

# MSP430x1xx Microcontrollers

## Low Power Modes

CPE 621 Advanced Microcomputer Techniques

Dr. Emil Jovanov

MSP430 low power concepts

1

## Low Power

- Power: A First-Class Architectural Design Constraint
  - Trevor Mudge, *IEEE Computer*, April 2001.
    - Internet, server farm
      - 25,000 square foot server farm, 8,000 servers
      - Power consumption → 2 MW
      - 25% of the managing cost
- Information technology consumes 8% of power in U.S.
  - Exponential growth, without control very soon more power than all other uses combined! (*Financial Times*, Fall 2000)
  - Alpha 21364 30W/cm<sup>2</sup> → 3 x hot plate

MSP430 low power concepts

2

# CPU Survey

<http://bwrc.eecs.berkeley.edu/CIC/summary/local/summary.pdf>

Processor	Source	Date (y-yr)	Bits (rd)	Clock (MHz)	SPEC-92		Units / hour	Pipe Stages (rd/clk)	Cache (rd)		Vdd (V)	Tch (µm)	MCM	Power (W)		Size (mm <sup>2</sup> )	Transistors (10 <sup>6</sup> )
					int	fp			KB	Assoc				peak	typ		
i386SX	[v,41]	88		16	2.4	8.9					1.5						
		89		20	3.3	7.3	1/1	4/m/m	-	-	5.0						0.275
	[v]	89	v/32	25	4.6	2.9					1.0						
		92		33	6.2	3.3								~2		45	
i486DX2	[v]	92	v/32	50	29.9	16.2	2/1	5/m/5?	8 u.	4	5.0	0.8	3	7.0	4.0	81	1.2
	[v,210]			66	59.6	12.8											
P5	[v]	93	v/32	66	70.4	33.1	3/2	5/m/6	8/8	2/2	5.0	0.8	3			298	3.1
	[2,10]	93		66	78	63.6								16.0			

  

Processor	Source	Date (y-yr)	Bits (rd)	Clock (MHz)	SPEC-92		SPEC-95		Units / hour	Pipe Stages (rd/clk)	Cache (rd)		Vdd (V)	Tch (µm)	MCM	Power (W)		Size (mm <sup>2</sup> )	Transistors (10 <sup>6</sup> )
					int	fp	int	fp			KB	Assoc				peak	typ		
Alpha	[v]	98		535			333	475								~60		30	35.2
	[v,697]	98	64	667			44	65	6/4	5/5/10	64/64	2/2	2			~72			
	[v]	98		700												~25			
	[v]	98	64						6/8	5/5/10						~23			
	[reg,by]	98		800												~30			
	[v]	99	64						6/8	5/5/10						~25			
21264	[v]	98													~18				
21364	[reg,by]	00		1000			-36	-128	6/5/4	6/6/7	64/64	2/2	4	1.5	~18	~6	~300	~300	100
21464	[reg]	03									64/64	2/2	4	1.5	~18				~250

MSP430 low power concepts

3

## Power Equations for CMOS Logic

Dynamic power consumption  
Power due to short-circuit current during transition  
Power due to leakage current

$$P = ACV^2f + \tau AVI_{\text{short}}f + VI_{\text{leak}}$$

A (activity of gates)  
Reduce the supply voltage, V  
Reduce threshold  $V_t$

$$f_{\text{max}} \propto \frac{(V - V_t)^2}{V}$$

$$I_{\text{leak}} \propto \exp\left(-\frac{qV_t}{kT}\right)$$

MSP430 low power concepts

4

## How can we reduce power consumption?

- Dynamic power consumption
  - charge/discharge of the capacitive load on each gate's output
  - frequency
- Control activity
  - reduce power supply voltage
  - reduce working frequency
  - turn off unused parts (module enables)
  - use low power modes
  - interrupt driven system
- Minimize the number of transitions
  - instruction formats, coding?

### Low-Power Concept: Basic Conditions for Burst Mode

The example of the heat cost allocator shows that the current of the non-activity periode dominates the current consumption.

	Measure	Process data	Real-Time Clock	LCD Display
$I_{AVG} =$	$I_{Measure}$	$+ I_{Calculate}$	$+ I_{RTC}$	$+ I_{Display}$
	$= I_{ADC} * t_{Measure} / T$	$+ I_{active} * t_{calc} / T$	$+ I_{active} * t_{RTC} / T$	$+ I_{Display}$
	$= 3mA * 200\mu s / 60s$	$+ 0.5mA * 10ms / 60s$	$+ 0.5mA * 0.5ms / 60s$	$+ 2.1\mu A$
	$= 10nA$	$+ 83nA$	$+ 4nA$	$+ 2.1\mu A$
$I_{AVG} \equiv$				<b>2.1<math>\mu A</math></b>

The **sleep current dominates** the current consumption!

The currents are related to the sensor and  $\mu C$  system. Additional current consumption of other system parts should be added for the total system current

## Low power - features

- Peak power
  - Possible damage
- Dynamic power
  - Non-ideal battery characteristics
  - Ground bounce, di/dt noise
- Energy/operation ratio
  - MIPS/W
  - Energy x Delay

MSP430 low power concepts

7

## Reducing power consumption

- Logic
  - Clock tree (up to 30% of power)
  - Clock gating (turn off branches that are not used)
  - Half frequency clock (both edges)
  - Half swing clock (half of Vcc)
  - Asynchronous logic
    - completion signals
    - testing
- Architecture
  - Parallelism (increased area and wiring)
  - Speculation (branch prediction)
  - Memory systems
    - Memory access (dynamic)
    - Leakage
    - Memory banks (turn off unused)
  - Buses
    - 32-64 address/data, (15-20% of power)
    - Gray Code, Code compression

MSP430 low power concepts

8

## Reducing power consumption #2

- Operating System
  - Finish computation “when necessary”
  - Scale the voltage
    - Application driven
    - Automatic
- System Architecture
  - Power efficient and specialized processing cores
  - A “convergent” architecture
  - Trade-off
    - AMD K6 / 400MHz / 64KB cache – 12W
    - XScale with the same cache 450 mW @ 600 MHz (40mW@150MHz)
    - 24 processors? Parallelism?
- Other issues
  - Leakage current – Thermal runaway
  - Voltage clustering (low  $V_{\text{threshold}}$  for high speed paths)

MSP430 low power concepts

9

## Operating Modes-General

The MSP430 family was developed for ultralow-power applications and uses different levels of operating modes. The MSP430 operating modes, give advanced support to various requirements for ultralow power and ultralow energy consumption. This support is combined with an intelligent management of operations during the different module and CPU states. An interrupt event wakes the system from each of the various operating modes and the RETI instruction returns operation to the mode that was selected before the interrupt event.

The ultra-low power system design which uses complementary metal-oxide semiconductor (CMOS) technology, takes into account three different needs:

- The desire for speed and data throughput despite conflicting needs for ultra-low power
- Minimization of individual current consumption
- Limitation of the activity state to the minimum required by the use of low power modes

MSP430 low power concepts

10

## Low power mode control

There are four bits that control the CPU and the main parts of the operation of the system clock generator:

- CPUOff,
- OscOff,
- SCG0, and
- SCG1.

These four bits support discontinuous active mode (AM) requests, to limit the time period of the full operating mode, and are located in the status register. The major advantage of including the operating mode bits in the status register is that the present state of the operating condition is saved onto the stack during an interrupt service request. As long as the stored status register information is not altered, the processor continues (after RETI) with the same operating mode as before the interrupt event.

## Operating Modes-General

Another program flow may be selected by manipulating the data stored on the stack or the stack pointer. Being able to access the stack and stack pointer with the instruction set allows the program structures to be individually optimized, as illustrated in the following program flow:

### Enter interrupt routine

The interrupt routine is entered and processed if an enabled interrupt awakens the MSP430:

- The SR and PC are stored on the stack, with the content present at the interrupt event.
- Subsequently, the operation mode control bits OscOff, SCG1, and CPUOff are cleared automatically in the status register.

### Return from interrupt

Two different modes are available to return from the interrupt service routine and continue the flow of operation:

- Return with low-power mode bits set. When returning from the interrupt, the program counter points to the next instruction. The instruction pointed to is not executed, since the restored low power mode stops CPU activity.
- Return with low-power mode bits reset. When returning from the interrupt, the program continues at the address following the instruction that set the OscOff or CPUOff-bit in the status register. To use this mode, the interrupt service routine must reset the OscOff, CPUOff, SCG0, and SCG1 bits on the stack. Then, when the SR contents are popped from the stack upon RETI, the operating mode will be active mode (AM).

## Operating Modes - Software configurable

---

There are six operating modes that the software can configure:

- ❑ Active mode AM; SCG1=0, SCG0=0, OscOff=0, CPUOff=0: CPU clocks are active
- ❑ Low power mode 0 (LPM0); SCG1=0, SCG0=0, OscOff=0, CPUOff=1:
  - CPU is disabled
  - MCLK is disabled
  - SMCLK and ACLK remain active
- ❑ Low power mode 1 (LPM1); SCG1=0, SCG0=1, OscOff=0, CPUOff=1:
  - CPU is disabled
  - MCLK is disabled
  - DCO's dc generator is disabled if the DCO is not used for MCLK or SMCLK when in active mode. Otherwise, it remains enabled.
  - SMCLK and ACLK remain active
- ❑ Low power mode 2 (LPM2); SCG1=1, SCG0=0, OscOff=0, CPUOff=1:
  - CPU is disabled
  - MCLK is disabled
  - SMCLK is disabled
  - DCO oscillator automatically disabled because it is not needed for MCLK or SMCLK
  - DCO's dc-generator remains enabled
  - ACLK remains active

## Operating Modes #2

---

- ❑ Low power mode 3 (LPM3); SCG1=1, SCG0=1, OscOff=0, CPUOff=1:
  - CPU is disabled
  - MCLK is disabled
  - SMCLK is disabled
  - DCO oscillator is disabled
  - DCO's dc-generator is disabled
  - ACLK remains active
- ❑ Low power mode 4 (LPM4); SCG1=X, SCG0=X, OscOff=1, CPUOff=1:
  - CPU is disabled
  - ACLK is disabled
  - MCLK is disabled
  - SMCLK is disabled
  - DCO oscillator is disabled
  - DCO's dc-generator is disabled
  - Crystal oscillator is stopped

## Operating Modes-Low Power Mode in details

---

### □ Low-Power Mode 0 and 1 (LPM0 and LPM1)

Low power mode 0 or 1 is selected if bit CPUOff in the status register is set. Immediately after the bit is set the CPU stops operation, and the normal operation of the system core stops. The operation of the CPU halts and all internal bus activities stop until an interrupt request or reset occurs. The system clock generator continues operation, and the clock signals MCLK, SMCLK, and ACLK stay active depending on the state of the other three status register bits, SCG0, SCG1, and OscOff.

The peripherals are enabled or disabled with their individual control register settings, and with the module enable registers in the SFRs. All I/O port pins and RAM/registers are unchanged. Wake up is possible through all enabled interrupts.

### □ Low-Power Modes 2 and 3 (LPM2 and LPM3)

Low-power mode 2 or 3 is selected if bits CPUOff and SCG1 in the status register are set. Immediately after the bits are set, CPU, MCLK, and SMCLK operations halt and all internal bus activities stop until an interrupt request or reset occurs.

Peripherals that operate with the MCLK or SMCLK signal are inactive because the clock signals are inactive. Peripherals that operate with the ACLK signal are active or inactive according with the individual control registers and the module enable bits in the SFRs. All I/O port pins and the RAM/registers are unchanged. Wake up is possible by enabled interrupts coming from active peripherals or RST/NMI.

## Operating Modes-Low Power Mode in details

---

### □ Low-Power Mode 4 (LPM4)

System Resets, Interrupts, and Operating Modes In low power mode 4 all activities cease; only the RAM contents, I/O ports, and registers are maintained. Wake up is only possible by enabled external interrupts.

Before activating LPM4, the software should consider the system conditions during the low power mode period. The two most important conditions are environmental (that is, temperature effect on the DCO), and the clocked operation conditions.

The environment defines whether the value of the frequency integrator should be held or corrected. A correction should be made when ambient conditions are anticipated to change drastically enough to increase or decrease the system frequency while the device is in LPM4.

## Operating Modes-Examples

- The following example describes entering into low-power mode 0.

```

;===Main program flow with switch to CPUOff Mode=====
BIS #18h,SR ;Enter LPM0 + enable general interrupt GIE
            ;(CPUOff=1, GIE=1). The PC is incremented
            ;during execution of this instruction and
            ;points to the consecutive program step.
.....
            ;The program continues here if the CPUOff
            ;bit is reset during the interrupt service
            ;routine. Otherwise, the PC retains its
            ;value and the processor returns to LPM0.
    
```

- The following example describes clearing low-power mode 0.

```

;===Interrupt service routine=====
.....
BIC #10h,0(SP) ;CPU is active while handling interrupts
              ;Clears the CPUOff bit in the SR contents
              ;that were stored on the stack.
RETI          ;RETI restores the CPU to the active state
              ;because the SR values that are stored on
              ;the stack were manipulated. This occurs
              ;because the SR is pushed onto the stack
              ;upon an interrupt, then restored from the
              ;stack after the RETI instruction.
    
```

## Operating Modes C Examples

- C – programming msp430x14x.h

- ...

```

/*****
 * STATUS REGISTER BITS
 *****/
#include "In430.h"

#define LPM0      _BIS_SR(LPM0_bits) /* Enter LP Mode 0 */
#define LPM0_EXIT _BIS_SR(LPM0_bits) /* Exit LP Mode 0 */
#define LPM1      _BIS_SR(LPM1_bits) /* Enter LP Mode 1 */
#define LPM1_EXIT _BIS_SR(LPM1_bits) /* Exit LP Mode 1 */
#define LPM2      _BIS_SR(LPM2_bits) /* Enter LP Mode 2 */
#define LPM2_EXIT _BIS_SR(LPM2_bits) /* Exit LP Mode 2 */
#define LPM3      _BIS_SR(LPM3_bits) /* Enter LP Mode 3 */
#define LPM3_EXIT _BIS_SR(LPM3_bits) /* Exit LP Mode 3 */
#define LPM4      _BIS_SR(LPM4_bits) /* Enter LP Mode 4 */
#define LPM4_EXIT _BIS_SR(LPM4_bits) /* Exit LP Mode 4 */
#endif /* End #defines for C */

/* Low Power Modes coded with
   Bits 4-7 in SR */
/* Begin #defines for assembler */
#ifndef __IAR_SYSTEMS_ICC
#define LPM0      CPUOFF
#define LPM1      SCG0+CPUOFF
#define LPM2      SCG1+CPUOFF
#define LPM3      SCG1+SCG0+CPUOFF
#define LPM4      SCG1+SCG0+OSCOFF+CPUOFF
/* End #defines for assembler */

#else /* Begin #defines for C */
#define LPM0_bits CPUOFF
#define LPM1_bits SCG0+CPUOFF
#define LPM2_bits SCG1+CPUOFF
#define LPM3_bits SCG1+SCG0+CPUOFF
#define LPM4_bits SCG1+SCG0+OSCOFF+CPUOFF
    
```