APPLICATION NOTE

AT07146: Low Power Design Consideration in Thermostat with SAM4L

Atmel 32-bit Microcontroller

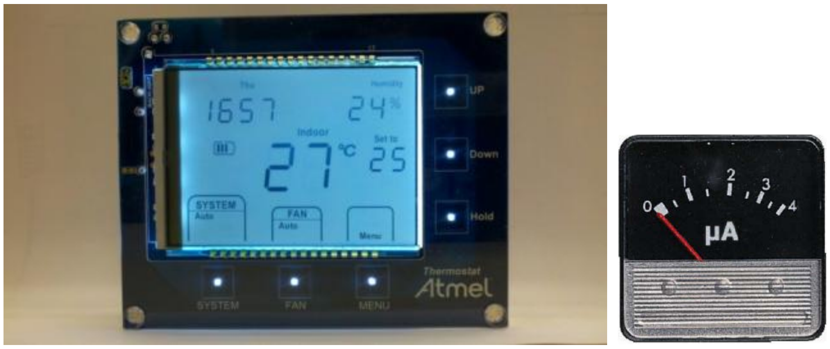
Introduction

This application note mainly describes the low power design consideration in a thermostat with SAM4L. The thermostat with SAM4L reference design can be found at www.atmel.com. The related application notes are: Atmel® AT03197: Thermostat with Touch and Wireless Connectivity Hardware User's Guide and Atmel AT03198: Thermostat with Touch and Wireless Connectivity - Software User's Guide.

Features

- Atmel picoPower® Technology
- Atmel® ATSAM4LC4C Microcontroller
- Tickless FreeRTOS™

Figure 1. Thermostat with picoPower Technology



Atmel started picoPower technology on the popular 8-bit AVR® products, extended it to 32-bit AVR products and it is available for Cortex®-M4 MCUs too. The SAM4L MCU is designed from the ground up for the lowest power consumption. Together with picoPower technology, the system level firmware and hardware designs in the thermostat give a good reference on how to achieve low power with an Atmel SAM4L device.

Table of Contents

1. Overview

Embedded with Atmel picoPower technology, the Atmel SAM4L microcontrollers redefine the power benchmark for Cortex-M4 processor-based devices. It delivers the lowest power in active mode (down to 90µA/MHz) as well as in sleep mode with full RAM retention (1.5µA) with the shortest wake-up time (down to 1.5µs). The SAM4L family is ideal for power-sensitive designs. For details about Atmel picoPower technology, refer to http://www.atmel.com/technologies/lowpower/picopower.aspx.

Figure 1-1. Atmel picoPower Technology on ARM Platform

In the thermostat reference design, several low power techniques of SAM4L device are used.

- Power Save Modes

  SAM4L supports multiple power save modes to achieve optimized power consumption in different low power modes. The thermostat is designed to work in low power WAIT mode as long as possible without any compromise on function and performance. The typical power consumption in WAIT mode is about 3.5µA.

- Power Scaling

  The Power Scaling technique is intended to scale the internal regulator output voltage. According to the requirements in terms of performance, operating modes, and current consumption, the user can select the Power Scaling configuration that fits the best with their application. For the thermostat, it is running in Power Scaling 1.

- Fast wake up

  There are many clock source options in SAM4L device. Flexible clock management is provided to select the best clock for the user application. When switching between low power modes and active mode, the fast wakeup time is required to achieve lower power. For the thermostat, RCFAST is used and it typically provides <1.5µs wakeup time.

- Efficient regulator mode

  By selecting proper regulator mode, we can get optimized power efficiency on SAM4L. For the battery powered thermostat, switching mode (BUCK) is used as it provides long battery life and good performance.

- Low power peripheral

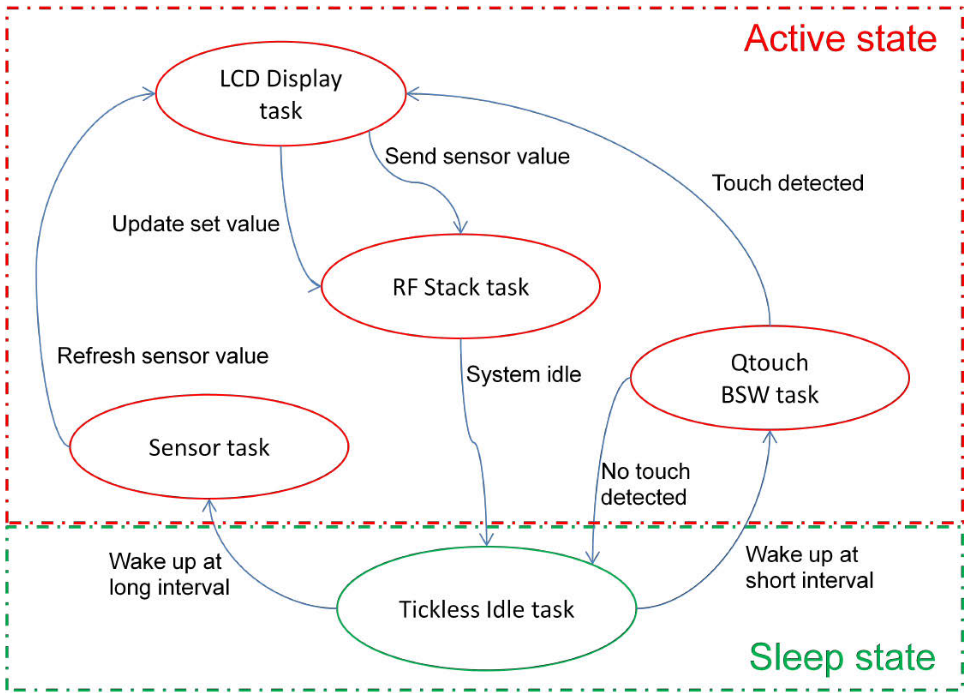
  The peripheral design on SAM4L is optimized for low power application. The CATB and LCDCA peripherals are used in the thermostat. CATB is used for touch buttons and LCDCA is used for thermostat display. Both of the peripherals are designed to be able to process input data or signals independently of the CPU, so CPU load is dramatically reduced. It means CPU can be put into low power mode most of the time and the total power consumption is greatly reduced.

2. Low Power Design Consideration

2.1 Thermostat Working State

For a microcontroller, the CPU usually consumes 40%~80% of total chip power consumption. To achieve low power, it is better to put CPU into low power mode as long as possible. As shown in Figure 2-1, the thermostat is designed to have two working states; active state and sleep state. In the sleep state, the device will be put into WAIT mode, in which the Cortex-M4 core and all clock sources are stopped except 32kHz which is enabled for the AST and LCDCA peripherals. To save power, the firmware is designed to stay in the sleep state as much as possible.

Figure 2-1. Thermostat Application Block Diagram



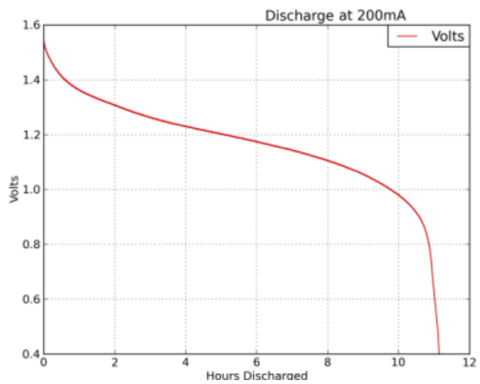
All the above application tasks and working states are managed by tickless FreeRTOS. When all the application tasks are blocked or suspended, FreeRTOS will run the idle task, in which the device will be put into WAIT mode until a task is required to run again. In the WAIT mode SleepWalking is supported and RAM contents are retained.

To realize tickless mode, SAM4L AST peripheral is used as tick source for FreeRTOS. AST peripheral is able to generate wakeup signal when the counter overflows, when the counter reaches the selected alarm value, or when the selected prescaler bit has a 0-to-1 transition. Here AST alarm is enabled as wake up source.

2.2 Power Supply Strategy

The thermostat is designed to work with two AA batteries. For commonly used AA alkaline batteries, the discharge curve is shown in Figure 2-2. The discharge current of the thermostat is far below 200mA, but the pattern of discharge curve should be similar. From the discharge curve, we can see ~80% of battery life is above 1.1V. And the cut off voltage of AA battery is around 1.0V. The voltage will drop quickly if it goes below the cut off voltage, which makes the battery unusable. So the valid input voltage for the thermostat is 2.2V ~ 3V.

Figure 2-2. Typical AA Alkaline Battery Discharge Curve



Depending on the input voltage range and final application, it is recommended to choose the most efficient power strategies for SAM4L application. As shown in Table 2-1, the embedded voltage regulator can be set in two modes and the device can be set in different Power Scaling configuration. When input voltage is above 2.3V, it is more efficient to set the embedded voltage regulator in switching mode. It is done by pulling high PA02 pin at POR33 reset. In this mode, a device running in Power Scaling 1 (PS1) configuration typically gives the most optimal power efficiency.

Table 2-1. SAM4L Power Strategy

<table>
<tr><th></th><th colspan="4">VDDIN Voltage</th></tr>
<tr><th></th><th>1.68V</th><th>1.80V</th><th>2.00V</th><th>2.30V … 3.60V</th></tr>
<tr><td>Switching Mode
(BUCK/LDOn
(PA02) =1)</td><td colspan="2">N/A</td><td>Possible but not efficient</td><td>Optimal power efficiency</td></tr>
<tr><td>Linear Mode
(BUCK/LDOn
(PA02) =0)</td><td colspan="3">Optimal power efficiency</td><td>Possible but not efficient</td></tr>
<tr><td>F_{CPUMAX}</td><td>12MHz</td><td colspan="3">Up to 36MHz In PS0
Up to 12MHz in PS1
Up to 48MHz in PS2</td></tr>
<tr><td>PowerScaling</td><td>PS1</td><td colspan="3">ALL</td></tr>
<tr><td>Typical power consumption in RUN mode</td><td colspan="3">212µA/MHz @ F_{CPU}=12MHz(PS1)
306µA/MHz @ F_{CPU}= 48MHz(PS2)</td><td>100µA/MHz @ F_{CPU}=12MHz(PS1) @ V_{VDDIN}=3.3V
180µA/MHz @ F_{CPU}=48MHz(PS2) @ V_{VDDIN}=3.3V</td></tr>
<tr><td>Typical power consumption in RET mode</td><td colspan="4">1.5µA</td></tr>
</table>

From the above analysis, in this thermostat application, the SAM4L embedded voltage regulator is set in switching mode and the device is configured to run in PS1 configuration to achieve most optimal power efficiency.

2.3 I/O Pin State

2.3.1 Sensor Pins

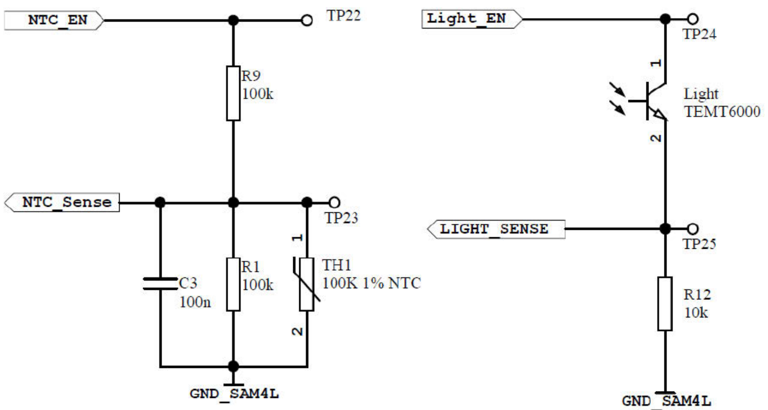
On the thermostat, there are three types of sensors:

- Temperature sensor
- Humidity sensor
- Ambient light sensor

To save power, the sensors are powered only when required. Most of the time (both active state and sleep state), the sensor circuits are not powered.

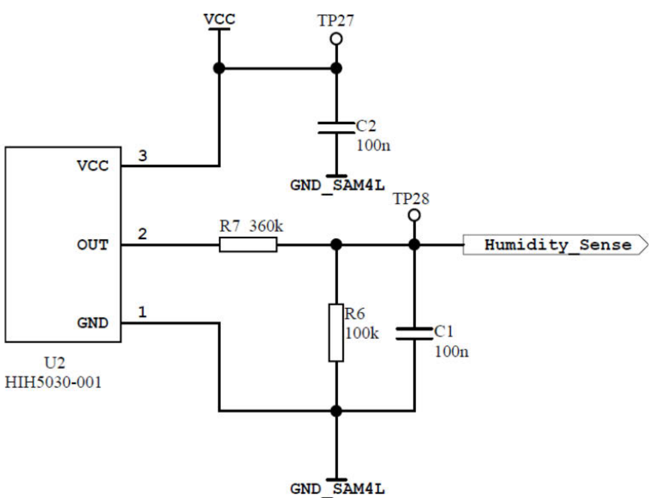
As shown below, the temperature sensor and ambient sensor are powered by NTC_EN and Light_EN which are connected to I/O pins directly. When required, the I/O pins will be set high to supply power to temperature sensor and ambient light sensor. After ADC conversion is done, the I/O pins are set low to cut off the power to the sensor circuit.

Figure 2-3. Temperature Sensor and Ambient Light Sensor



According to the humidity sensor datasheet, it is required to operate above 2.7V. For proper operation, the VCC of the humidity sensor is connected to a boost IC which provides 3.3V power. The boost IC is enabled by I/O pin and it is also enabled on demand.

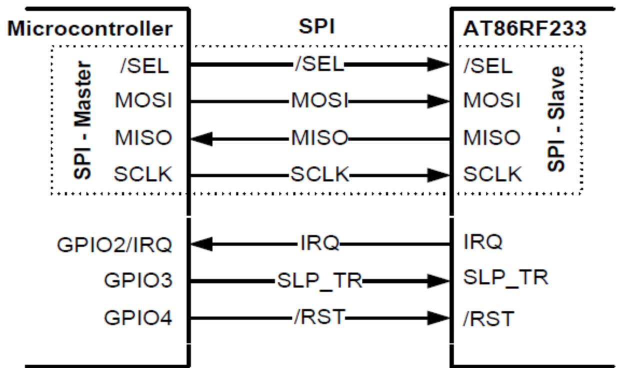
Figure 2-4. Humidity Sensor



2.3.2 AT86RF233 Interface Pins

An Atmel AT86RF233 is used in the thermostat to realize 2.4GHz wireless communication. The interface between SAM4L and AT86RF233 are SPI interface and several control pins.

Figure 2-5. SAM4L and AT86RF233 Interface



The thermostat works as an end device in a Light Weight Mesh network, so it is allowed to sleep. To save power, whenever the wireless communication is completed, the AT86RF233 should be put into SLEEP state. This is done by outputting high on AT86RF233 SLP_TR pin in TRX_OFF state. A low level on SLP_TR pin will wake up AT86RF233 from SLEEP state. Refer to AT86RF233 datasheet for more details of the operating mode state of AT86RF233. In the LwMesh stack, two functions; `void NWK_SleepReq(void)` and `void NWK_WakeupReq(void)`, provide sleep and wakeup functions for the AT86RF233 chip. There is no need to write new functions to operate the SLP_TR pins.

2.3.3 Buzzer Pin

A buzzer on the thermostat is used to indicate button operation. The buzzer is driven by 3kHz PWM signal from the SAM4L TC peripheral. As the PWM is generated by TC peripheral hardware, the I/O pin level should be set to low in firmware whenever buzzer is stopped. The buzzer off code is as below.

```
void buzzerOff(void)
{
      ioport_enable_pin(BUZZER_GPIO);     // Enable pin function to guarantee low
      if (tc_get_status(TC1, 0) & TC_SR_CLKSTA)
      {
            tc_stop(TC1,0);               // Stop TC
            buzzerStatus = 0;
      }
}
```

When buzzer is on, the buzzer pin should be controlled by TC peripheral hardware again. The buzzer on code is as below.

```
void buzzerOn(uint8_t on_time)
{
      ioport_disable_pin(BUZZER_GPIO);   // Disable pin function
      tc_start(TC1, 0);                  // Start TC
      buzzerOnTime = on_time;
      buzzerStatus = 1;
}
```

2.4 SAM4L LCD Controller Low Power Features

The LCDCA peripheral on SAM4L is power optimized. The following low power features are used in the thermostat:

- LCD driver active in power save mode for low power operation
- Low power waveform
- ASCII character mapping
- Configurable blink mode and frequency

With the above features, the thermostat can blink the LCD and display LCD contents in power save mode, and the time to update LCD content is minimized by the ASCII character mapping function. For more details about these features, refer to SAM4L datasheet LCDCA chapter.

To enable low power waveform, LCDCA should be configured as below during LCDCA initialization.

```
lcdca_cfg.lp_wave = true;                        // Enable low power waveform
…
lcdca_set_config(&lcdca_cfg);                    // Configure LCDCA
```

To use ASCII code mapping function, simply call the ASF API below. The parameters in this API decide how the ASCII code should be displayed on LCD screen.

```
void lcdca_write_packet(uint8_t lcd_tdg, uint8_t first_seg,
             const uint8_t *data, size_t width, uint8_t dir);
```

To blink the selected segment at a specified frequency, set the blink mode as below during LCDCA initialization.

```
lcdca_cfg.fc0 = 3;                               // Set frame counter
…
lcdca_set_config(&lcdca_cfg);                    // Configure LCDCA
…
lcdca_enable_timer(CONF_USERLCD_BLINK_TIMER);    // Enable frame counter
…
blink_cfg.lcd_blink_mode = LCDCA_BLINK_SELECTED;
blink_cfg.lcd_blink_timer = CONF_USERLCD_BLINK_TIMER;
lcdca_blink_set_config(&blink_cfg);              // Set blink mode
lcdca_set_blink_pixel(USERLCD_ICON_COLON);       // Select blink segment
lcdca_blink_enable();                            // Enable blink mode
```

For the complete code of LCDCA configuration and usage in the thermostat, find the zip file attached with Atmel AT03198: Thermostat with Touch and Wireless Connectivity - Software User's Guide.

2.5 SAM4L Capacitive Touch Module

The CATB peripheral on SAM4L is dedicated for capacitive touch application. The touch buttons on the thermostat are detected by this module. It also provides wheel and slider function together with Atmel QTouch® Library. The CATB performs acquisition, filtering, and detection of capacitive touch sensors. The QTouch Library will handle the sensor data from CATB. Thus the CPU load is reduced.

When CATB operation is completed or the device is put in power save mode, CATB can be stopped to save power. It is simply done by writing a zero to RUN bit in CR register. When touch detection is needed again, it can be started by writing a one to the same bit.

To stop CATB operation, the code is as below.

```
CATB->CATB_CR &= ~CATB_CR_RUN;
```

To start CATB operation, the code is as below.

```
CATB->CATB_CR |= CATB_CR_RUN;
```

Refer to the source file to know where to stop and start the CATB.

3. Power Consumption Calculation

3.1 Power Consumption Breakdown

To have an estimation of the thermostat power consumption, we can add up the CPU power consumption, peripheral power consumption and the power consumption of the on-board components. Note that CPU and most peripherals only work in active mode. To get an average current consumption, the time in active and power save mode should be considered.

Table 3-1. Thermostat CPU and Peripheral Power Consumption in PS1 RUN Mode

Item	Name	Typ Consumption [1] Units [µA/MHz]	Working Frequency [2] Units [MHz]	Power Consumption Units [µA]
CPU	CPU	100.00[3]	12.00	1200.00
Peripheral	SPI	1.10	12.00	13.20
-	TC	3.10	6.00	18.60
-	ADCIFE	1.60	12.00	19.20
-	CATB	1.50	80.00	120.00
-	LCDCA	2.20	12.00	26.40
-	PDCA	0.40	12.00	4.80
-	CHIPID	0.10	1.50	0.15
-	SCIF	3.10	1.50	4.65
-	FREQM	0.20	1.50	0.30
-	GPIO	3.40	1.50	5.10
-	BPM	0.40	1.50	0.60
-	BSCIF	2.30	1.50	3.45
-	AST	0.80	1.50	1.20
-	EIC	0.30	1.50	0.45
Oscillator	RCFAST			180.00
-	RC80M			330.00
Analog	Embedded Regulator			111.00
Sub-total				2039.10

Notes: 1. The data are from SAM4L datasheet. (42023E–SAM–07/2013)
2. Peripheral frequency can be adjusted by PBxSEL register.
3. CPU running a CoreMark algorithm Switching mode in PS1.

Table 3-2. Non-stop Peripheral and On-board Component Power Consumption

Item	Name	Comments	Power Consumption Units [µA]
Peripheral	LCD supply current		8.85
-	OSC32K		0.35
Components	LCD glass [1]		7.00
-	Battery sense circuit		0.97
-	Battery remove circuit		0.75

-	Boost IC output circuit		2.32
-	RF233 [1]	In Sleep mode	0.20
-	Boost IC [1]	Shutdown Current	3.00
-	LCD backlight transistor [1]	Zero Gate Voltage Drain Current and Gate-Body leakage current	2.00
Sub-total			16.24

Notes: 1. The data are from the components datasheet.

First we take a minimum working loop (QTouch button scan loop) as the calculation period. The average power consumption in this minimum working loop should be,

*(MCU RUN mode power consumption * RUN mode time + MCU power-save power consumption * Power-save time) / (RUN mode time + Power-save time) + Non-stop Peripheral and On-board Components power consumption*

Based on test on the thermostat board, the active time is about 1.6ms and sleep time is about 30ms for a minimum working loop. So we can get the average power consumption in a minimum working loop as below,

$$(2309.10 * 1.6 + 4.7 * 30) / (1.6 + 30) + 16.24 = 133.15\mu A$$

where 4.7(µA) is the typical power consumption in WAIT mode when AST and OSC32K are running.

In most of cases, a thermostat is placed somewhere in the house and user will not operate it frequently. So the battery life of a thermostat is determined mainly by the power consumption in this no-operation case. For this reference design, the thermostat will enable sensors, and then read sensors, update LCD and send out the sensor data wirelessly in 1 minute interval. Meanwhile it wakes up every 1 second to update the time. The active time to update the time is about 790µs according to the test. So the average power consumption is found by adding up the current below,

Average power consumption in minimum loop + sensor + read sensor/update LCD/send sensor data + time update

The result is,

$$133.15 + 770 * 5/60 + 13800 * 0.015/60 + 2039.10 * 0.00079 = 202.38\mu A$$

where 770(µA) is the sensor power consumption (note that the boost IC will work together with the humidity sensor), 0.015(ms) is the estimated RF transmission time and 13800(µA) is the AT86RF233 Tx power consumption.

3.2 Power Consumption Test Result

Here we list the measured power consumption result and compare it with the calculated result in above section.

Table 3-3. Power Consumption of the Thermostat

Item	Average Power Consumption / Units [µA]	
	Measured Result	Calculated Result
Minimum working loop (QTouch button scan loop)	158	133.15
No-operation case1	240	202.38

Notes: 1. The thermostat is working without user operation, i.e. LCD menu and backlight are not changed. Most of the time, a real thermostat will work in this state.

Compared with the calculation, the measured values are somewhat larger (~40µA). It is mainly because the data used in the calculation are typical values and some data are from different test cases. However, the calculation gives a good estimate for the power consumption in the thermostat reference design.

4. Revision History

Doc. Rev.	Date	Comments
42263A	03/2014	Initial document release

Atmel | Enabling Unlimited Possibilities®

Atmel Corporation
1600 Technology Drive
San Jose, CA 95110
USA
Tel: (+1)(408) 441-0311
Fax: (+1)(408) 487-2600
www.atmel.com

Atmel Asia Limited
Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
Tel: (+852) 2245-6100
Fax: (+852) 2722-1369

Atmel Munich GmbH
Business Campus
Parkring 4
D-85748 Garching b. Munich
GERMANY
Tel: (+49) 89-31970-0
Fax: (+49) 89-3194621

Atmel Japan G.K.
16F Shin-Osaki Kangyo Building
1-6-4 Osaki, Shinagawa-ku
Tokyo 141-0032
JAPAN
Tel: (+81)(3) 6417-0300
Fax: (+81)(3) 6417-0370