Network Embedded Systems
Sensor Networks – Fall 2013

Hardware

Marcus Chang, mchang@cs.jhu.edu

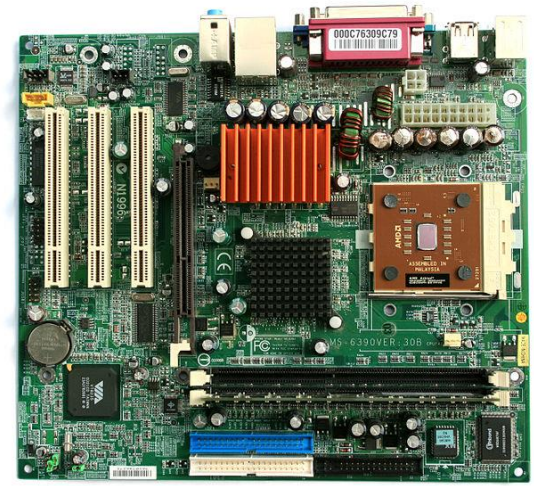
Embedded Systems

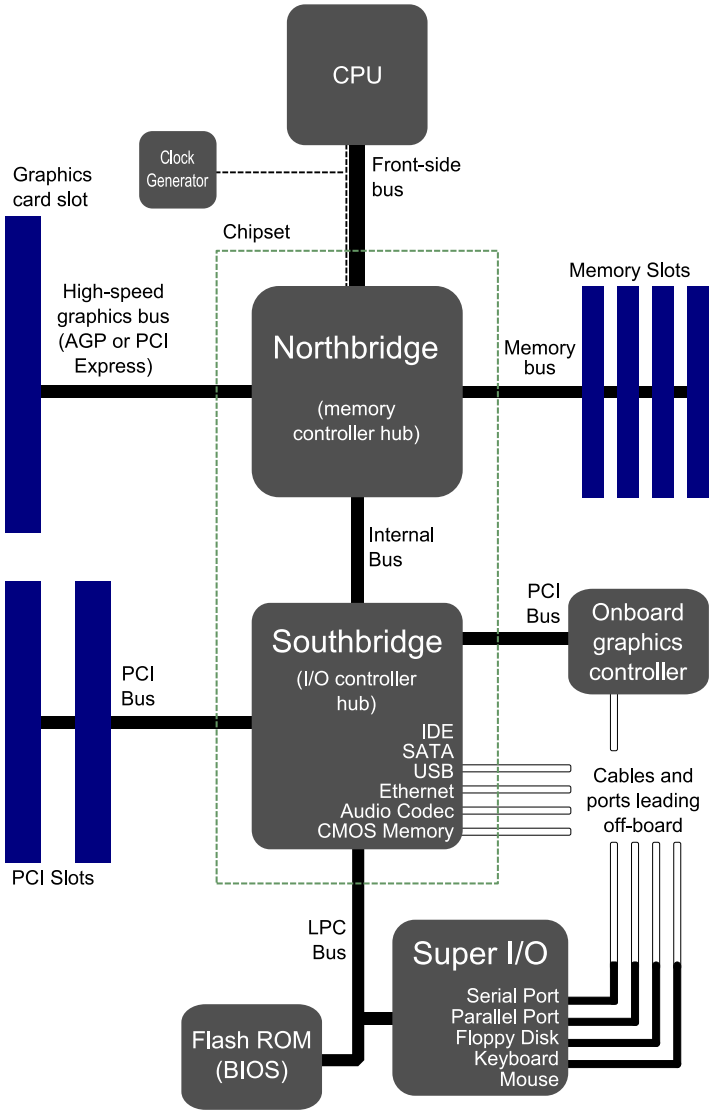
- Designed to do one or a few dedicated and/or specific functions
- *Embedded* as part of a complete device often including hardware and mechanical parts
- By contrast, a general-purpose computer, is designed to be flexible and to meet a wide range of end-user needs

- Wikipedia

Personal Computers

- Individual Components
  - CPU
  - Northbridge
    - RAM
    - GPU
  - Southbridge
    - Hard drive
    - WIFI
    - USB

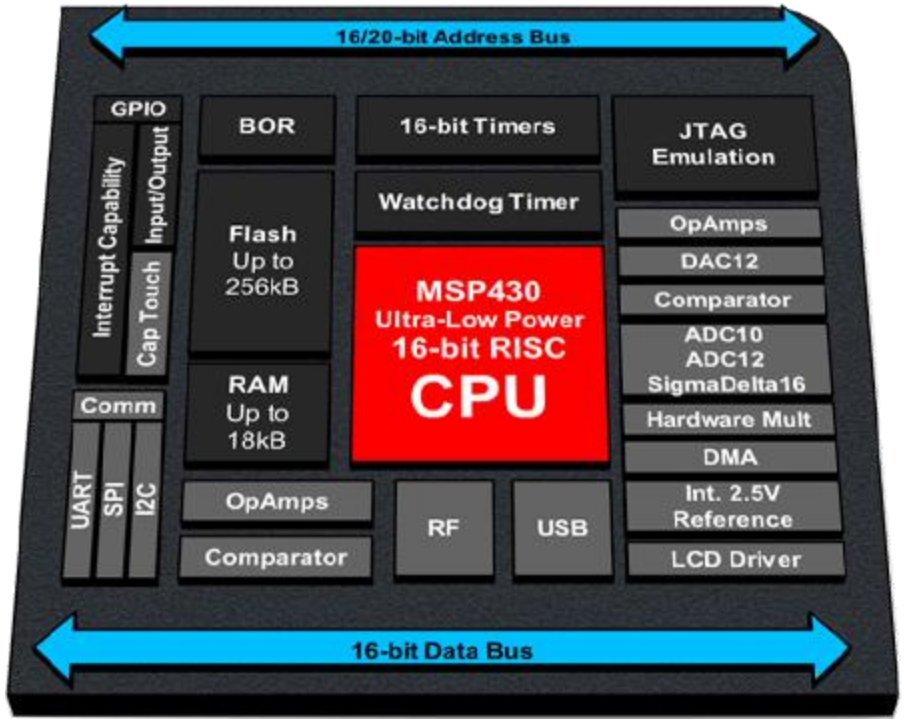




Images: Wikipedia

Microcontroller Unit (MCU)

- Multiple components in same chip
  - CPU
  - Volatile memory (e.g. RAM)
  - Persistent memory (e.g. Flash)
  - Analog-to-Digital Converters
  - Timers (alarms, stop-watches)
  - Peripheral interfaces
  - Radio
  - …



- Think of multiple cores, some with only one purpose but very high efficiency/low power consumption

Image: Texas Instruments

System on a Chip (SoC)

- Microcontrollers are systems on a single chip
  - However, often people think of SoC as more powerful than MCU
  - Qualcomm Snapdragon

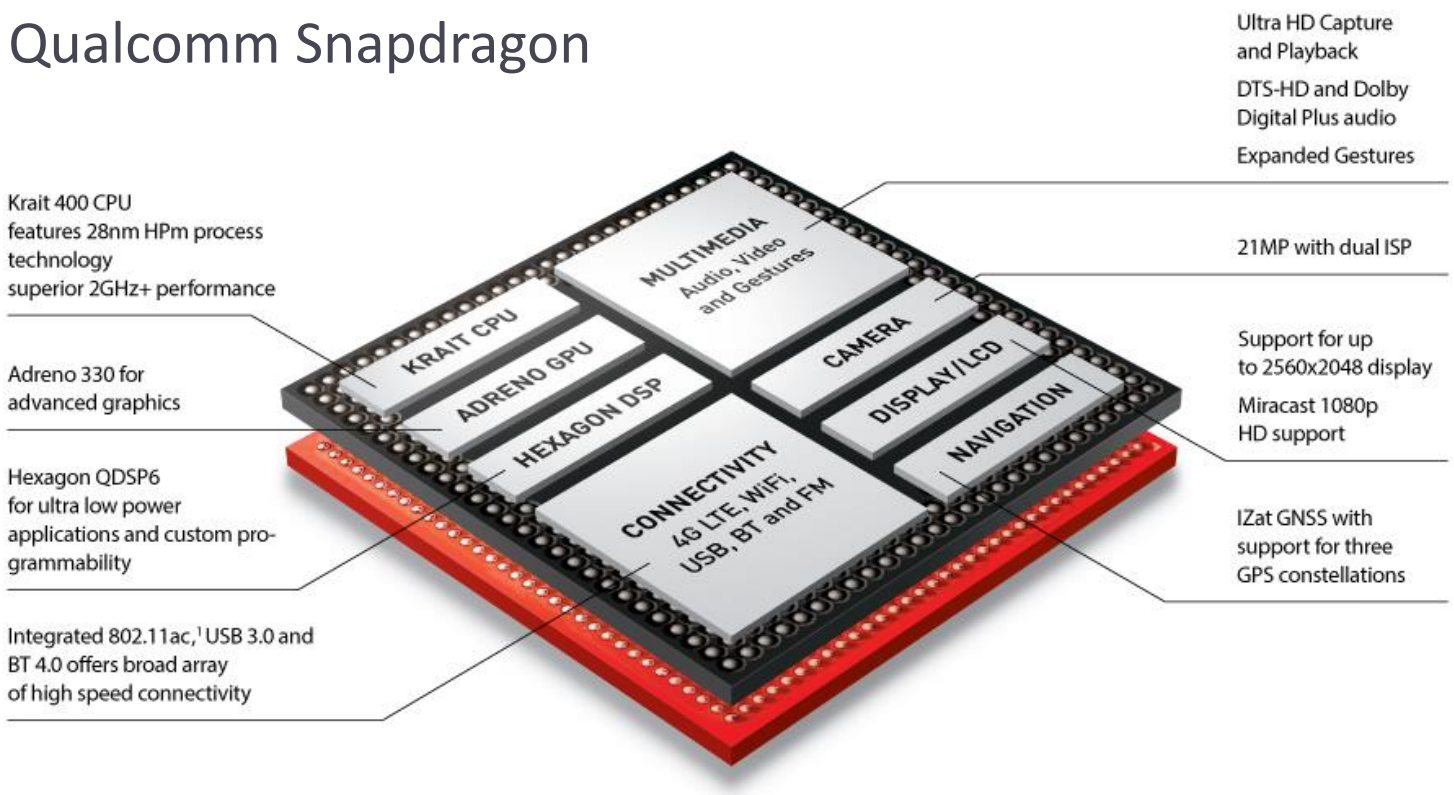


Image: Qualcomm

Famous Microcontrollers / System on a Chip

- Home appliances, Auto industry, Industrial systems
  - Intel 8051 – billions units shipped per year (2008)
  - Microchip PIC – 6 billion in total (2008)
  - Atmel AVR – (Arduino)
  - Texas Instruments MSP430

- Cell phones, Tablets
  - ARM Cortex
    - Apple A6
    - Qualcomm Snapdragon
    - Samsung Exynos

Telos rev. B, Telosb, Tmote Sky

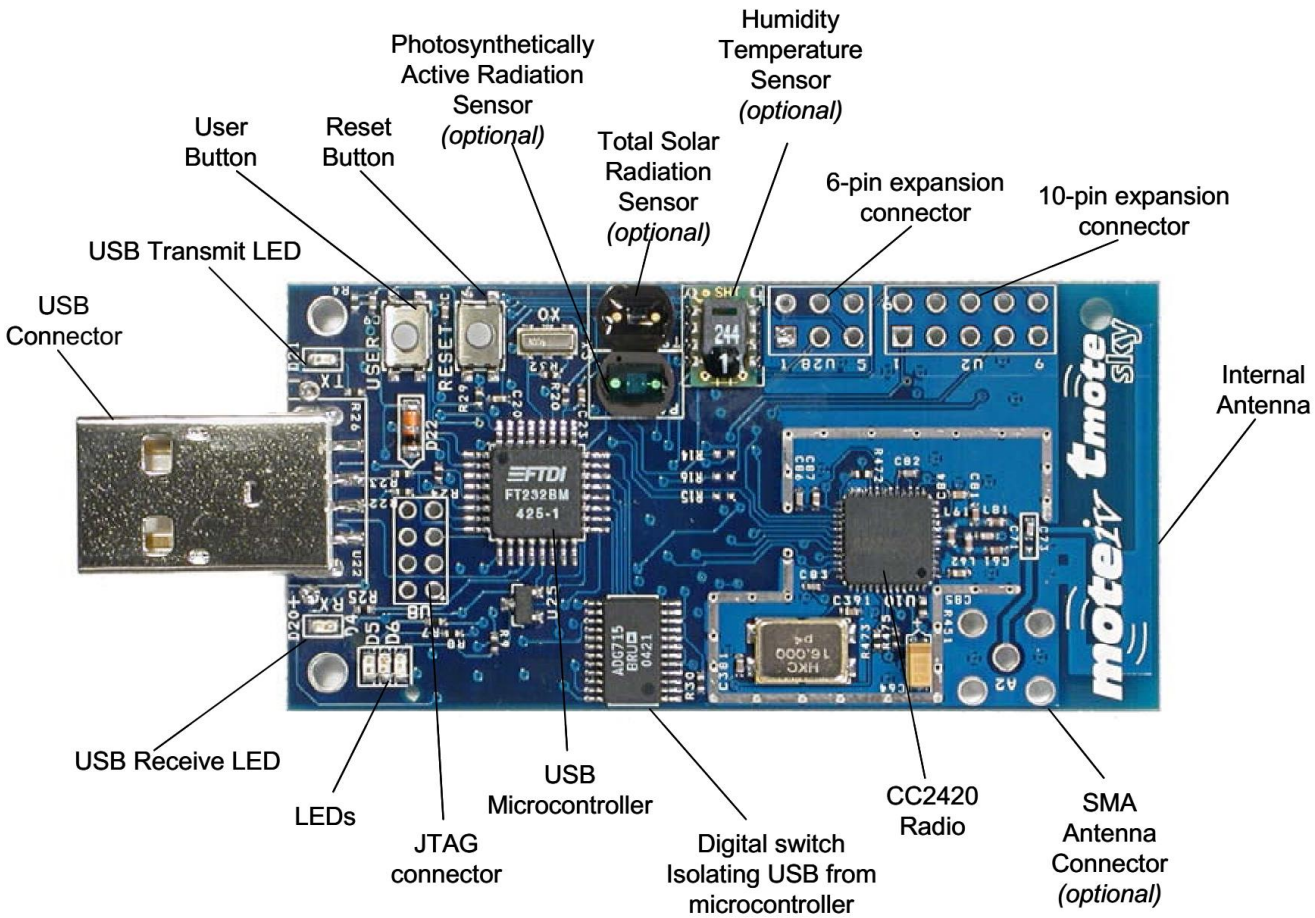


Image: MoteIV

Telos rev. B, Telosb, Tmote Sky

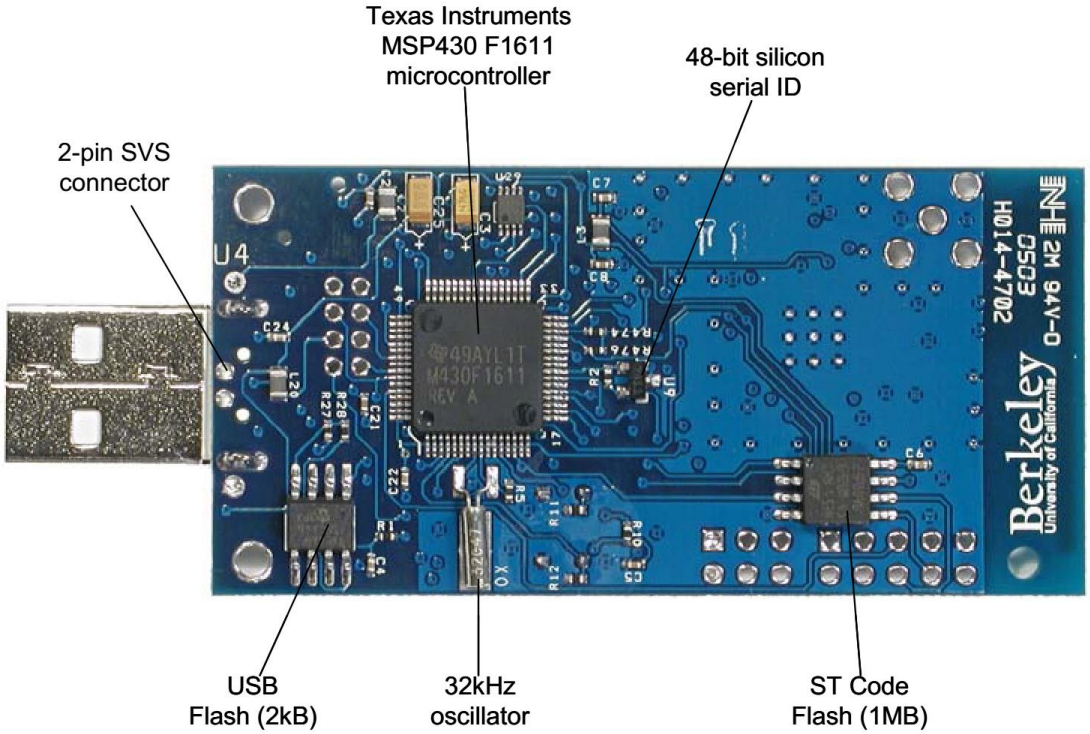


Image: MoteIV

Telosb Block Diagram

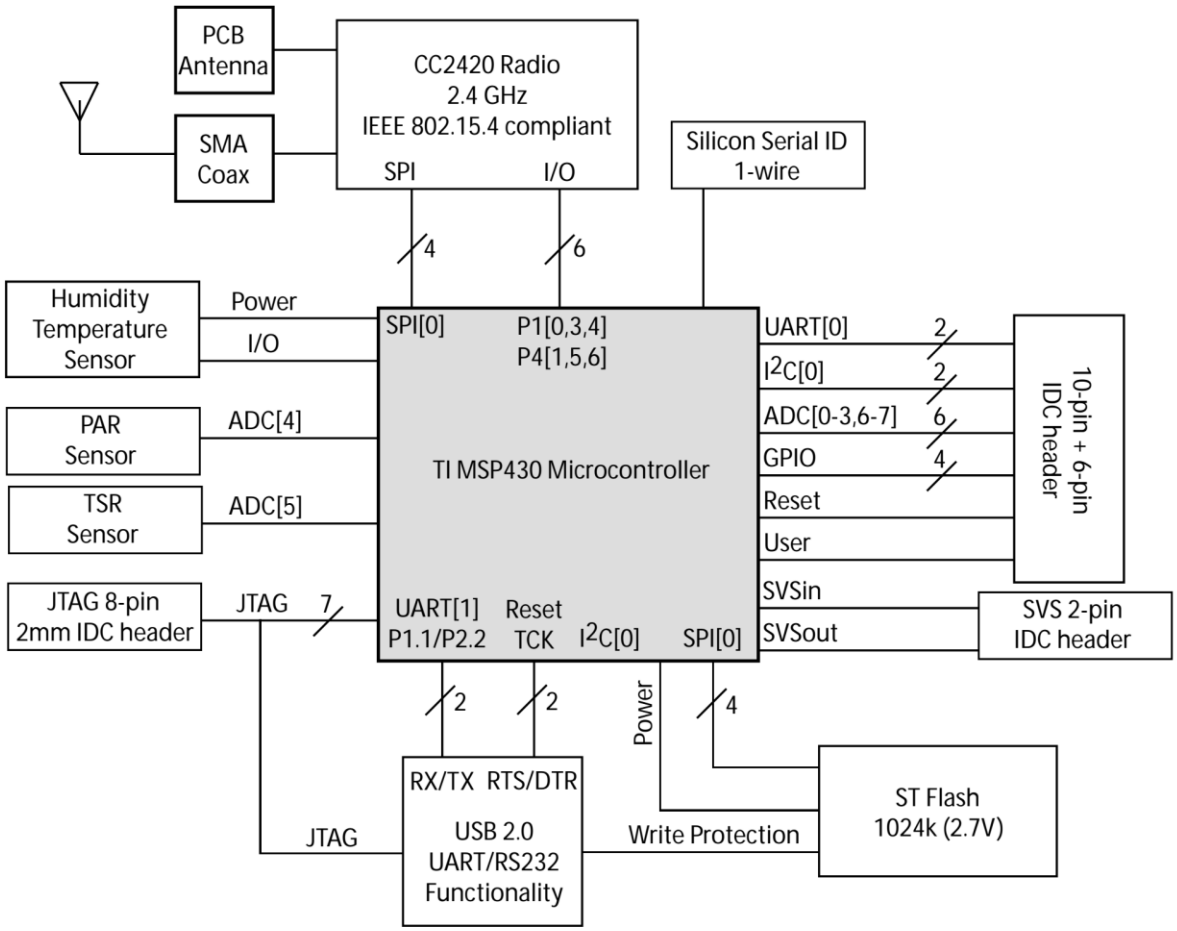


Image: MoteIV

Texas Instruments MSP430F1611 MCU

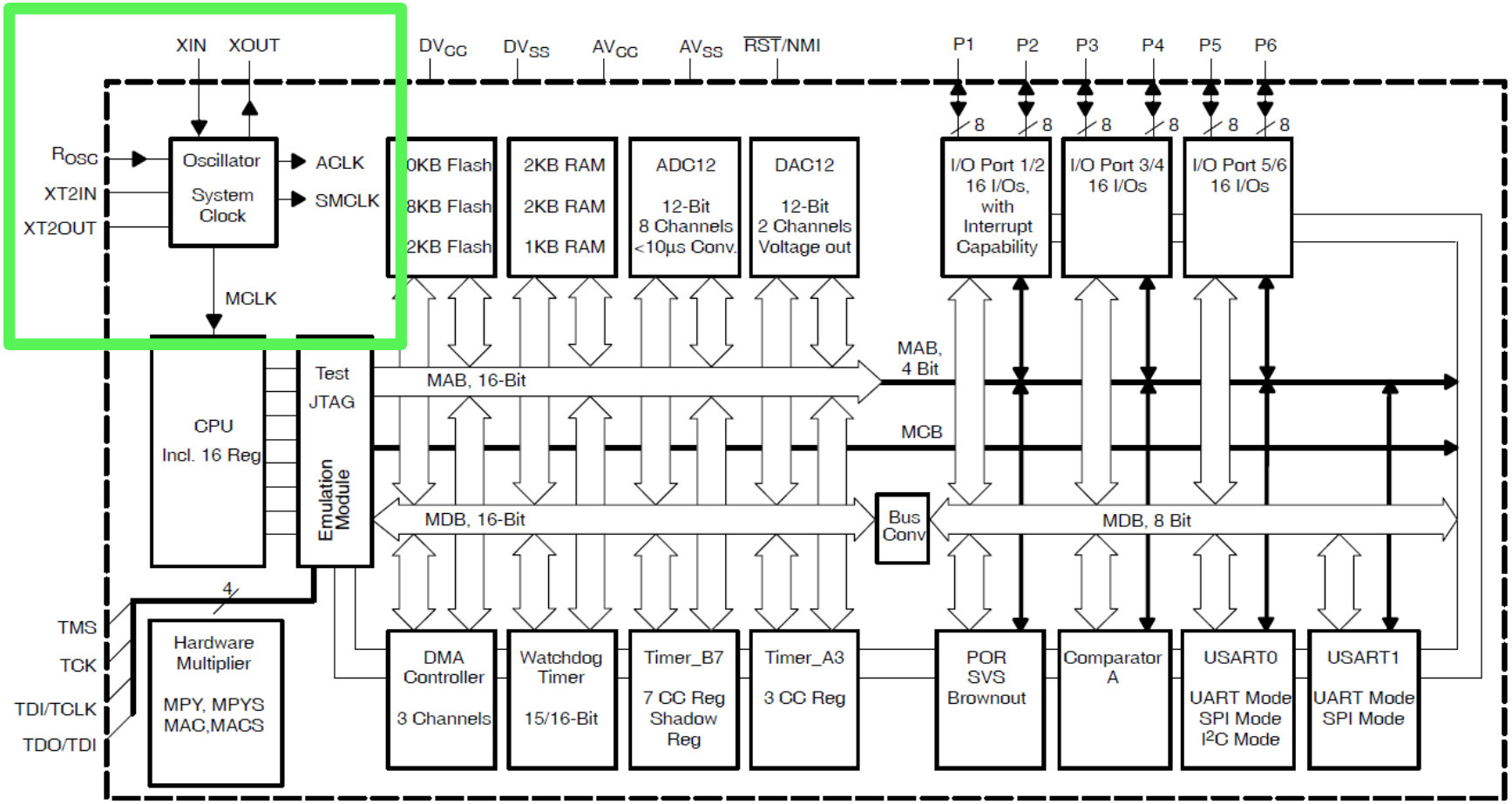
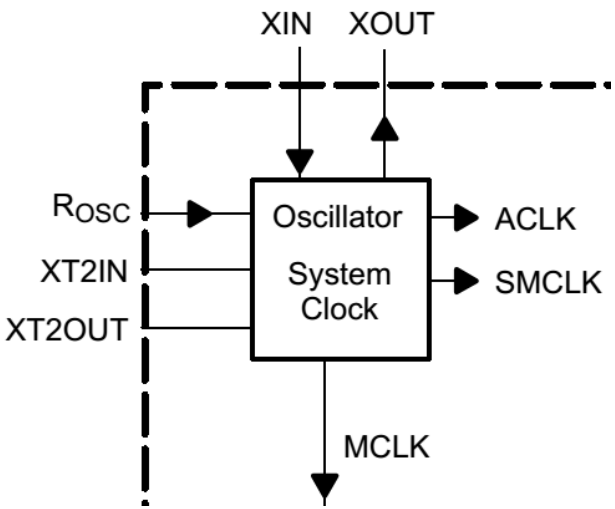


Image: Texas Instruments

MSP430 System Clock

- Drives <u>everything</u>



- Inputs:
  - Low-frequency crystal (kHz range)
  - High-frequency crystal (MHz range)
  - Digitally-Controlled Oscillator (DCO)

- Outputs:
  - Master Clock
  - Sub-Main Clock
  - Auxiliary Clock

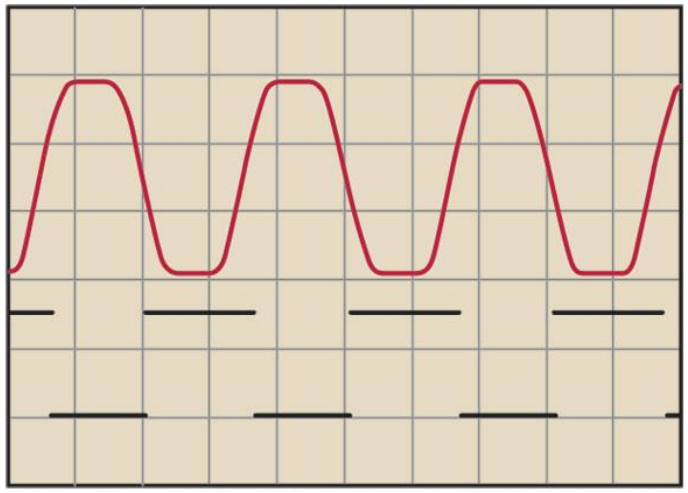


Image: Hubert Hagadorn

Telosb Clock Sources

- External Sources
  - Crystal (32 kHz)
- Precision: Crystal > DCO
  - Calibrate DCO with crystal
- Temperature:
  - Crystal, DCO change frequency
  - Store temperature specific calibration parameters

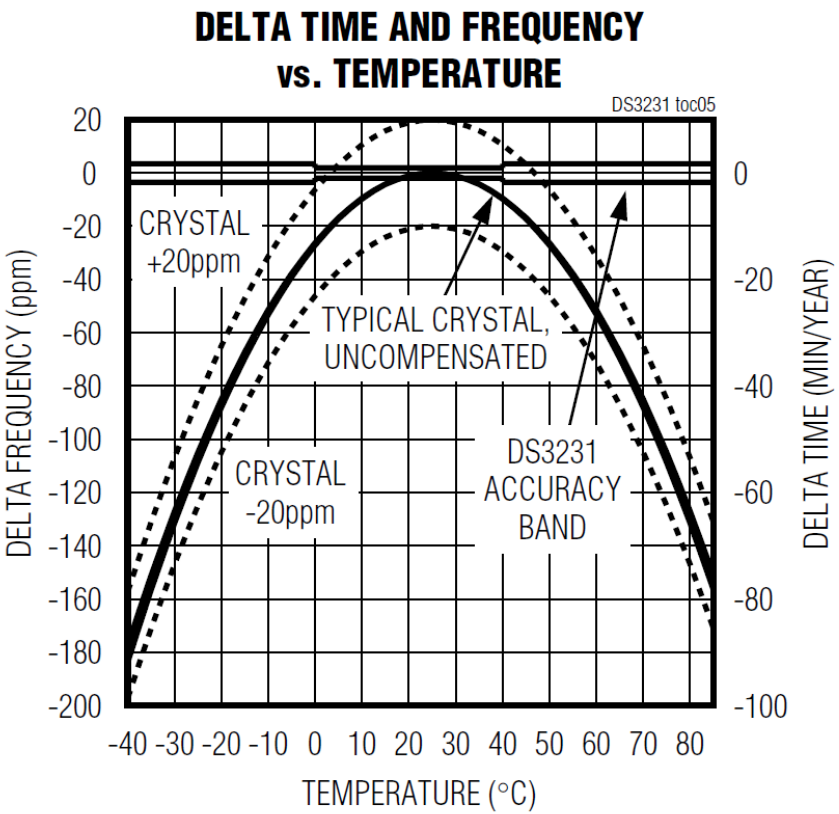


Image: Dallas Semiconductors

System Clock Configuration

- Input-to-Output
  - User configurable with constraints
  - CPU
    - Master System Clock
  - Peripherals
    - Master System
    - Sub-main Clock
    - Auxiliary Clock

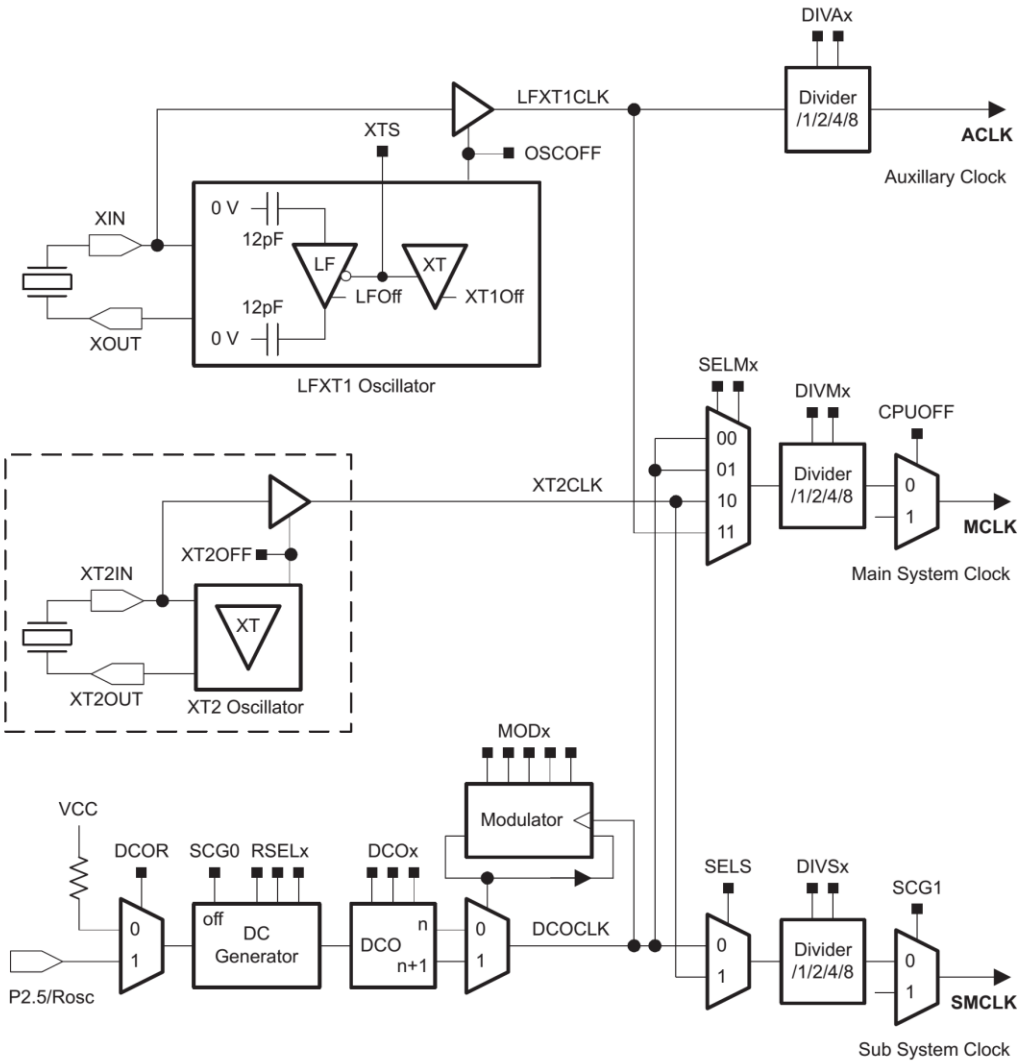


Image: Texas Instruments

MCU Power Operating Modes

- 5 Low-Power Modes in total
  - Active Mode: CPU active, all clocks active
  - LPM0: CPU disabled, SMCLK and ACLK active
  - LPM3: CPU disabled, ACLK active
  - LPM4: CPU, all clocks disabled

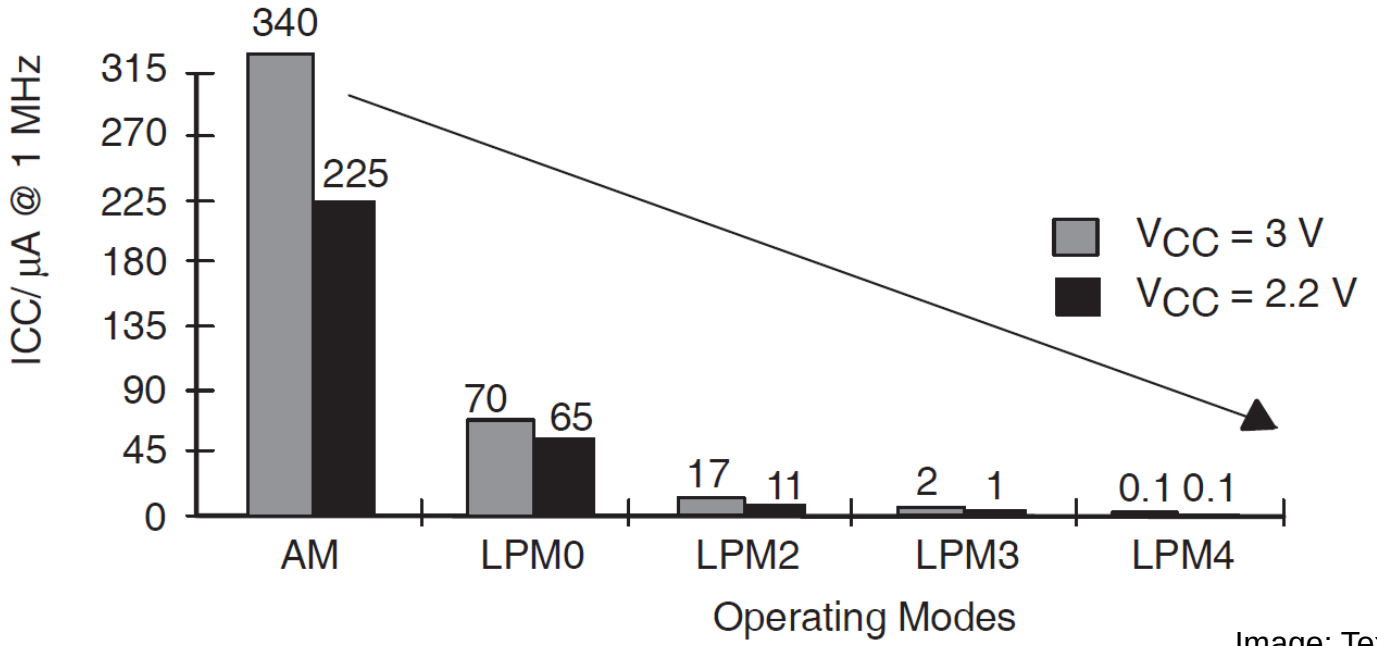


Image: Texas Instruments

System Clock Summary

- MCU components are driven by MCLK, SMCLK, ACLK
  - Changing one input can affect multiple outputs
  - Changing one output can affect multiple components
- The relation between the three are limited by hardware
  - Signals are multiplied and/or divided
- Peripheral components can have different requirements
  - E.g., high-speed timer vs. low speed communication
- **Choosing the clock source wisely can reduce power consumption significantly**

Timers

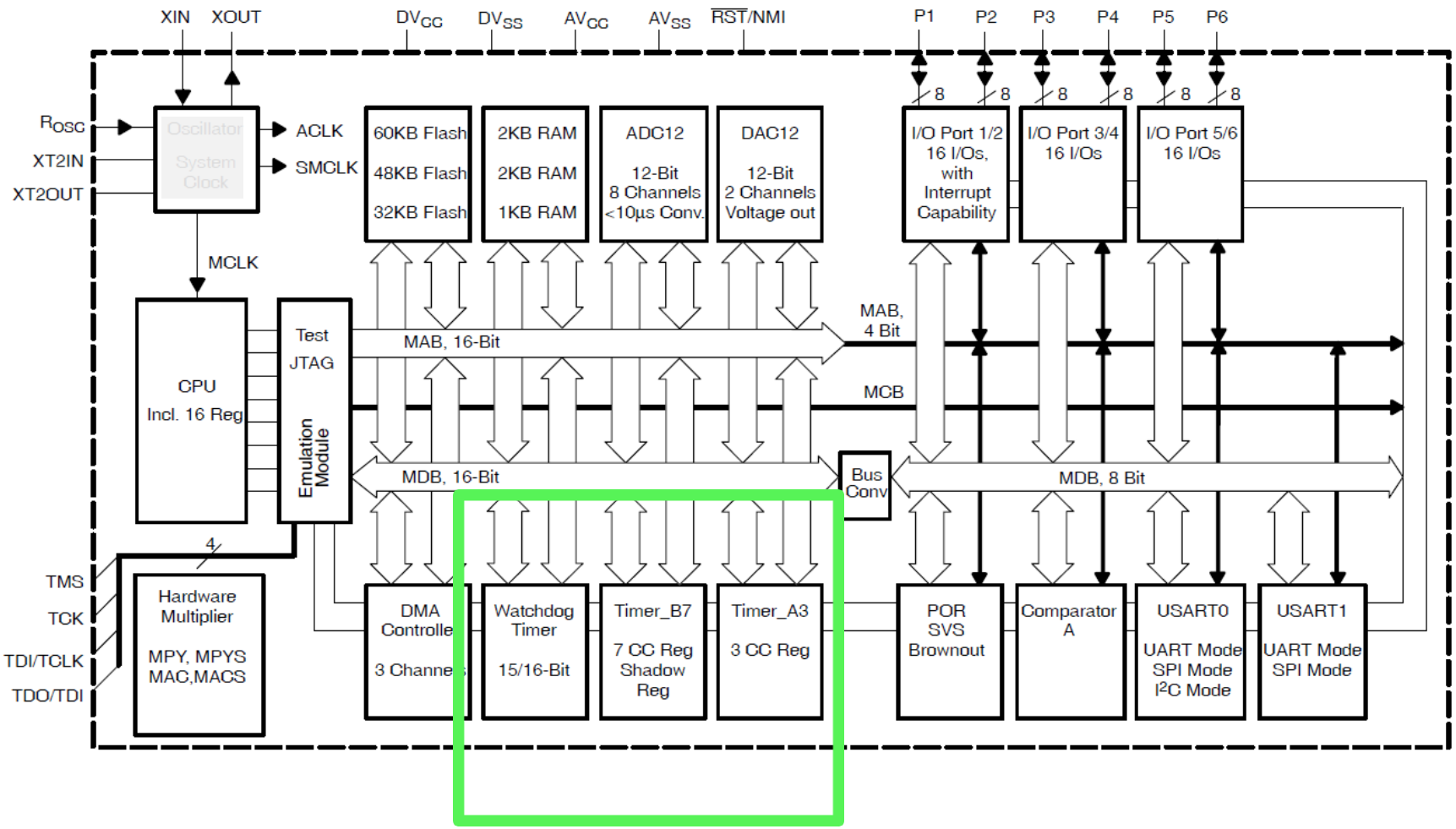
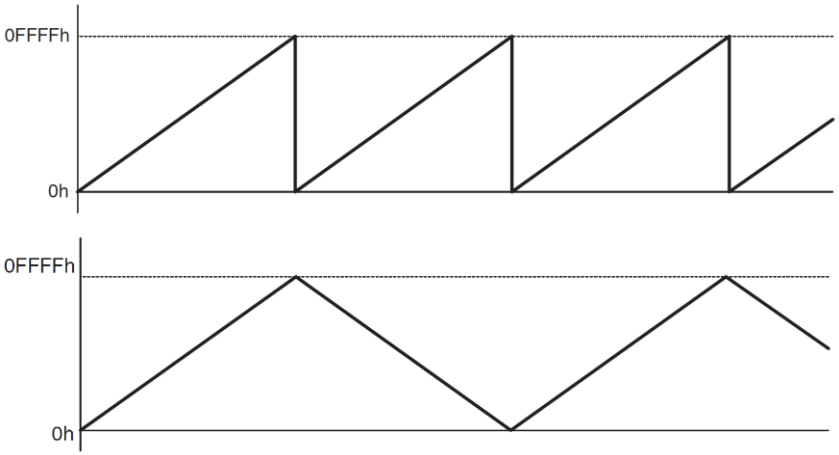


Image: Texas Instruments

Timer Module

- 16-bit counter connected to either SMCLK or ACLK
  - Counts number of clock cycles
    - Measure elapsed time
    - Timestamp events
  - Set alarms at specific points
  - Continuous/Up-down mode
    - Overflow



- Capture/Compare Module
  - Read counter when an event occurs
  - Generate event when counter reaches specific value

Image: Texas Instruments

Watchdog Timer

- Similar to Timer Module, except
- Microcontroller resets when alarm is triggered
  - Clear counter periodically to avoid it (i.e. ‘feed the watchdog’)
  - Useful to reset microcontroller to a known state

- The watchdog is enabled by default after reset

Analog Signals

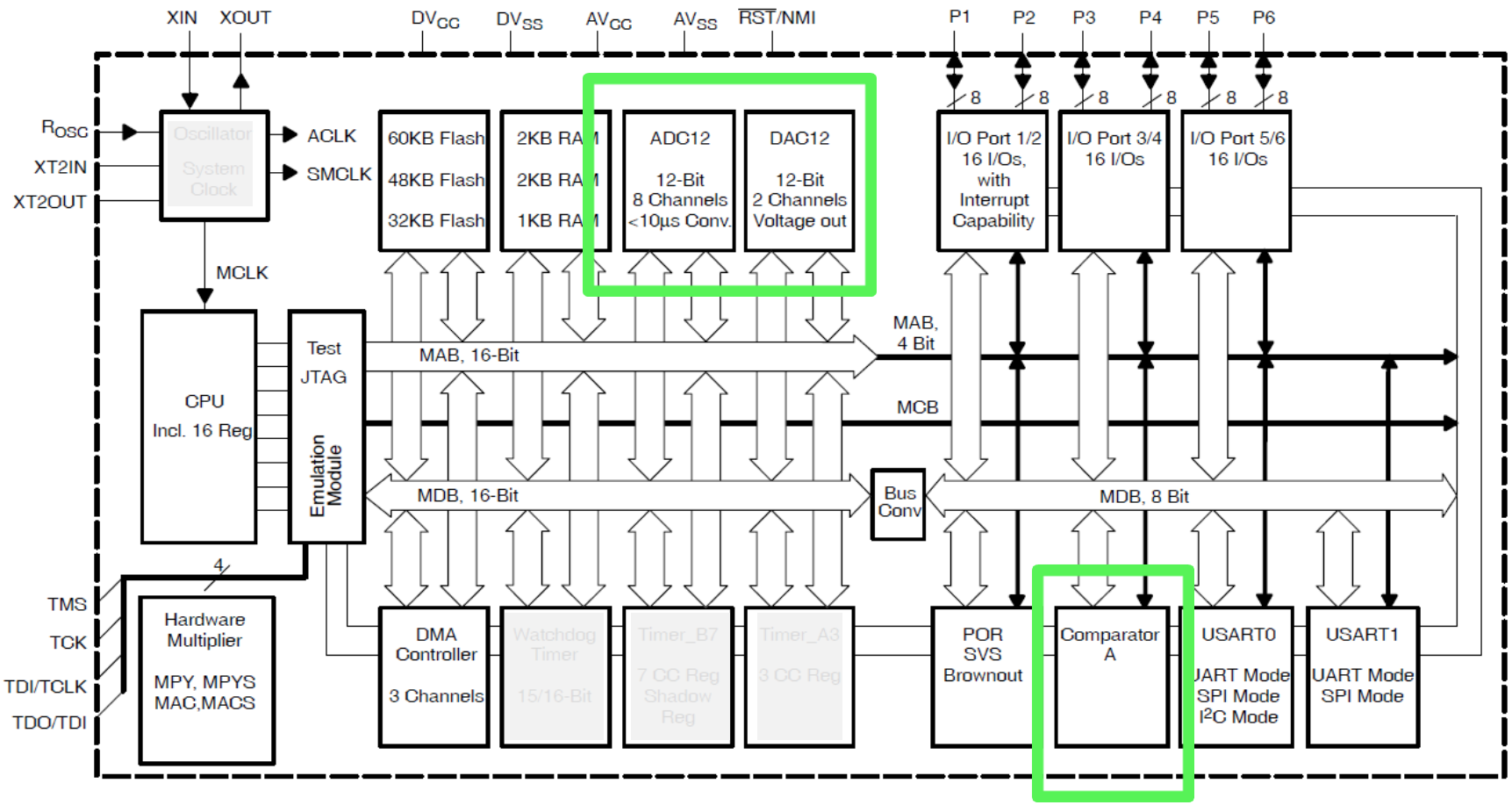


Image: Texas Instruments

Analog Signals

- 12-bit Analog-to-Digital Converter (ADC)
  - Measure input voltage as a fraction of the reference voltage
  - $$N_{ADC} = 4095 \times \frac{Vin - V_{R-}}{V_{R+} - V_{R-}}$$
  - Analog sensors output voltage
- Comparator
  - Compare voltage between two analog signals
- Digital-to-Analog Converter
  - Outputs voltage

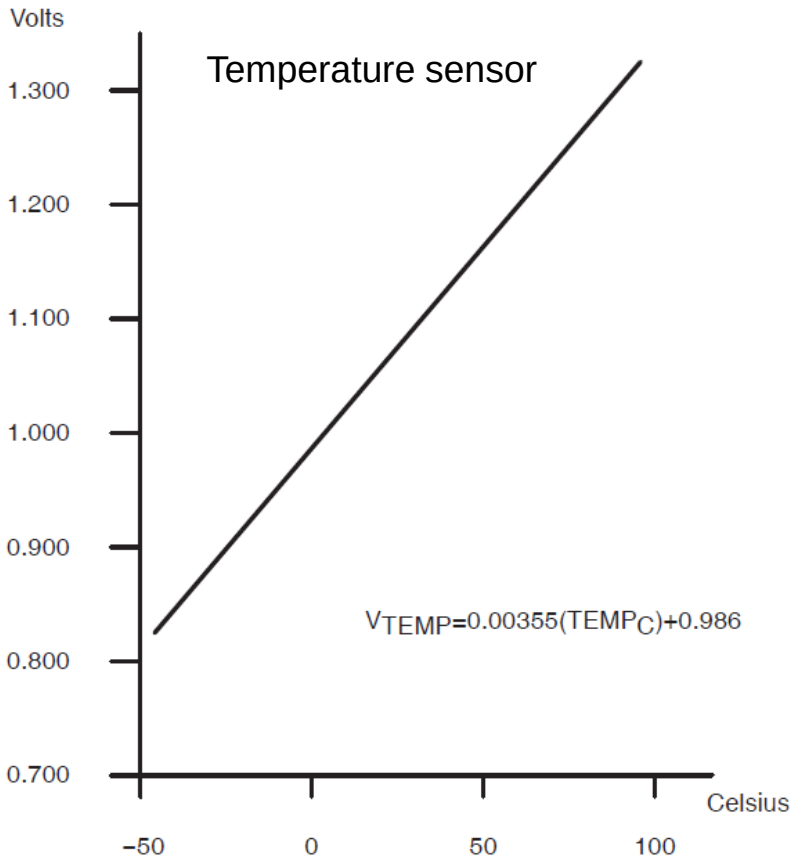


Image: Texas Instruments

External Components

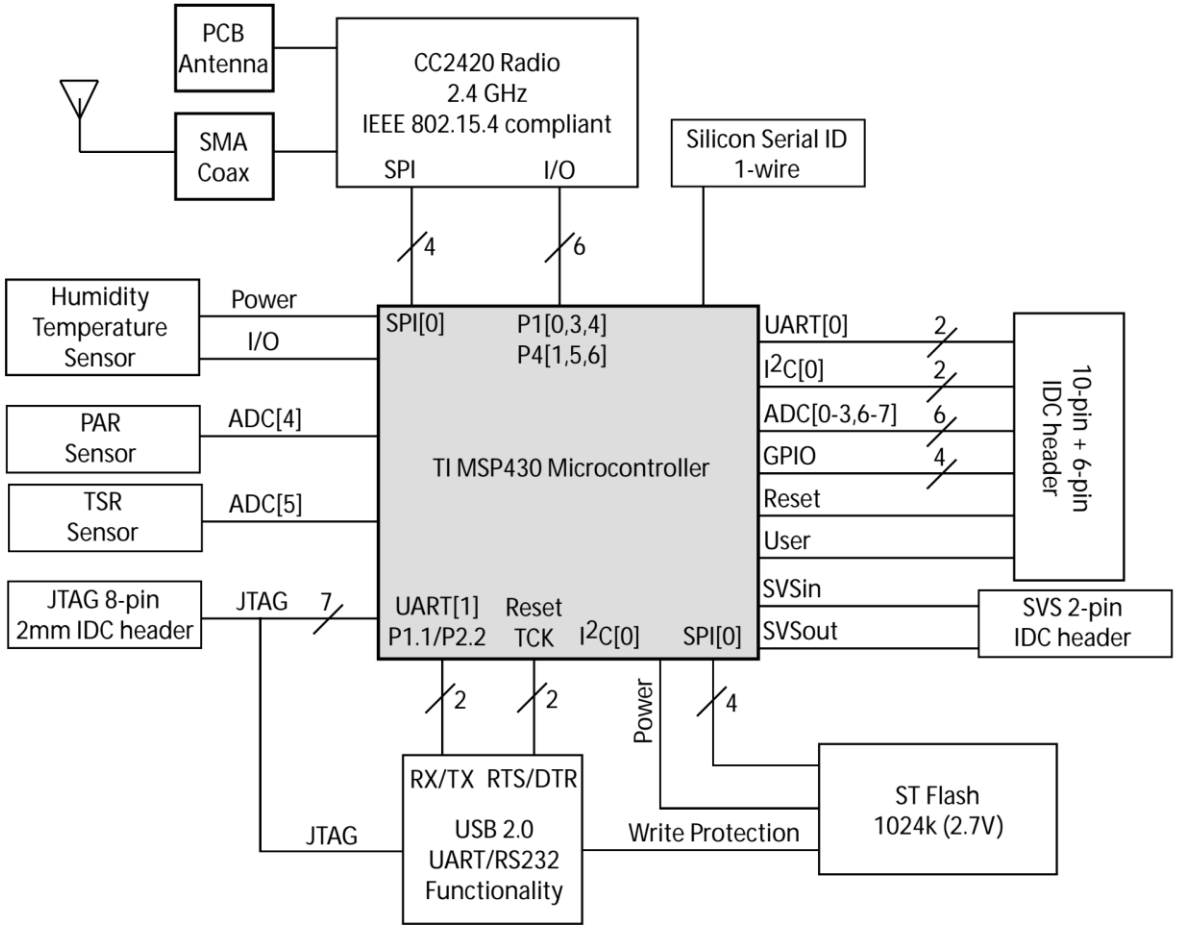


Image: MoteIV

General Purpose Input/Output Pins (GPIO)

- Input Pin
  - High voltage reads ‘1’ – low voltage reads ‘0’
    - High – MCU supply voltage minus Δ
    - Low – 0V plus Δ
- Output Pin
  - Write ‘1’ sets voltage high – write ‘0’ sets voltage low
- Interrupt capable Pin
  - Interrupt program flow when input value changes
    - E.g. press button to make voltage go low

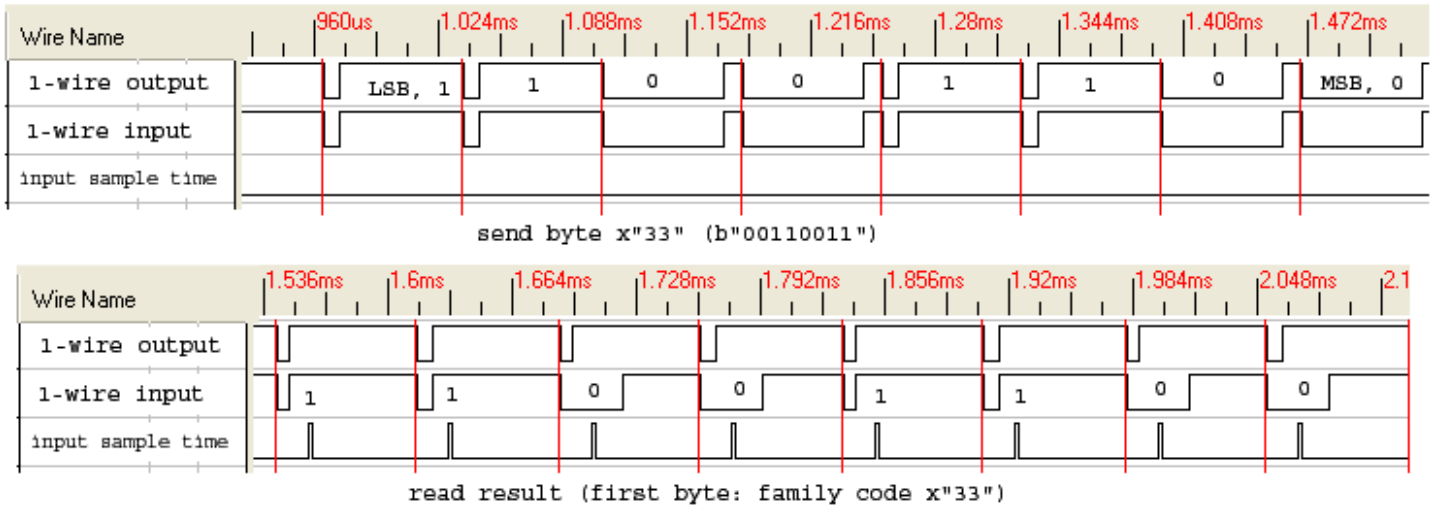
MCU to External Chip Communication

- GPIO pin
  - Control signals:
    - Turn peripherals on/off
    - Configure peripherals
  - Data streams:
    - Manually transmit 1 bit at a time – ‘bit banging’
    - Inefficient, keeps CPU in Active Mode

Images: Wikipedia

1-Wire Bus (Timer Module)

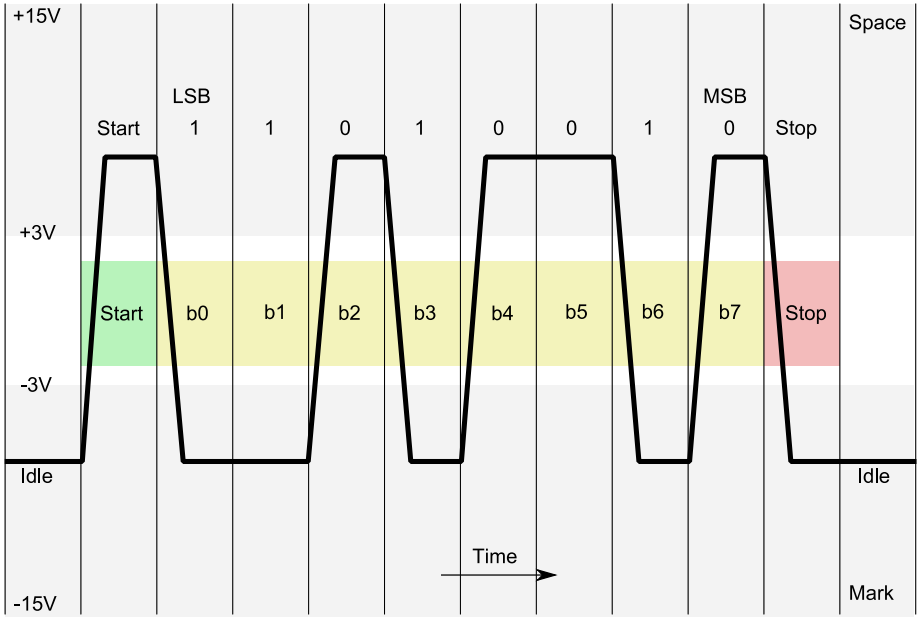
- 1 signal line, multiple devices (master/slaves)
  - Designed for parasitic power
  - 64-bit unique serial number/address
  - Master pulls wire low to transmit
    - Duration determines whether it is a ‘1’ or ‘0’



Images: Wikipedia

Universal Synchronous/Asynchronous Receive/Transmit Module (USART)

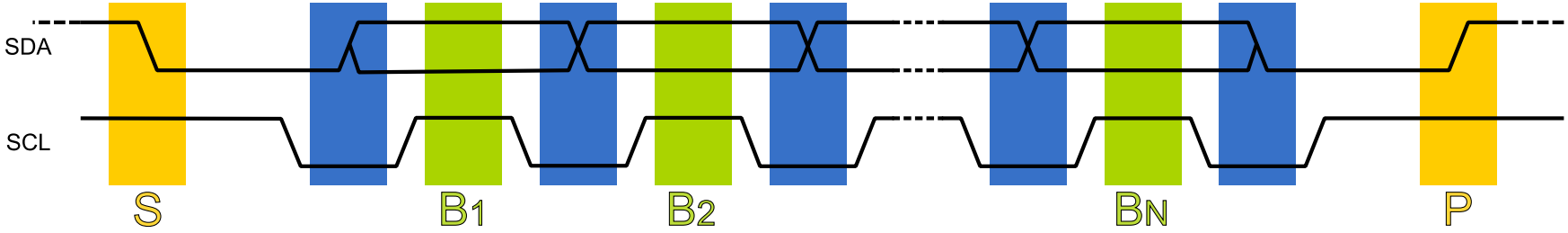
- UART mode
  - Universal asynchronous receiver/transmitter (UART)
  - Transfers 1 ASCII character (7 or 8 bit) at a time
  - Self clocked
  - Receiver oversamples
  - Very popular for interfacing with switches and routers



Images: Wikipedia

Universal Synchronous/Asynchronous Receive/Transmit Module (USART)

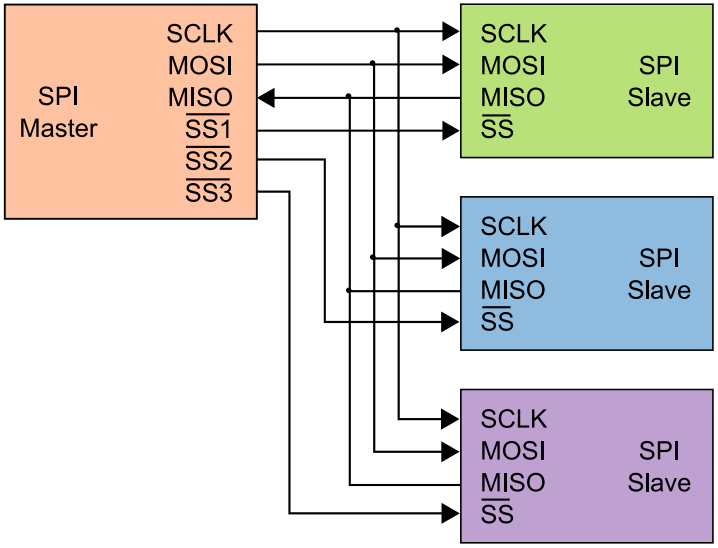
- Inter-Integrated Circuit (I2C) Mode
  - 2 signal lines, multiple devices (masters/slaves)
  - SCL – clock line (for timing)
  - SDA – data line
    - Speed: 400 kbit/s (although 3.4 Mbit/s standard exists)
  - Value is read/written on clock boundaries
    - 7-bit or 10-bit address space (1-bit used to signal read or write)



Images: Wikipedia

Universal Synchronous/Asynchronous Receive/Transmit Module (USART)

- Serial Peripheral Interface Bus (SPI) Mode
  - 3+1 signal lines, multiple devices (master/slaves)
    - Clock
    - MISO – master in, slave out
    - MOSI – master out, slave in
    - Slave select (enable slave when line is low)
  - Value is read/written on clock boundaries
  - Full duplex



Images: Wikipedia

Internal MCU Communication

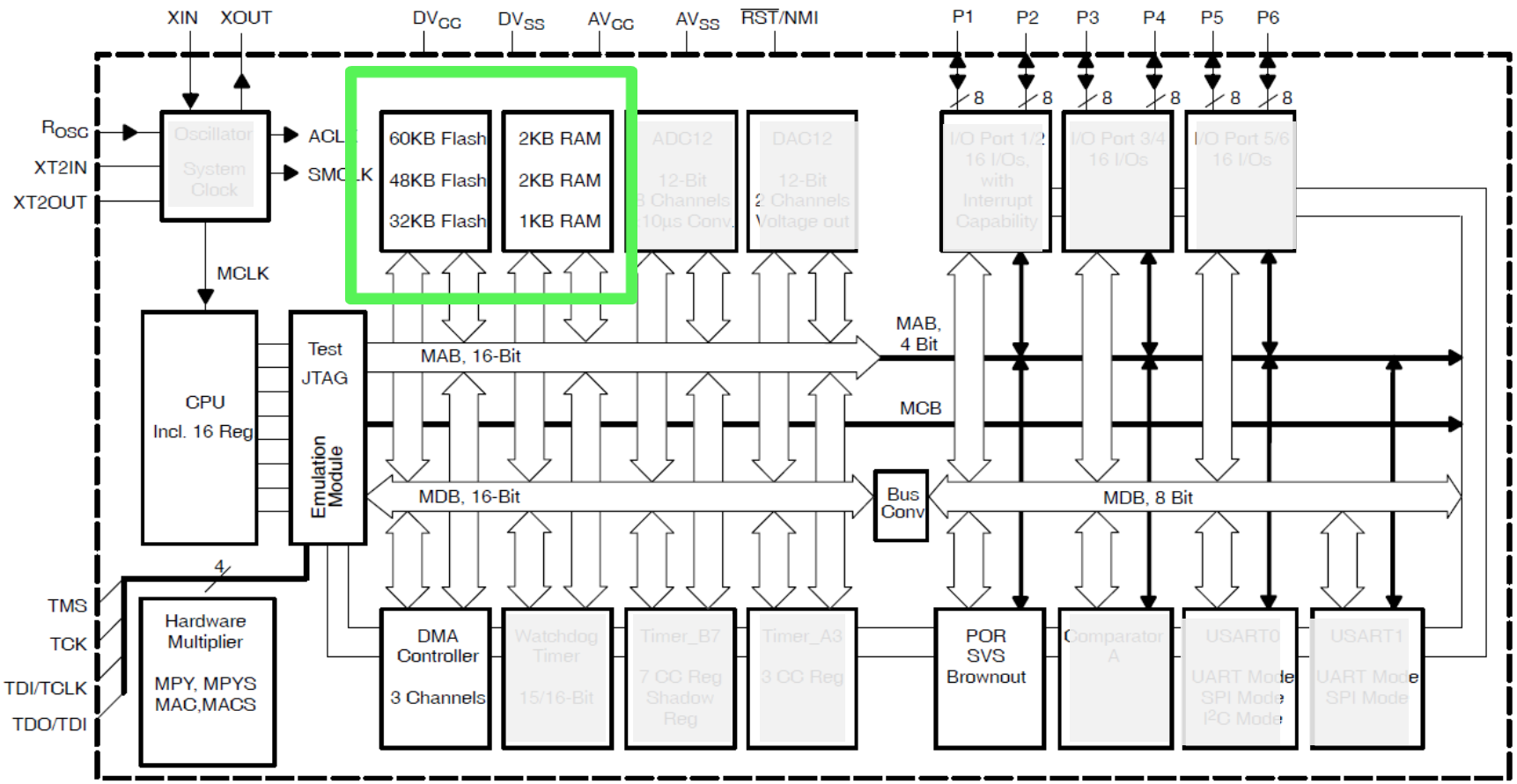
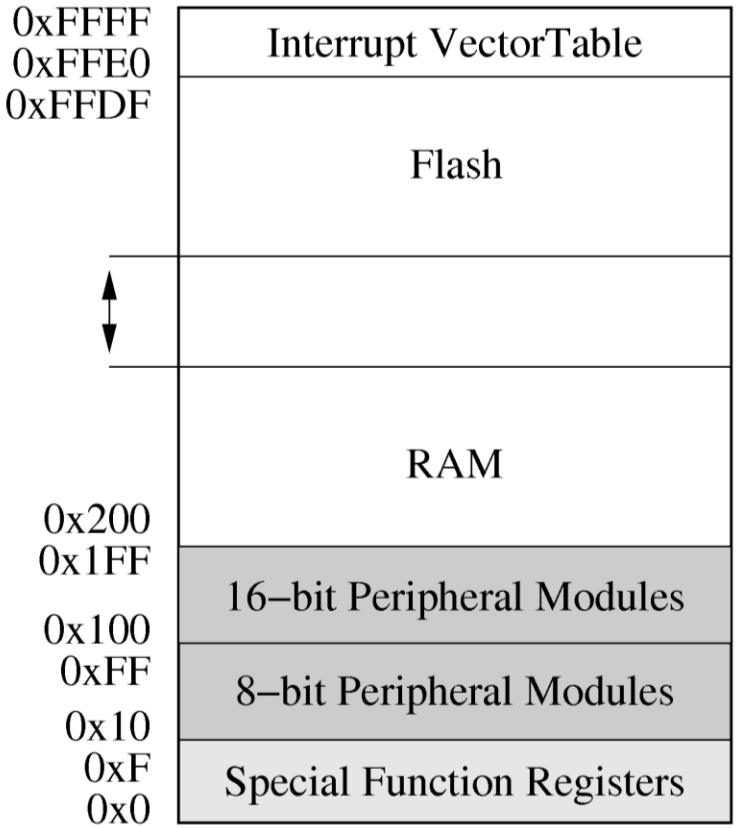


Image: Texas Instruments

Memory Map

- Shared Memory Space
- Memory Mapped I/O
  - Peripheral modules are accessed as regular variables (pointers)
  - Flash read, extension of RAM
  - Flash write, ‘peripheral module’
- Pros and cons
  - Global variables, no overhead
  - No memory protection



MSP430 Memory Map

Images: Doina Bucur

Recap: GPIO

Port	Register	Short Form	Address	Register Type	Initial State
P1	Input	P1IN	020h	Read only	–
	Output	P1OUT	021h	Read/write	Unchanged
	Direction	P1DIR	022h	Read/write	Reset with PUC
	Interrupt Flag	P1IFG	023h	Read/write	Reset with PUC
	Interrupt Edge Select	P1IES	024h	Read/write	Unchanged
	Interrupt Enable	P1IE	025h	Read/write	Reset with PUC
	Port Select	P1SEL	026h	Read/write	Reset with PUC

- 8 pins on one port
  - one bit per pin = 1 byte per port
  - E.g. set Port 1 pin 3 high:
    - **P1OUT |= 0x04; // set value before direction to avoid outputting old value**
    - **P1DIR |= 0x04; // high means pin is output**

Source: Texas Instruments

Recap: Timer Module

TACTL:

7	6	5	4	3	2	1	0
IDx		MCx		Unused	TACLR	TAIE	TAIFG
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	w–(0)	rw–(0)	rw–(0)

IDx	Bits 7-6	Input divider. These bits select the divider for the input clock. 00 /1 01 /2 10 /4 11 /8
MCx	Bits 5-4	Mode control. Setting MCx = 00h when Timer_A is not in use conserves power. 00 Stop mode: the timer is halted 01 Up mode: the timer counts up to TACCR0 10 Continuous mode: the timer counts up to 0FFFFh 11 Up/down mode: the timer counts up to TACCR0 then down to 0000h

- Start free-running timer:
  - **TACTL = (0x02 << 4);**

Image: Texas Instruments

Recap: Timer Module

Register	Short Form	Register Type	Address	Initial State
Timer_A control	TACTL	Read/write	0160h	Reset with POR
Timer_A counter	TAR	Read/write	0170h	Reset with POR
Timer_A capture/compare control 0	TACCTL0	Read/write	0162h	Reset with POR
Timer_A capture/compare 0	TACCR0	Read/write	0172h	Reset with POR
Timer_A capture/compare control 1	TACCTL1	Read/write	0164h	Reset with POR
Timer_A capture/compare 1	TACCR1	Read/write	0174h	Reset with POR
Timer_A capture/compare control 2	TACCTL2	Read/write	0166h	Reset with POR
Timer_A capture/compare 2	TACCR2	Read/write	0176h	Reset with POR
Timer_A interrupt vector	TAIV	Read only	012Eh	Reset with POR

- Read free-running timer:
  - **TACTL = (0x02 << 4); // start timer**
  - **localtime = TAR; // read value and store in 'localtime'**

Image: Texas Instruments

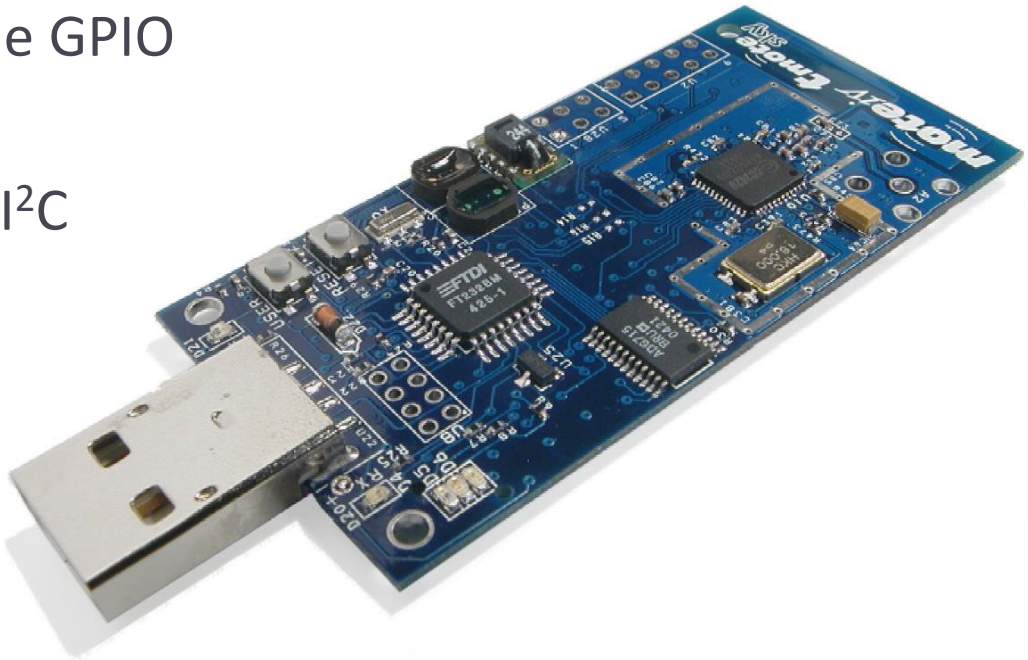
Missing Hardware Components

- What components are missing from the MCU?

- Floating Point Unit
  - Only found in high-powered systems
  - Software floating point: time and space overhead

Telosb Mote

- Microcontroller
  - Texas Instruments MSP430F1611
- Peripherals
  - 2x light sensors – ADC
  - Button – interrupt capable GPIO
  - Unique ID – 1-Wire
  - Temperature/humidity – I^2C
  - External Flash – SPI
  - Radio – SPI, int. GPIO
  - USB – FTDI – UART
  - LEDs – GPIO



Schedule

- Week 1: Introduction and Hardware
- Week 2: Embedded Programming
- Week 3: Medium Access Control
- Week 4: Link Estimation and Tree Routing
- Week 5: IP Networking
- Week 6: Near Field Communication
- Week 7: (seminar, no lecture)
- Week 8: Energy Management
- Week 9: Review and Midterm
- Week 10: Time Synchronization
- Week 11: Localization
- Week 12: Energy Harvesting
- Week 13: (seminar, no lecture)
- Week 14: TBD