TWL6030 Battery Charging

ABSTRACT

Smart phones, gaming handsets, and other handheld devices typically use rechargeable batteries. The TWL6030 device includes an efficient switched-mode charger with integrated switches in addition to the various regulators for the application processor. This document describes the use of the TWL6030 battery charger. The battery charging procedure together with charging related functions are described. The document also includes software examples. The specifications and the external components with the connections of the TWL6030 IC are described in the TWL6030 data sheet (TI literature number SWCS045), and the software registers are described in the TWL6030 Register Map, (TI literature number SWCU084).

Contents

List of Figures

1 Battery Charging

The TWL6030 device has an integrated switched-mode battery charger designed to charge the battery from a USB port. In addition, it can control an external battery charging IC (like BQ24156) during hardware-controlled charging and selects the priority of the chargers so that only one is enabled at time.

The block diagram of the USB charging electronics is shown in Figure 1. The figure shows the USB charging-related functions with external components, including the USB PHY which is a stand-alone device or integrated into application processor. The functions and external components related to VAC charging are described in Section 8.
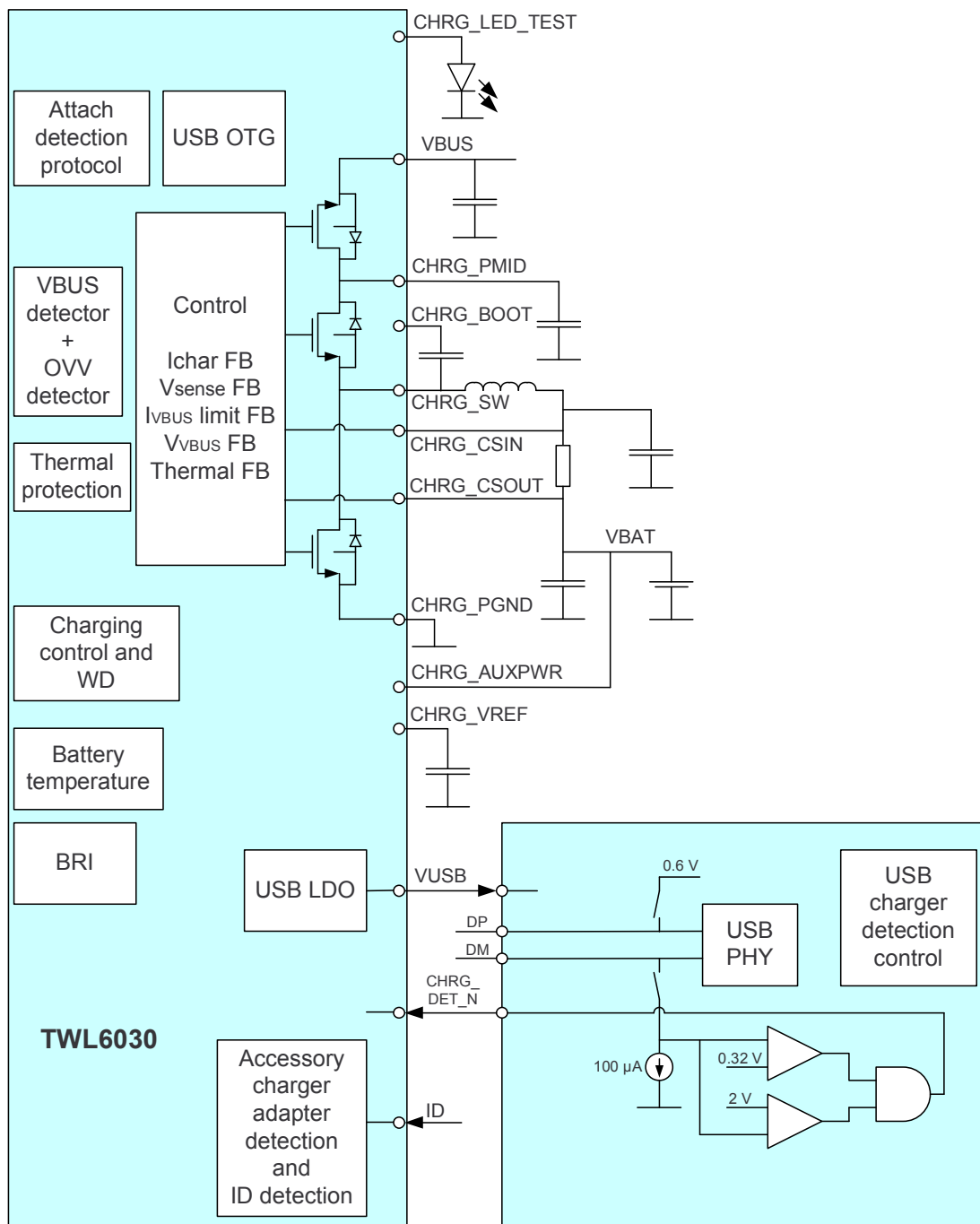

Figure 1. TWL6030 Battery Charger Block Diagram

2   Charging Profile

The TWL6030 device includes five analog loops that influence the charger output current:

- Constant current (CC) loop sensing the voltage accross the Rsense resistor and limiting the output current to the programmed level (VICHRG[3:0])
- Constant voltage (CV) loop sensing the battery voltage and limiting the output voltage to the programmed level (VOREG[5:0])
- VBUS voltage anticollapse loop sensing the VBUS voltage and preventing the VBUS voltage from dropping below the programmed level (BUCK_VTH[2:0])
- VBUS input current loop sensing the input current and limiting it below the programmed level (CIN_LIMIT[3:0])
- Thermal regulation loop limiting the power dissipation above 125°C temperature level

During the charging process all loops are enabled and the one that is dominating controls the output current. The battery charging profile consists of three phases: hardware controlled preconditioning, hardware controlled precharging, and software controlled full-charge phase. A typical charging profile and the different EPROM and software programmable parameters are shown in Figure 2.

The charging current is usually limited by the VBUS input current loop when charging from the standard downstream port because the limit must be set to 100 mA or 500 mA.

If the charging source cannot provide the current which the charger is drawing, the VBUS voltage decreases. The VBUS anticollapse loop senses the VBUS voltage and decreases the current so that the voltage does not fall below the programmed voltage level.
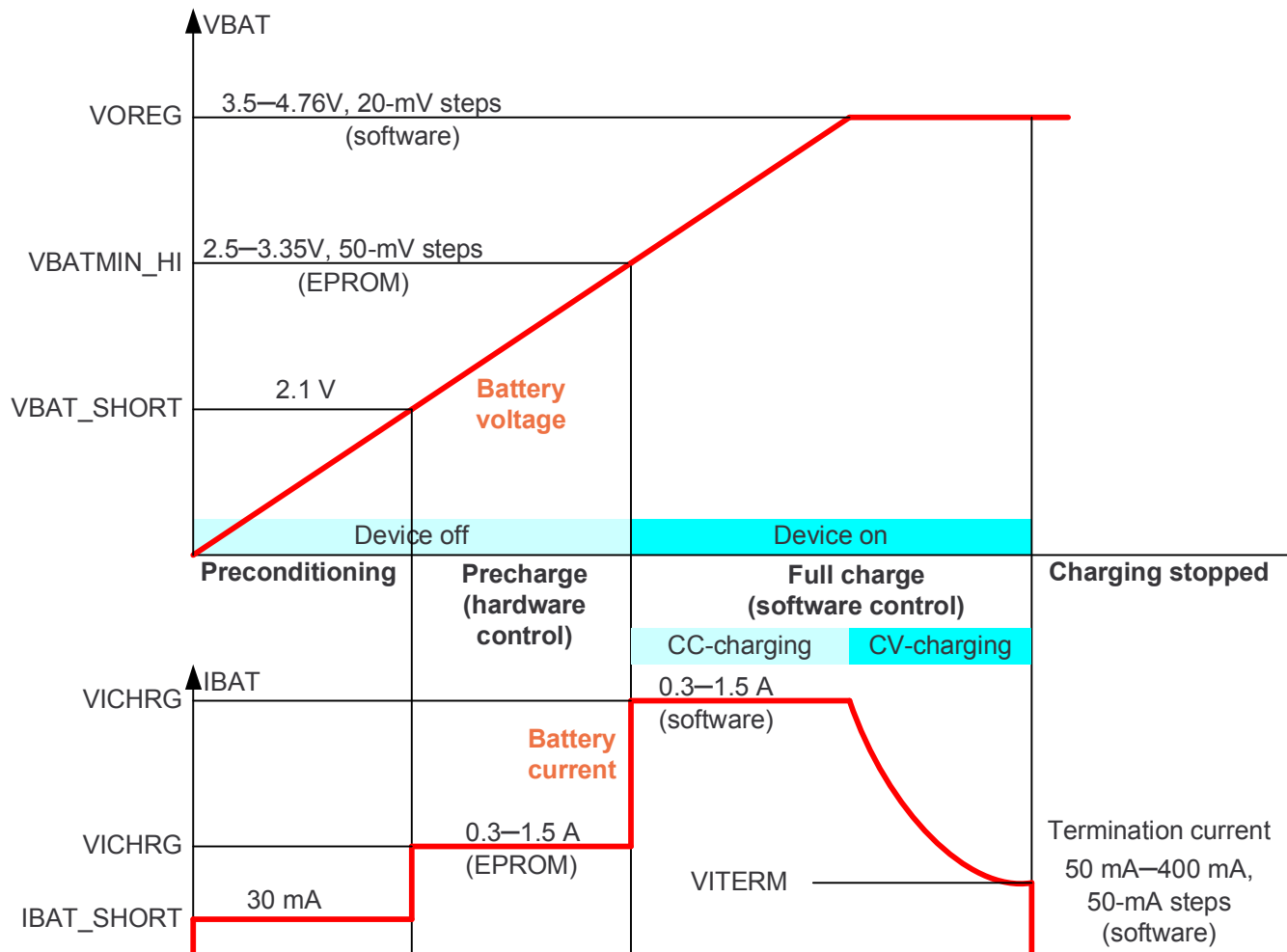

Figure 2. Typical Charging Profile

2.1 Preconditioning (Hardware Controlled)

The preconditioning phase is automatically enabled as soon as the charging source is detected and operates in a constant current charging mode. During preconditioning the battery voltage is below 2.1 V (VBAT_SHORT) and the charging current is limited to 30 mA (IBAT_SHORT). In this mode, the charger uses a linear charging operation mode. This phase detects a defective (shorted) battery and brings the battery voltage to a level acceptable for higher charging current. As soon as the battery voltage is above 2.1 V, a precharging phase is entered automatically.

2.2 Precharge (Hardware Controlled)

The precharging phase is automatically enabled when the charging source is detected, battery voltage is between VBAT_SHORT and VBATMIN_HI voltages, and the device is powered off. The battery is charged with a default VICHRG current which is programmable by EPROM. To fulfill the USB Battery Charging Specification, the VBUS input current is limited based on the USB charger detection signal (CHRG_DET_N) and accessory charger adapter (ACA) detection from the USB ID-line in the following way:

- Slow precharging for 2.1 V < VBAT < 3.54 V (VBUS current is limited to 92 mA from USB standard downstream port)
- Fast precharging for 2.1 V < VBAT < 3.54 V (VBUS current is limited to 470 mA and battery charging current set to default charging current value when USB charging port is detected)

The polarity of the CHRG_DET_N signal is defined by EPROM bit and is typically such that low level limits the input current to 92 mA.

When the battery voltage triggers the VBATMIN_HI threshold, the device is powered up and the charging enters the full-charge phase.

2.3 Full Charge (Software Controlled)

The full-charge phase is software controlled and it is entered when:

- The device is in power off with a battery voltage above VBATMIN_HI (the device is powered up and full-charge is started by software)
- The device is powered on and the charging source is detected
- The battery voltage is rising above VBATMIN_HI during precharge (the device is powered up and full-charge is started by software)

If the device is powered on when the charging source is detected, the TWL6030 device generates an interrupt and the charging is started by software. If the device is powered off or precharging is ongoing, the charging starts/continues with the default parameters until software takes control over the charging. During the full-charge phase, software sets the charging current (VICHRG) and voltage (VOREG) levels. During software-controlled charging, software must update the charging watchdog to continue the charging. This is a safety measure to ensure that software monitors the battery temperature and sets the charging current and voltage levels accordingly. When the battery is close to full charge, the charging automatically changes from CC charging into CV charging. The charging current decreases and when it triggers the termination current level (VITERM), the battery is full and the charging can be stopped. The termination detection can be programmed to stop the charging or just to send interrupt for software.

3 Safety Timer and Charging Watchdog

The TWL6030 device includes a safety timer for hardware-controlled preconditioning and precharging phases when the device is OFF, and a charging watchdog for software-controlled full-charge phase when the device is ON. The period of the safety timer depends on the charging control mode and the USB charger detection result, and it can be modified with EPROM bits.

In default settings the timer for the USB charging port is approximately 5 minutes, whereas for USB standard downstream port and customer-specific charger in VAC input the timer is 13 or 14 minutes. By using the EPROM bit it is possible to select double time (10 and 26 minutes) or fixed times for all charging sources (13 or 26 minutes).

During software-controlled charging, software can select the watchdog time up to 127 seconds. The transition from hardware-controlled charging to software-controlled charging occurs when the device is ON and the software updates for the first time the WDG_RST, WDT[6:0], VICHRG[3:0], or VOREG[5:0] bits. After that the software need to update the watchdog periodically in order to maintain the battery charging enabled. The different safety timer and watchdog times are summarized in Table 1.

Table 1. Battery Charging Safety Timer and Watchdog Times

EPROM Bits		Charging Control	Charging Source	Safety Timer / Watchdog	
CHWDT_DEP_DETN	CHWDT_DEP0			Minimum	Maximum
1	0	Hardware, device OFF	USB charger	5 min 4 s	5 min 36 s
1	0	Hardware, device OFF	Other sources	13 min 13 s	14 min 15 s
1	1	Hardware, device OFF	USB charger	10 min 10 s	11 min 12 s
1	1	Hardware, device OFF	Other sources	26 min 26 s	28 min 29 s
0	0	Hardware, device OFF	All sources	13 min 13 s	14 min 15 s
0	1	Hardware, device OFF	All sources	26 min 26 s	28 min 29 s
D.C.	D.C.	Software (programmable), device ON	All sources	0 s	127 s

4   Temperature Monitoring

JEITA requirements define the maximum battery charging current and voltage at different temperature ranges for Li-Ion batteries. The TWL6030 device supports the JEITA requirements with hardware-based temperature measurement gating the battery charging below and above the preset temperature values. Between these limits software must monitor the battery temperature using the integrated general purpose analog-to-digital converter (GPADC) and setting the charging current and voltage accordingly. Figure 3 shows the voltage and current limits at different temperatures.
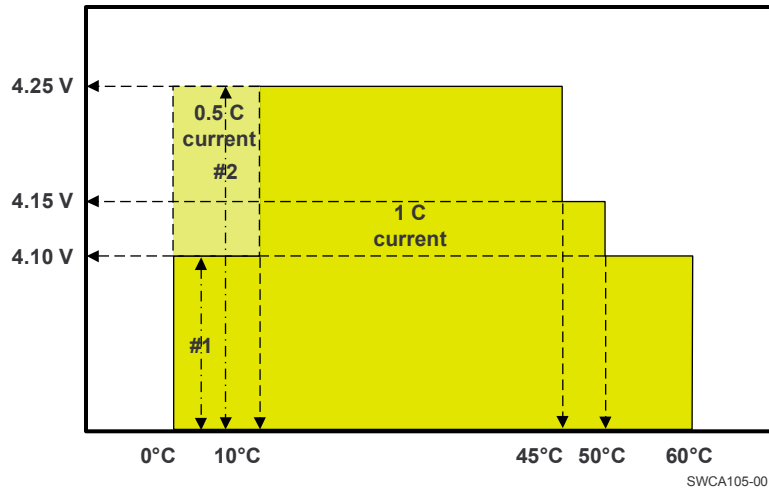

Figure 3. Charging Current and Voltage Limits at Different Temperatures

The figure allows two options for charging between 0°C and 10°C. #1 allows to charge up to 4.1 V with 1C current and #2 allows to charge up to 4.25 V with 0.5C current. The term 1C defines the charging current related to the battery capacity. For 1200 mA-h battery 1C corresponds to 1.2 A current.

The battery temperature is measured using an external NTC resistor. The measurement is enabled before the charging starts and the temperature is constantly monitored during charging. If the battery temperature is outside the valid range, the charging is gated; if the temperature returns inside the valid range, the charging continues. The gating of the charging can be disabled with an EPROM bit if needed. The module is enabled if VBUS or an external charger is detected. An interrupt is always generated when the battery temperature crosses the temperature limits in both directions. The interrupt generation can be masked if needed. Figure 4 shows the battery temperature measurement circuitry.
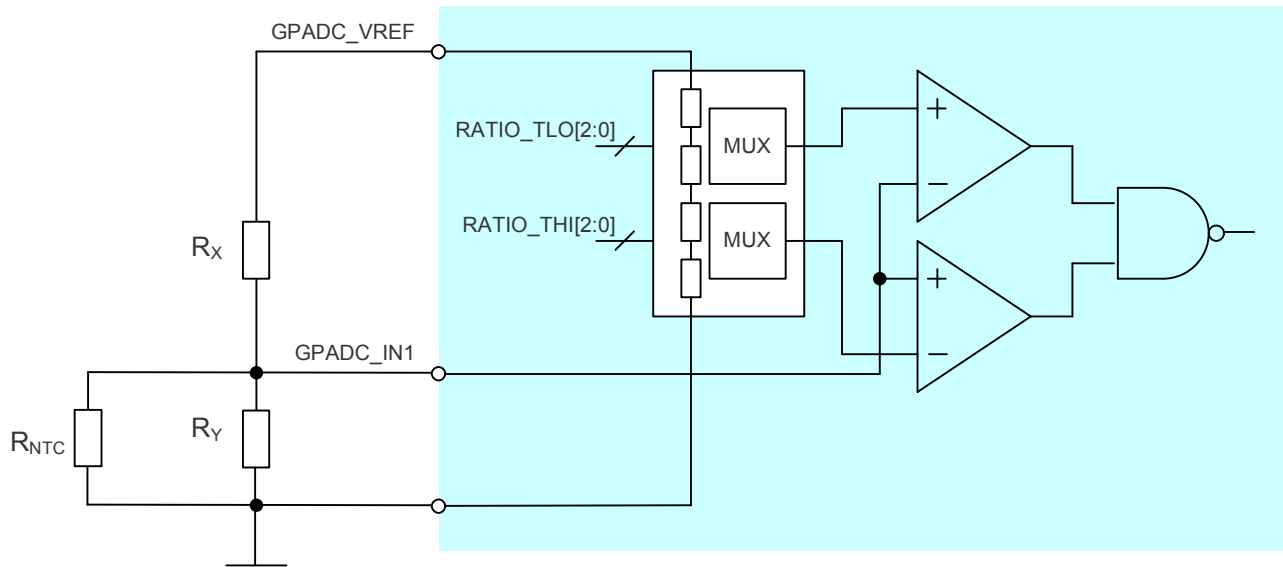

Figure 4. Battery Temperature Measurement

Because the NTC characteristic is highly nonlinear, it is combined with two resistors allowing linearization of its characteristic and making the sensitivity of the system more constant over a wide temperature range. The resulting voltage at GPADC_IN0 can be measured using the GPADC and is also monitored by two comparators that enable the charge of the battery only when the temperature is within a specified window, typically 0°C to 60°C. Resistors R_X and R_Y are used to set the desired temperature threshold levels.

5   Battery Presence Detection

The TWL6030 device supports battery detection. The presence of the battery can be detected with the GPADC_IN0 input signal. The interface has two different functions:
- Detect battery removal/presence
- Measure the size of the resistor connected to the GPADC_IN0 line in the battery pack using the GPADC battery pack removal is detected by a comparator that monitors GPADC_IN0

The battery pack must have a pulldown resistor (R_{BRI}) and the TWL6030 device has a current source (I_{BRI}) in the line. If the battery pack is removed, GPADC_IN0 rises above the comparator threshold level, the battery removal is detected, and the TWL6030 device sends an interrupt to the host processor and also drives BATREMOVAL pin. In addition, battery charging is gated if the battery is not present. Battery removal is detected with a comparator and a current source is supplied on the VRTC supply domain. This supply scheme allows the detection in a dead battery case configuration, because the VRTC can be supplied from the VBUS or VAC lines. The battery presence detection module is enabled during the charging and during the ACTIVE and SLEEP states. The charging gating function can be disabled by EPROM. Figure 5 shows a block diagram of the battery presence detection module.
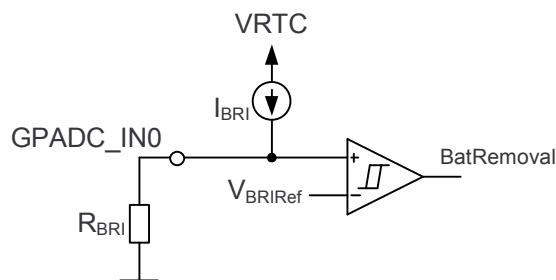

Figure 5. Battery Presence Detection

6   Charger Identification

The TWL6030 device has integrated ACA detection from the USB ID line and it supports external charger detection from D+/D– lines with CHRG_DET_N input signal. During preconditioning and precharge the TWL6030 device automatically delivers supply for USB PHY with USB low dropout voltage regultaror (LDO) so that the PHY can perform the charger detection from D+ and D– lines. The USB PHY indicates the detection result by the CHRG_DET_N signal, and the TWL6030 device changes the VBUS input current limit accordingly. In parallel, the TWL6030 device monitors the USB ID line, and if it detects ACA the VBUS input current limit is increased.

During full-charge phase the charger detection is not started automatically and software needs to take care of that.

7   USB Suspend

The TWL6030 device includes a SUSPEND_BOOT bit which is usable for USB suspend periods. The benefit of the bit is that it can be used to gate the charging without changing any charging parameters. When the suspend period ends, clearing the bit continues the battery charging.

8   External Charging IC Support

The TWL6030 device can be interfaced with an auxiliary stand-alone charger device to support the following use cases:
- Simultaneous battery charge from a non-USB charger (different connector) and OTG operating mode (the TWL6030 device internal USB charger used as VBUS supply)
- Charging from the sources not connected to USB connector

Figure 6 shows an example of an external charging IC supported by the TWL6030 device.
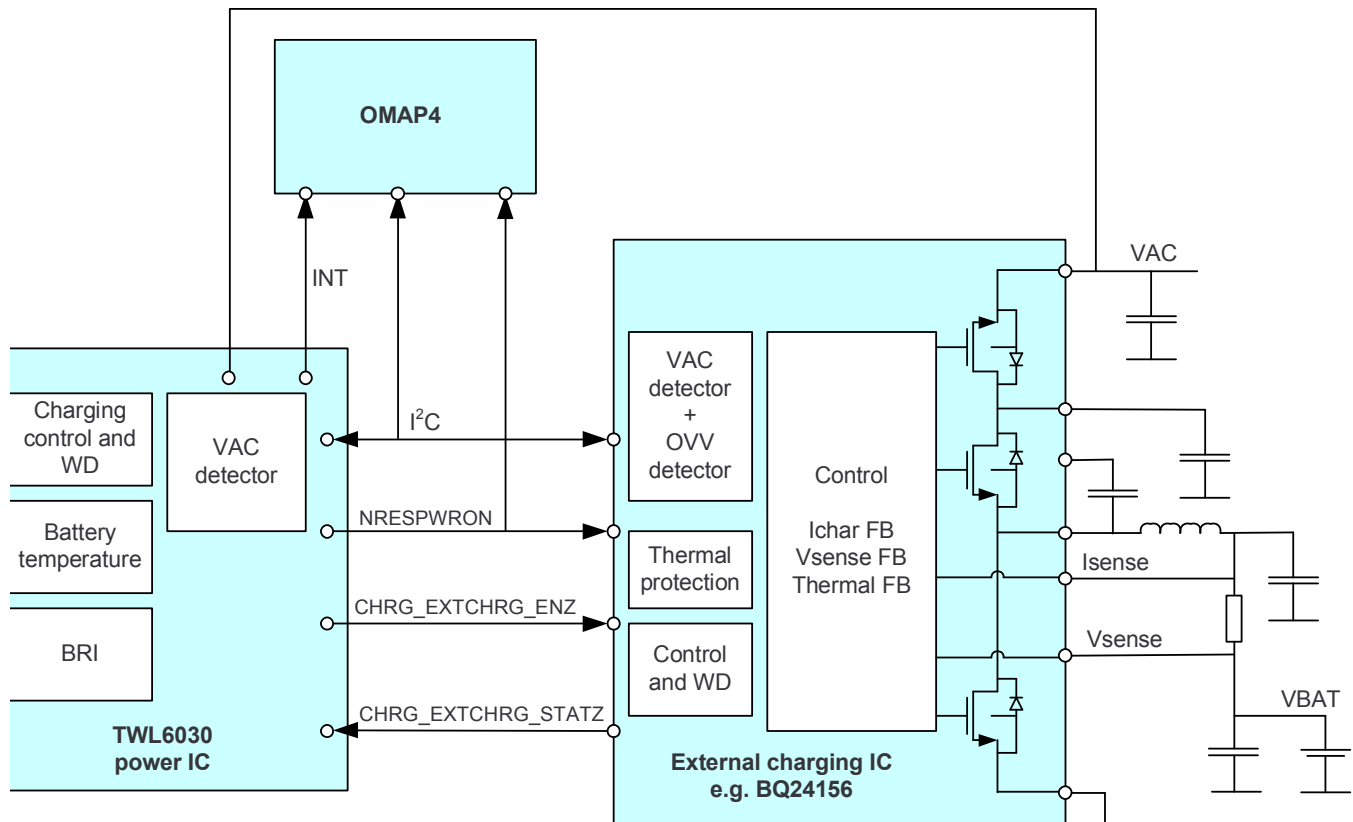
Figure 6. External Charging IC Interface

The external charging IC is enabled with 1.8-V CMOS level signal, CHRG_EXTCHRG_ENZ. Low logic level indicates that charging is enabled. Charging status is indicated with the CHRG_EXTCHRG_STATZ signal. The external charging IC pulls the signal down during charging. The integrated USB charger can be associated with an external VAC (wall) charger. For that reason, VAC wall charger input is connected to the TWL6030 device to define the charge priorities:

• When the VBUS is detected and the VAC is not detected, the USB charge starts.
• When the VAC is detected and the VBUS is not detected, the external charging starts.
• When the VBUS and VAC are detected, the USB charge starts only if the CHRG_DET_N pin is set high so the USB charge has a 475-mA input current limitation.
  – When CHRG_DET_N = 0 (100-mA input current USB limitation), the VAC wall charger is expected to be better (or equivalent) and thus is chosen as the default charge path.
  – When CHRG_DET_N = 1 (500-mA input current USB limitation), the precharge associated with a USB is expected to be sufficient for battery-level quick recovery if the USB charge path is chosen.

If there is a fault condition on a charger during hardware-controlled charging and the fault condition continues at least 2.5 seconds, the charging source is changed for a lower-priority charger. The change into a lower-priority charger only prevents the infinite looping between chargers. If only one charger is attached, the charger is not disabled in a fault condition, and if the fault condition does not disappear, the charging is terminated when the watchdog expires.

9   Limit Registers

During the full-charge phase, software sets the charging voltage and current. However, the TWL6030 device limits the current and voltage to a level that is defined in the limit registers. The limit registers in the TWL6030 device must be written just after the startup. Software must check the battery type and define the maximum charging current and voltage for the battery being used, write the limit values, and lock the limit registers with the LOCK_LIMIT bit, so that these cannot be changed when the device is powered on. The limit values are reset during power off by the NRESPWRON signal and they must be written by software during every power up. This ensures that third-party software or a virus cannot set a charging current or voltage that is too high. Figure 7 shows the structure of the limit and programming registers.
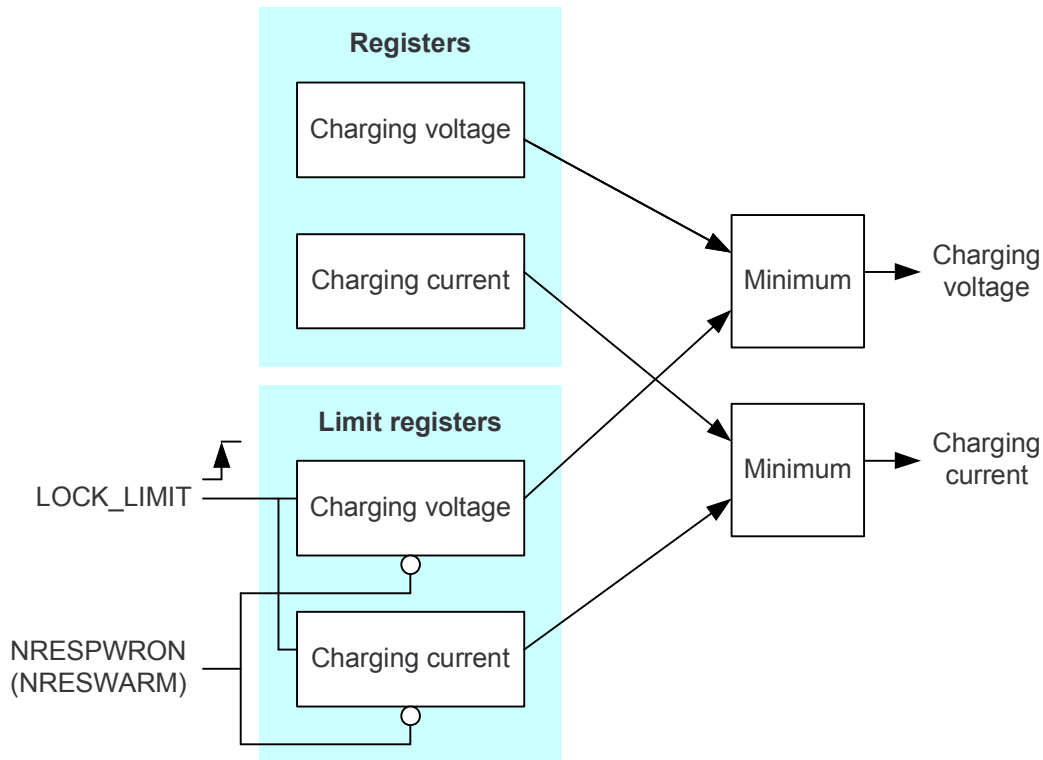
Figure 7. Charging Current and Voltage Limit Registers

10   Indication LED

The TWL6030 device has an indicator LED driver that indicates the battery charging is ongoing. During hardware-controlled charging, the LED driver is automatically enabled only if the USB charger or external charging IC is charging the battery, and it is turned off if the battery is not charged (for example, because of charger overvoltage). The supply for the charging indicator LED driver is generated from CHRG_PMID or VAC, depending on the active charging path. The CHRG_PMID pin is used instead of the VBUS line so that the LED indicator current is included into the VBUS input current limit.

During power on, the indicator LED indicates the battery charging by default. In addition, the software can also control the indicator LED regardless of the charging. The supply for the LED can be selected as CHRG_PMID, VAC, or CHRG_LED_IN. The current level can also be selected and the dimming function can be used. Dimming is done with a 128-Hz PWM signal, which has 255 linear steps. The LED output pin has a selectable pulldown when the module is disabled; the pulldown is enabled by default.

11 Charging Procedure

Software must define several parameters for safe battery charging. During every startup, software must detect the battery type and size, set the charging current and voltage limit registers, and lock the limit registers. This must be done even if the charger is unattached.

The charging period starts when the VBUS voltage or VAC voltage is detected. The default charging parameters are taken in use and if the charging parameters are changed, they are stored until the charging period ends (that is, charger is removed [VBUS and VAC voltages are not valid], charging watchdog expires, software terminates the charging, warm reset, or shutdown).

When software detects a charger or the device is powered up after charger detection, software must identify the charging source and set charging parameters accordingly. If the power source is in USB port, software must identify if the source is dedicated charging port (DCP), charging downstream port (CDP), ACA or standard downstream port (SDP). The USB port terms are defined in USB Battery Charging Specification (http://www.usb.org/developers/devclass_docs/Battery_Charging_V1_2.zip).

In case of standard downstream port, software must connect to USB host, enumerate and configure to 500-mA or 100-mA device to get permission to charge the battery. Software must set the VBUS input current limit accordingly and monitor suspend mode during charging and gate the battery charging during suspend with SUSPEND_BOOT bit.

If the source is dedicated charging port, charging downstream port, or ACA, the source should provide at least 500-mA current. Software must set the VBUS input current limit, battery charging current and voltage, VBUS anticollapse threshold, charging termination current, and termination options.

If the power source is VAC, the charging parameters are programmed into external charging IC.

Before selecting the charging source and enabling the battery charging, software must measure the battery temperature and adjust the charging current and voltage accordingly. Battery manufacturer's may have their own limitations which must be taken into account. After that software must update the charging watchdog and enable the charging.

Software must update the charging watchdog to maintain the battery charging process. Software can use, for example, a 1-minute timer to wake-up the software for the watchdog update. Software must check the battery temperature and adjust the charging current and voltage if needed before updating the charging watchdog. Figure 8 shows the battery charging process.

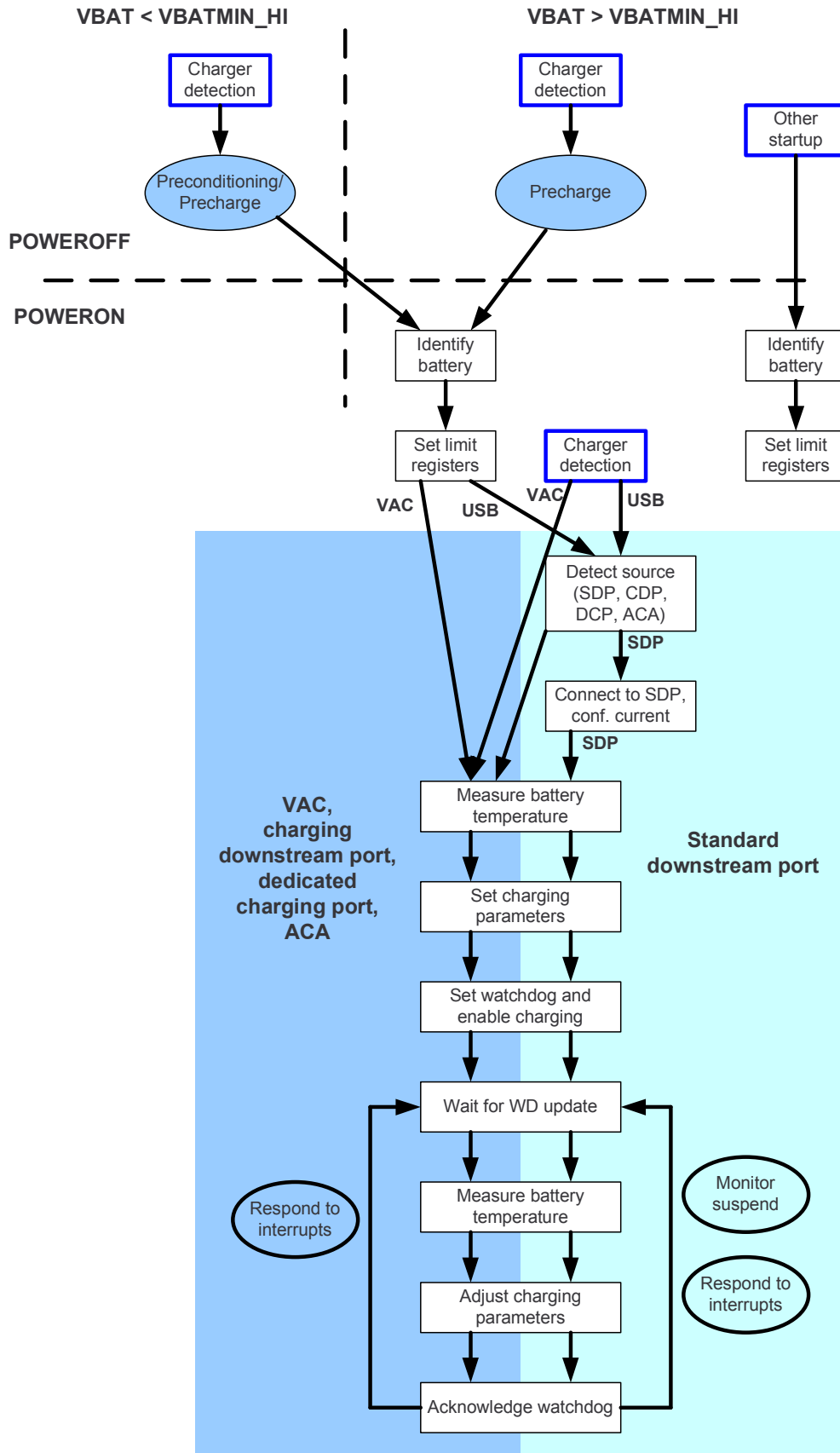**VBAT < VBATMIN_HI**   **VBAT > VBATMIN_HI**

Charger detection → Preconditioning/Precharge

Charger detection → Precharge

Other startup

POWEROFF

POWERON

Preconditioning/Precharge, Precharge → Identify battery

Identify battery → Set limit registers

Other startup → Identify battery → Set limit registers

VAC   **USB**   **VAC**   **USB**

Charger detection

Detect source (SDP, CDP, DCP, ACA)

SDP

Connect to SDP, conf. current

SDP

VAC, charging downstream port, dedicated charging port, ACA

Measure battery temperature

Set charging parameters

Set watchdog and enable charging

Wait for WD update

Respond to interrupts

Measure battery temperature

Adjust charging parameters

Acknowledge watchdog

Standard downstream port

Monitor suspend

Respond to interrupts

Figure 8. Battery Charging Process

During the battery charging process interrupts can occur which need software interaction. Interrupt handling is described in Section 12.

12   Interrupt Handling

Figure 9 shows the interrupt generation structure. The marking (M) means the signal is maskable so that it will not trigger an interrupt. However, the related status signal is still visible in the register.
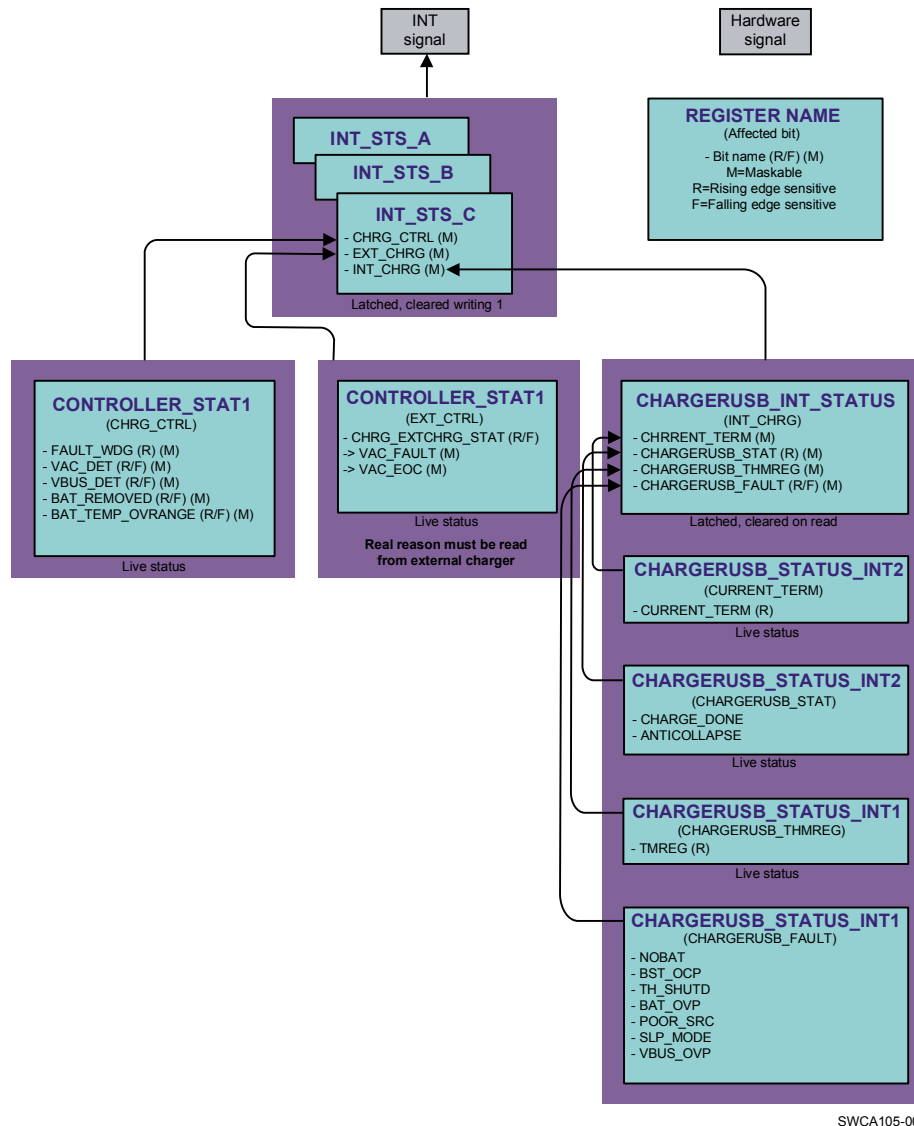
Figure 9. Interrupt Generation Architecture

When the INT interrupt signal is set, software must read the reason of the interrupt by reading the three interrupt status registers (INT_STS_A, INT_STS_B, and INT_STS_C). The battery charging related register bits (CHRG_CTRL, EXT_CHRG, and INT_CHRG) are in the INT_STS_C register. The source of the interrupt is

- CHRG_CTRL: The interrupt source is in the charger controller.
- EXT_CHRG: The interrupt source is in external charging IC.
- INT_CHRG: The interrupt source is in the TWL6030 battery charger.

The CHRG_CTRL indication can be further identified from the CONTROLLER_STAT1 register. This register shows the actual status of the different interrupt sources, so if the situation disappears before software can read it, software cannot know the real reason for the interrupt.

The origin of the external charger interrupt must be read from the external charging IC. The CHRG_EXTCHRG_STATZ bit shows the actual level of the status signal.

The source for the INT_CHRG interrupt must be further clarified in the CHARGERUSB_INT_STATUS register, which stores the latched information. The bits in the CHARGERUSB_INT_STATUS register are cleared by read access.

12.1 Sources of the Interrupt

The interrupt sources are described below, together with the actions software must take.

12.1.1 Charger Controller Interrupts

The charger controller interrupts are:

- **FAULT_WDG**: The charging watchdog has expired. Software must reset the timer in the CONTROLLER_WDG register. In addition, software must initialize the charging and restart it.
- **VAC_DET**: The VAC detection threshold voltage has been crossed. The VAC charger has been either inserted or removed. If the VAC charger is inserted, software can initialize and enable the charging and select the charging source using the CONTROLLER_CTRL1 register bits. The parameters for the VAC charging are programmed in the external IC.
- **VBUS_DET**: The VBUS detection threshold voltage has been crossed. The USB plug has been either inserted, removed, or the VBUS delivery has been started or stopped. If the USB plug is inserted, the software can initialize and enable the charging and select the charging source using the CONTROLLER_CTRL1 register bits.
- **BAT_REMOVED**: The battery is either removed or inserted. This battery detection is based on the pulldown resistor on the GPADC_IN0 line. This feature can be enabled/disabled using the EPROM bit.
- **BAT_TEMP_OVRANGE**: The battery temperature has crossed the minimum or maximum temperature limit set by the external resistors. The battery charging is gated outside the valid range automatically by hardware. This feature can be enabled/disabled using the EPROM bit.

12.1.2 External Charger Interrupt

The external charger interrupt is:

- **EXT_CHRG**: Interrupt from the external charger IC. The reason for the interrupt must be read from the external charger IC.

12.1.3 Internal Charger Interrupts

The internal charger interrupts are:

- **CURRENT_TERM**: The current termination level has been detected. The functionality must be controlled by software using the TERM and CHARGE_ONCE bits in the CHARGERUSB_CTRL1 and CHARGERUSB_CTRL3 registers.
- **CHARGE_DONE**: The charging termination current level is detected. Software can terminate the battery charging.
- **ANTICOLLAPSE**: The anticollapse loop limiting the VBUS input voltage drop is active.
- **TMREG**: The thermal regulation loop of the USB charger is active. The temperature of the IC must be monitored and if it becomes too high, some power dissipation must be removed by decreasing the charging current or by disabling functions.
- **NOBAT**: The battery is not detected by the USB charger. This feature is enabled/disabled using the EPROM bit.
- **BST_OCP**: The OTG boost supply over-current protection is active. There can be a short circuit or an excessively high load in the VBUS. The boost regulator must be disabled if the VBUS voltage is not increasing.
- **TH_SHUTD**: The temperature of the USB charger is higher than the threshold level and the battery charging is gated. The power dissipation must be decreased by reducing the charging current or by disabling some functions.
- **BAT_OVP**: The battery voltage is higher than the overvoltage threshold, and the charging is gated. Something may be wrong in the battery charging and it should be stopped.

- **POOR_SRC**: The VBUS voltage is between the system supply voltage and the minimum VBUS detection voltage. The voltage is too low for battery charging. The charging from VBUS must be terminated.
- **SLP_MODE**: The charger is in a low-power sleep mode. The VBUS voltage is below the sleep-mode entry threshold and VBUS is higher than the VBUS detection threshold.
- **VBUS_OVP**: The VBUS voltage is higher than the overvoltage threshold and the charging is gated. The voltage can be monitored with the ADC and if the high voltage situation continues, the charging from VBUS must be terminated by software.

13   Battery Charging Registers

Table 2 lists the registers that control battery charging.

Table 2. Charger Register Map

ID	Address	Register name	Description
49	0xE0	CONTROLLER_INT_MASK	Masking of interrupt sources
49	0xE1	CONTROLLER_CTRL1	Enable and select charger
49	0xE2	CONTROLLER_WDG	Charging watchdog
49	0xE3	CONTROLLER_STAT1	Status of signals
49	0xE4	CHARGERUSB_INT_STATUS	Status of interrupt sources
49	0xE5	CHARGERUSB_INT_MASK	Masking of interrupt sources
49	0xE6	CHARGERUSB_STATUS_INT1	Status of interrupt sources
49	0xE7	CHARGERUSB_STATUS_INT2	Status of interrupt sources
49	0xE8	CHARGERUSB_CTRL1	Control signals for charging and OTG supply
49	0xE9	CHARGERUSB_CTRL2	Termination current setting
49	0xEA	CHARGERUSB_CTRL3	Control signals for charging
49	0xEB	CHARGERUSB_STAT1	USB charger status (for debugging purposes)
49	0xEC	CHARGERUSB_VOREG	Charging voltage setting
49	0xED	CHARGERUSB_VICHRG	Charging current setting
49	0xEE	CHARGERUSB_CINLIMIT	VBUS input current limit setting
49	0xEF	CHARGERUSB_CTRLLIMIT1	Limit of charging voltage
49	0xF0	CHARGERUSB_CTRLLIMIT2	Limit of charging current, locking of the limit registers
49	0xF1	ANTICOLLAPSE_CTRL1	VBUS anticollapse settings
49	0xF2	ANTICOLLAPSE_CTRL2	VBUS anticollapse settings
49	0xF3	ANTICOLLAPSE_STAT1	VBUS anticollapse status
49	0xF4	LED_PWM_CTRL1	Indicator LED control
49	0xF5	LED_PWM_CTRL2	Indicator LED control

14 Software Examples

14.1 Setting the Limit Registers

1. Write (49h, 0xF0) <= 0x09 /* Write the maximum charging current (that is, 1000 mA).
2. Write (49h, 0xEF) <= 0x22 /* Write the maximum charging voltage (that is, 4.18 V).
3. Read (49h, 0xF0) /* Read the maximum charging current.
4. Read (49h, 0xEF) /* Read the maximum charging voltage.
5. Write (49h, 0XF0) <= 0x19 /* Lock the charging limit registers if readings are correct.

14.2 Enabling/Disabling the Charging

1. Write (49h, 0xE1) <= 0x10 /* Enable the USB charging.
2. Write (49h, 0xE1) <= 0x18 /* Enable the VAC charging.
3. Write (49h, 0xE1) <= 0x00 /* Terminate the battery charging.
4. Write (49h, 0xE8) <= 0b100x0000 /* Gate the current from VBUS for a while (that is, during suspend).
5. Write (49h, 0xE8) <= 0b000x0000 /* Continue to draw the current from VBUS after gating.

14.3 Setting/Adjusting the Charging Parameters

1. Write (49h, 0xED) <= 0x07 /* Set the USB charging current level (that is, 800 mA).
2. Write (49h, 0xEC) <= 0x1D /* Set the USB charging voltage level (that is, 4.08 V).
3. Write (49h, 0xE9) <= 0x20 /* Set the USB charging termination current level (that is, 100 mA).
4. Write (49h, 0xE8) <= 0bxxx10000 /* Enable the charge current termination detection.
5. Write (49h, 0xE8) <= 0bxxx00000 /* Disable the charge current termination detection.
6. Write (49h, 0xEA) <= 0bx1x0x00x /* USB charging is terminated when the termination current level is reached.
7. Write (49h, 0xEA) <= 0bx0x0x00x /* USB charging is not terminated when the termination current level is reached.
8. Write (49h, 0xEE) <= 0x09 /* Set the VBUS input current limit, that is (500 mA).
9. Write (49h, 0xF1) <= 0x04 /* Enable the analog anticollapse loop for the USB charger.
10. Write (49h, 0xF2) <= 0x60 /* Set the VBUS anticollapse loop voltage (that is, 4.44 V).

14.4 Finding the Interrupt Source

1. Read (49h, D2) /* Read the Status C register, charging related bits are 6, 5, and 4.
2. Write (49h, D2) <= 0bx000xxxx /* Clear the charging interrupt bits.

Using the structure in Figure 9, find the reason for the interrupt from the following registers:

1. Read (49h, E3) CONTROLLER_STAT1
2. Read (49h, E4) CHARGERUSB_INT_STATUS
3. Read (49h, E6) CHARGERUSB_STATUS_INT1
4. Read (49h, E7) CHARGERUSB_STATUS_INT2

Clear the interrupt bit if needed and continue with the corresponding action. Sometimes the reason of the interrupt can disappear before reading; in this case the charging can be continued normally.

14.5 Battery Voltage Measurement

1. Write (49h, 0x90) <= 0b0000xx10 /* Enable the GPADC.
2. Write (48h, 0xE4) <= 0b000xxx1x /* Enable the battery voltage scaler.
3. Write (49h, 0x2E) <= 0bxxxx1xxx /* Select the battery voltage scaler. VBAT/4 gives better accuracy if the result is not saturated.
4. Write (49h, 0x33) <= 0x08 /* Start the conversion.
5. Wait for the interrupt.
6. When the ADC conversion ready interrupt is detected:

   (a) Read (49h, 0x65), Read (49h, 0x66) /* Read the conversion result of channel 7.
   (b) Calculate the correct battery voltage level using the calibration bits.
   (c) Write (48h, 0xE4) <= 0b000xxx0x /* Disable the battery voltage scaler.
   (d) Write (49h, 0x90) <= 0b0000xx01 /* Disable the GPADC if there is no need for other measurements.

14.6 Battery Temperature UPDATE Measurement

1. Write (49h, 0x90) <= 0b0000xx10 /* Enable the GPADC.
2. Write (49h, 0x2E) <= 0bxxxxxxx1 /* Select the battery temperature measurement.
3. Write (49h, 0x33) <= 0x08 /* Start the conversion.
4. Wait for the interrupt.
5. When the ADC conversion ready interrupt is detected:

   (a) Read (49h, 0x59), Read (49h, 0x5A) /* Read the conversion result of channel 1.
   (b) Calculate the correct voltage level using the calibration bits.
   (c) Calculate the temperature level based on the NTC and limit resistor values.
   (d) Write (49h, 0x90) <= 0b0000xx01 /* Disable the GPADC if there is no need for other measurements.

14.7 Updating the Charging Watchdog

Write (49h, 0xE2) <= 0xBC /* Update the charging watchdog (that is, with 60 seconds).

14.8 Control the LED Indicator

1. Write (49h, F4) <= 0xC0 /* Set 75% dimming level (192/256) on the indicator LED.
2. Write (49h, F5) <= 0x0x11xxxx /* Set current level (when dimming is 256/256) for the indicator LED (that is, 5 mA).
3. Write (49h, F5) <= 0x01xxxxxx /* Disable the active pulldown from the LED indicator output.
4. Write (49h, F5) <= 0x00xxxxxx /* Enable the active pulldown from the LED indicator output when the LED indicator is disabled.
5. Write (49h, F5) <= 0x0xxx1001 /* Force the LED driver supply (that is, EXTPIN [CHRG_LED_IN]).
6. Write (49h, F5) <= 0x0xxxxx10 /* Disable the LED driver totally.

15 Reference Documents

Ref[1]	USB Battery Charging Specification, Revision 1.2: http://www.usb.org/developers/devclass_docs/Battery_Charging_V1_2.zip

16 Revision History

The following table summarizes the TWL6030 Battery Charging Application Note versions.

Note: Numbering may vary from previous verisons.

Version	Literature Number	Date	Notes
*	SWCA105	April 2011	See [1]
A	SWCA105A	July 2011	See [2]

[1] TWL6030 Battery Charging Application Note, (SWCA105) - initial release.
[2] TWL6030 Battery Charging Application Note, (SWCA105A):
- Remove NDA statement

IMPORTANT NOTICE