

Designing for Low Power with the MCF5213

by: Michael Norman
TSPG 32-bit Standard Product Applications

The MCF5213 represents a family of highly-integrated, 32-bit microcontrollers based on the V2 ColdFire® architecture. Featuring integrated Flash and SRAM memories and a rich set of integrated peripherals, the MCF5213 was created as a single-chip solution to the lack of a cost-sensitive, low-power, 32-bit microcontroller that fills the gap between existing 16-bit and 32-bit products.

The purpose of this application note is to highlight the features of the MCF5213 that should be considered when designing for power conscious applications. The information provided here is also pertinent to the MCF5211 and MCF5212.

Table of Contents

1	Introduction	2
2	Power Supplies	2
3	Clocking Options	3
	3.1 Reference Clock	3
	3.2 Phase-Locked Loop (PLL)	4
	3.3 Low-Power Divider (LPD)	5
	3.4 CLKOUT	6
4	Low-Power Modes	6
	4.1 Run	7
	4.2 Wait and Doze	7
	4.3 Stop	8
	4.4 Exiting Low-Power Mode	8
5	Peripheral Power Management	9
6	Memory Configuration	9
7	Other Considerations	10
8	Power Consumption Examples	11
	8.1 Test Setup	11
	8.2 Low-Power Mode Examples	12
	8.3 Loop-Execution Examples	13
	9 Estimating Power Consumption	14

1 Introduction

Understanding the power consumption of an embedded processor is an important piece of a new system design. Several factors come into play when estimating the power consumption of a processor: operating frequency, operating voltage, software requirements, I/O activity, and architectural features. The ColdFire architecture offers several features, such as an efficient instruction set and fast interrupt processing that help make it suitable for low-power applications. In addition, the MCF5213 family features peripheral clock gating, dynamically scalable system frequency, and low-power operating modes. This application note details these low-power features and provides some guidelines for estimating the power consumption of an MCF5213 application.

2 Power Supplies

The total power consumption of the MCF5213 microcontroller is comprised of the power required by the digital logic, I/O pads, PLL, and the analog to digital converter (ADC). The digital power is the power consumed by all the internal digital logic blocks including the CPU, on-chip memories, and integrated peripherals. This component is usually 80% or more of the total chip power. The I/O power is the power consumed by the I/O pads used to drive the microcontroller interface signals to external devices. This component of total power depends heavily on the system design and activity levels. The PLL power is the power consumed by the PLL circuitry and is usually small compared to the total device power.

The total power can be separated into two conceptual power components. Static power refers to the power dissipation that occurs even when the clocks in the system are not active. This power can be attributed to the leakage current that flows from every powered transistor. The static consumption depends on the core voltage and die-junction temperature. Static power increases as temperature increases and may vary from device to device due to fabrication process variations. When an application is active, the static power is a small percentage of the total power. The variance of this component has a larger effect on the total power when the microcontroller is placed into a low-power mode.

Dynamic power is the component of power consumption resulting from signal transitions. The dynamic power in a system is determined by the clock frequencies, the core voltage, and the software running on the device. Temperature and process variation have a minimal effect on the dynamic component of power.

The total power consumed by the MCF5213 is delivered via three power supply inputs, each with the same operating voltage. An on-chip low-power, fixed-voltage regulator provides internal power to the CPU, SRAM, Flash, I/O, and peripherals based on these external supplies.

Table 1. Voltage Supplies

Name	Abbreviation	Function	Operating Voltage Range (V)
PLL Analog Supply	V_{DDPLL} , V_{SSPLL}	Isolate the PLL analog circuitry from the normal levels of noise present on the digital power supply.	3.0 – 3.6
ADC Analog Supply	V_{DDA} , V_{SSA}	Isolate the ADC analog circuitry from the normal levels of noise present on the digital power supply.	3.0 – 3.6
Digital Supply	V_{DD} , V_{SS}	Digital power supply to core, peripherals, and I/O pads	3.0 – 3.6

The on-chip voltage regulator features an integrated power-on reset (POR) circuit that monitors the input supply and forces an MCU reset as the supply voltage rises. A low voltage detect (LVD) circuit monitors the supply voltage and can be configured to force a reset or interrupt condition if it falls below the LVD trip point. Please refer to the “Reset Controller Module” section of the *MCF5213 Reference Manual* (MCF5213RM) for details on the control of the LVD and its associated reset and interrupt.

3 Clocking Options

The dynamic power requirements of an application depend heavily on the operating frequency of the microcontroller and its integrated peripherals. The MCF5213 supports several features that allow control over the operating frequency. These features include a variety of reference clock inputs, a phase-locked loop (PLL), and a low-power clock divider. The clocking circuitry is diagrammed in [Figure 1](#), and these options are described in the following sections.

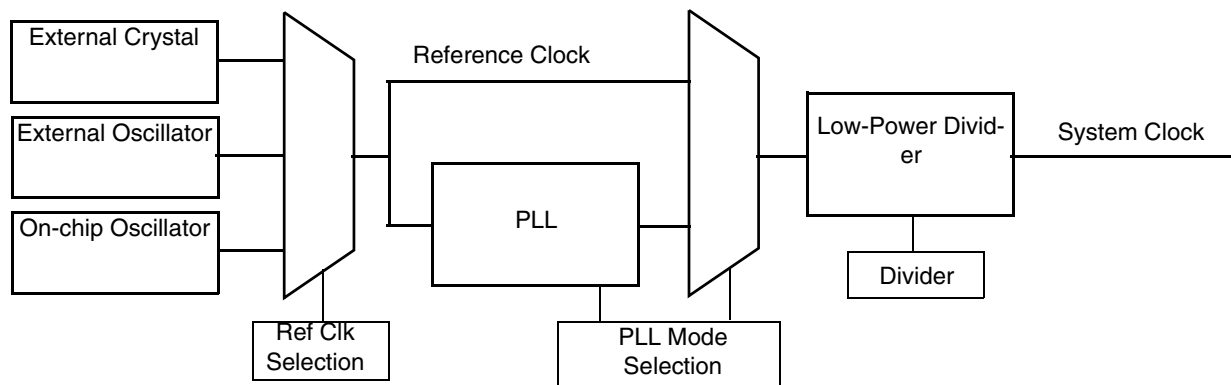


Figure 1. Clocking Options Block Diagram

3.1 Reference Clock

There are three different reference clocks supported by the MCF5213: external oscillator, external crystal, and on-chip oscillator. The CLKMOD0 and XTAL digital inputs are used to determine which reference clock is to be applied, as shown in [Table 2](#).

Table 2. Reference Clock Mode Configurations

CLKMOD0	XTAL	Clocking Mode
0	0 ¹	clock driven by external oscillator
0	1	clock driven by on-chip oscillator
1	n/a ²	clock driven by external crystal

NOTES:

¹ In this mode, the external clock is input on EXTAL

² In this mode, the XTAL and EXTAL pins are connected to the external crystal circuit.

The frequency range of the different reference clock modes is shown in [Table 3](#).

Table 3. Reference Clock Frequency Ranges

Reference Clock	Frequency Range (MHz)
External Oscillator	0.0 — 80.0
External Crystal	1.0 — 16.0
On-Chip Oscillator	8.0

3.2 Phase-Locked Loop (PLL)

The MCF5213 features a fully programmable PLL that, when enabled, synthesizes the system clocks. The PLL can multiply a 2 to 10-MHz reference clock by 4x to 18x, provided that the system clock frequency remains within specification. The multiplication factor is determined by the programmable MFD setting. A programmable reduced frequency divider (RFD) can reduce the system frequency by dividing the output of the PLL. With the PLL enabled, the system clock frequency generated by the PLL is determined by the following formula:

$$F_{\text{sys}} = F_{\text{ref}} \times 2(\text{MFD} + 2) / 2^{\text{RFD}}$$

The state of the CLKMOD1 digital input signal determines the default operation of the PLL following reset. If enabled at system reset, the PLL has a default multiplication factor of 6x. Following reset, the PLL can be enabled and disabled via software, and the MFD and RFD settings are fully programmable.

Table 4. PLL Reset Configuration

CLKMOD1	PLL Operation
0	Disabled
1	Enabled

When enabled, the power consumption of the PLL is dependent entirely on the MFD setting. [Table 5](#) and [Figure](#) show the relationship between the MFD settings and the power consumption on the V_{DDPLL} input supply.

Table 5. PLL Power Consumption

MFD	Power (mW)
0	4.1
1	5.0
2	6.1
3	7.2
4	8.4
5	9.5
6	10.9
7	12.0

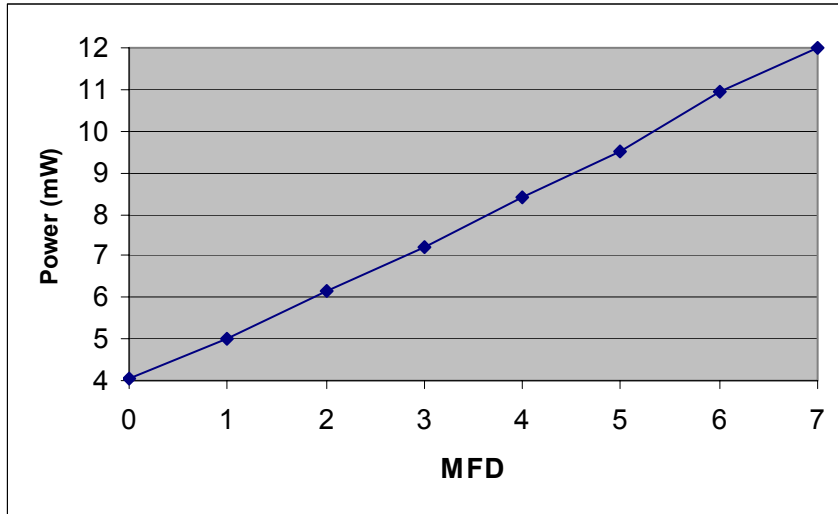


Figure 2. PLL Power Consumption

Note that when the PLL is disabled, the static power consumed on the V_{DDPLL} supply is about 0.4 mW at room temperature.

3.3 Low-Power Divider (LPD)

The system clocks of the MCF5213 are fed by the output of a divider circuit known as the low-power divider. If enabled, the LPD clock input comes from the PLL; otherwise, it comes directly from the reference clock. The division factor is a power-of-two number ranging from the default divider of 1 (2^0) to a maximum of 32,768 (2^{15}). See the “Clock Module” section of the *MCF5213 Reference Manual* (MCF5213RM) for details on programming the LPD.

Many systems only require the processor to operate at high performance levels periodically. Using the LPD, an application can conserve power by scaling the operating frequency to the minimum required by the current task. Table 6 shows some example system frequency ranges that can be accomplished using the PLL and LPD.

Table 6. LPD Clocking Examples

Ref Clk (MHz)	Max Sys Clk (MHz)	Min Sys Clk (Hz) ¹
1	1 ¹	30.5
2	36 ²	61
4	72 ²	122
8	80 ²	244
10	80 ²	305
16	16 ¹	488

NOTES:

- ¹ PLL disabled
- ² PLL enabled

3.4 CLKOUT

The MCF5213 replicates the system clock and drives it out as the PSTCLK/CLKOUT signal. This signal can be used to clock an external device or as the PSTCLK input to a BDM interface cable. To help save power, this signal should be disabled in systems where it is not required. There are two ways in which the PSTCLK/CLKOUT signal can be disabled:

1. Set the DISCLK bit in the SYNCR of the clocks module.
2. Boot the processor into JTAG mode via the JTAG_EN reset configuration signal.

Figure shows the estimated power consumption on the VDD supply attributed to the CLKOUT signal.

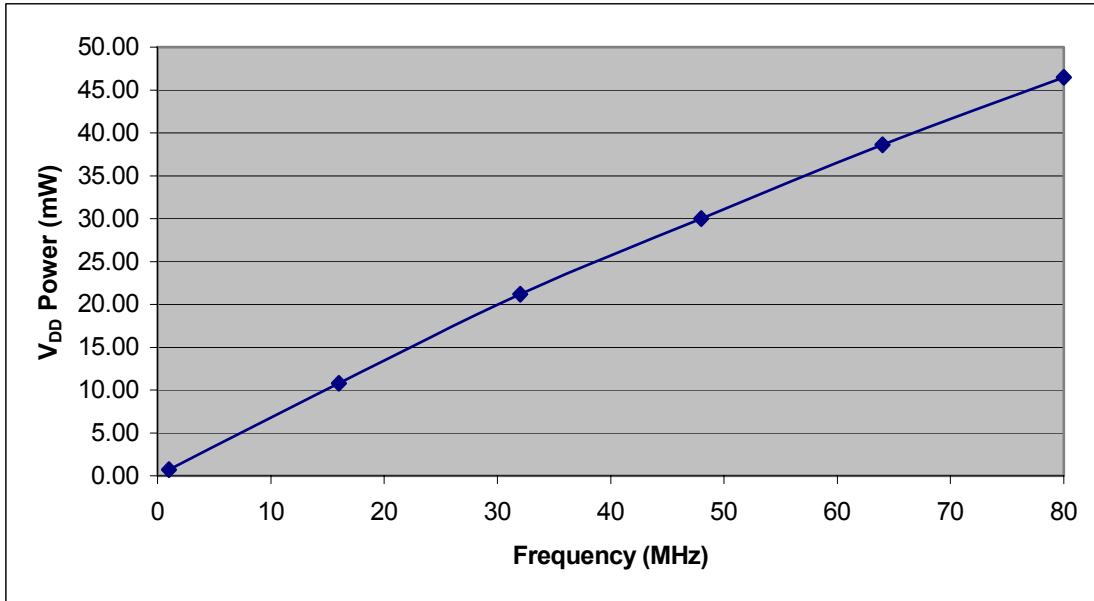


Figure 3. CLKOUT Power Consumption

4 Low-Power Modes

The MCF5213 features four low-power modes that are discussed here and detailed further in the “Power Management” section of the *MCF5213 Reference Manual* (MCF5213RM).

Like all ColdFire processors, the MCF5213 includes the stop instruction in its instruction set. This supervisor (privileged) instruction is used to stop the ColdFire core from fetching and executing instructions. Execution of the stop instruction by the MCF5213 can also put the system into a configurable low-power mode.

The four low-power modes are run, wait, doze, and stop. The state of the LPCR[LPMD] field determines which low-power mode is entered when the ENBSTOP bit in the LPICR is set and a stop instruction is executed. The LPICR, LPCR, and the various low-power mode settings are shown below.

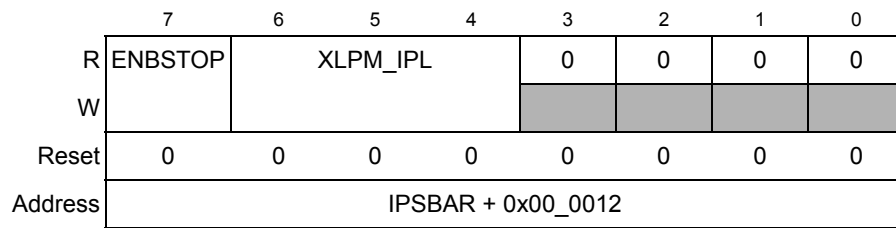


Figure 4. Low-Power Interrupt Control Register (LPICR)

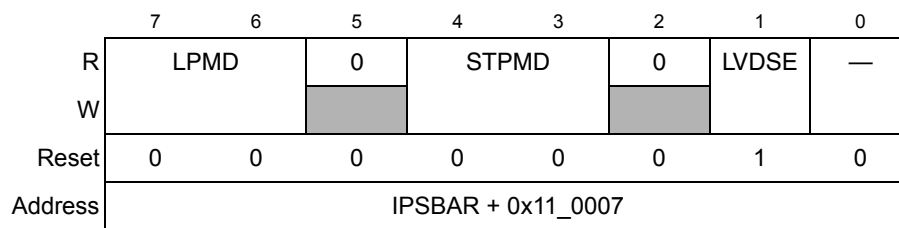


Figure 5. Low-Power Control Register (LPCR)

Table 7. Low Power Modes

LPCR[LPMD]	Mode
11	STOP
10	WAIT
01	DOZE
00	RUN

4.1 Run

In the run low-power mode, the CPU ceases to fetch and execute instructions; however, no clocks are disabled and power consumption is directly related to the system clock frequency. This mode provides a minimal amount of power savings.

4.2 Wait and Doze

In wait mode, the CPU is idle and the CPU and SRAM clocks are disabled until a wake-up event occurs. In this mode, all peripherals are still operational and may generate interrupts that will exit the low-power mode.

In doze mode, the CPU and SRAM clocks are disabled just as in wait mode; however, in this mode the programmable interrupt timers (PITs) may be programmed to shutdown (refer to the “Programmable Interrupt Timer Modules” section of the *MCF5213 Reference Manual* for more detail).

4.3 Stop

In stop mode, all system clocks are disabled. Therefore, no peripherals are operational and the only interrupt events that can wake the system out of low-power mode are interrupts from external devices (via the edge port).

The peripherals that are in operation should be gracefully terminated prior to entering stop mode. Upon exiting stop mode, most peripherals retain their pre-stop mode status and resume normal operation.

There are several options for enabling and disabling CLKOUT, the PLL, the on-chip oscillator (OCO) or crystal oscillator (OSC), and the low-voltage detect (LVD) logic in stop mode. The various options are shown in [Table 8](#).

Table 8. Stop Mode Operation

LPCR[STPMD]	Operation During Stop Mode				
	System Clocks	CLKOUT	PLL	OSC/OCO	LVD
00	Disabled	Enabled	Enabled	Enabled	Enabled
01	Disabled	Disabled	Enabled	Enabled	Enabled
10	Disabled	Disabled	Disabled	Enabled	Enabled
11	Disabled	Disabled	Disabled	Disabled	LPCR[LVDSE]

NOTE

Disabling the PLL and OSC/OCO during stop mode saves power at the expense of a longer wake-up period required for the OSC/OCO to restart and the PLL to lock.

4.4 Exiting Low-Power Mode

The system will remain in the low-power mode until a wake-up event occurs. Any reset and enabled interrupt request constitute a wake-up event. In order for an interrupt to generate a wake-up event, all the following conditions must be true:

- The priority level of the interrupt request (IRQ) is higher than the value programmed in the LPICR[XLPM_IPL]. The interrupt level setting of each interrupt request is configured in the interrupt controller module. See the *MCF5213 Reference Manual* for details.
- The priority level of the IRQ is higher than the interrupt level mask setting in the ColdFire core's status register (SR[I]). The desired SR value is programmed as part of the execution of the stop instruction. See the *ColdFire Family Programmer's Reference Manual* for details on the stop instruction.
- The IRQ must not be masked in the interrupt controller.
- The IRQ must be enabled at the module where the interrupt originates.

When in stop mode, all system clocks are disabled and the peripherals are not capable of generating interrupts; only resets and interrupts from an external device (via the edge port) are capable of generating

a wake-up event. Since the clock to the edge port will be disabled, the interrupt pins used for wake-up must be programmed to level sensitive instead of edge triggered.

5 Peripheral Power Management

The MCF5213 provides a means to entirely disable each peripheral individually. This allows an application to conserve power by turning off the clock to peripherals that are not used in the system or to dynamically turn peripherals on and off as needed.

The peripheral power management registers in the clock module control whether the clock to each peripheral is enabled or disabled.

Table 9 shows the approximate power consumption for each of the modules as a function of the system frequency.

Table 9. Approximate Peripheral Power Consumption

Peripheral	Power (mW/MHz)
DMA	0.12
UARTs	0.06
I ² C	0.05
QSPI	0.07
DTIMs	0.04
Ports	0.04
PITs	0.01
ADC	0.05
GPT	0.02
PWM	0.02
CAN	0.06

6 Memory Configuration

The MCF5213 is a single-chip solution with no external bus; the only memories to configure are the internal Flash and SRAM memory banks. However, there are some configuration options that should be considered when creating software for a power sensitive application.

The Flash and SRAM memories are configured in the FLASHBAR (a.k.a. RAMBAR0) and SRAMBAR (a.k.a. RAMBAR1) registers respectively. The RAMBARs provide programmable access control attributes for code and data accesses from user and supervisor modes. Power dissipation can be reduced by masking the appropriate accesses. For example, if the SRAM is used only for data operands, setting the mask bits for instruction fetches can decrease power consumption. Table 10 shows some typical RAMBAR settings.

Table 10. Typical RAMBAR Settings

Flash / SRAM Contents	RAMBARx[7:0]
Instructions Only	0x2B
Data Only	0x35
Instructions and Data	0x21

7 Other Considerations

The background debug module (BDM) on the MCF5213 is not controlled by the peripheral power management module. However, the largest power consumers in the BDM, the external PST/DDATA signals, do have a software programmable disable feature. The PST[3:0] and DDATA[3:0] signals provide processor status and debug information to an external debugging tool. These signals can be disabled to conserve power once a software application has been debugged and there is no more need for a performance monitoring or run-control tool.

The PST and DDATA signal can only be disabled by setting the CSR[PCD] bit. Software can access this BDM register with the WDEBUG instruction. The code snippet below provides an example. Please refer to the *ColdFire Family Programmer's Reference Manual* for more information.

NOTE

The MCF5213 also provides an ALLPST signal. This signal is the logical AND of PST[3:0] and is used by BDM interface cables to detect when the processor is halted. The ALLPST signal is also controlled by the CSR[PCD] setting.

```

/*****
* Assembly code that executes the debug instruction pointed to by the
* value passed in on the stack
*/
mcf5xxx_exe_wdebug:
_mcf5xxx_exe_wdebug:
    move.l    4(sp),a0
    wdebug.l (a0)
    rts

/*****
* C code that uses the WDEBUG functionality to enable/disable the PST

```

```

* and DDATA BDM signals
*/
uint16 wdebug_pstddata_off[] = {0x2C80, 0x0002, 0x0000, 0x0000};
uint16 wdebug_pstddata_on[]  = {0x2C80, 0x0000, 0x0000, 0x0000};

void
cpu_pstddata_disable(void)
{
    mcf5xxx_exe_wdebug(wdebug_pstddata_off);
}

void
cpu_pstddata_enable(void)
{
    mcf5xxx_exe_wdebug(wdebug_pstddata_on);
}

/*****/

```

8 Power Consumption Examples

The following sections provide example power consumption data for the MCF5213. These examples can be used to estimate the power consumption of custom applications.

NOTE

All data was collected on a small sampling of MCF5213 devices at room temperature with an operating voltage of 3.30V. Fluctuations in manufacturing process, voltage, and temperature are not reflected in these examples.

8.1 Test Setup

The examples in the following sections share a common setup:

- The CLKOUT and PST/DDATA signals are disabled.
- Examples executed from SRAM invalidate the FLASHBAR and disable the peripheral clock to the Flash.

Power Consumption Examples

- Examples executed from Flash enable the FLASHBAR for instruction and data access and enable the SRAMBAR for data access only.
- All peripheral clocks are disabled except for the edge port and interrupt controller (and possibly Flash as specified above).

8.2 Low-Power Mode Examples

Table 11 shows some example power consumption measurements taken on the M5213EVB. Again, note that these measurements were collected on a small sample of MCF5213 devices at room temperature and do not reflect variations due to common factors such as supply voltage and temperature variations.

Table 11. Power Consumption in Low-Power Modes¹

		System Frequency (MHz)					
		1	16	32	48	64	80
Flash	RUN	13.0	75.7	95.0	112.1	129.8	147.0
	WAIT/DOZE	12.4	67.3	78.8	88.0	98.0	107.7
	STOP 0 & 1	8.4	9.9	11.7	13.4	15.5	17.4
	STOP 2	7.1					
	STOP 3 (LVDSE = 1)	6.1					
	STOP 3 (LVDSE = 0)	0.23					
SRAM	RUN	9.1	25.2	42.4	59.1	76.3	93.2
	WAIT/DOZE	8.6	17.3	26.9	36.2	45.8	55.4
	STOP 0 & 1	8.1	9.6	11.5	13.1	15.1	17.2
	STOP 2	6.8					
	STOP 3 (LVDSE = 1) ²	5.8					
	STOP 3 (LVDSE = 0)	0.13					

NOTES:

¹ Data shown is in mW on the V_{DD} supply voltage

² LVDSE is the enable bit for the low-voltage detector. Refer to section 4.3.

The CLKOUT signal is disabled by our common test setup. Stop modes 0 and 1 are identical, because the status of CLKOUT is the only difference between the two. Similarly, the wait and doze modes are identical, because the peripheral clock to the PITs is disabled.

Stop modes 2 and 3 eliminate all sources of dynamic power and thus are not dependent on the system clock frequency. A graph of power consumption versus frequency for the remaining modes is shown in Figure .

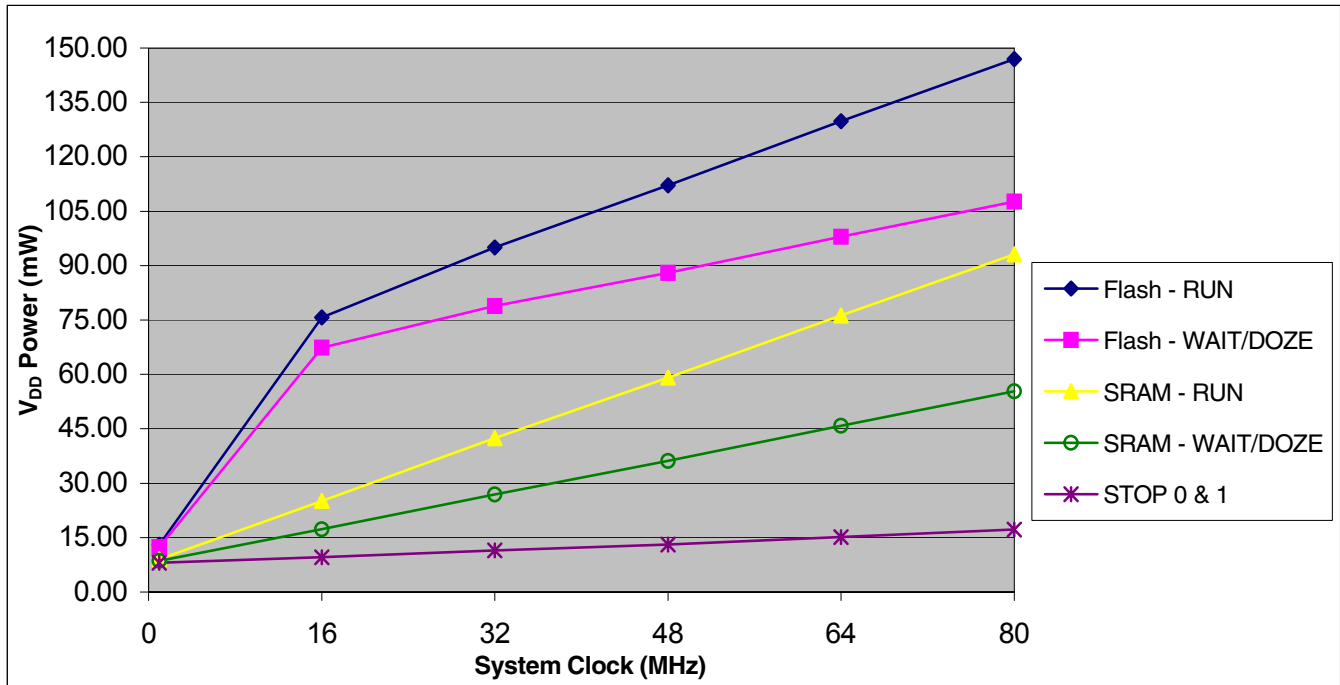


Figure 6. Power Consumption of Low-Power Modes

8.3 Loop-Execution Examples

Figure shows the power consumed while the ColdFire core executes some simple loops. These two loops, NOP and while(1), serve as good bounds for the minimum and maximum power consumed when the CPU is active.

Execution of the NOP instruction performs no operation. However, the NOP instruction does not begin execution until all pending bus cycles have completed, thus synchronizing the pipeline and preventing instruction overlap. Therefore, execution of this single instruction requires multiple clock cycles and is a very inefficient use of the CPU. For this reason, the NOP loop is a good approximation of a lower bound for the power consumed by the CPU when it is not in a low-power mode. The assembly code for the NOP loop used in this example is:

```

loop:
    nop
    nop
    nop
    nop
    bra loop
    
```

Estimating Power Consumption

The while(1) loop on the other hand consists of a single branch instruction that loops back to itself:

```
loop:
    bra loop
```

Each loop iteration requires only two clock cycles and requires no operand fetches. This tight loop is very efficient and can be used as a reasonable approximation of an upper bound for the CPU power consumption.

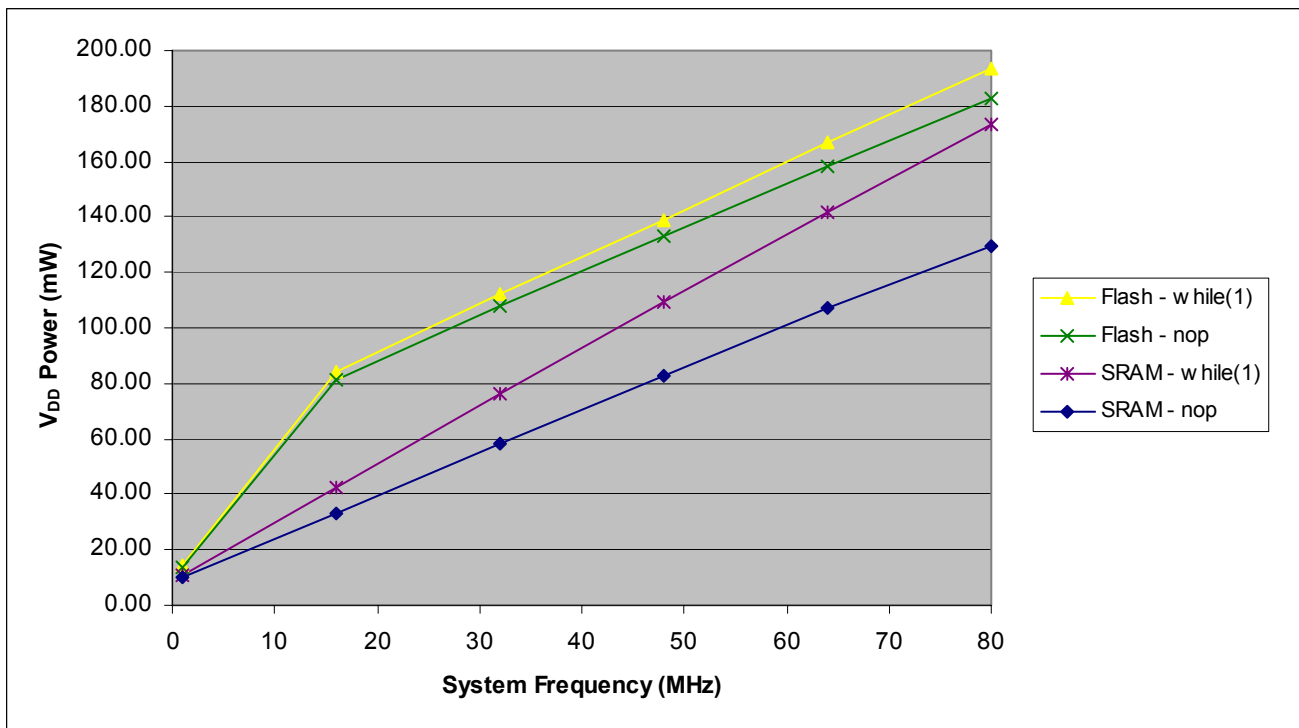


Figure 7. Loop Execution Power Consumption

9 Estimating Power Consumption

The examples provided in the previous section are very simple. However, they can be used to generate power consumption estimate for a custom MCF5213 application.

The loop execution examples above can be used to bound the power consumed by the ColdFire core, the interrupt controller, the external interrupts, and the Flash. Add to that the consumption of each enabled peripheral at the desired operating frequency (refer to [Table 9](#)), and the result is the power consumed from the V_{DD} supply in an active configuration.

The application should make use of the low-power modes for any periods of inactivity. Using the same peripheral consumption data (if applicable) along with the low-power mode estimations will provide the power consumed from the V_{DD} supply in an inactive configuration.

The average V_{DD} power consumption could then be estimated using the percentage of time spent in each configuration and the estimated consumption of those configurations.

Similarly, the average PLL supply consumption can be estimated for all configurations.

Finally, the total power consumed by the custom application is simply the sum of the power consumed from each power supply.

How to Reach Us:

Home Page:

www.freescale.com

E-mail:

support@freescale.com

USA/Europe or Locations Not Listed:

Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

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)
support@freescale.com

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 Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:

Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© Freescale Semiconductor, Inc. 2006. All rights reserved.