Power rails provide power to the processor blocks that require different voltages. The power rails are generated by using the DC-DC or LinReg power supplies. The voltage rating of each power rail is determined by the target field voltage and LinReg offset in the rail control register. When the rail is powered by the DC-DC converter, the target field determines the output voltage. In the case of LinReg powered rails, the sum of target field voltage and LinReg offset determines the output voltage. Also, each rail has a brownout voltage threshold that is below the target voltage. A system brownout occurs when the output voltage drops to the threshold voltage which indicates that the system load has exceeded the available power. The i.MX28 processor is configured to generate an interrupt or power down when a brownout occurs.

The different power rails are described in Table 2.

Table 2. i.MX28 Power Rails

Power Rails	Operating Range (V)	Power Supply	Description
VDDD	0.8–1.575	Directly from DC-DC or LinReg from VDDA	Provides power to the digital core such as CPU and clocks
VDDA	1.5–2.275	Directly from DC-DC or LinReg from VDDIO	Provides power to the analog core and audio components, and acts as a power source for the VDDMEM rail
VDDIO	2.8–3.6	Directly from DC-DC or LinReg from VDD5V	Provides power to the non-external memory I/O peripherals such as NAND Flash
VDDMEM	1.1–1.75	LinReg from VDDA	Provides power to the external memory

3.3.1 Target Voltages

The target voltage setting determines the output voltage. This setting specifies the number of voltage steps above a base voltage. The VDDD, VDDA, and VDDMEM rails have a step size of 25 mV while the VDDIO rail has a step size of 50 mV. Each of these power rails has a different base voltage.

Equation 1 gives the formula to calculate the output voltage.

$$voltage = base\ voltage + (step\ size \times number\ of\ steps) \quad \textbf{Eqn. 1}$$

NOTE

For easier mathematical calculations, voltages are written in millivolts (mV).

3.3.1.1 Setting VDDD, VDDA, and VDDIO Target Voltages

The procedure to set the target voltage is similar for the VDDD, VDDA and VDDIO rails. This example describes the procedure to set the target voltage for the VDDD rail to 1.45 V. The same procedure can be followed to set the target voltage for the VDDA and VDDIO rails.

The steps to set the target voltage for the VDDD rail to 1.45 V are as follows:

1. Calculate the number of steps required to reach 1450 mV:
   Applying the corresponding values in Equation 1, 1450 = 800 + (25 × number of steps)
   Therefore, number of steps = 26
2. Write the target voltage setting to the control register:
   `HW_POWER_VDDDCTRL[TRG] = 26`

3.3.1.2 Setting VDDMEM Target Voltage

The VDDMEM rail is used to power external memories. By default, this rail is not enabled. Therefore, this rail should be enabled before it is used.

The steps described in this section provide instruction to set the VDDMEM rail to 1.5 V (using a step size of 25 mV) and enabling the output. Irrespective of the VDDD, VDDA and VDDIO rails, the VDDMEM rail requires an additional step to activate pull-down loads. This ensures that the rail powers up from the ground.

NOTE

As the power source for the VDDMEM rail is VDDA, the target VDDMEM voltage should not be set higher than the VDDA voltage.

The steps to set the target voltage for the VDDMEM rail are as follows:

1. Calculate the number of steps required to reach 1500 mV:
   Applying the corresponding values in Equation 1, 500 = 1100 + (25 × number of steps)
   Therefore, number of steps = 16
2. Write the target voltage setting to the control register:
   `HW_POWER_VDDMEMCTRL[TRG] = 16`
3. Activate the pull-down loads on the external memory power pins. This allows the rail to power up from the ground:
   `HW_POWER_VDDMEMCTRL[PULLDOWN_ACTIVE] = 1`
4. Turn ON the VDDMEM current limiter and LinReg:
   — Set `HW_POWER_VDDMEMCTRL[ENABLE_ILIMIT] = 1`
   — Set `HW_POWER_VDDMEMCTRL[ENABLE_LINREG] = 1`
5. Wait at least for 500 µs to allow the rail to ramp up before proceeding to Step 6
6. Disable the current limiter in the LinReg. The current limiter is disabled as the limiter is required only to meet the USB inrush specifications and is active by default:
   `HW_POWER_VDDMEMCTRL[ENABLE_ILIMIT] = 0`

3.3.1.3 Setting VDD4P2 Target Voltage

The 4P2 power source converts the 5 V power to a voltage level that can be served as the input to the DC-DC converter. Generally, this voltage level is set to 4.2 V. However, there are cases where this voltage value should be changed. Therefore, four predefined voltage levels—4.2 V, 4.1 V, 4.0 V, and 3.9 V—and one battery-matching voltage level are defined. The battery-matching voltage level gives the same output voltage that is measured across the battery.

NOTE

The battery-matching voltage level should be set only when a battery is available in the range of 2.9–4.2 V.

The steps to set the target voltage for the VDD4P2 rail are as follows:

1. Determine the voltage level for the output:
   4.2 V output is converted to `0x0` bit field setting
2. Write the target voltage setting to the control register:
   `HW_POWER_DCDC4P2 [TRG] = 0`

3.3.2 Brownouts

Brownouts occur when the voltage on a rail drops below the target voltage setting. This is caused due to heavy output loads or insufficient input power. The i.MX28 PMU uses the brownout detectors that allows software or hardware to handle the brownouts. The software is notified about a brownout condition through interrupt so that the software can act accordingly. The hardware automatically shuts down the device when it is detected with a brownout condition.

3.3.2.1 Master Brownout Shutdown

All the hardware power down events are controlled by a master switch. These events include the brownouts and resets. However, this does not include the watchdog reset. The master shutdown is managed by software that allows the device to shut down when required.

The master shutdown is a single bit. Because this bit can turn OFF a device, the register that holds this bit requires an unlock key for the data to be written into the register. Therefore, an optional bit is provided to turn OFF the master shutdown. When this bit is set, the hardware power down is disabled.

The steps to enable and disable the master shutdown switch and power down the device are as follows:

1. Generate register masks and disable the master shutdown switch and power down the device. The unlock key should be placed in the upper 16 bits of the mask:
   - `ENABLE_MASK = (0x3E77 << 16) | 0x0 = 0x3E770000`
   - `DISABLE_MASK = (0x3E77 << 16) | 0x2 = 0x3E770002`
   - `PWD_MASK = (0x3E77 << 16) | 0x1 = 0x3E770001`
2. The master shutdown switch is enabled or disabled or the device is powered down as follows:
   - To enable the master shutdown switch, write ENABLE_MASK to the register:
     `HW_POWER_RESET = ENABLE_MASK`

— To disable the master shutdown switch, write `DISABLE_MASK` to the register:

```
HW_POWER_RESET = DISABLE_MASK
```

— To power down the device, write `PWD_MASK` to the register:

```
HW_POWER_RESET = PWD_MASK
```

3.3.2.2 Brownout Thresholds

Brownout thresholds determine the level that the comparators should detect a brownout. The threshold value is set in a register field that resides in one of the control registers. However, the calculation of the actual value to the written value varies with the type of brownout.

Power-Rail Brownout

The power-rail brownout threshold is an offset or margin from the target voltage register field. The actual value of the brownout threshold is calculated by determining the target output voltage and subtracting the brownout voltage margin.

The formula to calculate the voltage margin is given in Equation 2.

$$MARGIN = (STEP\_SIZE \times NUM\_STEPS) \quad \textbf{Eqn. 2}$$

The steps to set the VDDD brownout margin to 100 mV are as follows:

1. Calculate the number of 25 mV steps required to reach the brownout margin:
   Applying the corresponding values in Equation 2, 100 = (25 × NUM_STEPS)
   Therefore, NUM_STEPS = 4
2. Write the margin to the brownout offset field in the rail control register:

```
HW_POWER_VDDDCTRL[BO_OFFSET] = 4
```

Battery Brownout

The battery brownout threshold is an absolute voltage level, unlike in the case of power rails which is an offset value. The available range for the battery brownout threshold is 2.4–3.64 V with a 40 mV resolution. The battery brownout is triggered when the battery voltage reaches the threshold value and is configured to generate an interrupt that notifies the application that the battery is low.

The formula to calculate the battery brownout threshold is similar to the equation to set a power-rail target voltage and is given in Equation 3.

$$BO\_VOLT = BASE\_VOLT + (STEP\_SIZE \times NUM\_STEPS) \quad \textbf{Eqn. 3}$$

The steps to set the battery brownout level to 3.2 V are as follows:

1. Calculate the number of steps that are required to reach the brownout level to 3.2 V (3200 mV):
   Applying the values in Equation 3, 3200 = 2400 + (40 × NUM_STEPS)
   NUM_STEPS = 20
2. Write the brownout setting to the register bit field:

```
HW_POWER_BATTMONITOR[BRWNOUT_LVL] = 20
```

5 V-Source Brownout

In the i.MX28 PMU, the 5 V-source brownout is called the VBUS droop brownout. The 5 V-source brownout level is a voltage threshold and is an absolute voltage value as in the case of the battery brownout. There are four available settings for the 5 V-source brownout—4.3 V, 4.4 V, 4.5 V, and 4.7 V. Each setting has a 50 mV hysteresis to prevent chattering. The VBUS droop brownout generates an interrupt when the 5 V source drops below the threshold. This is useful to determine if the load on the 5 V source is excessive.

The steps to set the VBUS droop brownout to 4.3 V are as follows:

1. Select threshold and determine the corresponding register setting:
   4.3 V is converted to `0x0` bit field setting
2. Write the threshold setting to the register bit field:
   `HW_POWER_5VCTRL[VBUSDROOP_TRSH] = 0x0`

4P2 Brownout

The brownout threshold range for the 4P2 rail is 3.6–4.375 V with a resolution of 25 mV. The 4P2 brownout occurs when the 4P2 voltage reaches the brownout level. The device is configured to generate an interrupt when the 4P2 source drops to the threshold level. The 4P2 brownout voltage can be calculated using Equation 3 which is also used to calculate the battery brownout threshold.

The steps to set the 4P2 brownout voltage to 3.8 V are as follows:

1. Calculate the number of steps required to adjust the brownout level to the target threshold using Equation 3.
2. Write the brownout setting to the register bit field:
   `HW_POWER_DCDC4P2[BO] = 8`

3.3.2.3 Automatic Hardware Shutdown on Brownout

In addition to the interrupt generation, brownouts can also be configured to automatically power down the i.MX28 processor through hardware depending on the availability of the 5 V supply. The power rails, VDDD, VDDA, and VDDIO, battery, and 5 V source have this feature. Each of these five brownout detectors has an enable bit to control the power locally. These enable bits are controlled by the master enable switch.

In cases where the 5 V supply is powered-ON, it is possible only for the 5 V brownout automatic shutdown to be triggered. In this case, automatic shutdown due to the power-rail (VDDD, VDDA, and VDDIO) and battery brownouts are disabled. Therefore, it should be noted that the brownout interrupt is enabled regardless of the availability of the 5 V supply.

Automatic 5 V-Brownout Shutdown

The automatic 5 V brownout shutdown uses the VBUSVALID threshold instead of the VBUSDROOP threshold. The 5 V brownout shutdown is enabled only when the PMU is not configured to handle a 5 V removal event. This feature is enabled only during the initialization and is disabled when the handoff-to-battery is set. It is also important to note that the automatic hardware shutdown is not activated until the VBUS voltage level exceeds the VBUSVALID threshold.

The steps to enable the automatic 5 V brownout shutdown are as follows:

1. Configure the VBUSVALID threshold to the desired level:
   `HW_POWER_5VCTRL[VBUSVALID_TRSH] = Threshold Level`
2. Enable the local brownout enable for the 5 V supply:
   `HW_POWER_5VCTRL[PWDN_5VBRNOUT] = 1`
3. Enable the master brownout switch. Refer to Section 3.3.2.1, “Master Brownout Shutdown,” for the instruction.

Automatic Battery Brownout Shutdown

The automatic battery brownout shutdown uses the battery brownout threshold. The battery brownout shutdown is enabled when the battery-to-5V handoff is not set or battery is very low and the power loss is extremely high. Though the automatic shutdown is deactivated when the 5 V supply is powered-ON, the battery brownout detection that generates an interrupt is enabled.

The steps to enable the automatic battery brownout shutdown are as follows:

1. Follow Step 1 in Automatic 5 V-Brownout Shutdown to set the battery brownout voltage to the desired level.
2. Enable the local brownout detection for the battery:
   `HW_POWER_BATTMONITOR[BRWNOUT_PWD] = 0`
3. Deactivate the battery brownout shutdown when the 5 V supply is powered-ON:
   `HW_POWER_BATTMONITOR[PWDN_BATTBRNOUT_5VDETECT_ENABLE] = 1`
4. Enable the local brownout shutdown enable for battery:
   `HW_POWER_BATTMONITOR[PWDN_BATTBRNOUT] = 1`
5. Enable the master brownout switch. Refer to Section 3.3.2.1, “Master Brownout Shutdown,” for the instruction.

Automatic Power-Rail Brownout Shutdown

The power rails, VDDD, VDDA, and VDDIO have automatic shutdown functionality. The VDDMEM rail is generated from the VDDA rail and any brownout on the VDDMEM rail affects the VDDA rail. Therefore, the automatic shutdown should be triggered during the device boot up before the software interrupt handling is enabled or when the software handling is slow. The automatic power-rail shutdown can also be used in the final production builds to shut down a device on rail brownouts.

The steps to enable automatic shutdown on the VDDIO rail are as follows:

1. Follow the steps described in Power-Rail Brownout to set the brownout voltage.
2. Enable the local brownout shutdown enable for the VDDIO rail:
   `HW_POWER_VDDIOCTRL[PWDN_BRNOUT] = 1`
3. Enable the master brownout switch. Refer to Section 3.3.2.1, “Master Brownout Shutdown,” for the instruction.

3.3.2.4 Software Brownout Interrupts

The generation of the interrupts by the brownout detectors allows the software to handle the brownout conditions and find solutions to rectify them. These interrupts are sent through the ICOLL block and are configured as Fast Interrupt Requests (FIQ).

The steps to set the VDDD software brownout interrupt are as follows:

1. Follow the steps described in Power-Rail Brownout to set the brownout threshold for the VDDD rail to 100 mV below the target voltage level.
2. Clear the VDDD brownout interrupt status bit and write to the VDDD brownout IRQ status field in the CLR register:
   `HW_POWER_CTRL_CLR[VDDD_BO_IRQ] = 1`
3. Initialize the interrupt handler in the ICOLL block.
4. Enable the VDDD brownout interrupt:
   `HW_POWER_CTRL[ENIRQ_VDDD_BO] = 1`
5. Disable the VDDD brownout power down:
   `HW_POWER_VDDDCTRL[PWDN_BRNOUT] = 0`

4 Power Subsystem

This section describes the subsystem of the i.MX28 PMU.

4.1 Powering the Supplies

The DC-DC and LinReg power supplies receive their power from the 5 V supply or Li-ion battery. This section describes the steps to configure a power supply for the current power source. Here, the DC-DC converter should be used maximum number of times to achieve the best power efficiency.

4.1.1 LinRegs Powered by 5 V Supply

When a device is powered from a 5 V supply, the LinRegs also get powered from them. Also, it is less efficient to use the 4P2 source to power the DC-DC converter though the 4P2 source provides a stable power source. Therefore, during initialization, the LinRegs are turned ON when the 5 V supply is enabled while the DC-DC converter is turned OFF.

The steps to configure the LinRegs to be sourced from the 5 V supply are as follows:

1. Configure the voltage offset for the LinRegs to 25 mV below the target voltage
2. Set the power-rail brownout voltage
3. Set the power-rail target voltage

NOTE

Compensate for the LinReg offset to ensure that the output is at the desired voltage level.

4.1.2 DC-DC Powered by Battery

By default, the DC-DC converter is powered from the battery when a device is powered by pressing PSWITCH. In this case, the VDDA and VDDIO rails are powered from the DC-DC converter. However, the VDDD rail is powered from the LinReg. Therefore, during the power block initialization, the VDDD rail is switched to the DC-DC power source by disabling the LinReg and enabling the DC-DC output.

The steps to configure the DC-DC converter to be sourced from a battery are as follows:

1. Follow the steps described in Section 3.2.2.2, “Enabling Automatic Battery Voltage Input,” to enable the automatic battery voltage input for the DC-DC converter.
2. Configure the LinReg voltage offset to 25 mV below target voltage.
3. Enable the DC-DC converter when the 5 V supply is powered-ON. This prevents the DC-DC converter from turning OFF when the converter is detected with a 5 V supply. It is important to set the LinReg voltage properly before this step is performed.
4. Set the power-rail brownout voltage.
5. Set the power-rail target voltage.

4.1.3 DC-DC Powered by 5 V Supply

When the 5 V supply is powered-ON, it is recommended to power the DC-DC converter from the 4P2 rail, which is powered by the 5 V source. The DC-DC converter is the most efficient power supply in the PMU and allows fast battery charging as more current flows to the battery.

The steps to configure the DC-DC converter to be powered from the 5 V supply are as follows:

1. Enable the automatic battery voltage input to the DC-DC control logic
2. Configure the DC-DC control logic 5 V detection to use the VBUSVALID 5 V-detection method
3. Configure the LinReg voltage offset to 25 mV for all the rails
4. Enable the 4.2 V power source
5. Enable the DC-DC converter while the 5 V supply is powered-ON
6. Set the supply-rail brownout voltage
7. Set the supply-rail output voltage

4.2 Transition Between Power Sources

In an application, the power source changes dynamically. This is described in Figure 8 which shows the power states and transitions among them.

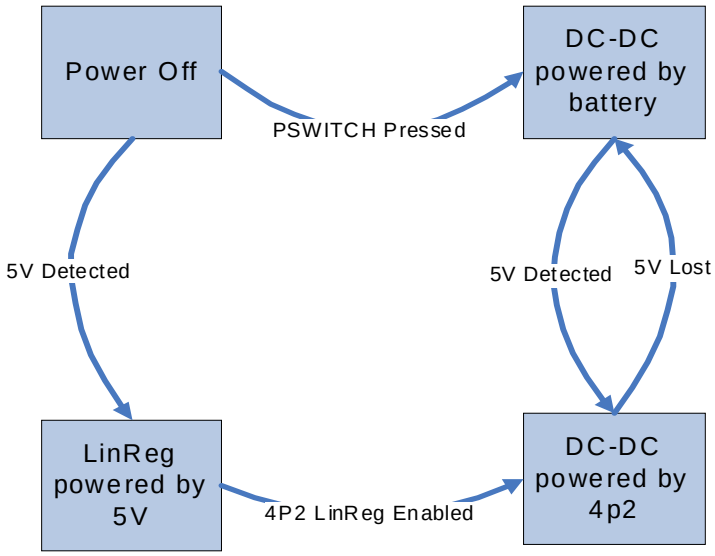

Figure 8. Power Source States and Transitions

The Power-Off state is an unpowered state while the other three states are power-ON states. These three power-on states are discussed in Section 3, "Getting Started with Power Subsystem." This section describes the transitions among the power-on states and between the end states. Here, the two DC-DC powered states are considered as the end states and the application leaves these states when it is detected with a change in the 5 V supply (or when the device is powered OFF). The transition among the Power Off, LinReg powered by 5 V, and DC-DC powered by 4P2 states are described in Section 3.2.1.2, "Enabling LinReg Supply," as it is a natural transition to the end state.

4.2.1 Transition from Battery to 5 V Power Source

The battery to 5 V power transition (or batt-to-5V handoff for short) happens in the software. As the power sources are available, the software can decide on the usage of the 5 V power. Therefore, there is no time limit to perform the batt-to-5V handoff.

The steps for the transition from the battery to 5 V power are as follows:

NOTE

It is assumed that application is currently in the DC-DC powered by battery state.

1. Enable the 5 V-detection interrupt to trigger the 5 V insertion. To trigger on insertion, configure the 5 V-interrupt polarity bit for the 5 V supply.
2. Wait for the 5 V insertion.
3. When the application is detected with 5 V, enable the 4P2 LinReg and switch the DC-DC converter to use the 4P2 LinReg as the input.

4. Enable the 5 V-detection interrupt to trigger on the 5 V removal so that the application is prepared for the next transition when the 5 V supply is disconnected.

4.2.2 Transition from 5 V Supply to Battery

The 5 V to battery power transition (or 5V-to-batt handoff for short) is allowed only a limited time to perform the switching operation. Here, when the 5 V supply is lost, the DC-DC converter should switch to the battery power source before the power rails begin to drop its voltage. Therefore, the device is allowed only a short time period for this transition.

To keep the power stable during the transition, the first half of the transition is performed in the hardware. The DC-DC control logic, which uses the 4P2 LinReg, is configured to use the 4P2 if its output voltage is greater than 85% of the battery voltage. When the 5 V supply is powered-OFF, the 4P2 rail drops and the battery voltage becomes the input source. At this point, the power rails are stable and the DC-DC converter has a stable power source. However, the application is not prepared for the next transition and is not aware of the power-source transition.

The 5 V-detection interrupt is triggered when the 5 V supply is powered-OFF. The application should now configure the 5 V-detection interrupt for the 5 V insertion to prepare for the next transition.

The steps for the transition from the 5 V to battery power are as follows:

NOTE

It is assumed that application is currently in the DC-DC powered by 4P2 state.

1. Configure the DC-DC control logic to select the greater between the battery output voltage and 4.2 V.
2. Configure the 5 V-detection interrupt so that the interrupt is triggered when the application is disconnected from the 5 V supply.
3. Wait for the 5 V removal notification through interrupt. When the interrupt is detected, the hardware automatically switches from the 4P2 power source to battery.
4. Disable the 4P2 LinReg:
   ```
   HW_POWER_DCDC4P2[ENABLE_4P2] = 0
   ```
5. Configure the 5 V-detection interrupt to trigger the 5 V insertion so that the application is prepared for the next transition when the 5 V supply is powered-ON.

4.3 Software-Brownout Interrupt

The software-brownout interrupts facilitate the application to handle any adverse power events. Each brownout types has a different level of severity. Therefore, each brownout should be handled in different ways. This section describes the recommended ways to handle the brownouts.

4.3.1 Software Battery Brownout Interrupt

The battery brownout interrupt is triggered when the battery voltage reaches a low level. It is recommended to use at least two thresholds that are managed by the application.

These thresholds are described as follows:

- First threshold is a low battery voltage that permits the battery to supply power normally. However, it gives a low battery warning to the user. The low battery voltage threshold is set in the range of 3.2–3.4 V.
- Second threshold is a voltage level where the device is powered down immediately. The threshold voltage is set around 3.0 V.

The common sequence of events for the software battery brownout are as follows:

1. The battery brownout is enabled and the high threshold is set as the brownout voltage level.
2. The battery voltage drops as the device is used.
3. The battery brownout generates an interrupt. The interrupt handler detects the high threshold. Then, the application re-arms the interrupt and sets the low threshold as the brownout voltage level. The handler also sends a low-battery notification to the application.
4. The battery voltage continues to drop unless a charger is fixed to the battery.
5. The battery brownout generates an interrupt. The handler detects that the low threshold has been reached and shuts down the device.

4.3.2 5 V-Detection Interrupt

The 5 V-detection interrupt is triggered when the 5 V supply is connected or disconnected from the application.

The 5 V-detection interrupt helps in the following ways:

- Helps to configure the power sources properly
- Provides information to the battery charger logic for enabling and disabling the charger
- Determines when to begin looking for a USB connection

4.3.3 Handling Power-Rail Brownouts

Power-rail brownouts occur when the battery or 5 V rail experiences a brownout or one of the rails is heavily loaded. The power-rail brownout is a fatal condition in the PMU and should be handled immediately by powering down the device. This is configured as an automatic hardware power down or handled in the interrupt handler by resetting the processor.

5 Battery Charging Subsystem

The i.MX28 battery charger has a two-stage operation—constant current and constant voltage. In the constant current state, the charge current is set by the application and the battery voltage is set below 4.2 V. As the battery receives current, the battery voltage rises. When the battery voltage reaches 4.2 V, the battery charger hardware lowers the current while keeping the battery voltage at 4.2 V. This begins the constant voltage stage. The hardware continues to lower the current until an internal current sensor detects that the battery current has reached a pre-determined threshold which is set by the application. This ends the battery charging process.

Figure 9 shows the constant current and constant voltage operation of the charger.

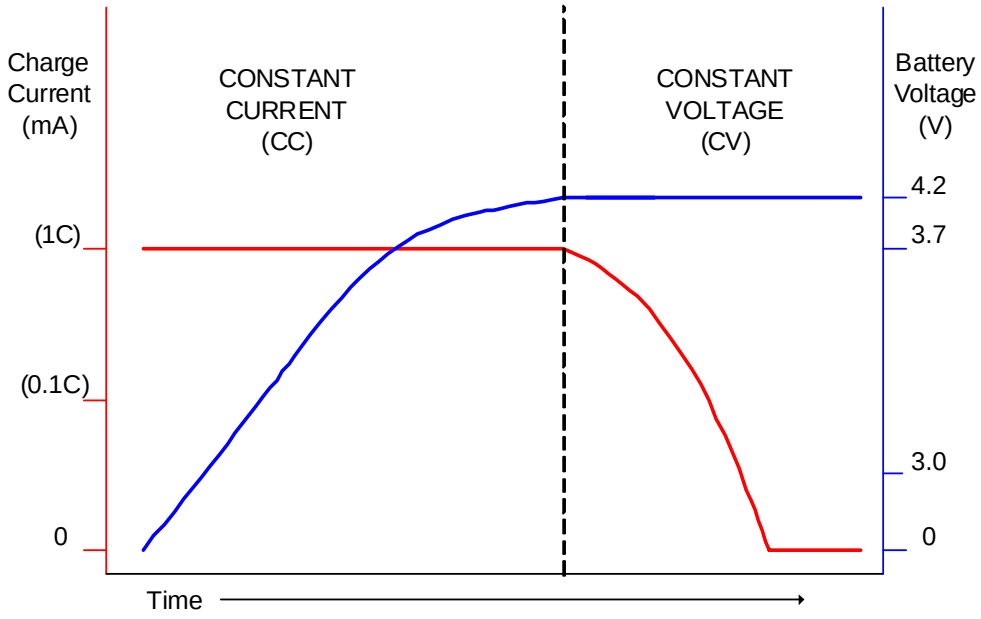

Figure 9. Constant Current and Constant Voltage Operation

The 4P2 LinReg and battery charger should share the 5 V current. The 4P2 LinReg is given priority over the 5 V current and is the main controller of the current. The 4P2 rail has a master current limiter and enable switch to control the 5 V current. These control fields should be configured before the charger runs as expected. For more information about the current limiter and enable switch, refer to Section 5.2.3, "Managing Charger and 4P2 Rail."

5.1 Configuring the Charger

The battery charger has two different current settings:

- Charge current—maximum amount of current that the charger can output
- Stop current—threshold current set by the application, which indicates that the battery is full

Both these settings should be configured before charging a battery.

5.1.1 Setting Charge Current

The charge current is represented by 6 bits where each bit corresponds to a particular value of current that is described in milliampere (mA). The charge current can have values in the range of 0–780 mA with a resolution of 10 mA depending on the selected value.

Table 3 gives the current corresponding to the position of the bits that represents the stop current.

Table 3. Bit to Charge Current Conversion

Bit Number	Charge Current (mA)
0	10
1	20
2	50
3	100
4	200
5	400

The procedure to set the charge current to 650 mA is as follows:

1. Convert the 650 mA value to the register setting:
   Start the conversion by subtracting the largest setting and working down until the current is 0 mA or smallest setting is reached.
   Therefore, the register setting that corresponds to 650 mA is `0b110100`.
2. Write the register setting to the battery charge current field:
   `HW_POWER_CHARGE[BATTCHRG_I] = 0b110100`

5.1.2 Setting Stop Current

The stop current is a setting that signals a flag to stop the charging process. The stop current is represented by four bits where each bit corresponds to a particular value of current that is described in mA. The stop current can have value in the range of 0–180 mA with a resolution of 10 mA depending on the selected value.

Table 4 gives the current corresponding to the position of the bits that represents the stop current.

Table 4. Bit to Stop Current Conversion

Bit Number	Stop Current (mA)
0	10
1	20
2	50
3	100

The procedure to set the charge current to 120 mA is as follows:

1. Convert the 120 mA value to the register setting:
   Start the conversion by subtracting the largest setting and working down until the current is 0 mA or smallest setting is reached.
   Therefore, the register setting that corresponds to 120 mA is `0b1010`.
2. Write the register setting to the battery charge current field:
   `HW_POWER_CHARGE[STOP_ILIMIT] = 0b1010`

5.1.3 Enabling Battery Charger

When the 4P2 LinReg is active, the charger can be enabled by toggling a register field member. When the 4P2 LinReg is not enabled, the battery charger is enabled in a two-step process. The two-step process is described in Section 5.2.3, "Managing Charger and 4P2 Rail." The instruction to enable the charger in the one-step process is as follows:
Enable the charger by clearing the charger power-down bit:

```
HW_POWER_CHARGE[PWD_BATTCHRG] = 0
```

5.2 Battery Charger

The battery charger can be used as long as the 5 V source is powered-ON. The charger shares the 5 V current with the 4P2 rail and generally, this does not cause any problem in the system. Current arbitration is required only when the 4P2 rail is heavily loaded.

5.2.1 Battery Charging Process

Battery charging is a multi-step process. The first few steps determine and configure the charge and stop currents. Then, the charger is initialized to begin the charging process. In the last steps, the current to the battery is monitored and the charging process is stopped when the charge current is reached.

The steps for battery charging are described as follows:

NOTE

Ensure that the 4P2 rail is enabled to maximize the current flow into the battery.

1. Determine the charge current and configure the charger setting:
   The maximum current is determined by the battery manufacturer and is provided in the battery data sheet. If the charge current is set at 600 mA, then set
   `HW_POWER_CHARGE[BATTCHRG_I] = 0x30`
2. Determine the stop current and configure the charger setting:
   This value is determined by the battery manufacturer. It is typically 10% of the maximum charge current and is provided in the battery data sheet. For this example, use 60 mA as the stop charge current, as the charge current is set at 600 mA (10% of the maximum charge current).
   Therefore, set `HW_POWER_CHARGE[STOP_ILIMIT] = 0x5`
3. Follow the step mentioned in Section 5.1.3, "Enabling Battery Charger," to enable the battery charger
4. Monitor the charger status flag to determine if the charger has reached the stop current threshold:
   This should be monitored for several times as the DC-DC converter can cause spurious results to this flag
5. End the charging process by disabling the battery charger. This is accomplished by setting the power down charger bit:
   `HW_POWER_CHARGE[PWD_CHARGE] = 1`

5.2.2 Battery Detection

Battery detection is useful for production line situations where the storage media is soldered to the board before the battery. If the battery is not detected, the application behavior changes and performs as if the battery is never used as a power source.

The steps for battery detection are as follows:

1. Enable the automatic battery voltage update for the DC-DC control logic. This is performed to achieve an accurate battery voltage measurement.
2. Configure the LinReg for the 5 V power source. This is the default configuration when the 5 V supply powers ON the device.
3. Disable the battery-brownout power down and enable the battery-brownout comparator:
   — Set `HW_POWER_BATTMONITOR[PWD_BATTBRNOUT] = 0`
   — Set `HW_POWER_BATTMONITOR[BRWNOUT_PWD] = 0`
4. Enable the fast-settling bit for the battery detection:
   `HW_POWER_REFCTRL[FASTSETTLING] = 1`
5. Read the `HW_POWER_STS[BATT_BO]` bit:
   — If the bit is 1, battery brownout has occurred and the battery is not available
   — If the bit is 0, a battery is attached to the system
6. Reset the fast-settling bit to complete the battery detection:
   `HW_POWER_REFCTRL_CLR[FASTSETTLING] = 1`

5.2.3 Managing Charger and 4P2 Rail

As the battery charger and 4P2 rail share the 5 V current, arbitration is required to control the current flow. As the 4P2 rail has more control over the 5 V current, the 4P2 rail is equipped with the master current limiter and enable switch. Though the battery charger has its own current limiter and enable switch, both have less priority over the master settings.

Table 5 gives the master and local switches of the 4P2 rail and battery charger, respectively.

Table 5. 4P2 Rail and Battery Charger Switches

4P2 Rail Master Switches	Battery Charger Local Switches
HW_POWER_5VCTRL[CHARGE_4P2_ILIMIT]	HW_POWER_CHARGE[BATTCHRG_I]
HW_POWER_5VCTRL[PWD_CHARGE_4P2]	HW_POWER_CHARGE[PWD_BATTCHRG]

NOTE

To manage the current limiter and enable switch, ensure that the master switches are configured properly before the battery charger local switches are used.

6 Thermal Overload Protection

The i.MX28 processor integrates a thermal-overload protection circuit to shut down the device when the on-die temperature exceeds a programmed target value. In the i.MX28 processor, a bipolar-based voltage generator tracks the on-die temperature and generates a temperature-dependant voltage. This voltage is compared with a fixed voltage level. This triggers the device shutdown if the temperature-dependant voltage is greater than the fixed voltage.

The thermal-overload protection circuit is configured by the HW_POWER_THERMAL register. By default, this circuit is disabled. The default thermal trip point for this circuit is 115 ºC and can be programmed up to 150 ºC in steps of 5 ºC. To ensure the proper operation of the system, the die temperature should not exceed 85 ºC. The system operation at a higher temperature can cause permanent damages to the i.MX28 processor and the proper operation cannot be guaranteed.

6.1 Enabling Thermal Overload Protection

The steps to enable the thermal overload protection with the thermal trip point at 115 ºC are as follows:

1. Convert 115 ºC to the corresponding register setting:
   The formula to calculate the threshold temperature is provided in Equation 4:

$$THRESHOLD\_TEMP = BASE\_TEMP + (STEP\_SIZE \times NUM\_STEPS) \quad \textbf{Eqn. 4}$$

   Applying the corresponding values in Equation 4, 115 = 115 + (5 × NUM_STEPS)
   Therefore, NUM_STEPS = 0
2. Write the threshold temperature to the thermal reset register field:
   `HW_POWER_THERMAL [TEMP_THRESHOLD] = 0`
3. Enable the thermal overload protection:
   `HW_POWER_THERMAL [PWD] = 0`

NOTE

The LRADC channels, 8 and 9, are dedicated for the internal die temperature sensing. They should also be enabled for the thermal overload protection. For more information, refer to the LRADC section of the *i.MX28 Applications Processor Reference Manual* (MC1MX28RM).

7 Optimizing Power Subsystem

The i.MX28 processor offers many designed features to minimize the power usage, which mainly include the clock frequency manipulation. The designed features also include dynamic hardware changes that customize the PMU. This feature not only saves the power, but also provides better system performance.

7.1 Enabling Features

This section describes the steps that are required to enable the features that minimize the power usage and situations where these features are applied.

7.1.1 Interrupt Wait

The interrupt wait feature reduces the power consumed by the CPU clock. This feature gates the CPU clock when the processor is idle. To enable the interrupt wait, an assembly instruction is added to the application scheduler task.

The steps to enable this feature are as follows:

1. Add the following instruction before the scheduler becomes idle:
   ```
   mcr p15, 0, r0,  c7, c0,  4
   ```

NOTE

At this point, there should not be any tasks to be executed in the scheduler.

2. Enable the interrupt wait feature in the hardware:
   ```
   HW_CLKCTRL_CPU[INTERRUPT_WAIT] = 1
   ```

7.1.2 HCLK Auto Slow

HCLK auto-slow feature reduces the HBUS frequency when it is not used for any activities.

NOTE

The type of activity or traffic and reduced HBUS frequency are configurable.

The activity types that are monitored by the HCLK auto-slow logic are as follows:

- Data Co-Processor (DCP)
- Pixel Pipeline (PXP)
- High Speed ADC
- APBH Direct Memory Access (DMA)
- APBX DMA
- AMBA High-Speed Bus Architecture (AHB) master
- Less than three AHB masters
- CPU data access
- CPU instruction access

The steps to enable these feature are as follows:

1. Select the activity types that are to be monitored:
   - `HW_CLKCTRL_HBUS[DCP_AS_ENABLE]`
   - `HW_CLKCTRL_HBUS[PXP_AS_ENABLE]`
   - `HW_CLKCTRL_HBUS[ASM_EMIPORT_AS_ENABLE]`
   - `HW_CLKCTRL_HBUS[APBHDMA_AS_ENABLE]`
   - `HW_CLKCTRL_HBUS[APBXDMA_AS_ENABLE]`
   - `HW_CLKCTRL_HBUS[TRAFFIC_JAM_AS_ENABLE]`
   - `HW_CLKCTRL_HBUS[TRAFFIC_AS_ENABLE]`

— HW_CLKCTRL_HBUS[CPU_DATA_AS_ENABLE]

— HW_CLKCTRL_HBUS[CPU_INSTR_AS_ENABLE]

2. Select the slow-divide mode divider. This value divides the HCLK frequency by the configured amount. For example, to divide the HCLK frequency by 2:
   HW_CLKCTRL_HBUS[SLOW_DIV] = 2
3. Enable the auto-slow mode for the HBUS:
   HW_CLKCTRL_HBUS[AUTO_SLOW_MODE] = 1

7.1.3 Setting Power Transistor Size

The power transistors are modified during the run-time to optimize the DC-DC converter. This provides better performance and reduces the power consumption. Three settings are available for the power transistors—half-size Field Effect Transistor (FET), normal-size FET, and double-size FET. The half-size FET setting is used in low-power modes where the high resistance of the power FETs is acceptable. This setting disables half the power FETs. The double-size FET setting is used in high-power conditions to make the DC-DC converter more robust to handle heavier loads. The half-size and double-size FET options have their own enable bit. Though the half-size and double-size setting can be enabled simultaneously, it should not be performed as they do not cancel out.

NOTE

Enabling the normal-size FET setting is not same as enabling the half-size and double-size setting simultaneously.

The codes to enable the respective settings are as follows:

- Half-size FET:
  HW_POWER_MINPWR[DOUBLE_FETS] = 0
  HW_POWER_MINPWR[HALF_FETS] = 1
- Double-size FET:
  HW_POWER_MINPWR[HALF_FETS] = 0
  HW_POWER_MINPWR[DOUBLE_FETS] = 1
- Normal-size FET:
  HW_POWER_MINPWR[HALF_FETS] = 0
  HW_POWER_MINPWR[DOUBLE_FETS] = 0

7.1.4 PFM Mode

The DC-DC power usage can be reduced during low-power situations by using the Pulse-Frequency Modulation (PFM) mode. This mode is enabled only when the voltage outputs of the DC-DC converter are lightly loaded. The PFM mode is not used in high-power situations because the PMU cannot supply power reliably to the rails from the DC-DC converter. As the PFM mode causes high transient noise, this mode should be used only in deep-sleep or standby modes where the PMU is inactive.

NOTE

Ensure that all clocks are configured to the 24 MHz crystal state before the device is switched to the PFM mode.

The following code enables the PFM mode:

```
HW_POWER_MINPWR[EN_DC_PFM] = 1
```

The PFM mode should be one among the first power savings features to be disabled when the device exits its deep sleep or standby mode.

8 Conclusion

The i.MX28 processor offers a highly-efficient and comprehensive power supply. The instruction and descriptions provided in this application note should help any device that uses the i.MX28 processor to maximize its power performance, which extends the battery life.

9 References

The references for this application note are as follows:

- *i.MX28 Applications Processor Reference Manual* (MC1MX28RM)
- *i.MX28 Reference Schematics*

10 Revision History

Table 6 provides a revision history for this application note.

Table 6. Document Revision History

Rev. Number	Date	Substantive Change(s)
0	09/2010	Initial release.

THIS PAGE INTENTIONALLY LEFT BLANK

THIS PAGE INTENTIONALLY LEFT BLANK

THIS PAGE INTENTIONALLY LEFT BLANK

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or
+1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku
Tokyo 153-0064
Japan
0120 191014 or
+81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor
Literature Distribution Center
1-800 441-2447 or
+1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor
@hibbertgroup.com

Document Number: AN4199
Rev. 0
09/2010