PSoC Short Course Day 1 INTRODUCTION TO PSoC®

Q1 2013

Before we begin

Install PSoC Creator (if you haven't already)

- Copy PSoCCreatorSetup_2.1.exe from thumbdrive onto local machine
- Launch PSoCCreatorSetup_2.1.exe
- Follow installation prompts (typical)
- Raise hands for issues / questions

Load Presentation and Labs

- Copy Intro_To_PSoC5.Bundle01.cywrk.Archive01.zip to local machine
- Unzip file to preferred folders
- Double Click ñIntro_To_PSoC5.Bundle01.cywrkò

Agenda

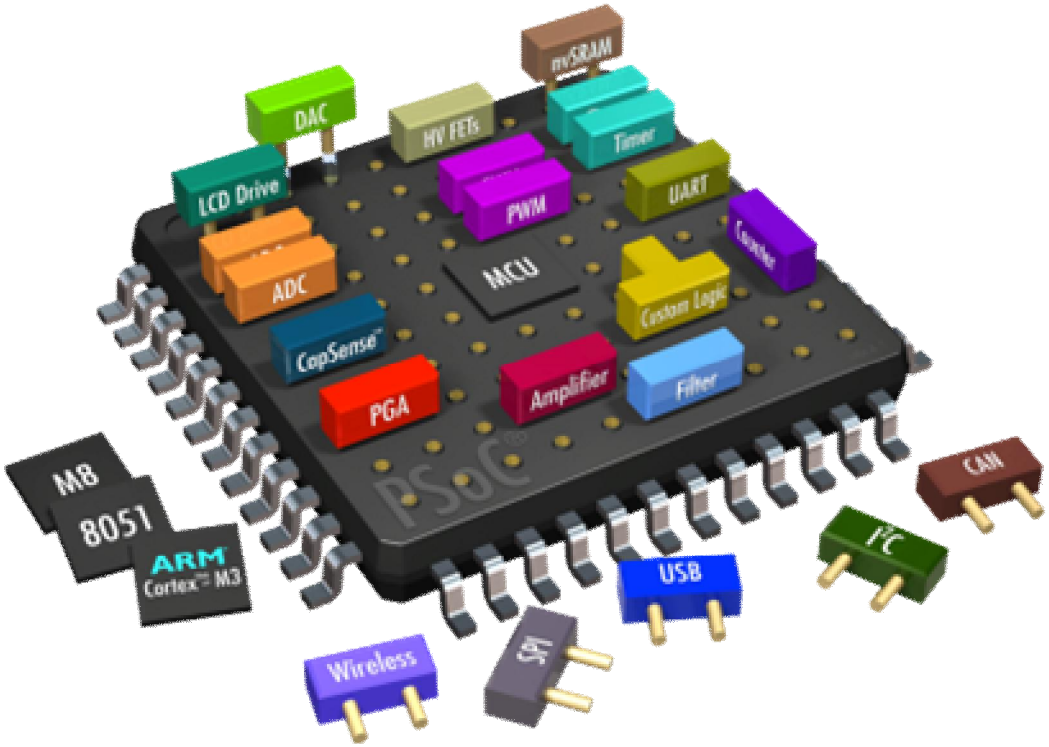
Time	Topic
10:00 ï 10:15 am	Course Introduction and Overview
10:15 ï 11:45 am	PSoC Architecture Overview
11:45 ï 1:00 pm	Lunch
1:00 ï 2:00 pm	SW Demo and Architecture Overview Lab
2:00 ï 2:30 pm	System Resources
2:30 ï 3:00 pm	System Resources Lab
3:00 ï 3:30 pm	Break
3:30 ï 4:00 pm	Digital Peripherals
4:00 ï 4:30 pm	Digital Peripherals Lab
4:30 ï 5:00 pm	Analog Peripherals
10:00 ï 10:45 am (Wed.)	Analog Peripherals Lab

Section Objectives

At the end of this section you will be able to

- Understand the major differences between PSoC 1/3/5
- Understand the high-level architecture of PSoC 3/5
- Understand the CPU, Digital, Analog and Programmable Routing / Interconnect Systems of PSoC 5

WHAT IS PSoC®?



__10__ years+ of explosive PSoC growth

Thousands of active PSoC customers

Over One billion PSoC units shipped

PSoC is Everywhere!

(including over 700 universities)

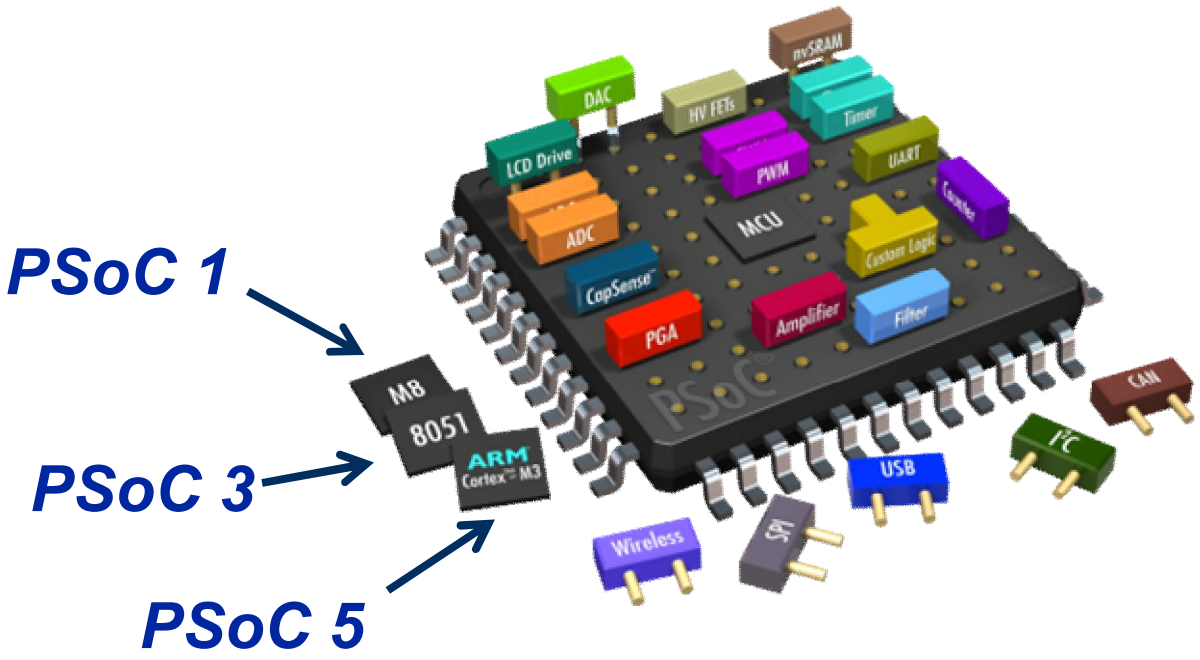
PSoC is a true programmable embedded SoC integrating configurable analog and digital peripheral functions, memory and a microcontroller on a single chip.

PSoC Is Everywhere!

HANDHELD DEVICES	APPLIANCES	INDUSTRIAL	ENTERTAINMENT/ DISPLAYS	SECURITY/ MONITORING
				

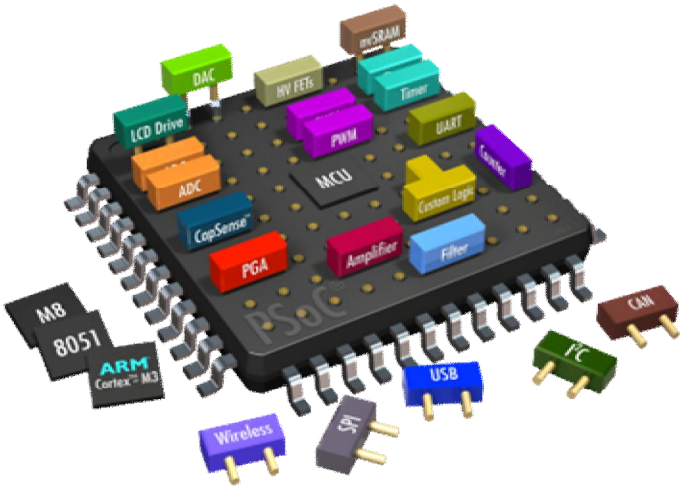
TOYS/GAMING	DIGITAL PHOTOGRAPHY	SPORTS/ FITNESS	COMPUTERS	PRESENTER TOOLS	HOME THEATER
					

PSoC - Future of Embedded Design



PSoC is the world's only programmable embedded SoC integrating configurable analog and digital peripheral functions, memory and a microcontroller on a single chip.

PSoC 1 Device Overview



M8C Microcontroller

Up to 24 MHz, 4 MIPS

Flash Memory

4 KB to 32 KB for program storage

SRAM

256B to 2 KB for data storage

Configurable Analog Functions

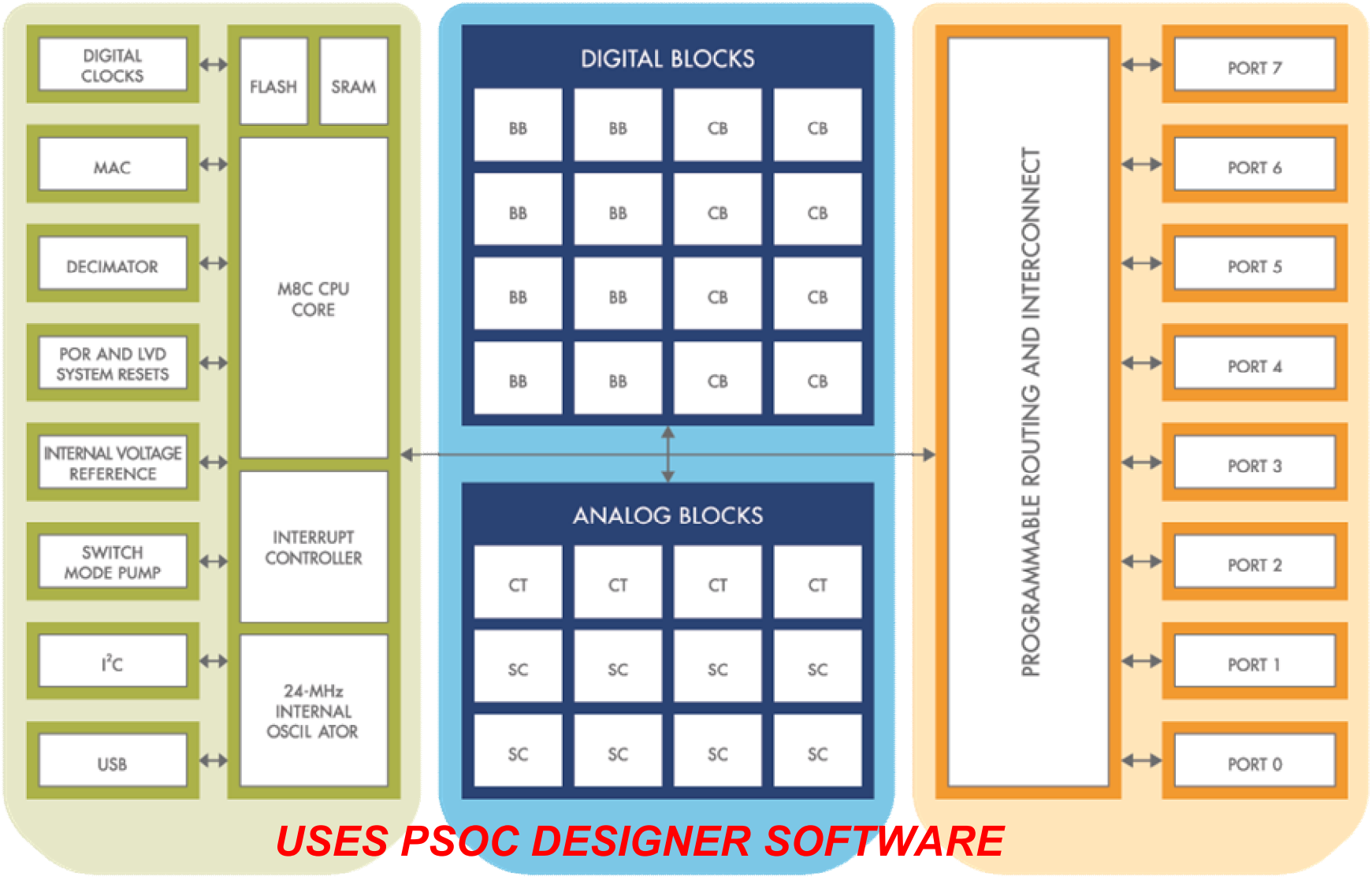
Implement ADCs, DACs, filters, amplifiers, comparators, etc.

Configurable Digital Functions

Implement timers, counters, PWMs, UART, SPI, IrDA, etc.

USES PSOC DESIGNER SOFTWARE

PSoC 1 Architecture

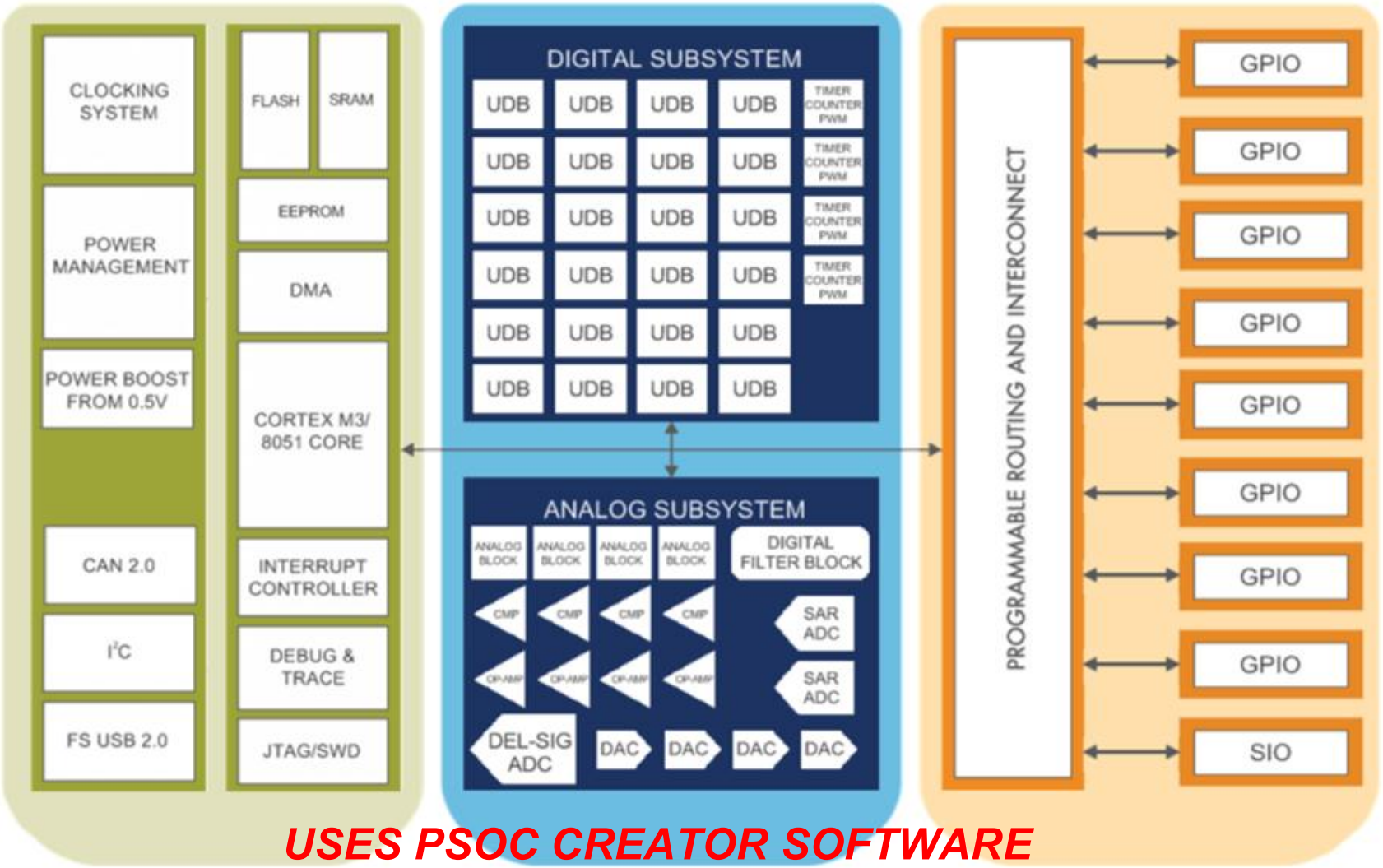


USES PSOC DESIGNER SOFTWARE

INTRODUCTION TO PSOC 3/5

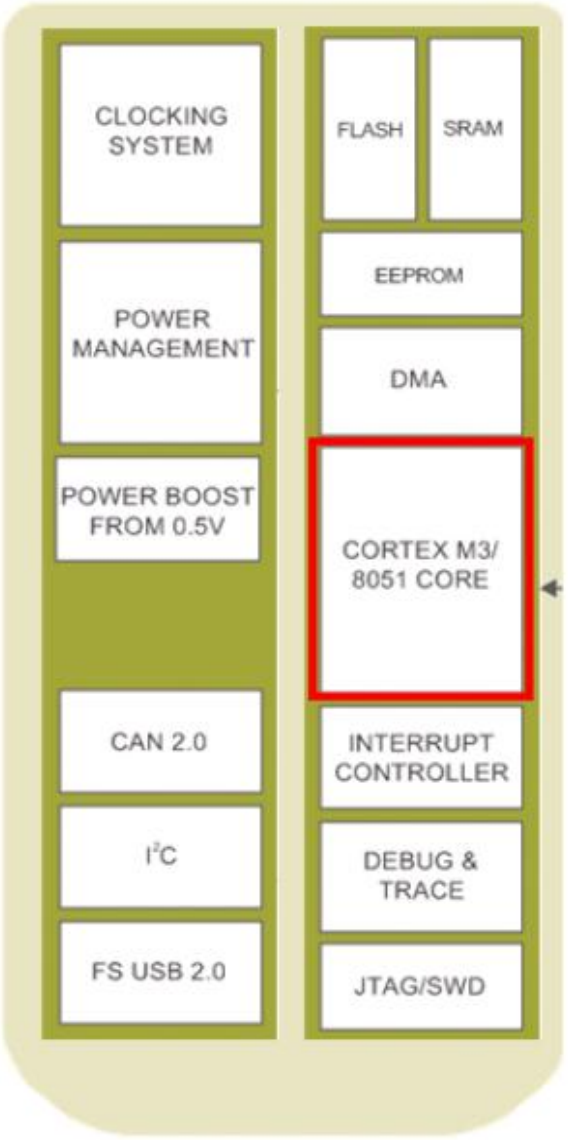
ARCHITECTURE OVERVIEW

PSoC 3 / PSoC 5 Platform Architecture



USES PSOC CREATOR SOFTWARE

CPU Subsystem



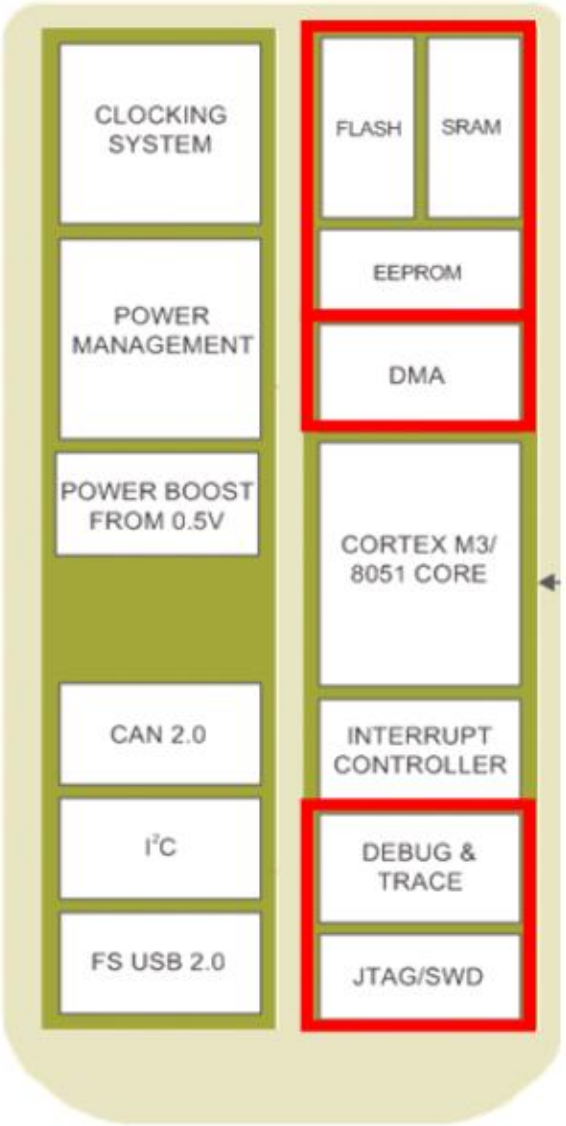
ARM Cortex-M3

- Industryôs leading embedded CPU company
- Broad support for middleware and applications
- Upto 67 MHz; 83 DMIPS
- Enhanced v7 ARM architecture
- Thumb2 Instruction Set
- 16- and 32-bit Instructions (no mode switching)
- 32-bit ALU; Hardware multiply and divide
- Single cycle 3-stage pipeline; Harvard architecture

8051

- Broad base of existing code and support
- Upto 67 Mhz; 33 MIPS
- Single cycle instruction set

CPU Subsystem



High Performance Memory

- Flash Memory with ECC
- High ratio of SRAM to flash
- EEPROM

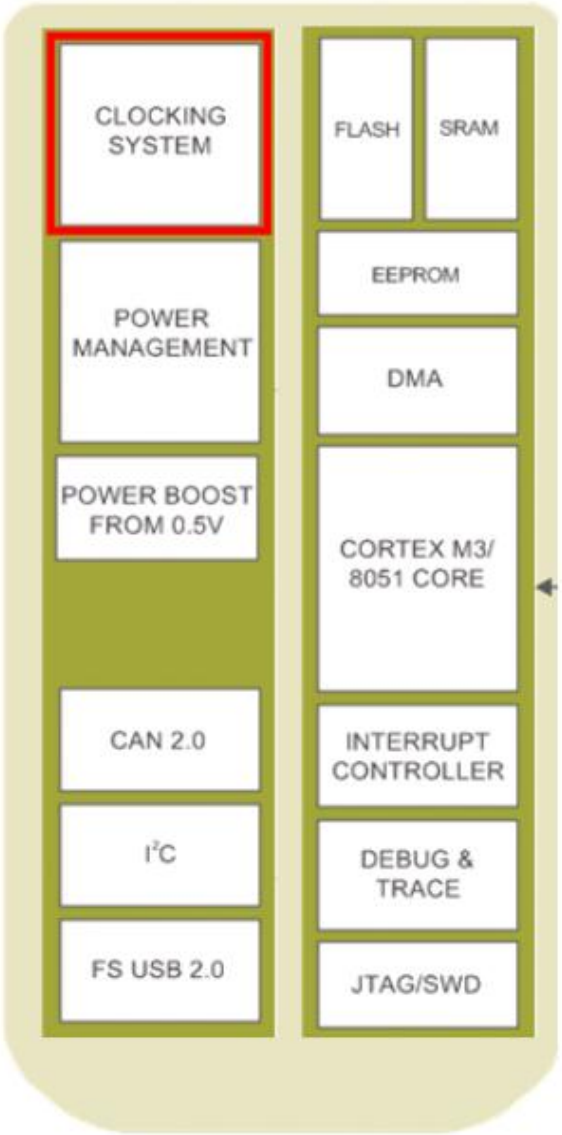
Powerful DMA Engine

- 24-Channel Direct Memory Access
- Access to all Digital and Analog Peripherals
- CPU and DMA simultaneous access to independent SRAM blocks

On-Chip Debug and Trace

- Industry standard JTAG/SWD (Serial Wire Debug)
- On-chip trace

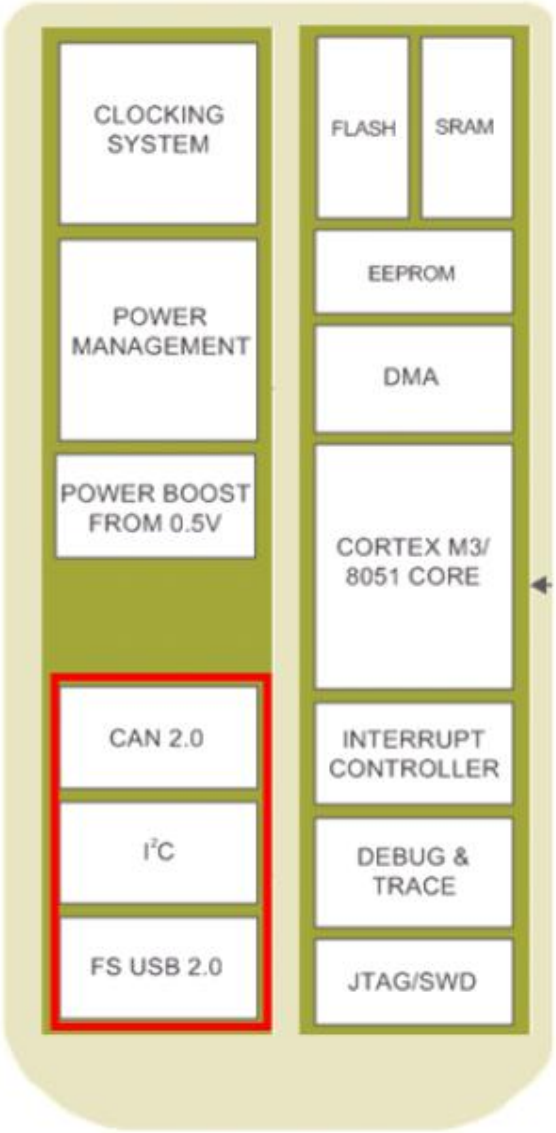
CPU Subsystem



Clocking System

- Multiple Clock Sources
  - Internal Main Oscillator
  - External clock crystal input
  - External clock oscillator inputs
  - Clock doubler output
  - Internal low speed oscillator
  - External 32 kHZ crystal input
  - Dedicated 48 MHz USB clock
  - PLL output
- 16-bit Clock Dividers
  - 8 Digital Domain
  - 4 Analog Domain
- PSoC Creator Configuration Wizard
- PSoC Creator auto-derives clocking sources / dividers

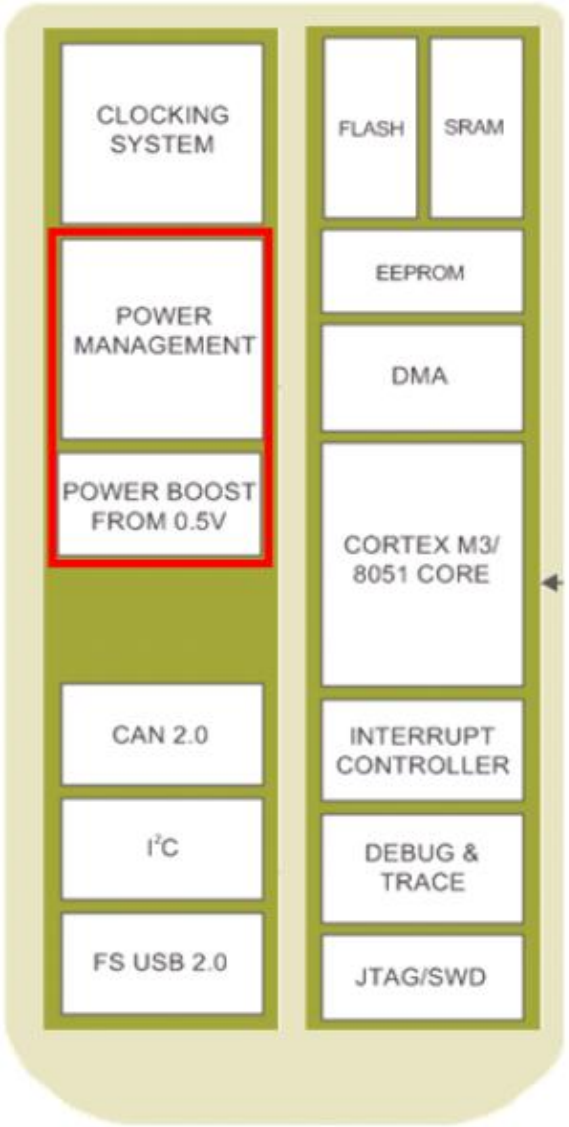
CPU Subsystem



Dedicated Communication Peripherals

- Full Speed USB Device
  - 8 bi-directional data end points + 1 control end point
  - No external crystal required
  - Drivers in PSoC Creator for HID class devices
- Full CAN 2.0b
  - 16 RX buffers and 8 TX buffers
- I2C master or slave
  - Data rate up to 400 kbps
  - Additional I2C slaves may be implemented in UDB array

CPU Subsystem



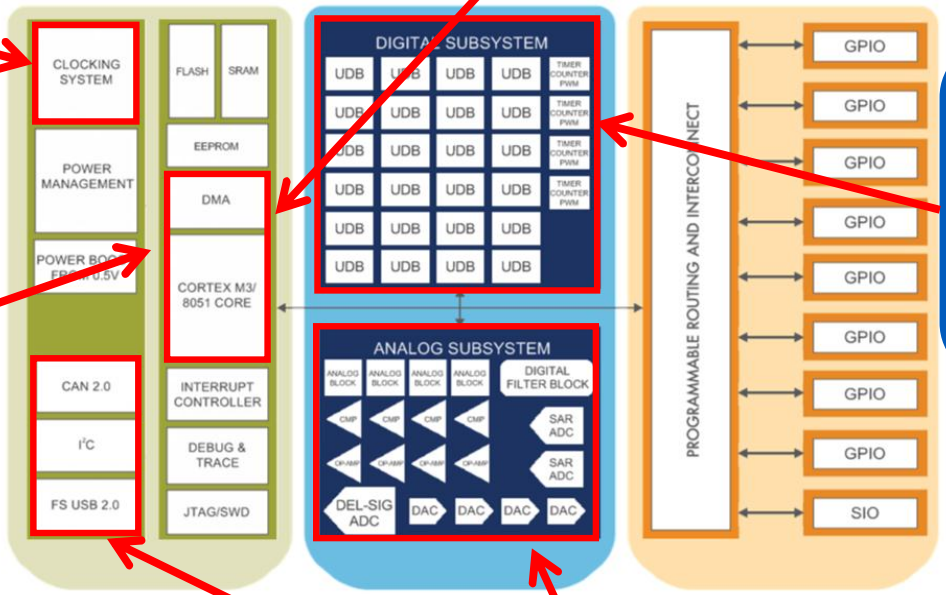
Power Management

- Industryôs Widest Operating Voltage
  - 0.5V to 5.5V with full analog/digital capability
- High Performance at 0.5V
  - PSoC 3 @ 67 MHz
- 4 Power Modes (Active, Alternate Active, Sleep and Hibernate)

Designed for Low Power / Low Voltage

On-board DMA Controller
Direct Memory transfer between peripherals offloads CPU operation, lowering power consumption

Highly configurable clock tree
Flexible, automated clock gating



Universal Digital Blocks
Implement features in hardware that reduce CPU processing requirements, lowering power consumption

Cached Operations
Execution from flash memory is improved by caching instructions (PSoC 5 only)

Precise CPU Frequencies
PLL allows 4032 different frequencies, tunable power consumption

Integrated Analog, Digital and Communications Peripherals
Reduce external component counts and lower overall **system** power consumption

Low Power Modes

PSoC 3

Power Mode	Current	Code Execution	Digital Resources Available	Analog Resources Available	Clock resources Available	Wakeup Sources	Reset Sources
Active	1.2 mA @ 6 MHz	Yes	All	All	All	N/A	All
Sleep	1 uA	No	I2C	Comparator	ILO/ kHzECO	Comparator, PICU,I2C, RTC,CTW, LVD	XRES, LVD, WDR
Hibernate	200 nA	No	None	None	None	PICU	XRES

PSoC 5LP

Power Mode	Current	Code Execution	Digital Resources Available	Analog Resources Available	Clock resources Available	Wakeup Sources	Reset Sources
Active	3.1 mA @ 6 MHz	Yes	All	All	All	N/A	All
Sleep	2 uA	No	None	None	ILO	CTW	XRES
Hibernate	300 nA	No	None	None	None	N/A	XRES

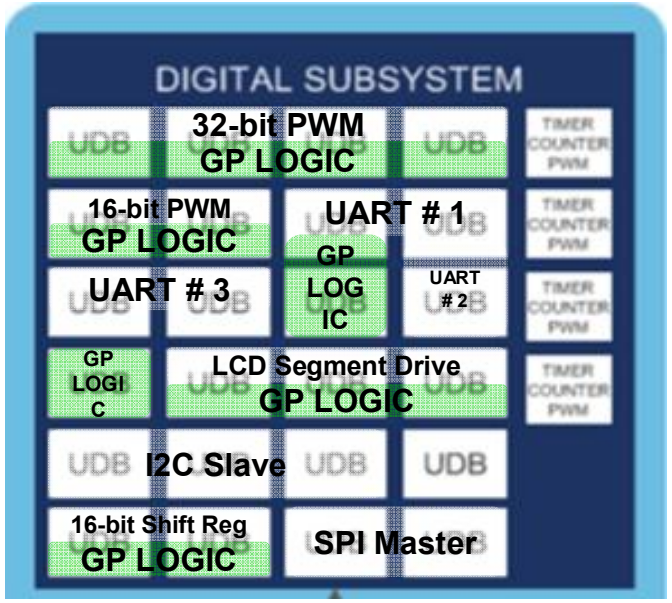
Power Management enabled in PSoC Creator

- Provides easy to use control APIs for quick power management
- Allows code and register manipulation for in-depth control

Digital Subsystem

Universal Digital Block Arrays (UDBs)

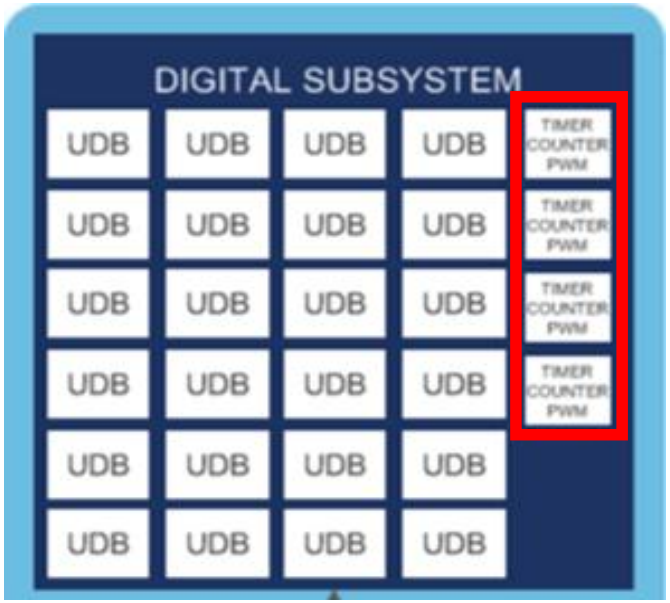
- Flexibility of a PLD integrated with a CPU
- Provides hardware capability to implement components from a rich library of pre-built, documented and characterized components in PSoC Creator
- PSoC Creator will synthesize, place and route components automatically as well as provide static timing analysis
- Fine configuration granularity enables high silicon utilization
- DSI routing mesh allows any function in the UDBs to communicate with any other on-chip function/GPIO pin with 8- to 32-bit data buses



Digital Subsystem

Organized 8/16-bit Timer/Counter/PWM Blocks

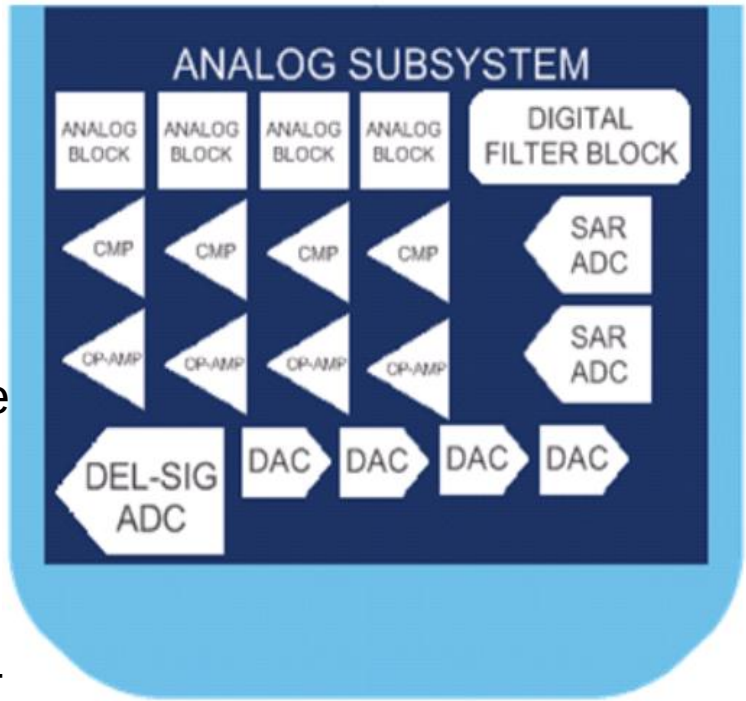
- Provides nearly all of the features of a UDB based timer, counter or PWM
- PSoC Creator provides easy access to these flexible blocks
- Each block may be configured as either a full featured 8-bitTimer, Counter or PWM. Two blocks may be combined to make it 16-bit
- Programmable options
  - Clock, enable, reset, capture, kill from any pin or digital signal on chip
  - Independent control of terminal count, interrupt, compare, reset, enable, capture and kill synchronization
- Plus
  - Configurable to measure pulse-widths or periods
  - Buffered PWM with dead band and kill



Analog Subsystem

Configurable Analog System

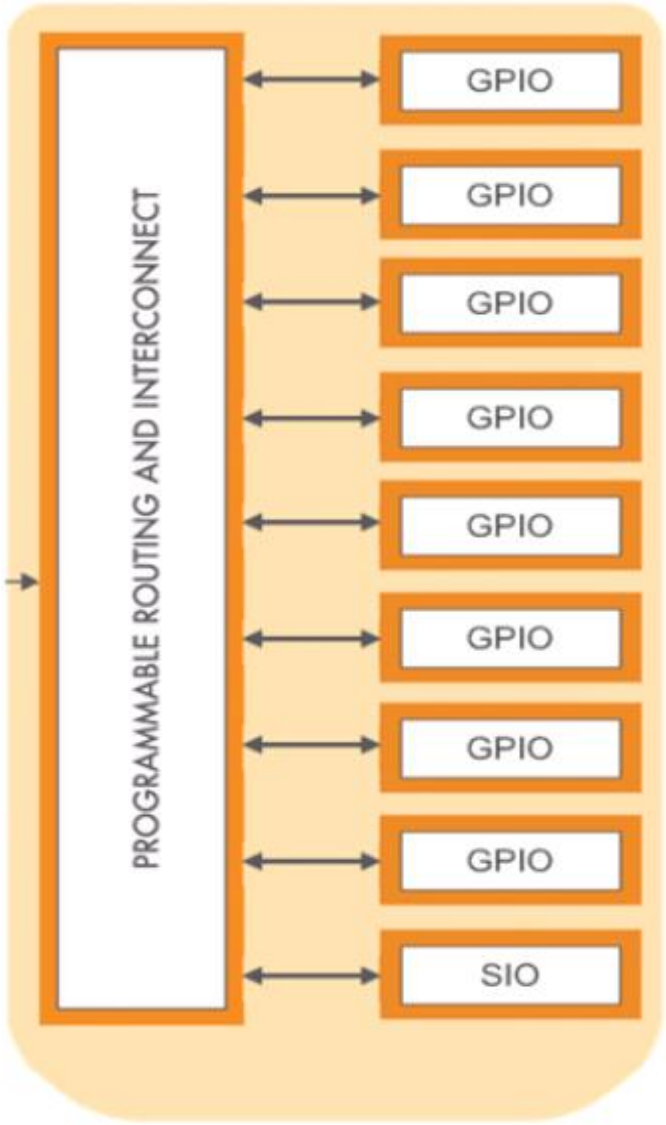
- Flexible Routing: All GPIO are Analog Input/Output
- +/- 0.1% Internal Reference Voltage
- Delta-Sigma ADC: Up to 20-bit resolution
  - 16-bit at 48 ksps or 12-bit at 192 ksps
- SAR ADC: 12-bit at 700 ksps
- DACs: 8-bit resolution, current and voltage mode
- Low Power Comparators
- Opamps (25 mA output buffers)
- Programmable Analog Blocks
  - Configurable PGA (up to X50), Mixer, Trans-Impedance Amplifier, Sample and Hold
- Digital Filter Block: Implement HW IIR and FIR filters
- CapSense Touch Sensing enabled



Programmable Routing/Interconnect

Input / Output System

- Three types of I/O
  - GPIO, SIO, USBIO
- Any GPIO to any peripheral routing
- Wakeup from sleep on analog, digital or I2C events
- Programmable slew rate reduces power and noise
- Eight different configurable drive modes
- Programmable input threshold capability for SIO
- Automatic and custom/lock-able routing in PSoC Creator



Four separate I/O voltage domains

- Interface with multiple devices using one PSoC 3 / PSoC 5 device

Review

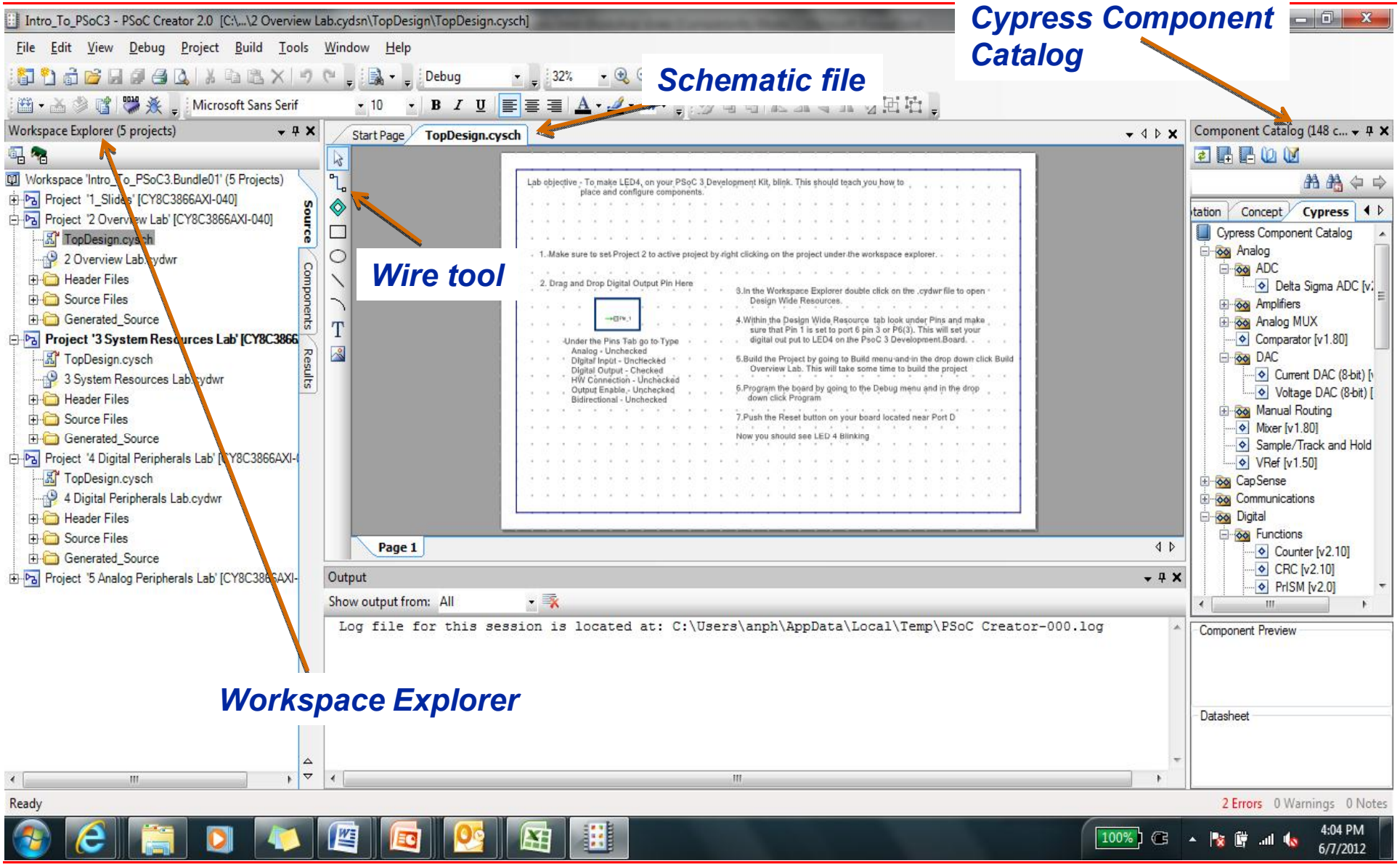
You should now be able to:

- Understand PSoC Architecture
- Understand the CPU, Digital, Analog and Programmable Routing and Interconnect Subsystems

INTRODUCTION TO PSOC 3 AND PSOC 5

ARCHITECTURE OVERVIEW LAB

PSoC Creator 2.0 Interface

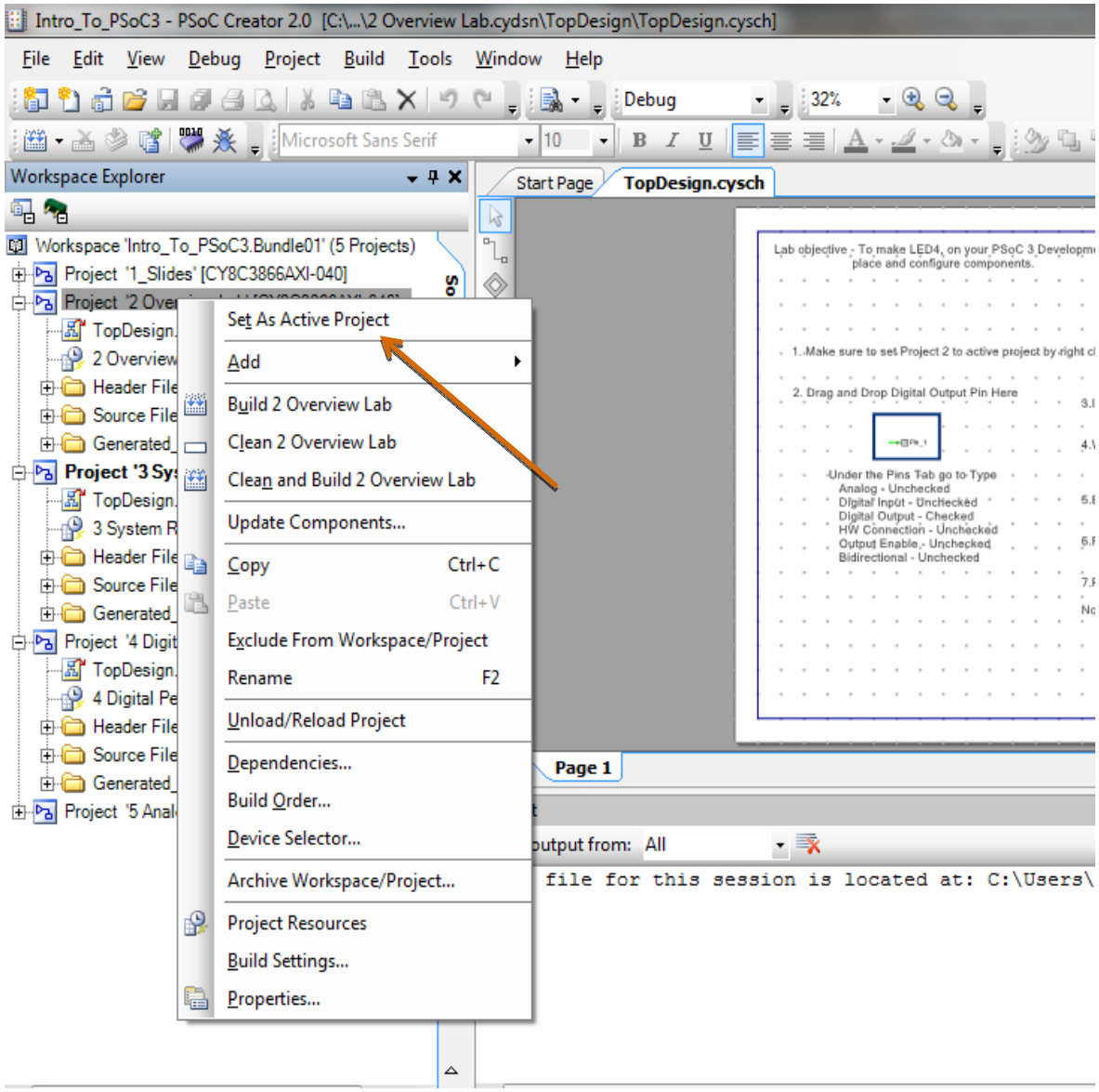


Architecture Overview Lab

Lab Objective

- To make LED4 on your PSoC 3 Development Kit blink.
- To learn how to place and configure components in PSoC Creator

Architecture Overview Lab

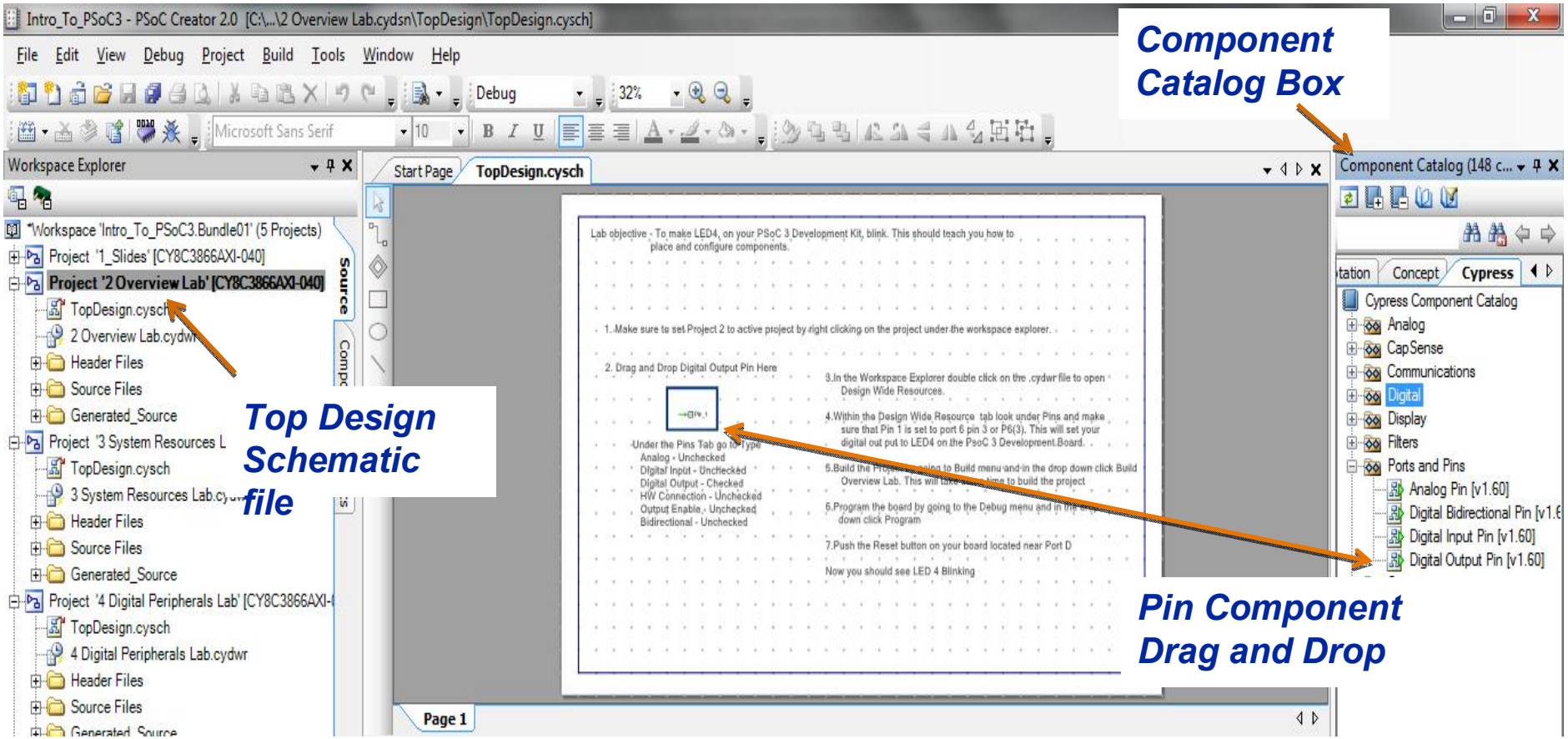


1. Open LAB 1 from the flash drive

2. Expand the ñ+òsign to the left of Project 2 title to view project files. Double-click the schematic file ñTopDesign.cyschòto open it

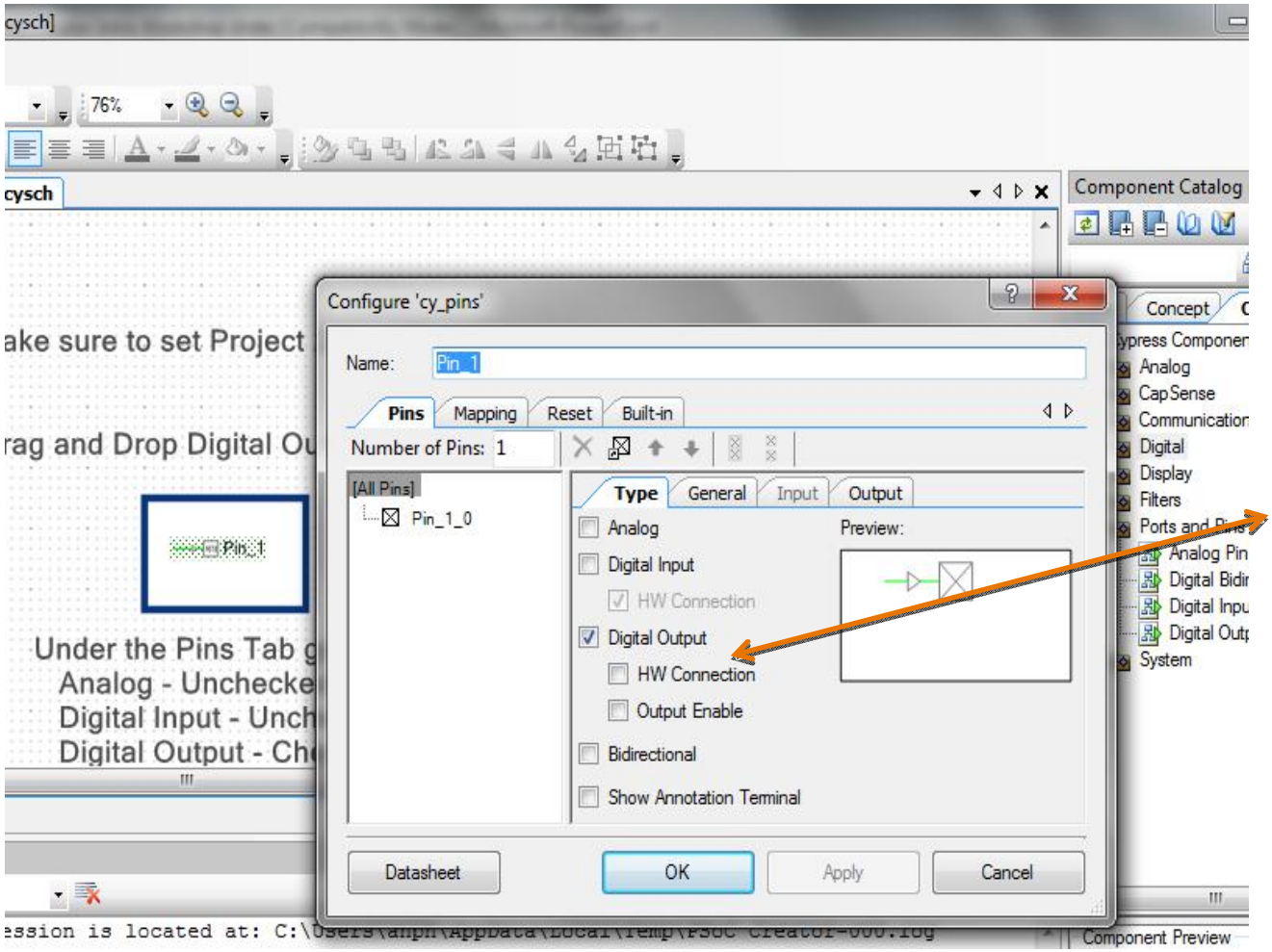
Architecture Overview Lab

3. From the ñComponent catalogò on the right side of the screen, drag & drop ñDigital Output Pinò under ñPorts and Pinsò into the box (as shown above)



Architecture Overview Lab

4. Double-click the component to open it in configuration mode and check the configuration as follows:



Analog -
Unchecked

Digital Input ï
Unchecked

Digital Output ï
Checked

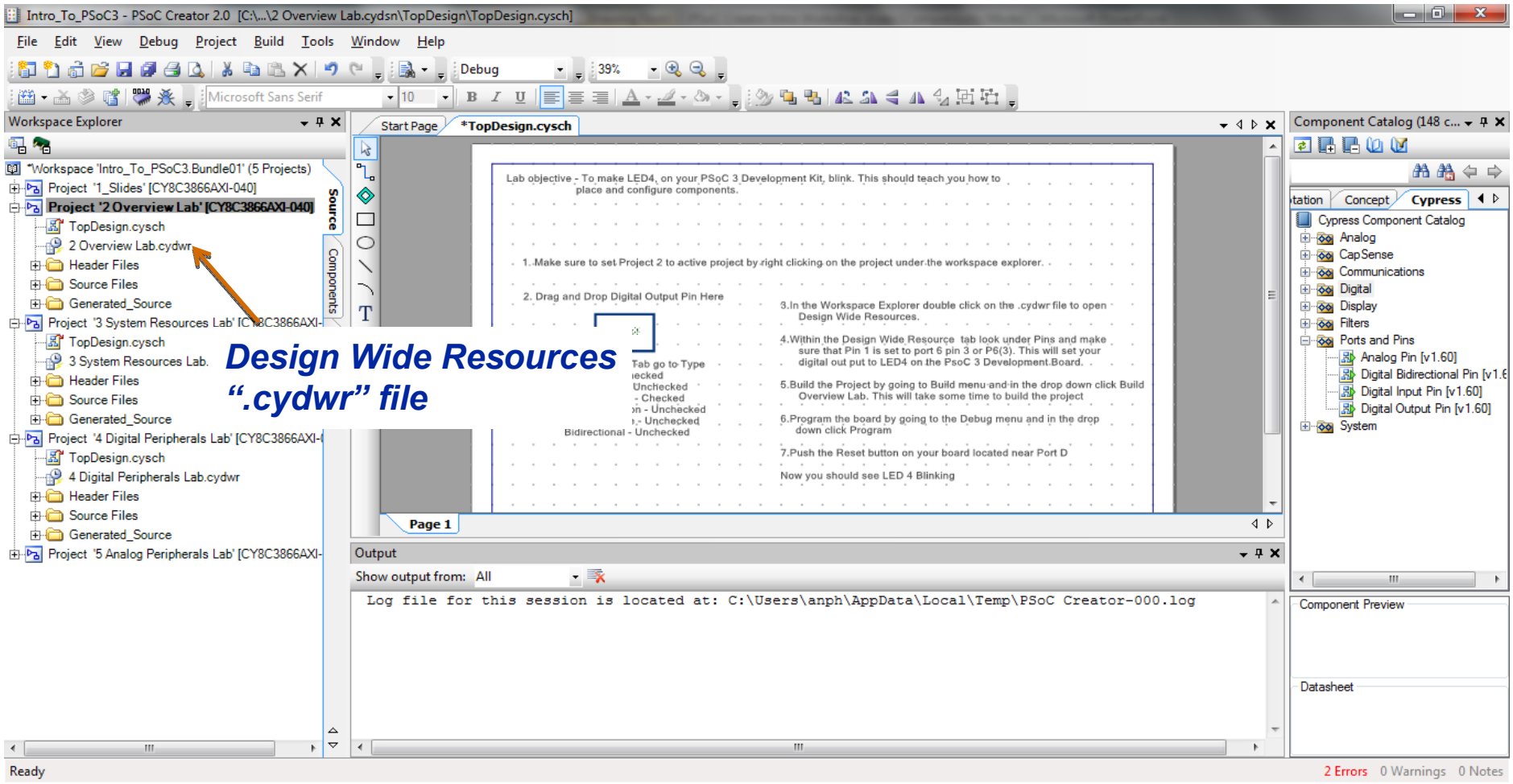
HW Connection ï
Unchecked

Output Enable ï
Unchecked

Bidirectional ï
Unchecked

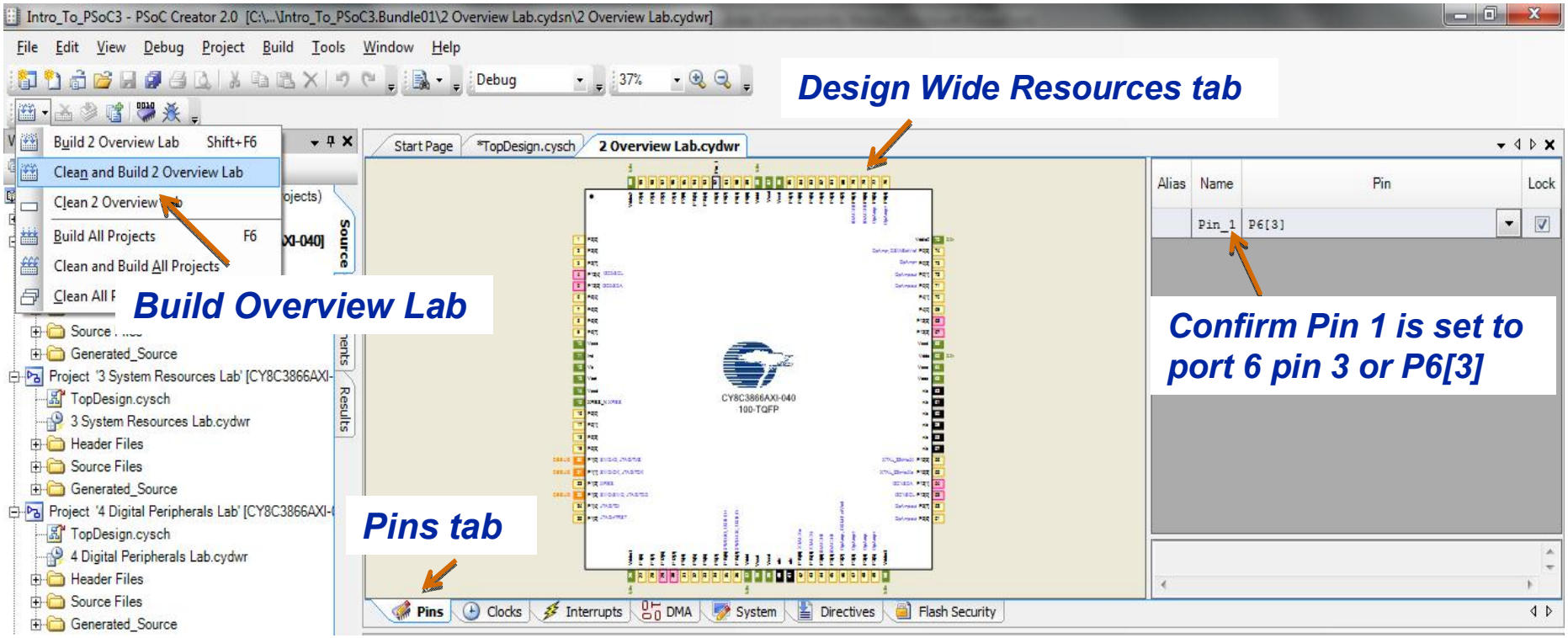
Architecture Overview Lab

5. In the Workspace Explorer double click on the .cydwr file to open Design Wide Resources



Architecture Overview Lab

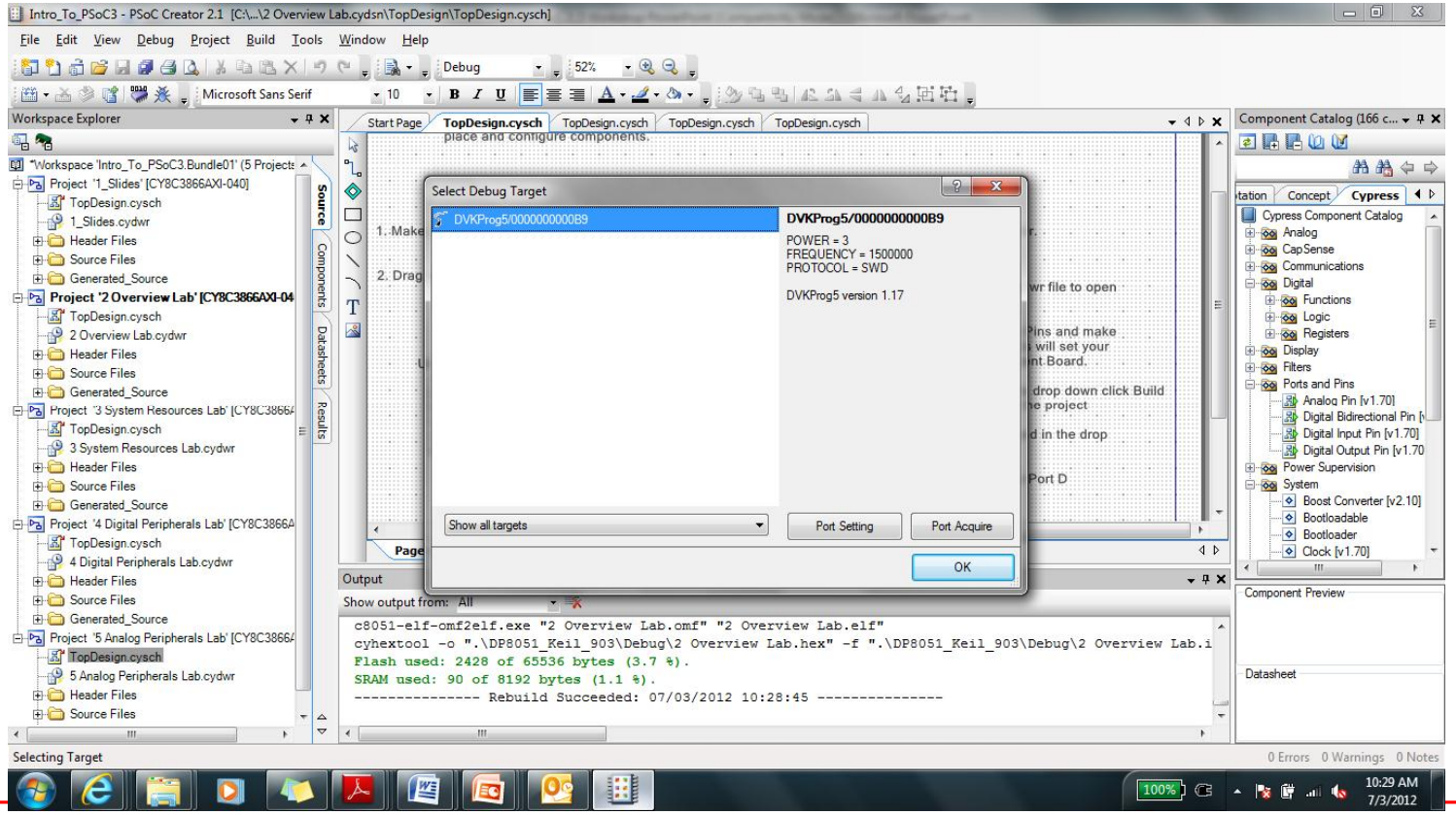
6. Within the Design Wide Resources tab, select the Pins tab (below the chip)
7. On the right hand side of the screen, make sure that Pin 1 is set to port 6 pin 3 or P6[3]. This will set your digital output to LED4 on the PSoC Development Board.
8. Build the Project by going to the Build menu and in the drop down click ñBuild Overview Labò. This will take a minute to build the project.



Architecture Overview Lab

9. Program the board by going to the Debug menu and click Program from the drop down list. Programming should take just a minute. You may have to select your kit and follow the steps to click on ñPort Acquireò
10. Push the Reset button on your board located near Port D

Verify that you see LED 4 Blinking.

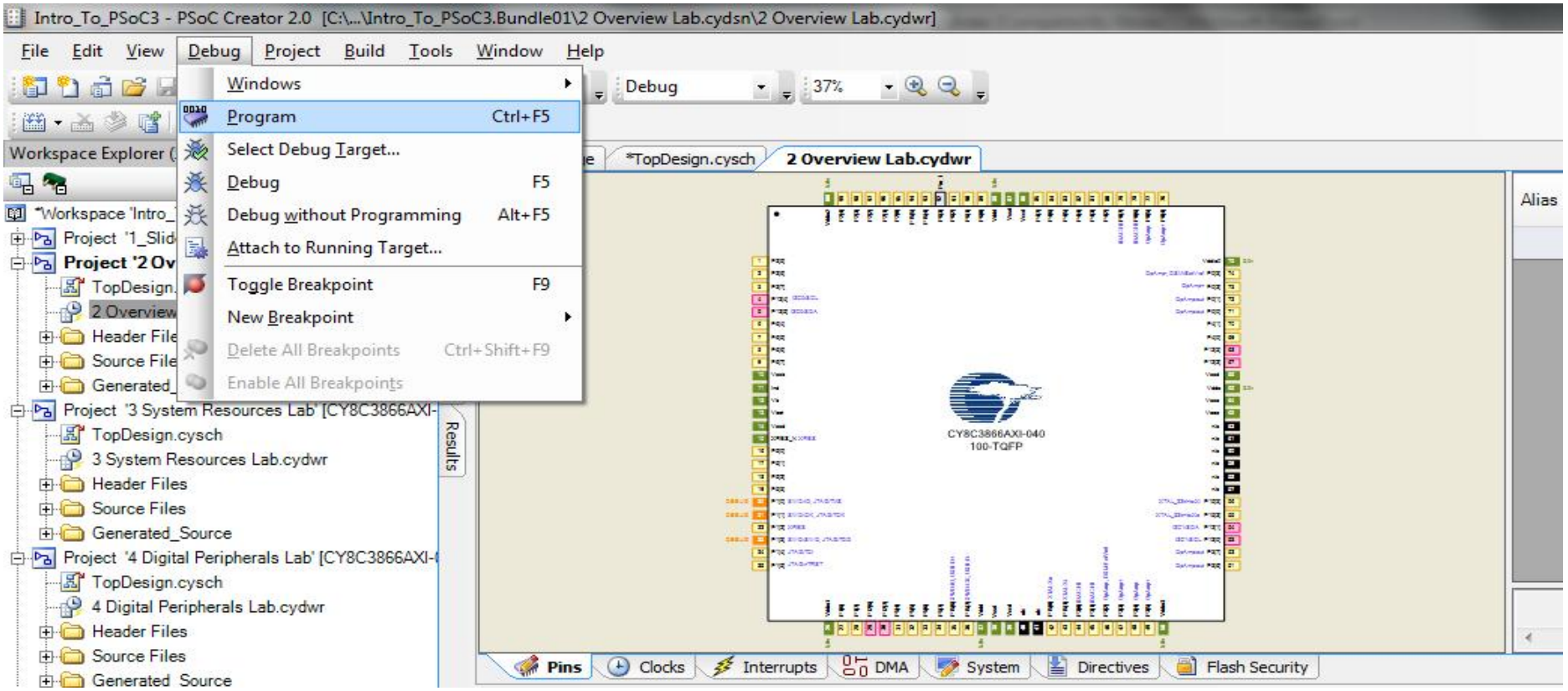


Architecture Overview Lab

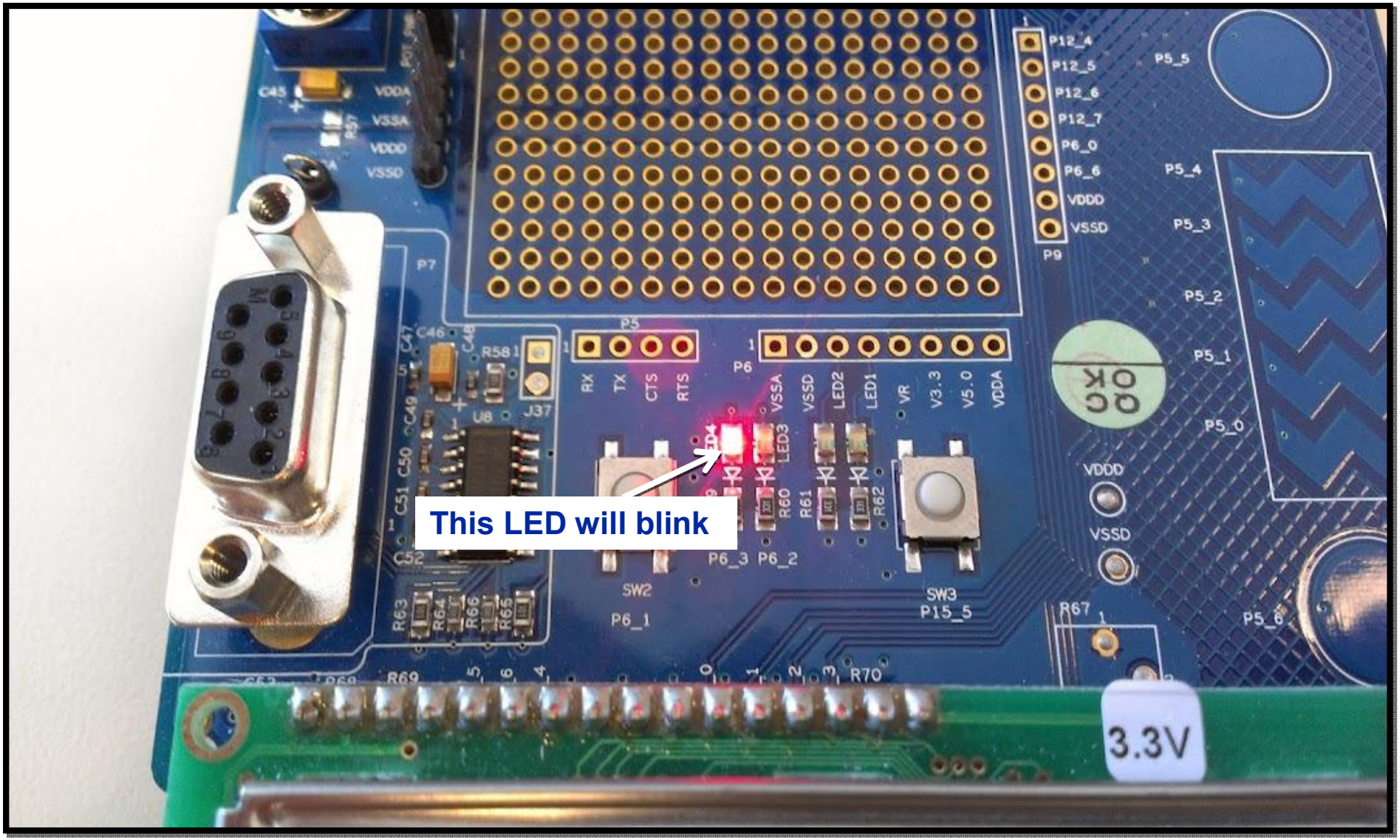
9. Program the board by going to the Debug menu and click Program from the drop down list. Programming should take just a minute. You may have to select your kit and follow the steps to click on ñPort Acquireò

10. Push the Reset button on your board located near Port D

Verify that you see LED 4 Blinking.



Architecture Overview Lab



INTRODUCTION TO PSOC 3 AND PSOC 5

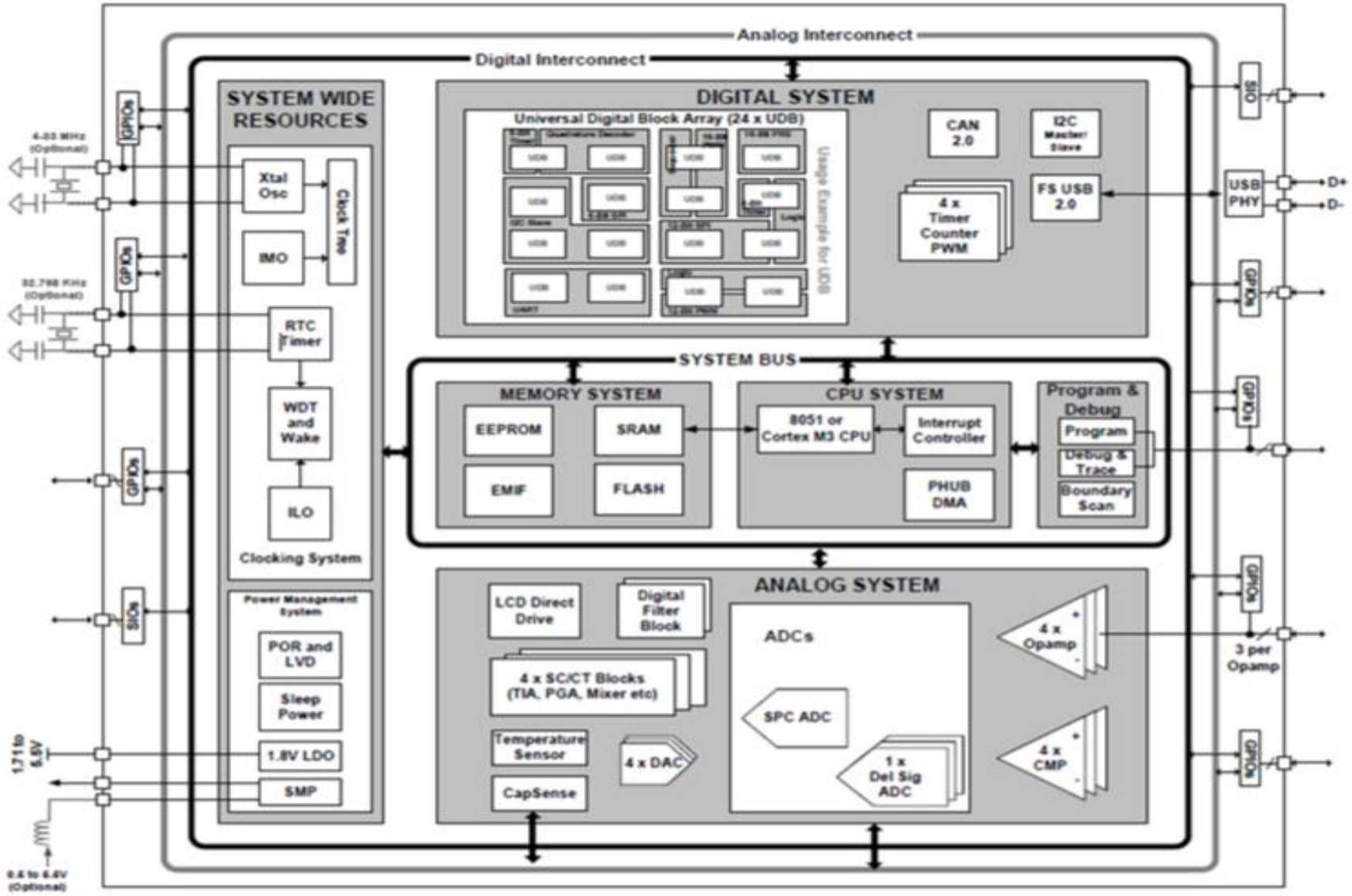
SYSTEM RESOURCES

Section Objectives

At the end of this section you will be able to

- Understand the system block diagram of PSoC 3 / PSoC 5 devices
- Understand and use the PSoC 3 / PSoC 5 System Resources, including:
  - Power System
  - Programming and debugging
  - Configuration and boot process
  - Resets
  - Clocking
  - Memory and Mapping
  - DMA and PHUB
  - I/O
  - Interrupts

System Block Diagram



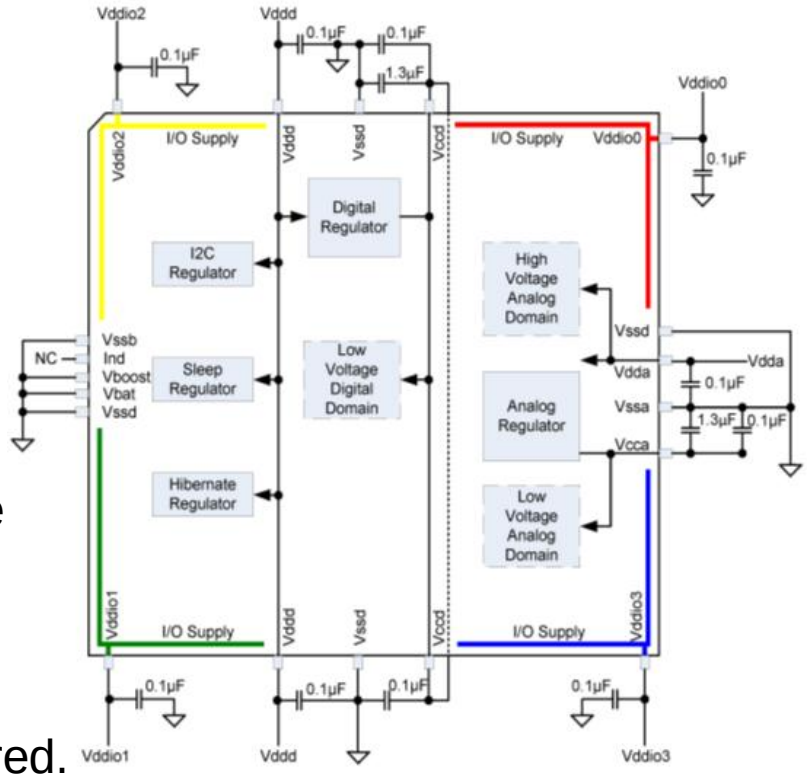
Power System and Supplies (no boost)

Standard Power Configuration

- No boost pump
- Vdda Vddd >= Vddio0/1/2/3
- Vdda = 1.8 ï 5.5V

Supply Rules and Usage

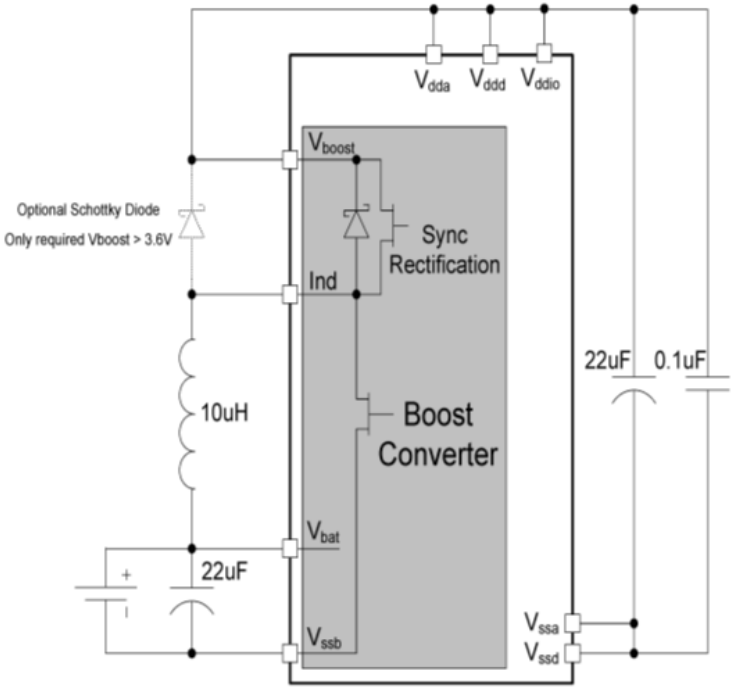
- Vdda: Must be highest voltage in system. Supplies analog high voltage domain and core regulator
- Vddd: Supplies digital system core regulators
- Vcca: Output of the analog core regulator. External 1.1uF capacitance to ground is required.
- Vccd: Output of the digital core regulator. External 1.1uF capacitance to ground is required. Both Vccd pins must be tied together on the PCB and share the 1.1uF capacitance to ground
- Vddio0/1/2/3: Independent I/O supplies. May be any voltage in the range of 1.8V to Vdda



Power System (with boost) (Only PSoC 3)

Boost Converter Configuration

- Used to generate up to 5.0 V (Vout)
- Battery voltage as low as 0.5 V(Vbat)
- Output voltage and current limit based on input voltage and boost ratio
- 75 mA max current
- 0.5 ï 0.8 V Vbat provides max of 1.95 V Vout
- Schottky diode required when Vout is > 3.6V
- Synchronous rectification maximizes efficiency
- Boost may be used to power external circuits
- Only supported for PSoC 3



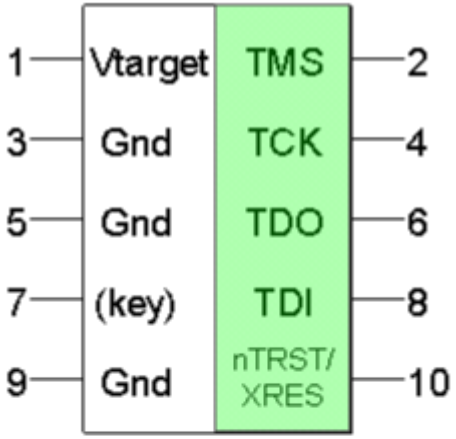
If boost not used

- Vssb, Vbat and Vboost must be tied to ground
- Ind left floating

Programming and Debug Interfaces

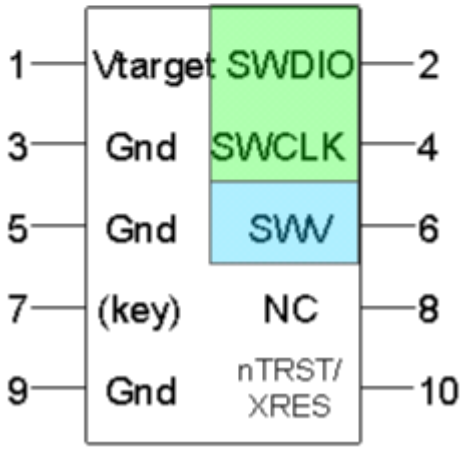
JTAG

- Legacy 4-wire Interface
- Supports all programming and debug features



Serial Wire Debug (SWD)

- Standard 2-wire interface for all CY tools and kits
- Supports all programming and debug features wth same performance of JTAG
- Default debug interface in PSoC Creator



Programming and General Features

Flash operations

- Erase all
- Erase block ï 256 blocks per device, independent of Flash size
- Program block
- Set block security
  - Unprotected ï No protection
  - Factory Upgrade ï Prevents external read
  - Field Upgrade ï Prevents external read and write
  - Full Protection ï Prevents external read and write as well as internal write

General Features available through JTAG/SWD

- I/O boundary scan through JTAG interface
- Enable/Disable JTAG and SWD interfaces
- On Chip Debug features enabled/disabled by firmware

Reset Sources

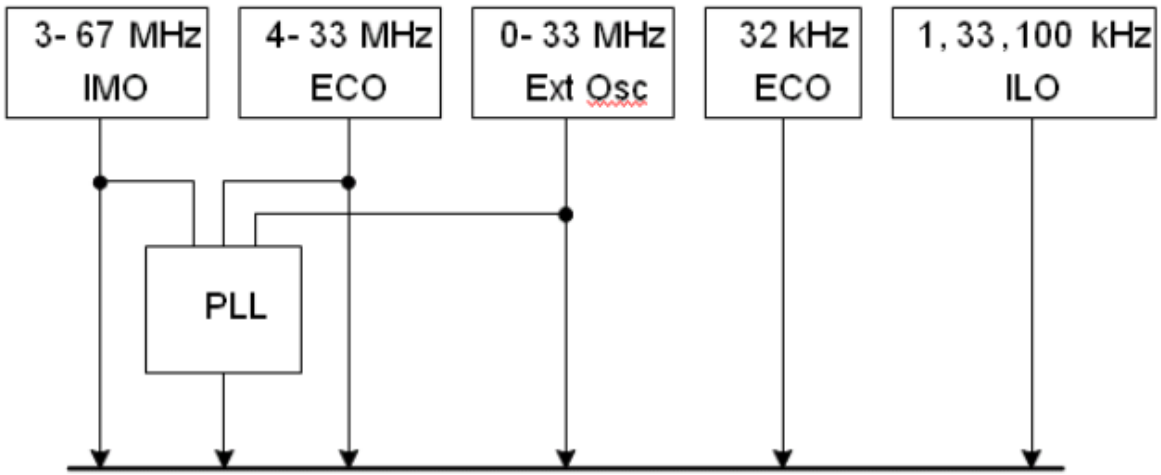
- PPOR ï Power on Reset
- XRES ï External Reset
- PRES ï Under Voltage on external supplies Vddd, Vdda (Precise Low Voltage Reset)
- PRES ï Under Voltage on internal supplies Vccd, Vcca
- AHVI ï Over Voltage on Vdda (Analog High Voltage Interrupt)
- HRES ï Hibernate mode under voltage detect
- SRES ï User software and/or hardware generated reset
- WRES ï Watchdog Timer reset

- JTAG or SWD interface generated reset

Clocking Sources

- Internal Main Oscillator (PSoC 3): 3-62 MHz (+/-1% at 3 MHz; +/-7% at 62 MHz)
- Internal Main Oscillator (PSoC 5): 3-74 MHz (+/-1% at 3 MHz; +/-7% at 74 MHz)
- PLL Output: 12.67 MHz (can not use 32 KHz crystal)
- External clock crystal input: 4-33 MHz
- External clock oscillator inputs: 0-33 MHz
- Clock doubler output: 12-48 MHz
- Internal Low speed Oscillator: 1 kHz, 33 kHz and 100 kHz

- External 32 kHz crystal input for RTC

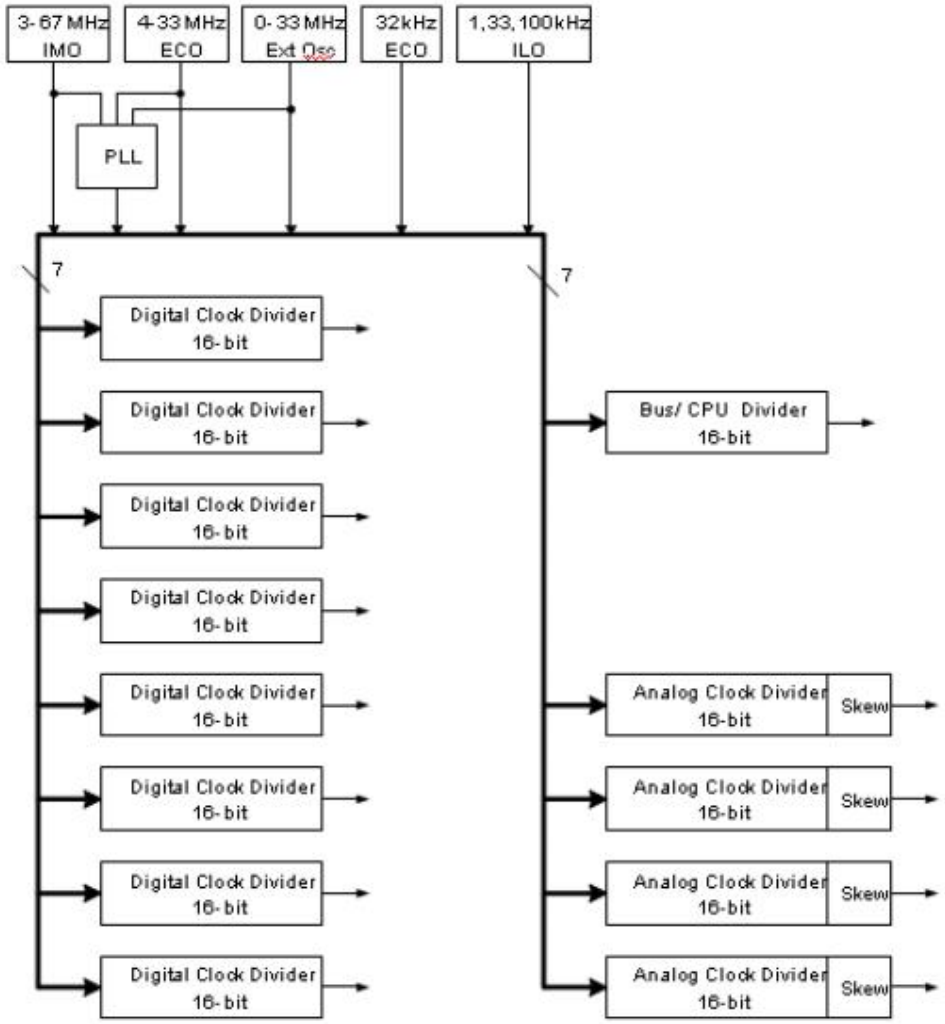


Clock Distribution

Clock Dividers

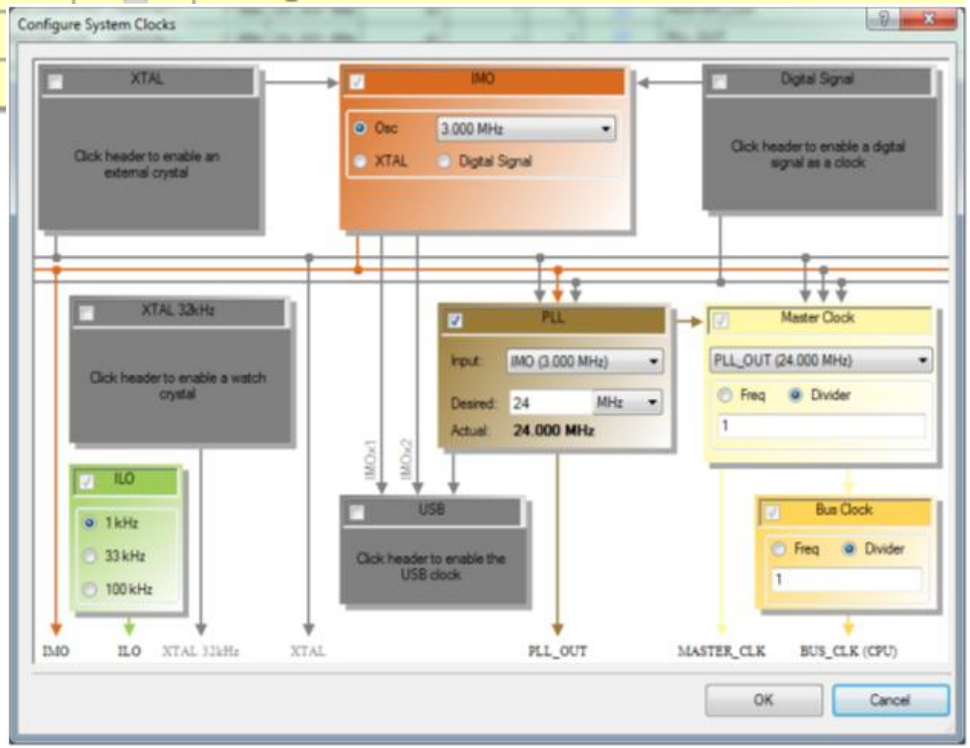
- 16-bit dividers
- 8 clock source inputs
- 8 digital domain clock dividers
- 4 analog domain clock dividers
  - Provide skew control to reduce digital switching noise
- 1 CPU divider

UDB's can be used to create additional digital clocks



System Clock Setup

Type	Name	Domain	Desired Frequency	Nominal Frequency	Accuracy (%)	Tolerance (%)	Divider	Start on Reset	Source Clock
System	USB_CLK	DIGITAL	48.000 MHz	? MHz	±0	-	1	☐	IMOx2
System	Digital_Signal	DIGITAL	? MHz	? MHz	±0	-	0	☐	
System	XTAL_32KHZ	DIGITAL	32.768 kHz	? MHz	±0	-	0	☐	
System	XTAL	DIGITAL	33.000 MHz	? MHz	±0	-	0	☐	
System	ILO	DIGITAL	? MHz	1.000 kHz	-50, +100	-	0	☑	
System	IMO	DIGITAL	3.000 MHz	3.000 MHz	±1	-	0	☑	
System	BUS_CLK (CPU)	DIGITAL	? MHz	24.000 MHz	±1	-	1	☑	MASTER_CLK
System	MASTER_CLK	DIGITAL	? MHz	24.000 MHz	±1	-			
System	PLL_OUT	DIGITAL	24.000 MHz	24.000 MHz	±1	-			



Easy to configure clock options using graphical configuration tool

Clock Management

Clocks allocated to dividers in clock tree

Clocks have software APIs to dynamically change frequency

Note: Reuse existing clocks to preserve resources

Double click clock to see component window

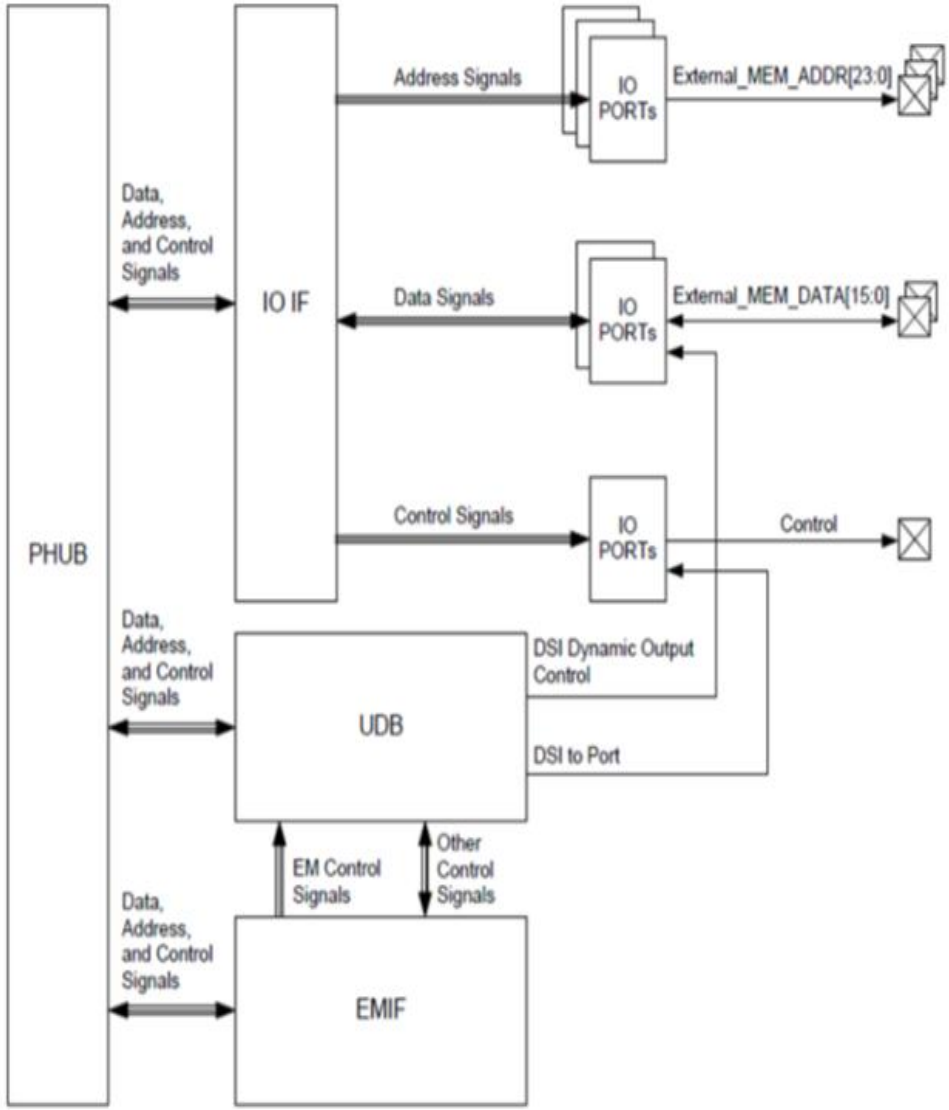
Clock Distribution

EMIF Supports:

- Sync SRAM
- Async SRAM
- Cellular RAM
- NOR Flash

EMIF Usage:

- Data only
- 8- or 16-bit data bus
- 8-, 16- or 24-bit address bus
- Max throughput 11-16 MHz depending on configuration details



Flash

Flash Blocks

- 256 blocks in all devices ï 64 KB flash has 256-byte block size
- Each block may be set to 1 of 4 protection levels of increasing security
  - Å Unprotected ï Allows internal and external reads and writes
  - Å Factory Upgrade ï Prevents external read
  - Å Field Upgrade ï Prevents external read and write
  - Å Full Protection ï Prevents external read and write as well as internal write
- Flash is erased and programmed in block units

Specs

- Code executes out of Flash
- Flash-writes block CPU unless executing from cache
- 20 year minimum retention
- 10 K minimum endurance
- 15 ms block erase + write time

Error Correcting Code (ECC)

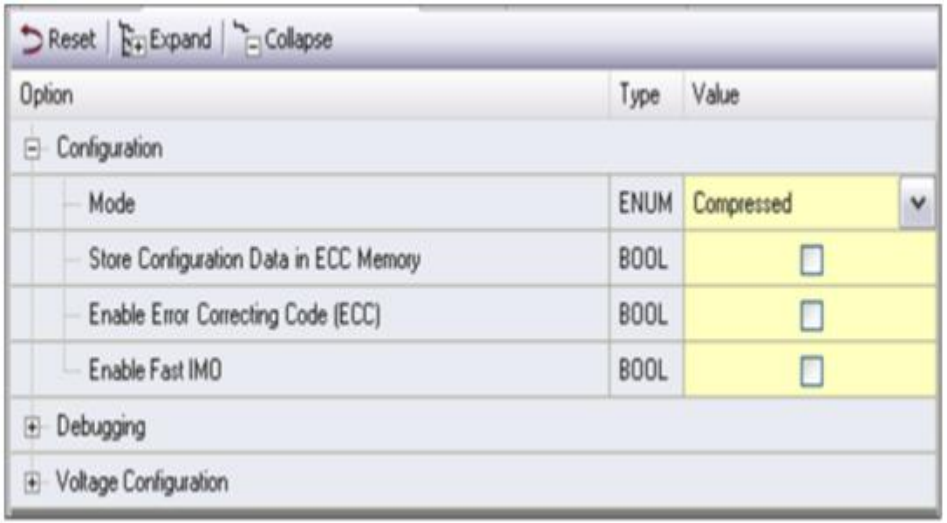
ECC = Flash Memory Error Correction

- Required for some high reliability designs (e.g. automotive and medical)
- Detects and corrects 1 bit of error per 8 bits
- Detects but does not correct 2 bits of error
- Correction is automatic; interrupt and flag bit are set
- 1 byte of ECC data for each 8 bytes of Flash data (1 row)
- 64 KB device includes + 8 KB of ECC memory for 72 KB total

8 KB is used for configuration data storage if ECC not used (default)

- ECC memory is mapped into contiguous region in peripheral space
- ECC memory may also hold user data
- Code cannot execute out of ECC memory

Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	ECC
Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	ECC
Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	ECC
Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	ECC
Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	ECC
Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	ECC



EEPROM

2 KB of EEPROM are provided

Code cannot execute out of EEPROM

EEPROM Specs:

- EEPROM writes do not block CPU execution
- 20 year minimum retention
- 100K minimum endurance
- 2 ms single byte erase + write time
  - Å Supports single byte erase and writes (read / modify / write row)
  - Å May erase or write up to 16 consecutive bytes (1 row) at the same time

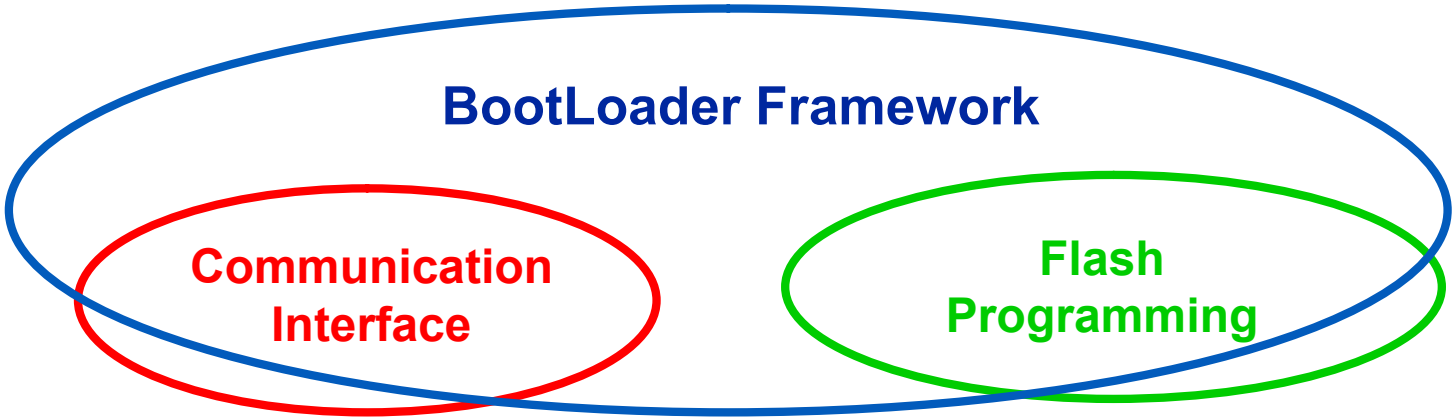
Bootloaders

Single Bootloader Supports

- I2C
- UART
- USB
- Others as required

Bootloader Integration

- Bootloader platform allows easy customization
- No bootloader programmed in parts at factory
- PSoC Creator integrates bootloader support seamlessly; just another component



Direct Memory Access (DMA)

- 24 hardware channels
- 8 priority levels with minimum bandwidth guarantees

Priority Level	Minimum Guaranteed Bus Bandwidth
0	100%
1	100%
2	50%
3	25%
4	12.50%
5	6.30%
6	3.10%
7	1.50%

- 128 Transaction Descriptors (TD) tell channel what to do
  - Å 2kB of dedicated SRAM holds all TD data
- Multiple channels or TDs may be chained or nested
- Configurable burst size
- DMA between peripherals on same spoke limited to 1-byte burst length

GPIO - I/O Digital Features

Independent supply rails

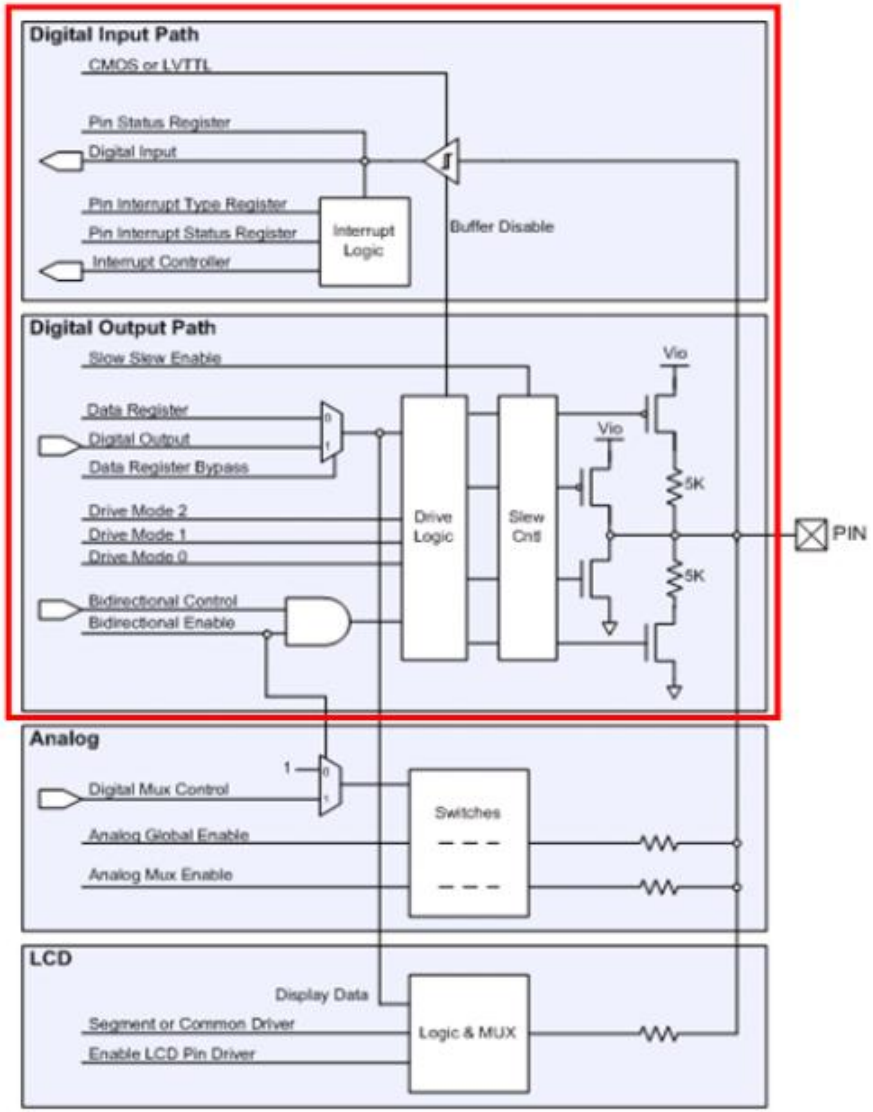
- Each quadrant of device has separate Vddio supply
- GPIO Vddio must be <=Vdda

Logic level maximum current

- 8 mA sink
- 4 mA source

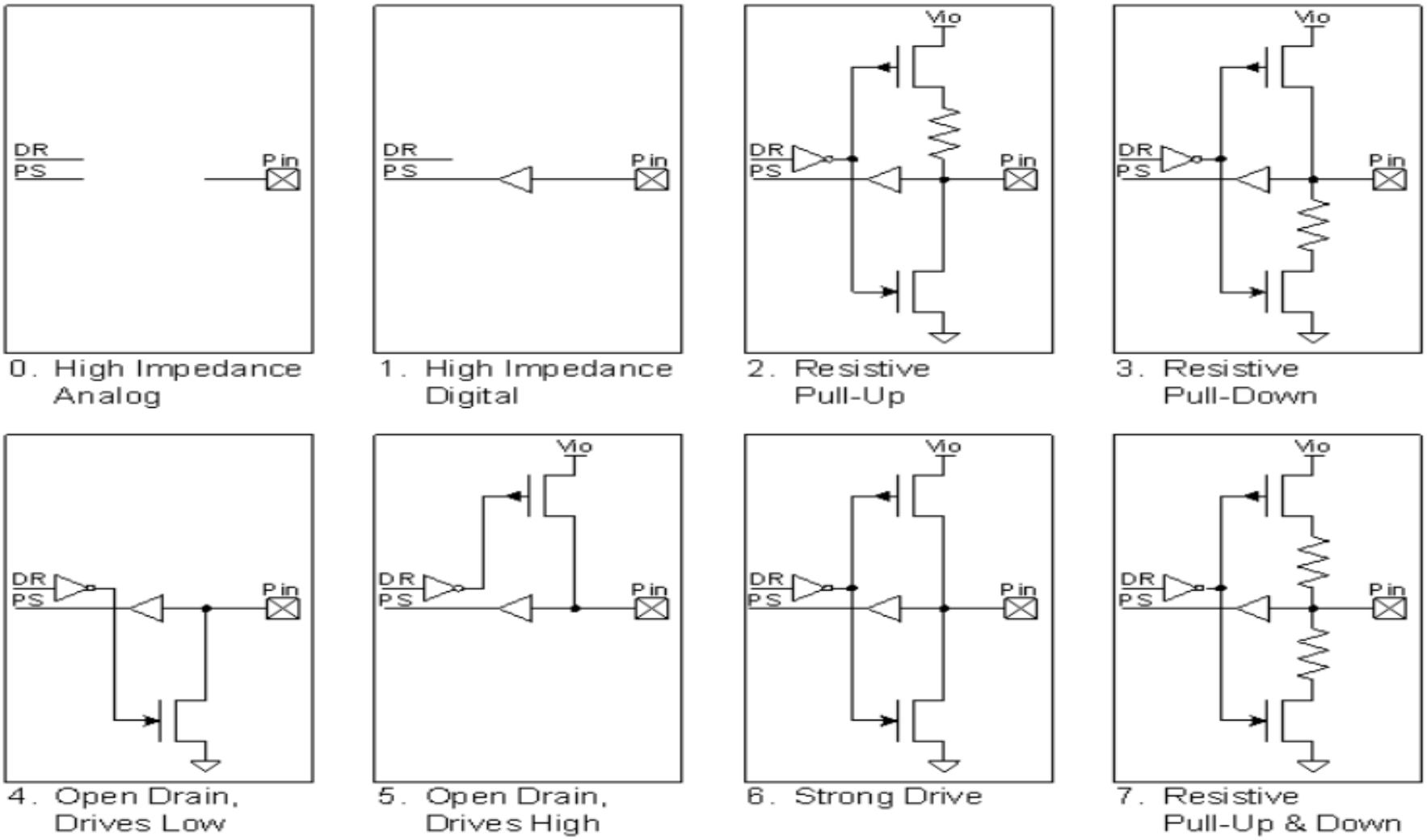
Pin maximum current

- ~25 mA sink
- ~25 mA source



GPIO - I/O Digital Features

8 Drive Modes



GPIO - Interrupts

Each GPIO port has:

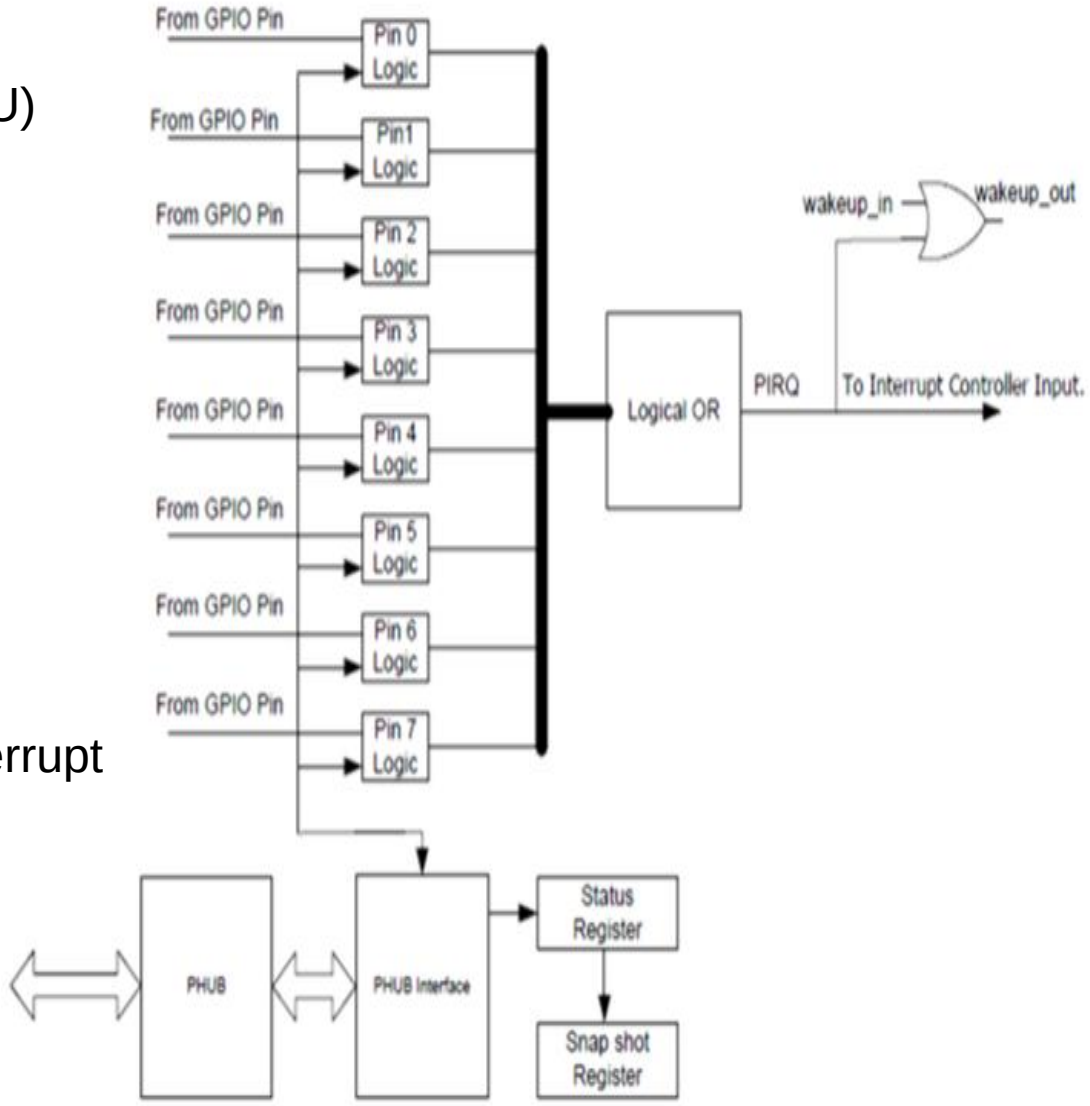
- Port Interrupt Control Unit (PICU)
- Dedicated Interrupt vector

Interrupt on:

- Rising edge
- Falling dege
- Any edge

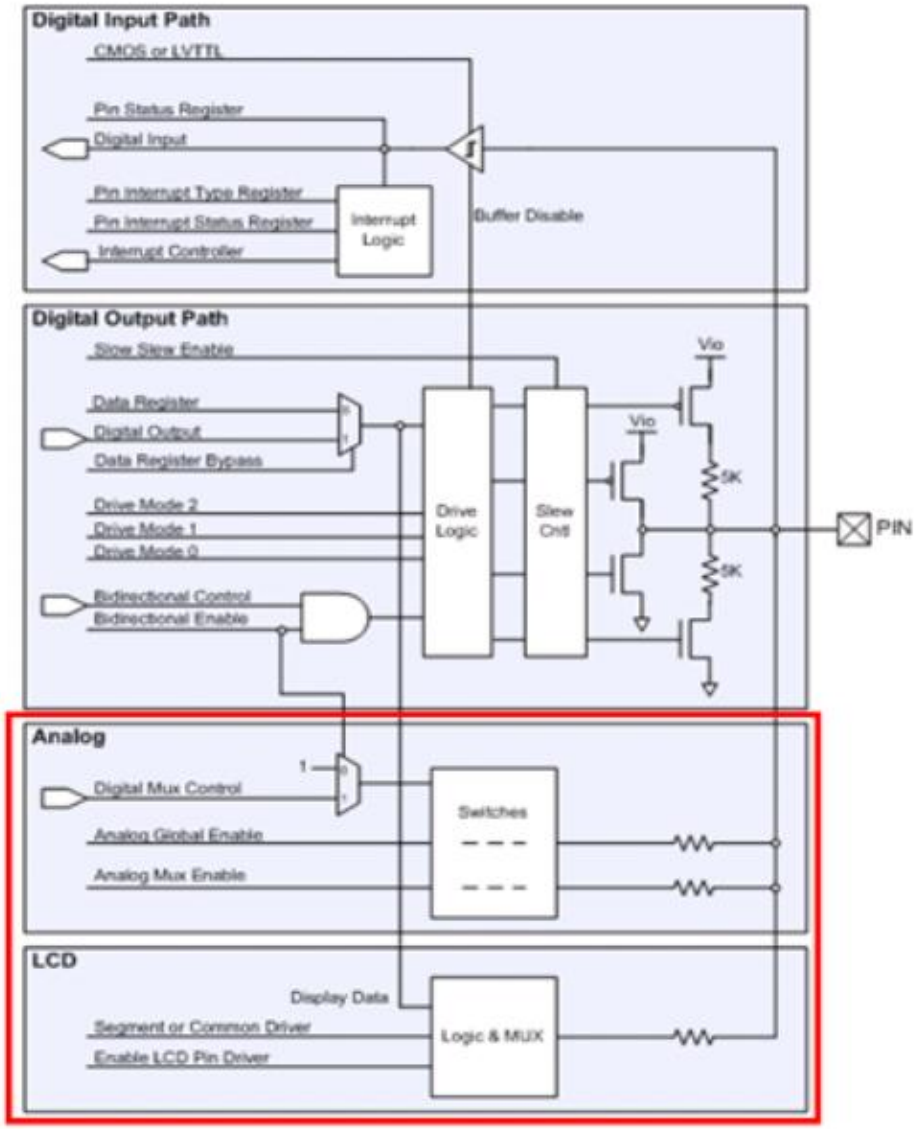
Status Rgister

- Latches which pin triggered interrupt
- Available for firmware read
- Read clear



GPIO - I/O Analog Features

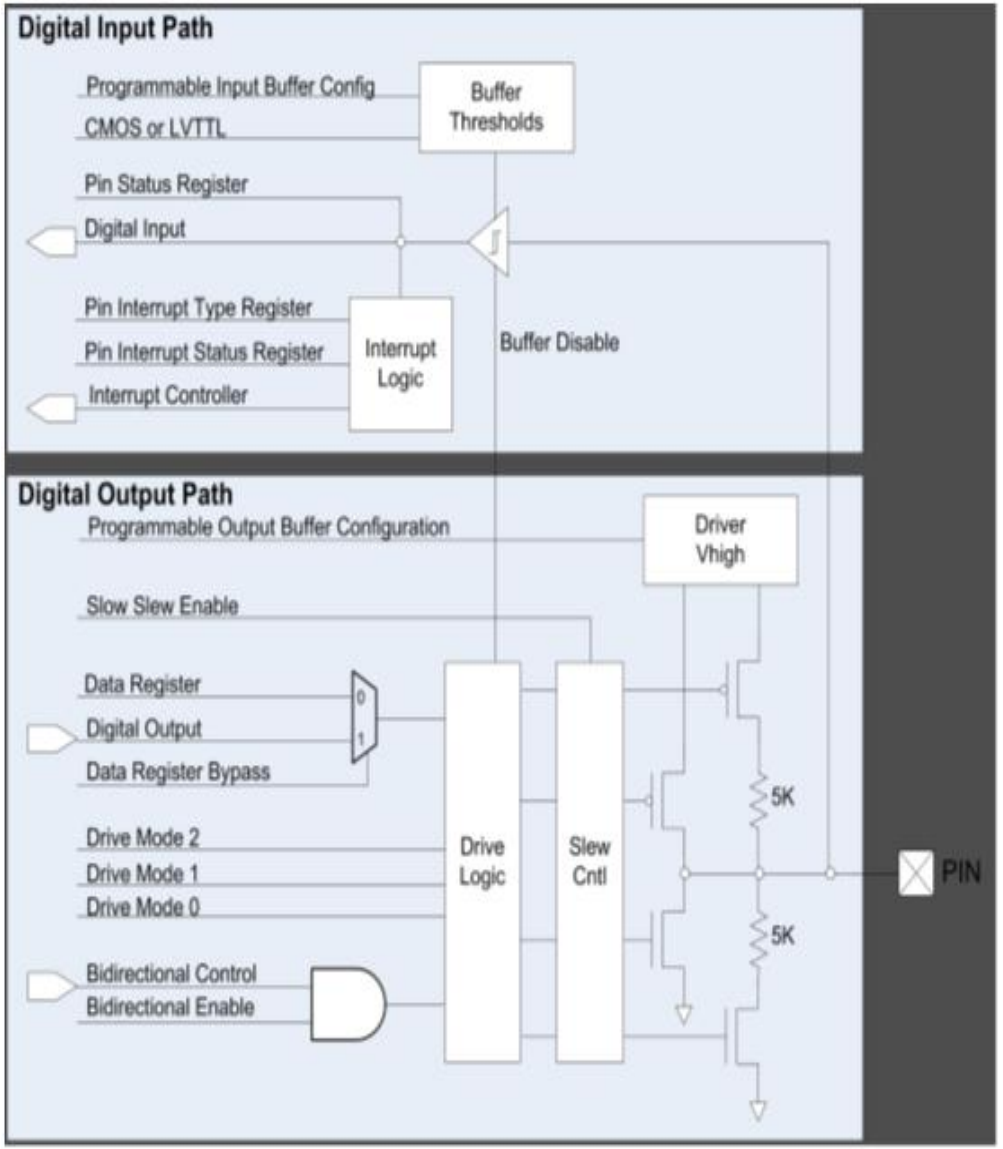
- All pins inputs and outputs
- Supports two independent analog connections at each pin
- CapSense Touch Sensing
- LCD char/segment drive
- Hardware controlled analog mux at pin
- Some pins have additional routing features:
  - OpAmps
  - High Current DAC mode



SIO (Special I/O) Features

Same as GPIO with exceptions:

- 5.5 V tolerant at all Vdda levels
  - Å Hot Swap
  - Å Overvoltage tolerance
- Configurable drive and sense voltage levels
  - Å Basic DAC output
  - Å High Speed CMP input
- Logic level max current
  - Å 25 mA sink
  - Å 4 mA source
- Pin max current
  - Å ~50 mA sink
  - Å ~25 mA source
- No Analog
- No LCD char/segment drive
- No CapSense touch sensing

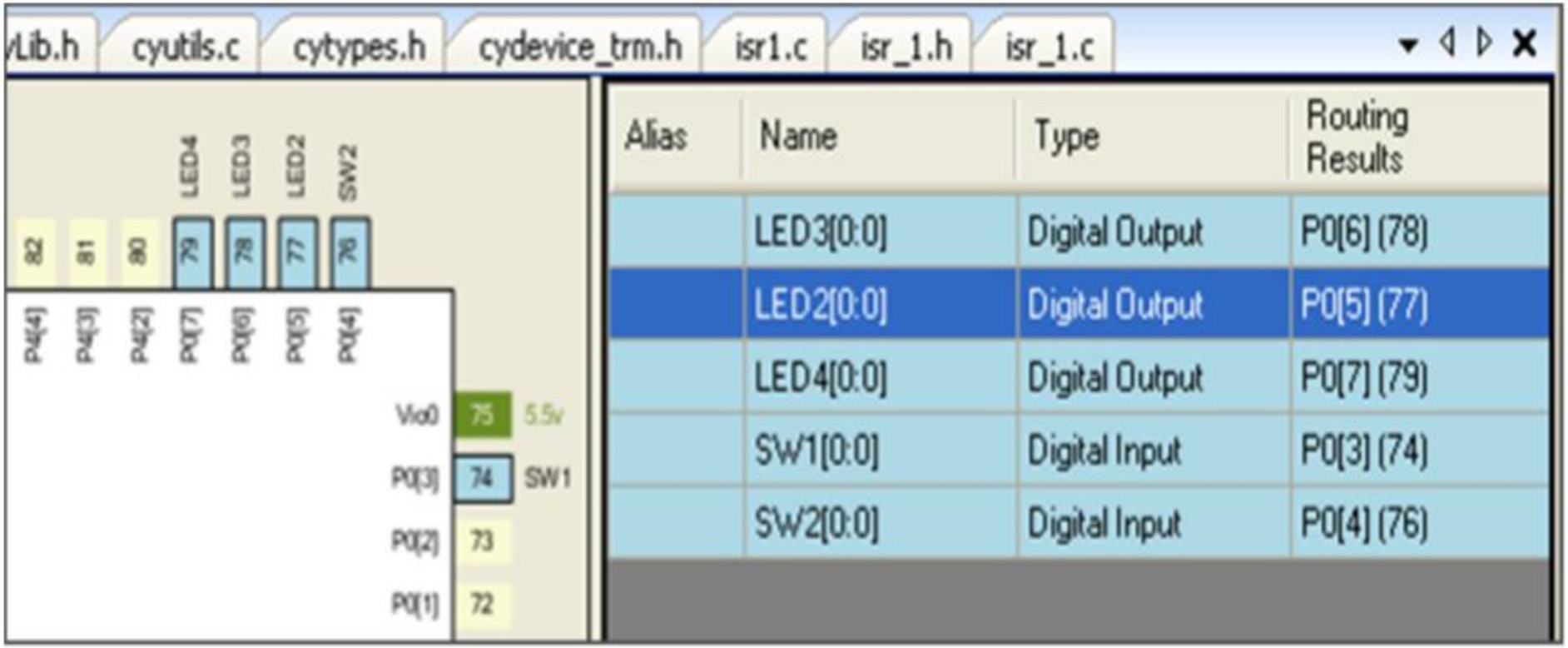


Pin Management

PSoC Creator can select pins automatically

- Lock pins when device pin out is finalized

Manual override in DWR file



Interrupts

Interrupt Controller

- 32 interrupt vectors
- Dynamically adjustable vector addresses
- 8 priority levels
- Each vector supports one of these sources
  - Å Fixed function DMA, DSI (UDB) route

PSoC 8051

- 32 interrupt vectors vs. standard 8051 is five

ARM Cortex-M3

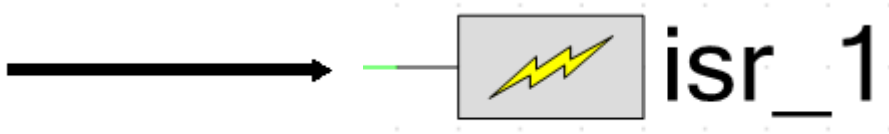
- 32 interrupts + 15 exceptions
- Tail chaining

Int #	Fixed Function	DMA	UDB
0	LVD	DMA output	UDB signal (DSI)
1	ECC	DMA output	UDB signal (DSI)
2	Boost Buck	DMA output	UDB signal (DSI)
3	Sleep (Pwr Mgr)	DMA output	UDB signal (DSI)
4	PICU[0]	DMA output	UDB signal (DSI)
5	PICU[1]	DMA output	UDB signal (DSI)
6	PICU[2]	DMA output	UDB signal (DSI)
7	PICU[3]	DMA output	UDB signal (DSI)
8	PICU[4]	DMA output	UDB signal (DSI)
9	PICU[5]	DMA output	UDB signal (DSI)
10	PICU[6]	DMA output	UDB signal (DSI)
11	PICU[7]	DMA output	UDB signal (DSI)
12	PICU[8]	DMA output	UDB signal (DSI)
13	Comparator Int	DMA output	UDB signal (DSI)
14	Switched Cap Int	DMA output	UDB signal (DSI)
15	I2C	DMA output	UDB signal (DSI)
16	CAN	DMA output	UDB signal (DSI)
17	Timer/Counter0	DMA output	UDB signal (DSI)
18	Timer/Counter1	DMA output	UDB signal (DSI)
19	Timer/Counter2	DMA output	UDB signal (DSI)
20	Timer/Counter3	DMA output	UDB signal (DSI)
21	USB SOF Int	DMA output	UDB signal (DSI)
22	USB Arb Int	DMA output	UDB signal (DSI)
23	USB Bus Int	DMA output	UDB signal (DSI)
24	USB Endpoint[0]	DMA output	UDB signal (DSI)
25	USB Endpoint[1]	DMA output	UDB signal (DSI)
26	USB Endpoint[2]	DMA output	UDB signal (DSI)
27	USB Endpoint[3]	DMA output	UDB signal (DSI)
28	USB Endpoint[4]	DMA output	UDB signal (DSI)
29	DFB Int	DMA output	UDB signal (DSI)
30	Decimator Int	DMA output	UDB signal (DSI)
31	Reserved	DMA output	UDB signal (DSI)

Interrupt Component

GUI-based Configuration

Double click clock to see component window



API

isr_1_Start() ï Configures and enables the interrupt. Typically the only API required to be called

Advanced APIs

- isr_1_SetVector() ï Dynamically change vector address
- isr_1_SetPriority() ï Dynamically change vector priority
- isr_1_GetPriority() ï Read current priority
- isr_1_Enable() ï Enable interrupt vector
- isr_1_GetState() ï Return current state of interrupt vector enable
- isr_1_Disable() ï Disable interrupt vector
- isr_1_SetPending() ï Force a pending interrupt
- isr_1_ClearPending() ï Clear a pending interrupt

Review

You should now be able to:

- Understand the system block diagram of PSoC 3 / PSoC 5 devices
- Understand and use the PSoC 3 / PSoC 5 System Resources, including:
  - Power system
  - Programming and debugging
  - Configuration and boot process
  - Resets
  - Clocking
  - Memory and Mapping
  - DMA and PHUB
  - I/O
  - Interrupts

INTRODUCTION TO PSOC 3 AND PSOC 5

SYSTEM RESOURCES LAB

System Resources Lab

Lab Objective

- To convert an output from the potentiometer into a digital number using the ADC
- To display the digital number on the LCD Screen on PSoC Development Kit

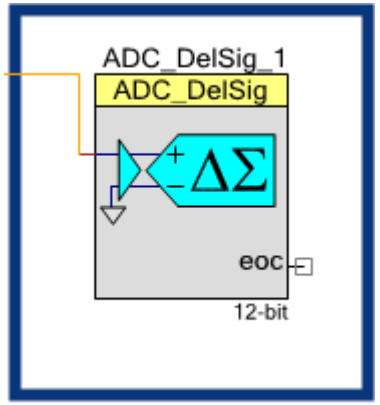
System Resources Lab

Instructions:

1. Open LAB 2
2. Place the Analog Pin from Component Catalog as shown here in the adjacent box

3. Place the Delta Sigma ADC in the box as shown below.
4. Double-click on the component to open it in the configuration mode and make the following changes:

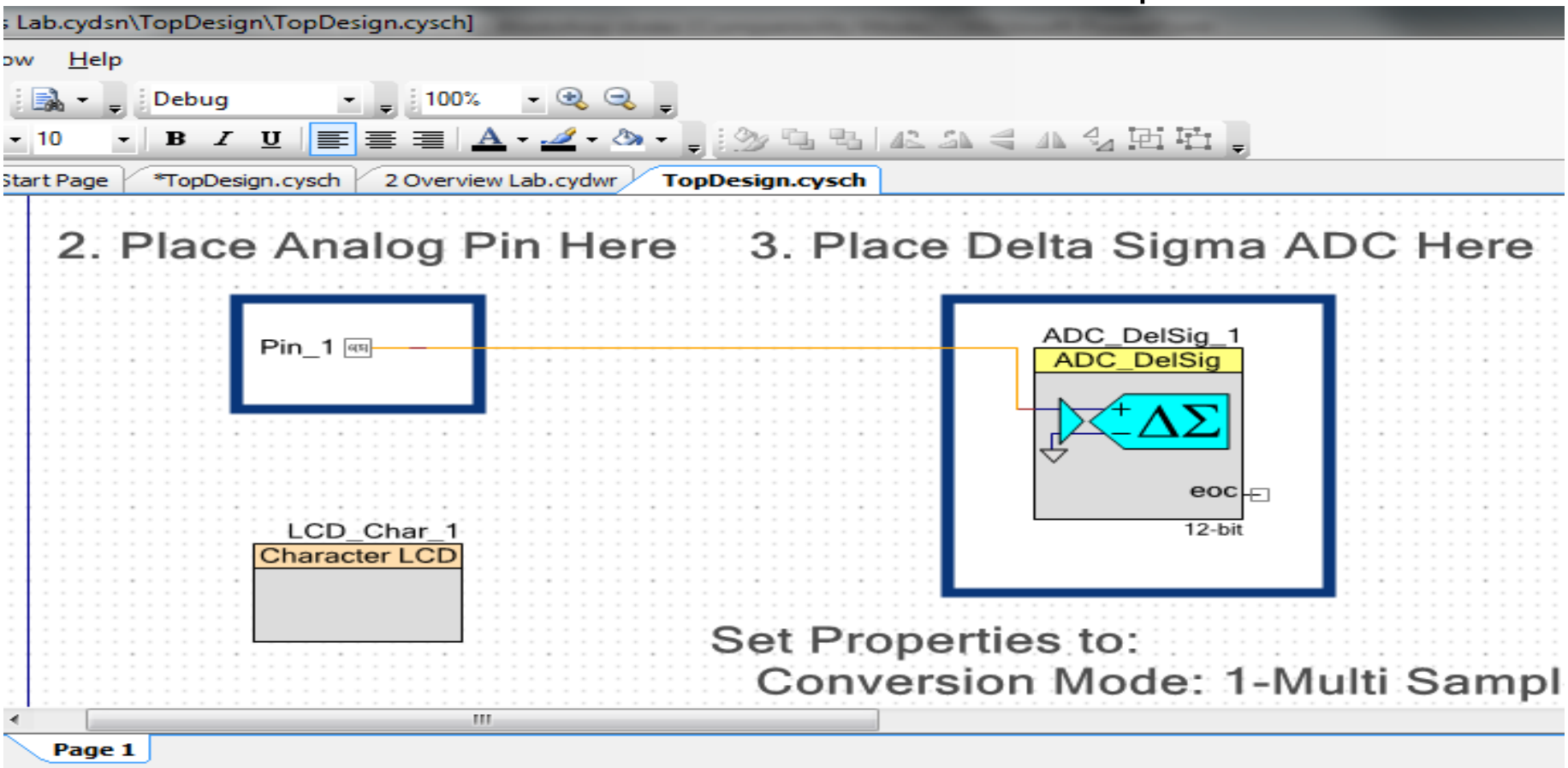
Set Properties to:



Conversion Mode: 1-Multi Sample	Input Mode: Single
Resolution: 12-Bits	Input Range: Vssa to Vdda
Conversion Rate: 1000 SPS	Buffer Gain: 1
Clock Frequency: 131 kHz (Calculated value)	Buffer Mode: Rail to Rail

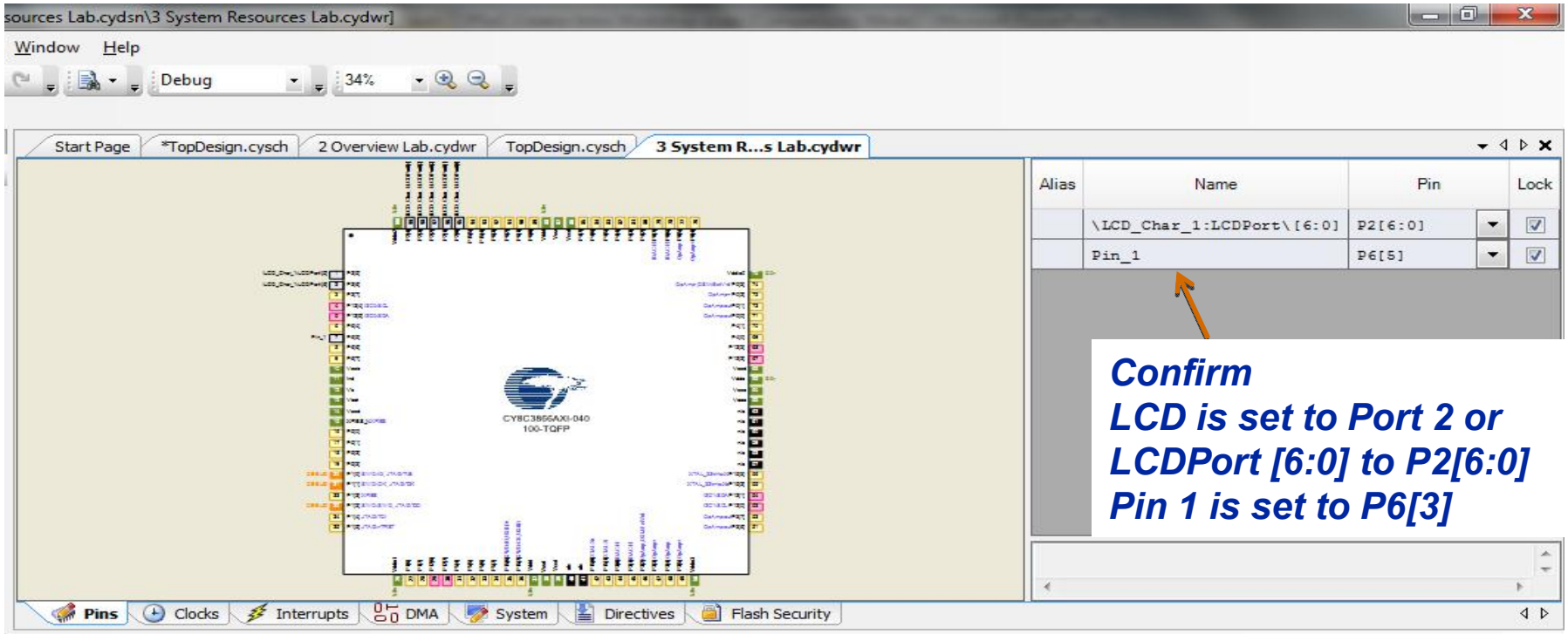
System Resources Lab

5. Use the wire tool found on the left side of the worksheet (shown in slide 23) to connect Pin 1 to the ADC input. To use the wire tool hover over the connections until an óXôappears, then click to make the connection.
6. Your final schematic should look like this when complete



System Resources Lab

7. In the Workspace Explorer double click on the .cydwr file to open Design wide Resources (as explained in Overview lab)
8. In the Design wide Resources tab locate the section for pins on the right
9. Connect the LCD to Port 2 by assigning LCDPort[6:0] to P2[6:0]. Connect Pin 1 to the potentiometer (Port 6, Pin 5) by assigning Pin_1 to P6(5).

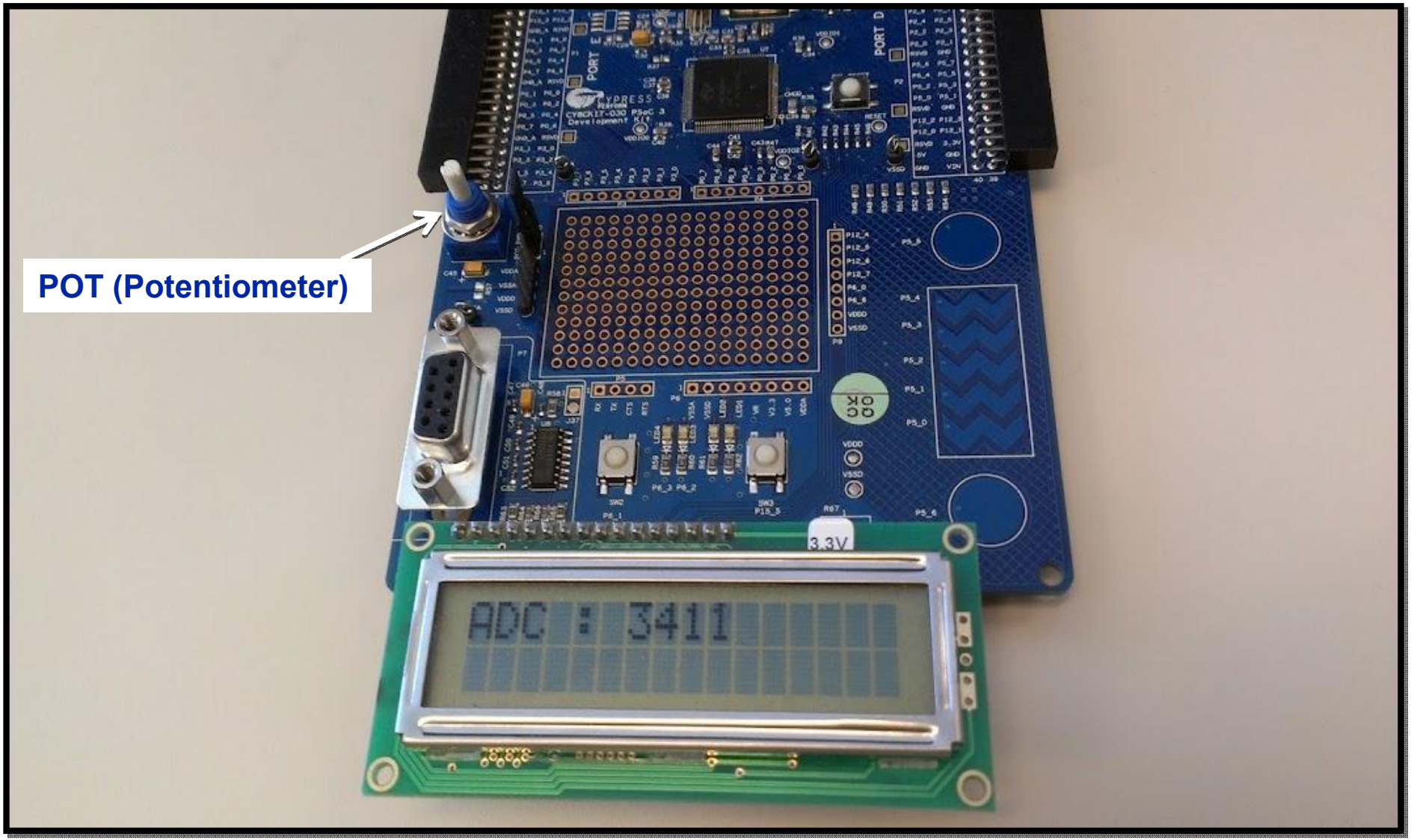


System Resources Lab

10. Build the project by going to the Build menu selecting Build System Resources Lab. This will take some time to build the project.
11. Program the board by going to the Debug menu and in the drop down click Program
12. Push the Reset button on your board located near Port D

Verify: When you turn the POT you should see the ADC values change

System Resources Lab



INTRODUCTION TO PSOC 3 AND PSOC 5

DIGITAL PERIPHERALS

Section Objectives

At the end of this section you should be able to:

- Understand Universal Digital Blocks (UDBs) in PSoC 3 / PSoC 5
- Use and implement digital peripherals with PSoC Creator

Digital Subsystems

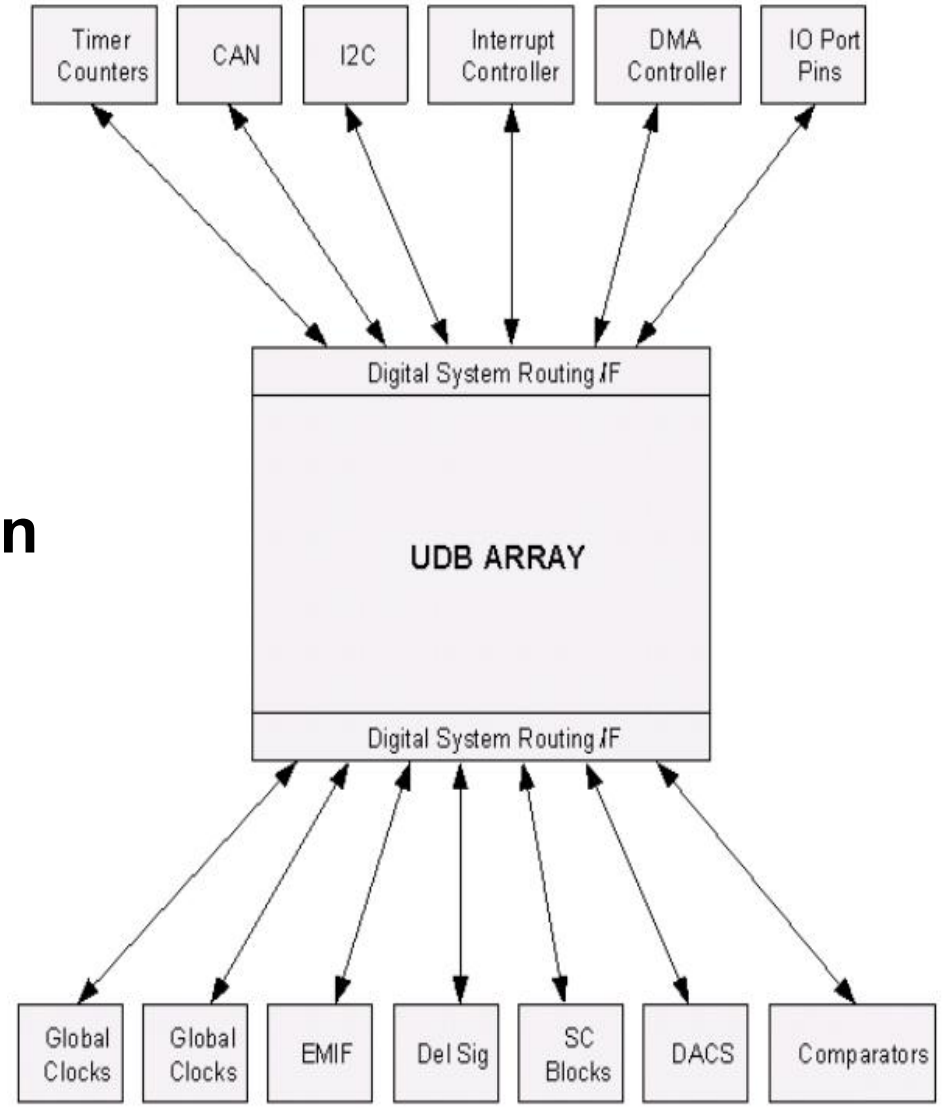
Fixed-function Peripherals

- Counter/Timer/PWMs, I2C, USB, CAN

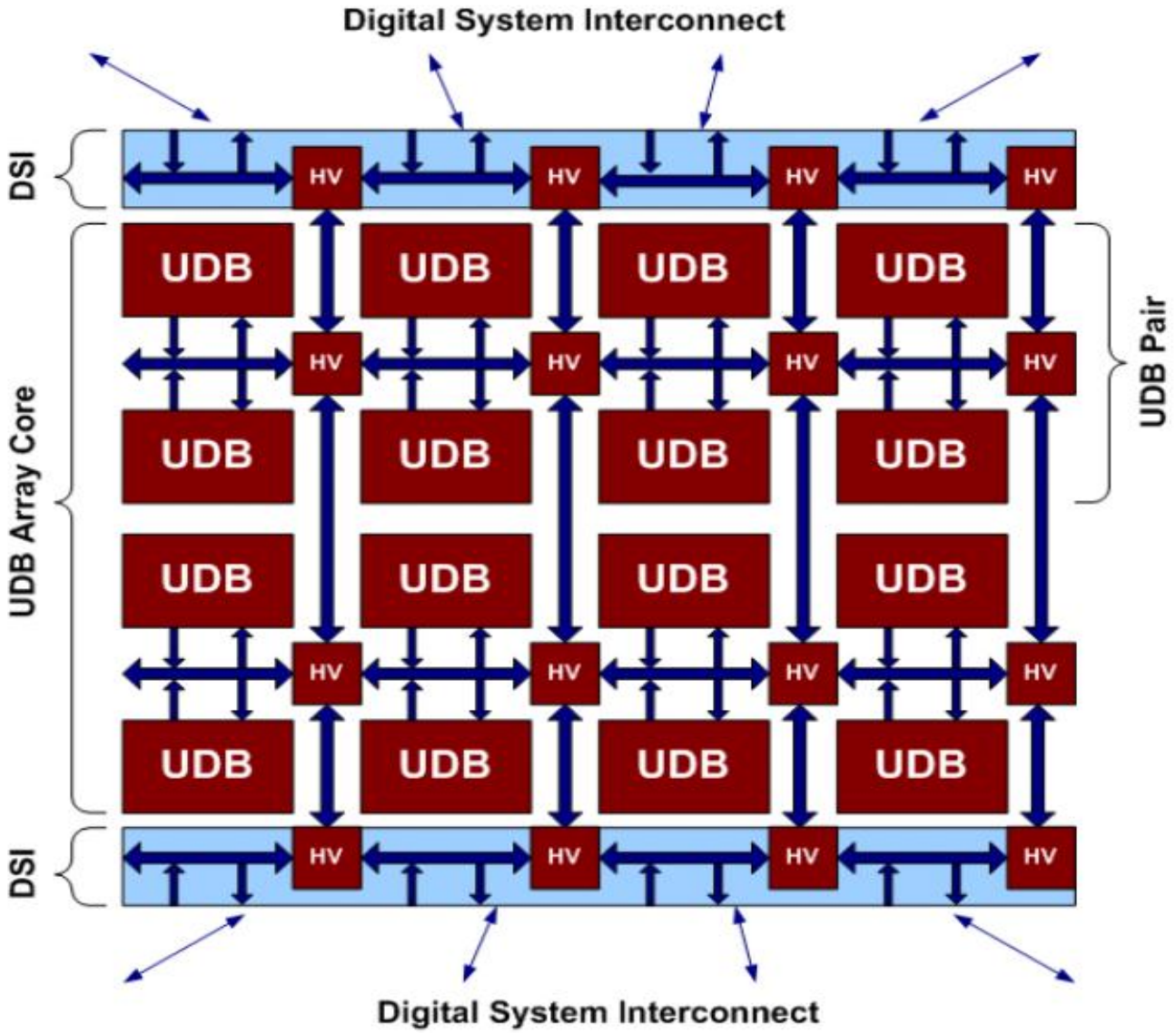
Universal Digital Blocks

Digital Interconnects Between

- Clocks
- IO pins
- Interrupts
- DMA
- External Memory
- Analog system



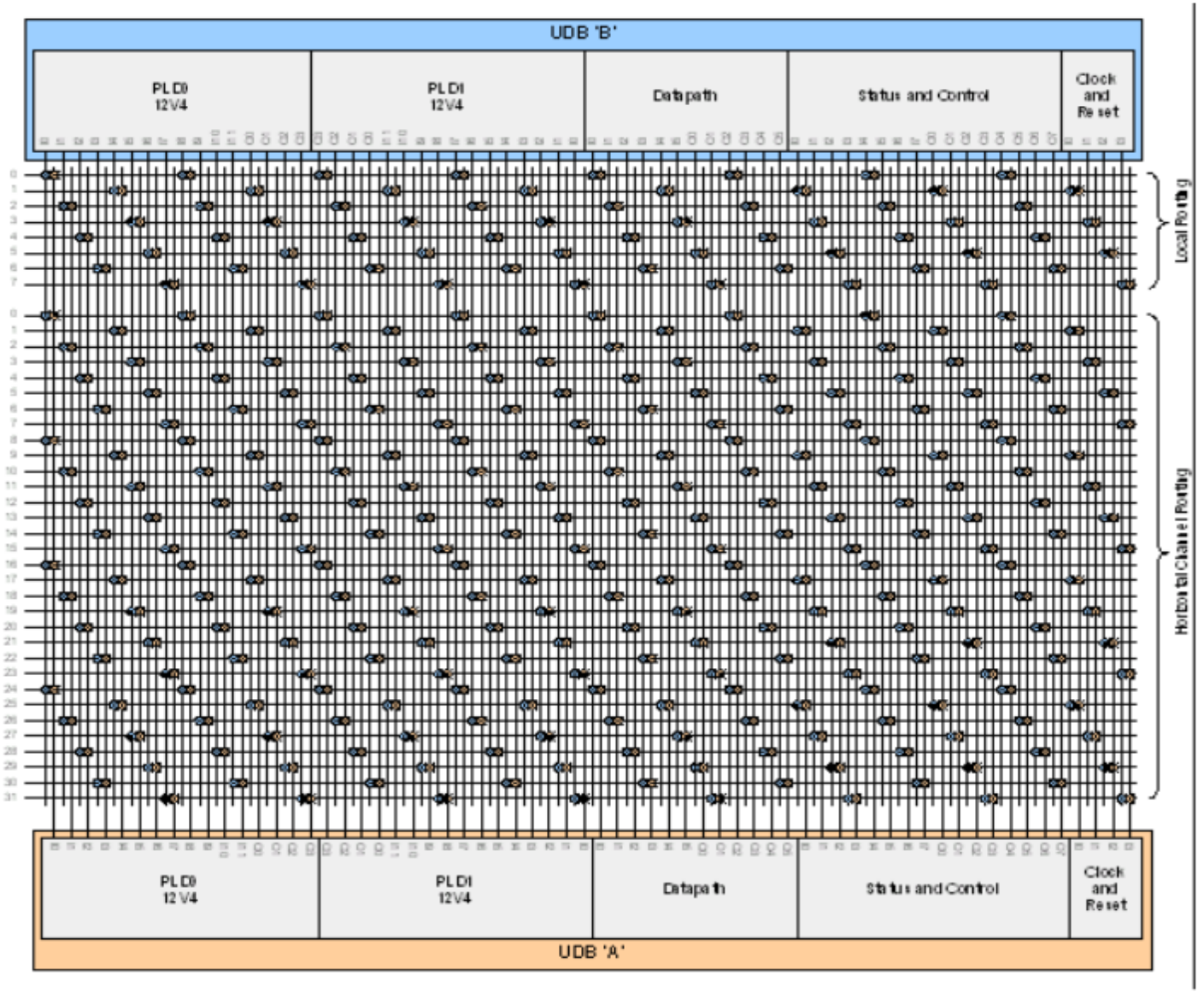
UDB Array


Digital System Interconnect
DSI
HV
UDB
UDB Array Core
UDB Pair
DSI
Digital System Interconnect

Digital Signal Interconnects (DSI)

Routing Example:

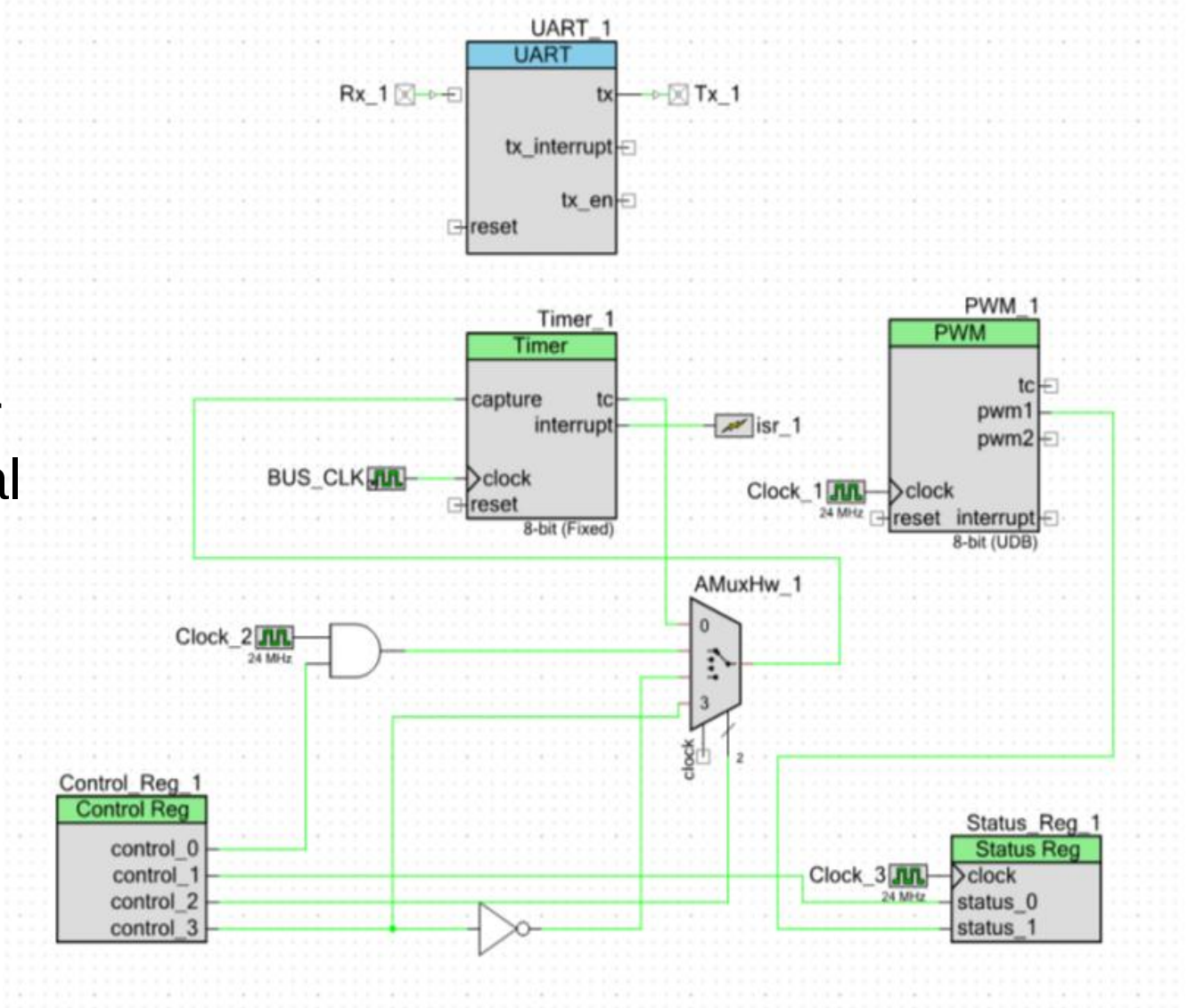
- Single UDB Pair
- 7000 DSI registers



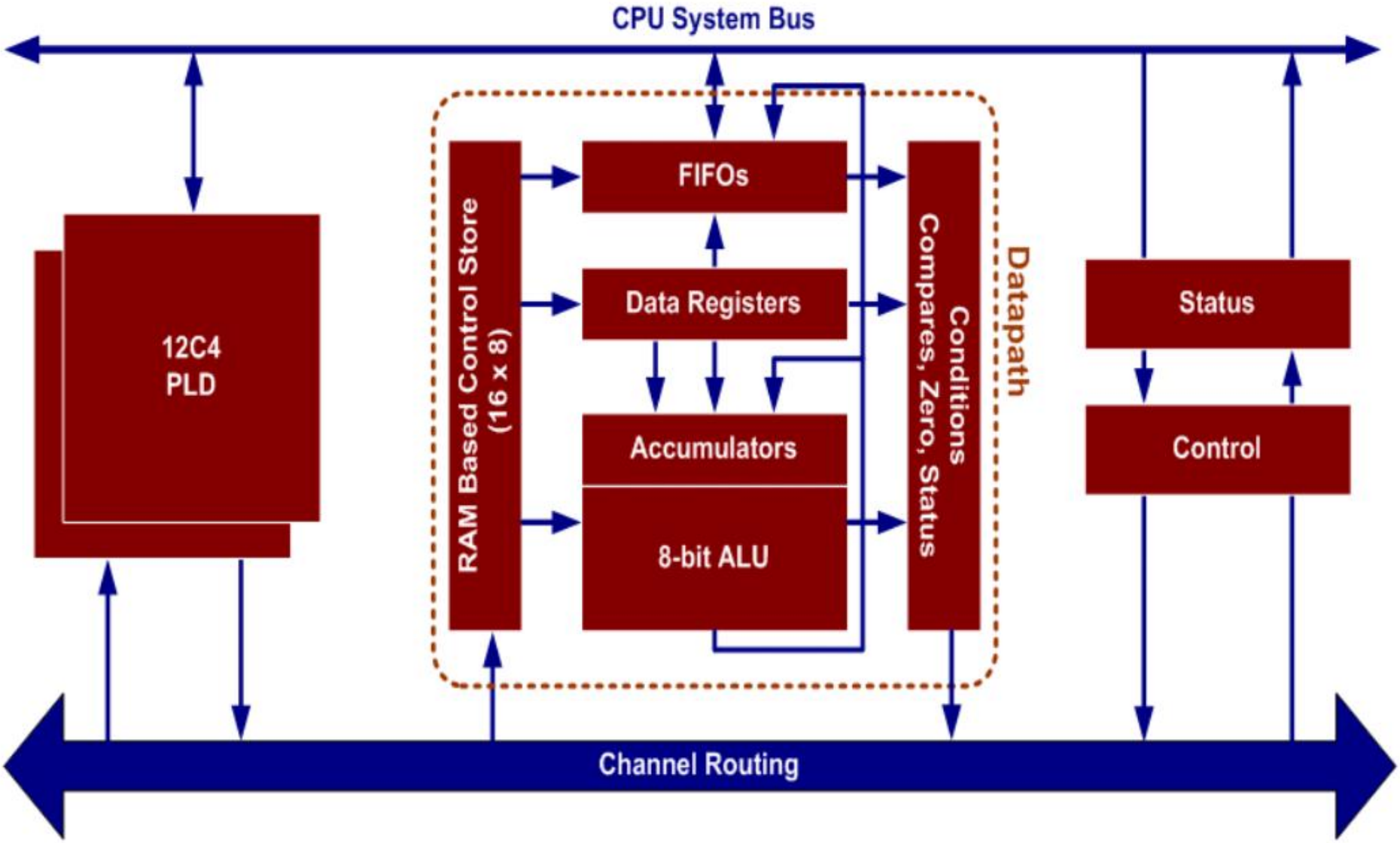
DSI, Automated

PSoC Creator enables:

- Schematic entry
- Automatic place and route
- Optimizations for analog and digital routing



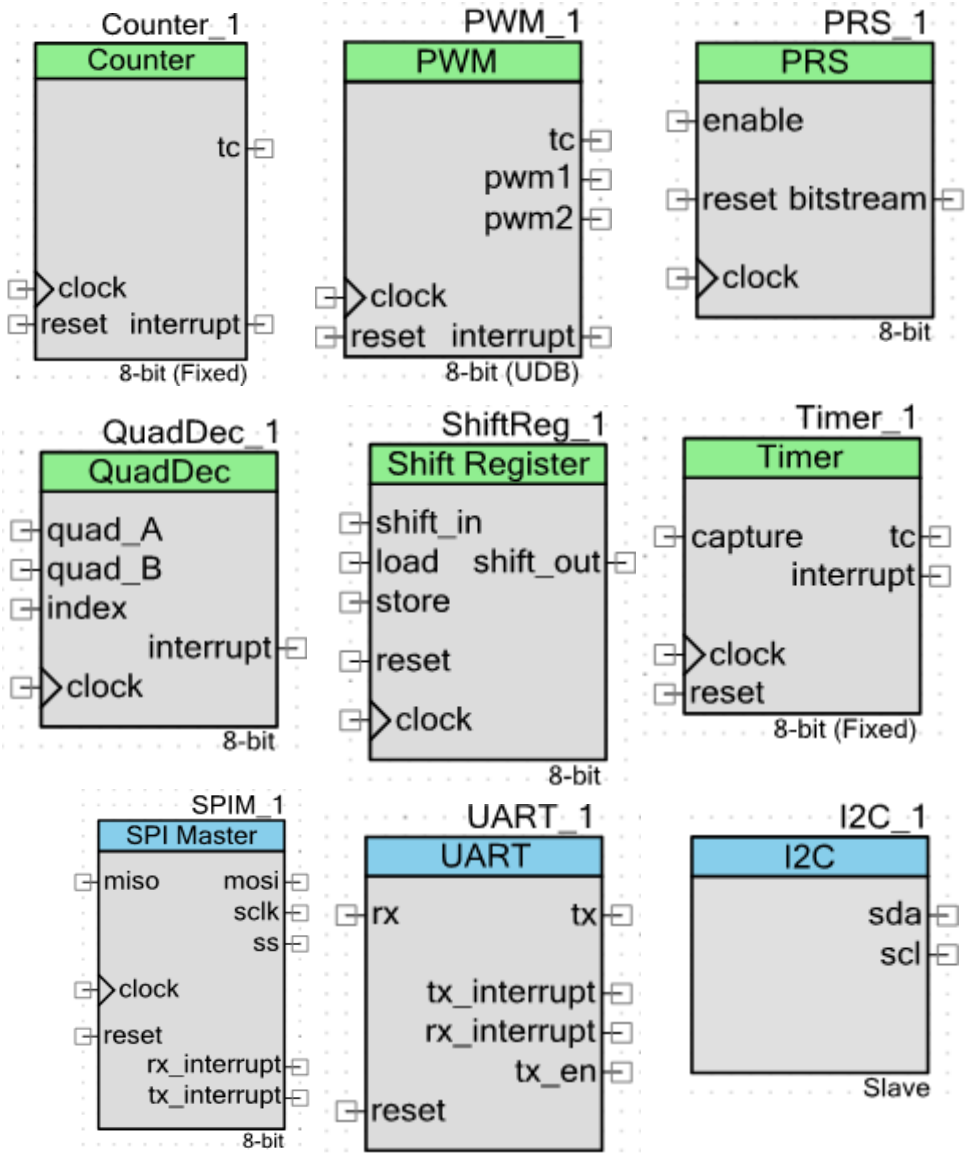
UDB Block Diagram



Digital Peripherals

Sample of Digital Peripherals:

- Counter
- Timer
- PWM
- PRS
- I2C
- USB
- UART
- SPI
- CAN
- Char/Segment LCD Drive



Counter

Fixed/UDB Counters:

- General-Purpose counter
- Continuous, reload on reset, terminal count, or one shot mode
- Compare options: < <= > >= =
- Enable, reset, capture inputs
- Compare, TC, interrupt outputs
- Interrupts on various events

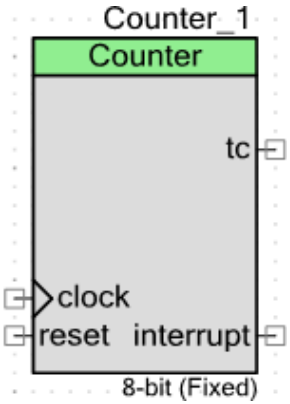
Fixed-Function Counters:

- 4 Available
- 8- or 16-bit
- Down counter only
- Single capture register

UDB-Based:

- 8-, 16-, 24- or 32-bit
- Many options:
  - Enable, count, capture, compare
  - 4-deep capture FIFO

Double Click
Counter to configure

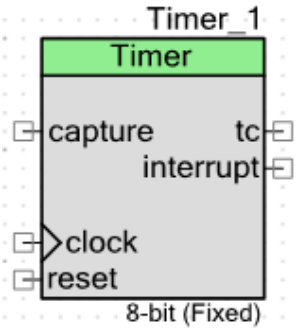


Timer

Fixed/UDB Timers:

- General-Purpose timer for measuring time between H/W events
- Capturing times of events
- Periodic pulse or interrupt
- Continuous, reload on reset, terminal count or one shot mode
- Enable, reset, capture and trigger inputs
- Compare, terminal count, interrupt outputs
- Interrupts on various events

Double Click Timer to configure



Fixed-Function Counters:

- 4 Available
- 8- or 16-bit
- Capture on rising edge only
- Single capture register

UDB-Based:

- 8-, 16-, 24- or 32-bit
- Many options:
  - Enable, trigger, capture
  - Capture counter and capture interrupt CTs
  - 4-deep capture FIFO

PWM

Fixed/UDB PWMs:

- General-Purpose PWM for motor control, LED brightness, etc.
- 8- or 16-bit
- Compare options: < <= > >= =
- Configurable deadbands
- Enable, trigger and kill inputs
- Biphase, TC and interrupt outputs
- Interrupts on various events

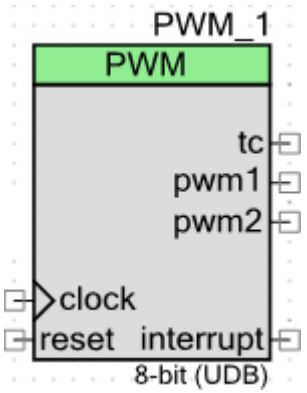
Fixed-Function PWMs:

- 4 Available
- 1 PWM output
- Left-aligned
- Hardware deadband and output kill

UDB-Based:

- 1 or 2 PWM outputs
- Left, right, center or dual-edge align
- Many options:
  - Enable, trigger, kill and compare

Double Click PWM to configure



I2C

Fixed/UDB I2Cs:

- I2C slave, master or multi-master
- Hardware or firmware address decode
- 7- or 10-bit addressing (10-bit F/W only)
- Bus stalling / clock stretching
- SMBus supported w/additonal firmware
- Routes SDA/SCL to any GPIO/SIO pins
- Interrupts for variety of bus events

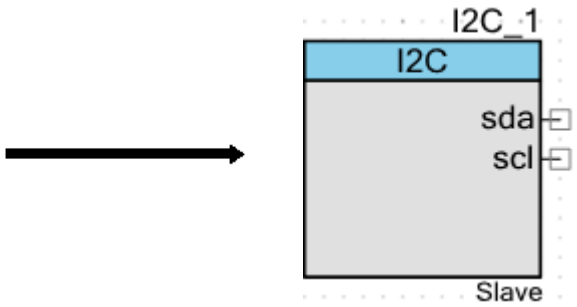
Fixed-Function I2C:

- 1 Available
- Standard 100 Kbps or Fast 400 Kbps
- Responds Sleep Low-Power mode if dedicated SIO connections used

UDB-Based:

- Adds high-speed mode; up to 3.4 Mbps

Double Click I2C to configure



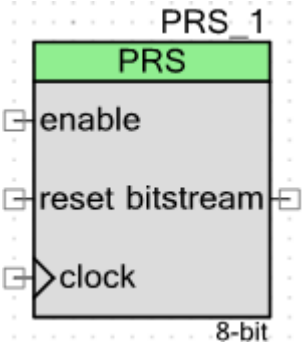
Pseudo Random Sequencer

Pseudo Random Sequencer (PRS)

Specs:

- 2- to 64-bit sequence length
- Serial output bit stream
- Continuous or single step run modes
- Standard or custom polynomial
- Standard or custom seed value
- Enable input provides synchronized operation with other components
- Computed pseudo-random number can be read directly from the linear feedback shift register (LFSR)

Double Click PRS to configure

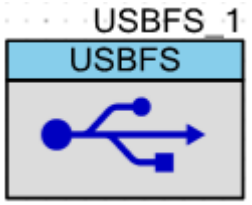


USB

Specs:

- Fixed-function, USB 2.0 Full Speed (12 Mbps) peripheral
- 8 unidirectional endpoints
- Shared 512 byte buffer
- Transfer Types: Control, Interrupt, Bulk, Isochronous
- DMA access / capable
- Wake from sleep

Double Click USB to configure

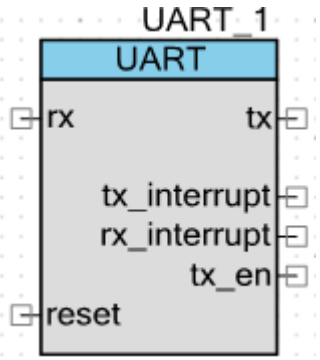


UART

Specs:

- Full-Duplex, Tx only and Rx only
- 5- to 9-data bits
- 110- to 921600-bps or arbitrary up to 4 Mbps
- Rx and Tx buffers 1- to 255-bytes
- Framing, Parity and Overrun error detection
- 9-bit address mode with hardware address detection
- Optional Tx enable for RS-485

Double Click UART to configure

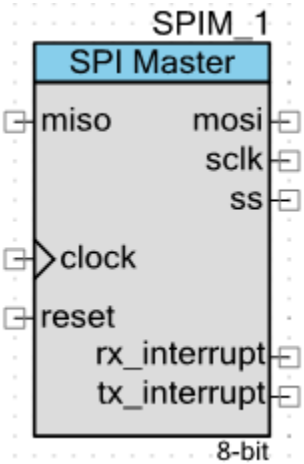


SPI

Specs:

- Master or Slave
- Data rates to 33 Mbps
- 2- to 16-bit data width
- 4 SPI modes
- LSB or MSB first
- 1- to 255-byte Rx and Tx buffers
- Hardware Slave Select generation
- Supports 3-wire mode

Double Click SPI to configure

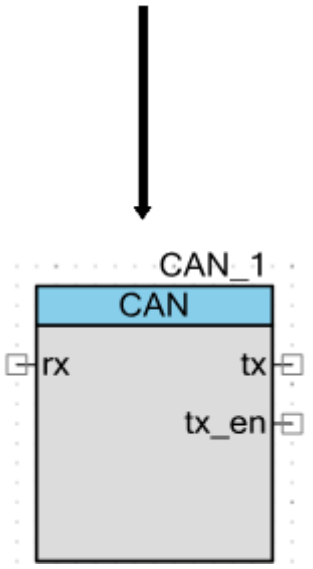


CAN

Specs:

- CAN 2.0A/B spec. compliant
  - Å Remote Transmission Request (RTR) support
  - Å Programmable bit rate up to 1Mbps
  - Å External CAN PHY connects to any GPIO
- Transmit path:
  - Å 8 transmit message buffers
  - Å Programmable priority for each
- Receive path:
  - Å 16 receive message buffers
  - Å 16 acceptance filters/masks
  - Å DeviceNet addressing support
  - Å Option to link multiple receive buffers to/from a hardware FIFO

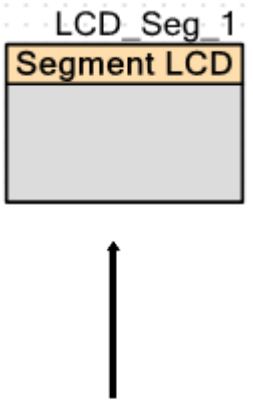
Double Click CAN to configure



LCD Drive

Specs:

- Drives up to 736-segments (16-commons/46-front plane)
- Up to 62 total LCD drive pins; commons and segment lines mapped to any GPIO
- High multiplex ratio of up to 1/16 for max 16 segments
- Type A (standard) and Type B (low power) waveforms supported
- Wide operating voltage range supported (2V to 5.2V) for LCD panels
- Static, ½, 1/3, ¼, 1/5 bias voltage levels
- Vias voltage generation using dedicated DAC, and internal resistor leader
- Up to 128 levels of software controlled contrast
- Ability to move display data from memory to LCD via DMA
- Adjustable LCD refresh rate from 10 Hz to 150 Hz



Double Click LCD to configure

Review

You should now be able to:

- Understand Universal Digital Blocks (UDBs) in PSoC 3 / PSoC 5
- Use and implement digital peripherals with PSoC Creator

INTRODUCTION TO PSOC 3 AND PSOC 5

DIGITAL PERIPHERALS LAB

Digital Peripherals Lab

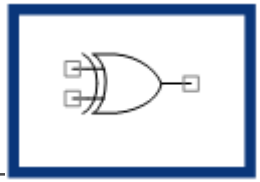
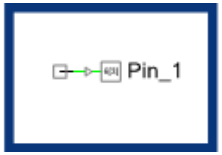
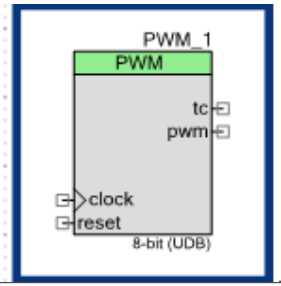
Lab Objective

- To use two PWMs to change the duty cycle on the LED to make it a ñBreathingò LED using your PSoC Development kit for this.

Digital Peripherals Lab

Instructions:

1. Open LAB 3 from the flash drive
2. Find the following components from the Component catalog on the right side of the screen and place them in the boxes given on the ñTop Design.cyschòschematic file. These are shown below:

Logic Low ó0ô 	XOR Gate 
Clock 	Digital Output Pin 
PWM 	

Digital Peripherals Lab

3. Place the components inside the right boxes shown on the schematic file
4. Double click on the clock component to open it in configuration mode and set it to 50 kHz
5. Double click on the PWM component to open it in configuration mode and make the following changes to its properties

Set Properties to:

Implementation: UDB	CMP Value 1: 128
Resolution: 8-Bits	CMP Type 1: Less
PWM mode: One Output	Dead Band : Disabled
Period: 255	Under Advanced tab: Interrupts: None

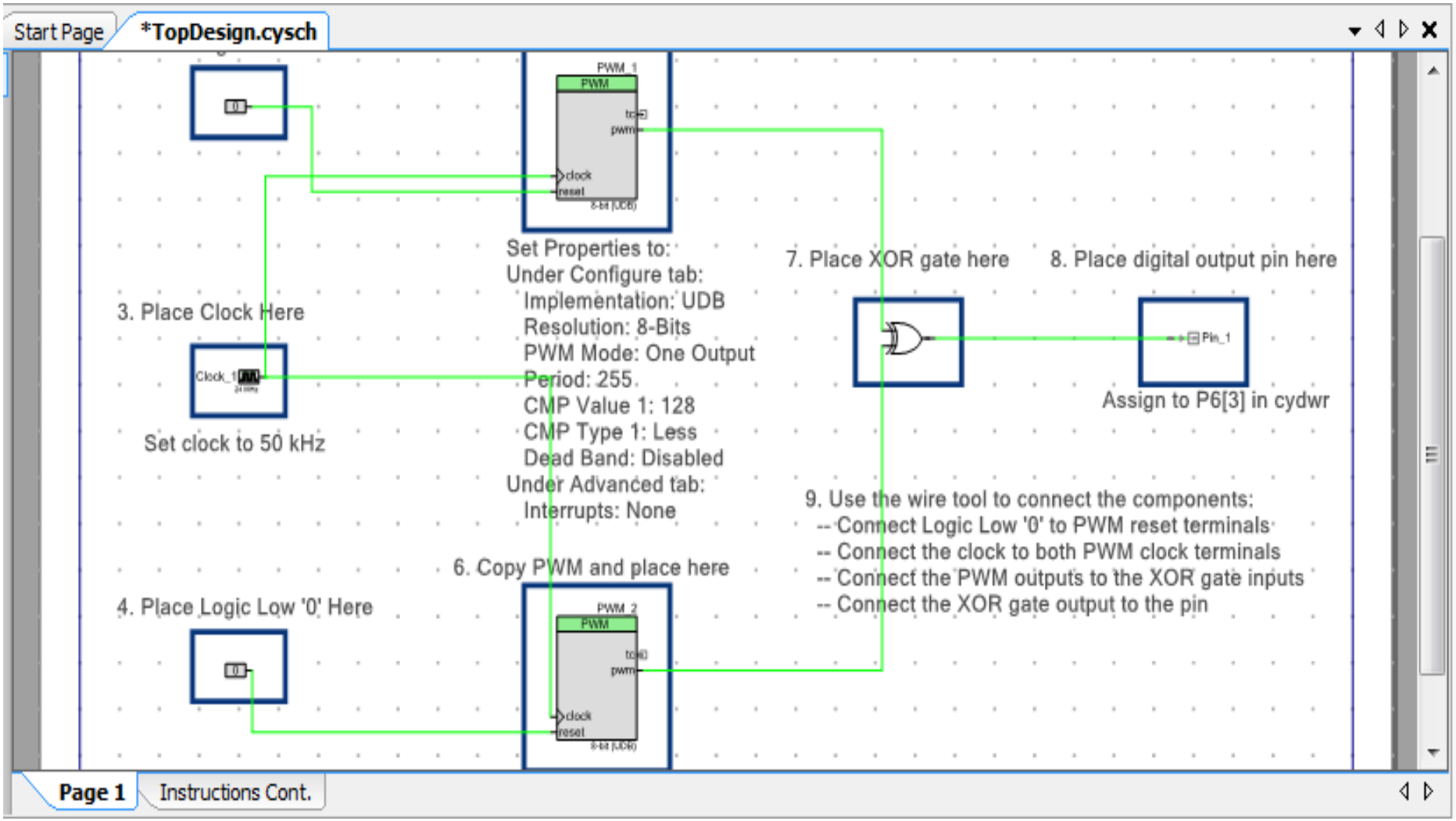
6. Copy the first PWM from the top box into the bottom box and change the ñPeriodò to 254

Digital Peripherals Lab

6. Use the wire tool to make the following connection between the components:
   - Connect Logic Low (0) to PWM reset terminals
   - Connect the clock to both PWM clock terminals
   - Connect the PWM outputs to the XOR gate inputs
   - Connect the XOR gate output to the pin.

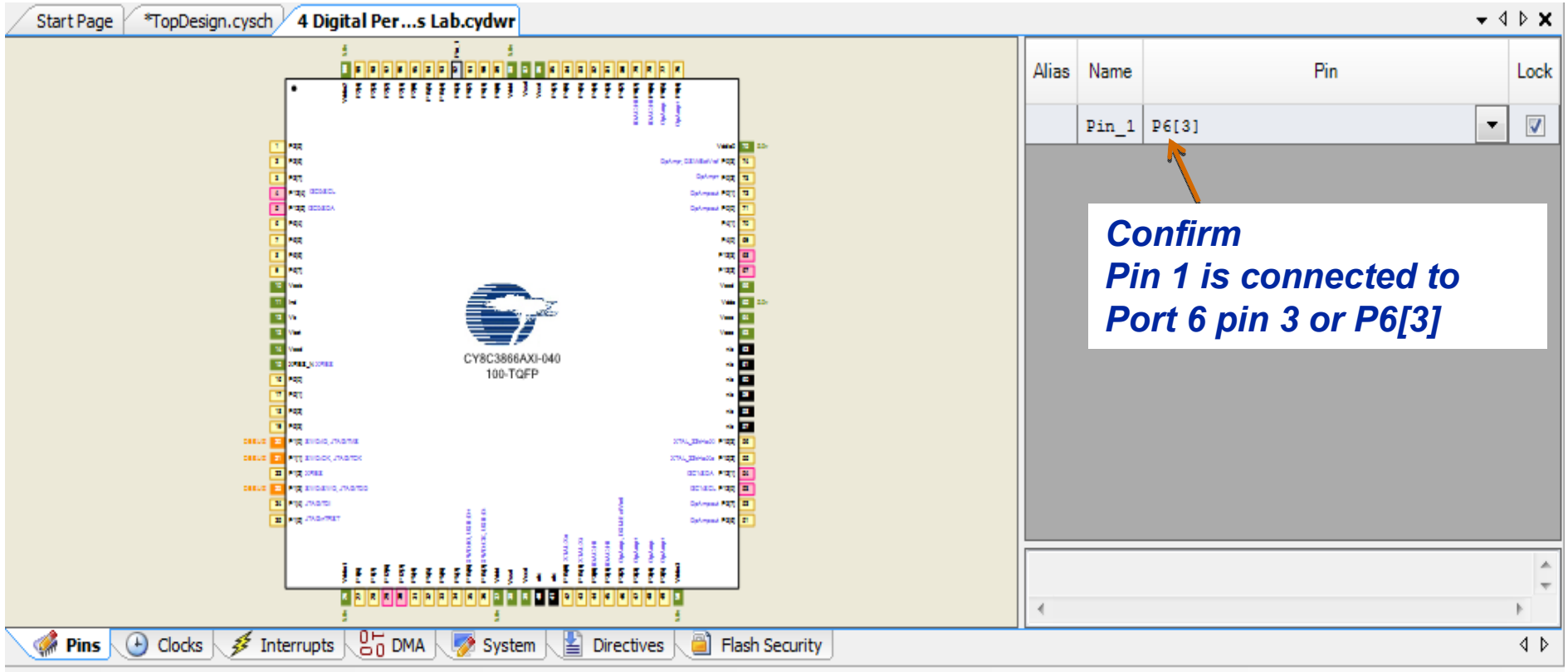
Digital Peripherals Lab

7. Your final schematic should look like this when complete



Digital Peripherals Lab

8. In the Workspace Explorer double click on the .cydwr file to open Design wide Resources (as explained in Overview lab)
9. In the Design wide Resources tab locate the section for pins on the right
10. Configure the I/O so that Pin 1 is connected to Port 6 pin 3 or P6[3].



Digital Peripherals Lab

11. Build the project by going to the Build menu selecting Build Digital Peripherals Lab or pressing Shift + F6. This will take some time to build the project.
12. Program the board by going to the Debug menu and in the drop down click Program
13. Push the Reset button on your board located near Port D

    Verify: LED 4 is seen ñBreathingò

Digital Peripherals Lab

INTRODUCTION TO PSOC 3 AND PSOC 5

ANALOG PERIPHERALS

Section Objectives

At the end of this section you will be able to

- Understand analog in PSoC 3 / PSoC 5
- Use and understand analog peripherals in PSoC Creator

Analog Subsystem

Routing

Multiplexers

Comparators

Opamps

DACs (V & I)

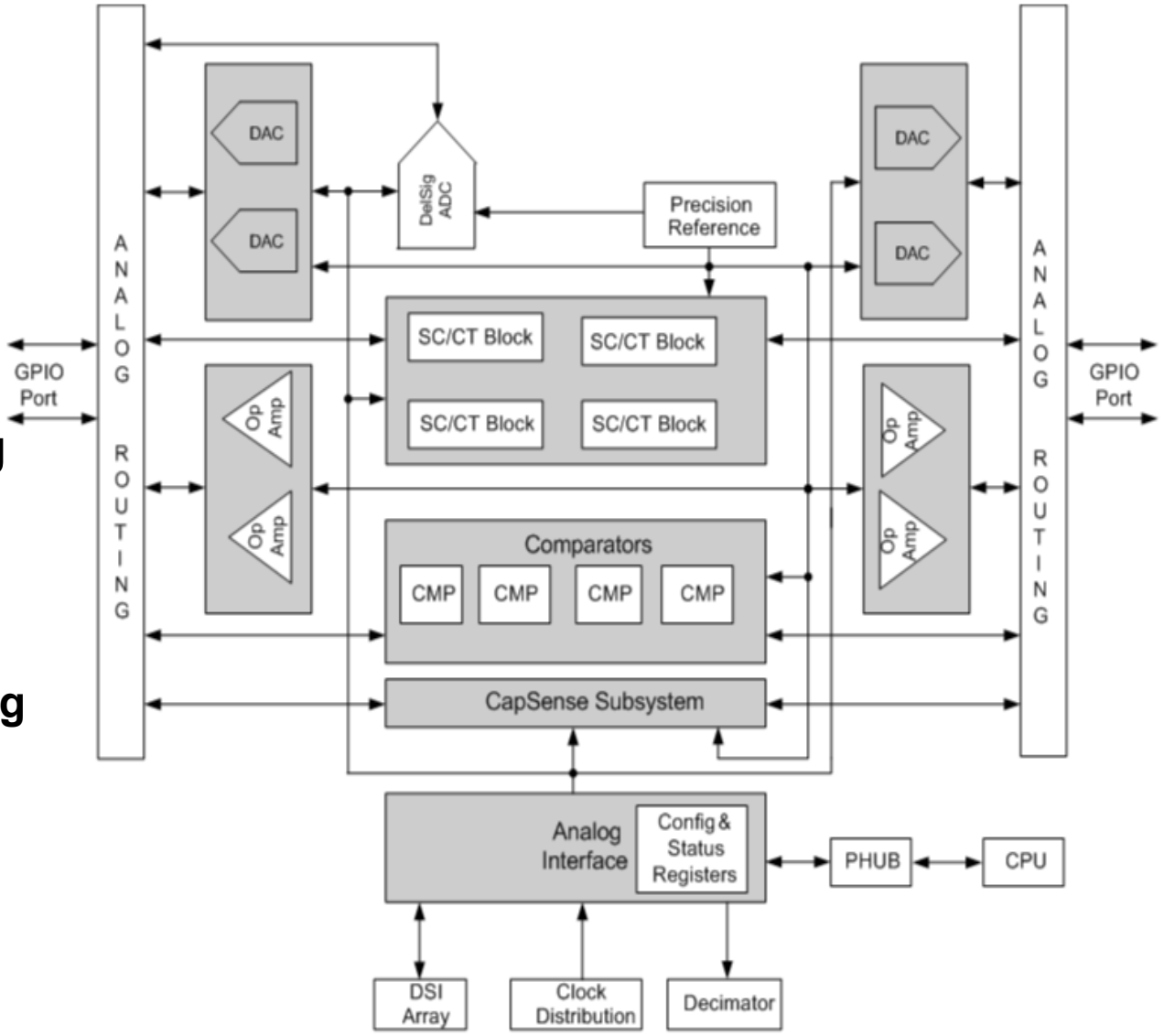
DeltaSigma ADC

Programmable Analog

- PGA
- TIA
- Mixer

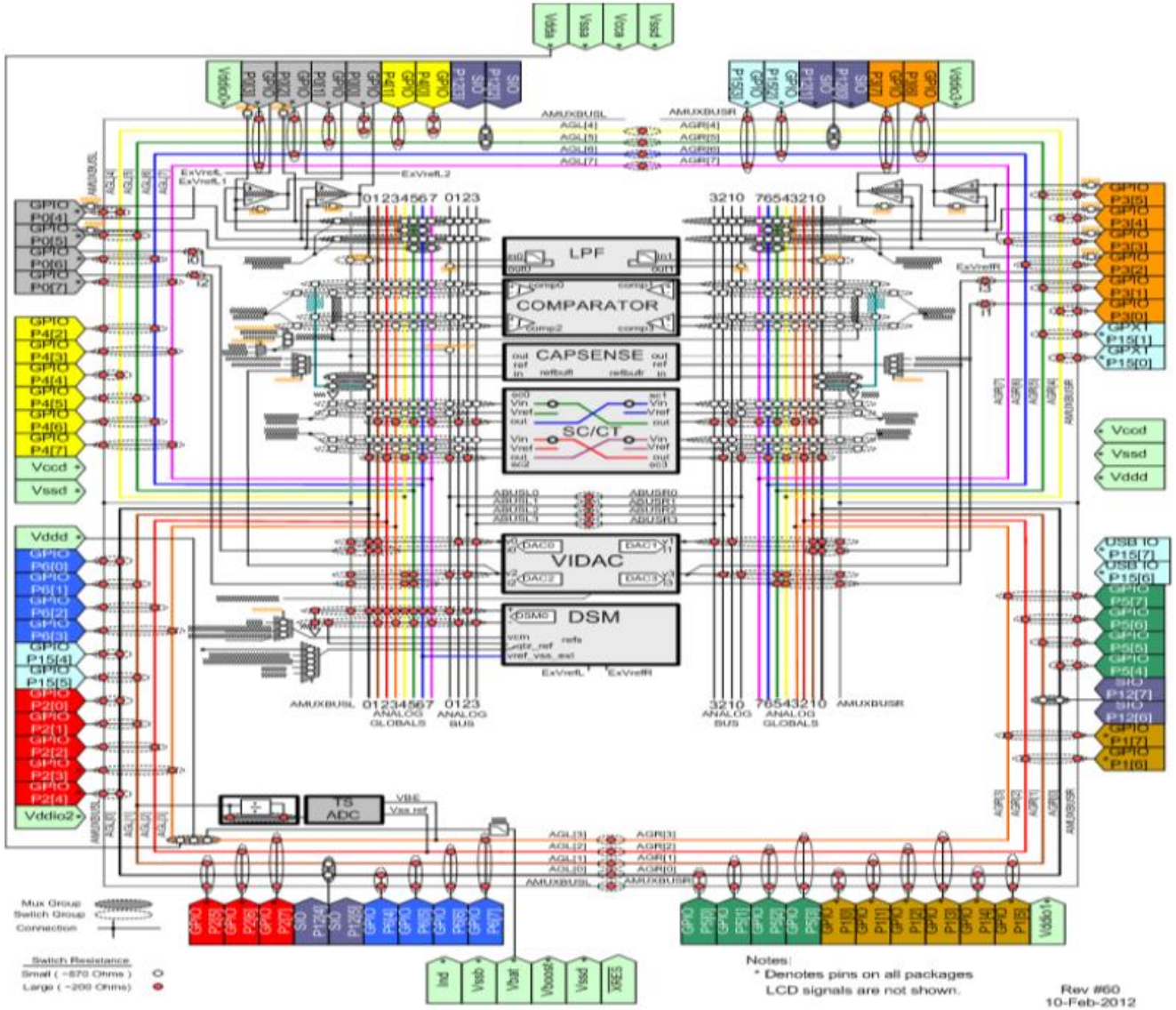
CapSense Touch Sensing

Digital Filters



Analog Matrix

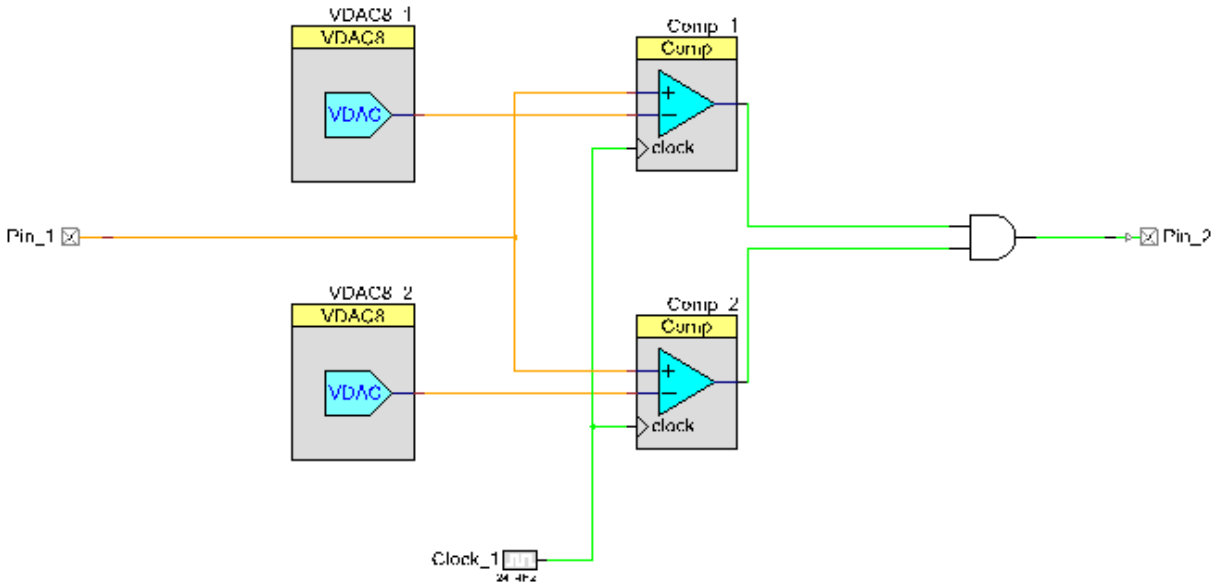
More than 320 switches !



Analog Matrix

PSoC Creator enables:

- Schematic Entry
- Automatic place and route
- Optimizations for analog and digital routing



Window compatator example

Routing with Muxes

Analog Multiplexer Specs:

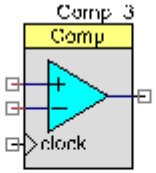
- Bi-directional
- Either single ended or differential
- Actual routing hidden from user
- May have more than one connection at a time
- Analog routing may be controlled by digital subsystem

Double Click MUX to configure

Analog Peripheral: Comparator

Specs:

- Up to four per device
- Speeds:
  - Fast ï 80 ns / 250 uA
  - Slow ï 55 us / 6 uA
- Accuracy:
  - 2 mV fast mode
  - Zero-adjust; Internal VDAC
- Hysteresis:
  - 10 mV nominal
  - May be enabled or disabled

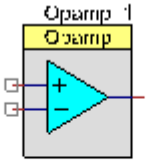


Double click to configure comparator

Analog Peripheral: Opamp

Specs:

- Up to four per device
- Speeds:
  - Fast ï 80 ns / 250 uA
  - Slow ï 55 us / 6 uA
- Accuracy:
  - 2 mV fast mode
  - Zero-adjust; Internal VDAC
- Hysteresis:
  - 10 mV nominal
  - May be enabled or disabled



Double click to configure Opamp

Analog Peripheral: VDAC8 and IDAC8

Features

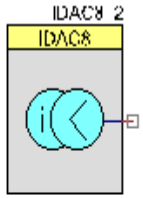
- DAC data source may be data register or DAC Bus
- CPU or DMA may write to data register
- Data Strobe from data register write or Strobe input
- Clock or UDB may be used for Strobe
- IDAC8 and VDAC8 are the same block with V and I inputs

Double click to configure DAC, also show data source

Analog Peripheral: IDAC8

Specs:

- Source or sink
- Ranges:
  - Å0 to 31.875 uA (125 nA/l
  - Å0 to 255 uA (1 uA/bit)
  - Å0 to 2.04 mA (8 uA/bit)
- Power (ICC):
  - Å100 uA max slow mode
  - Å500 uA max fast mode

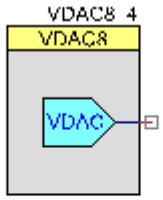


Double click to configure IDAC, also show Polarity

Analog Peripheral: VDAC8

Specs:

- Ranges:
  - 0 to 1.02V (4mV/bit)
  - 0 to 4.08V (16mV/bit)
- Output R = ~16 k ohms (4 volt range)
  - Must be buffered for external use
  - Some internal loads don't require buffering
- Power (ICC):
  - 100 uA max slow mode
  - 500 uA max fast mode
- Speed:
  - 1 Msps (1V mode)
  - 250 ksps (4V mode)

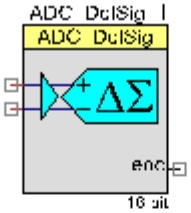


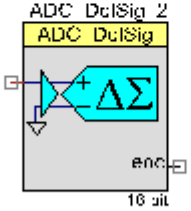
Double click to configure DAC

Analog Peripheral: Delta Sigma ADC

Specs:

- High speed, high resolution ADC
- Selectable resolutions (8- to 20-bit)
- Several input ranges
- High-impedance input buffers
  - Programmable gain (1,2,4,8)
  - Chopper mode for low offset
  - Internal reference may be bypassed
- Single and differential input modes
- Wide range of sample rates 10 to 375K
- Multiple reference sources
- Drop, connect, and go!





Double click to configure ADC

Programmable Analog Peripherals

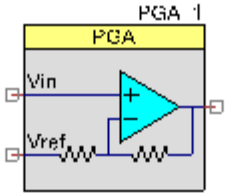
Specs:

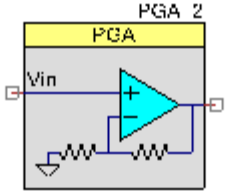
- Combination of functions from PSoC1 programmable analog catalog
- Switched-Capacitor (SC)-based analog
- Continuous Time blocks (CT)-based analog

Analog Peripheral: PGA

Specs:

- Programmable Gain Amplifier
- Amplify signals without external components
- Gain: 1x to 50x
- Vin and Vref to any pin





Accuracy:

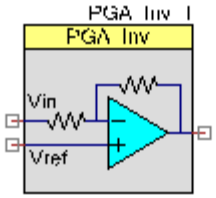
- Gain: +/- 5%
- Vos: 10 mV

Double click to configure PGA

Analog Peripheral: PGA_Inv

Specs:

- Inverting PGA
- Amplify signals without external components
- Gain: -1x to -49x
- Vin and Vref to any pin



Double click to configure PGA

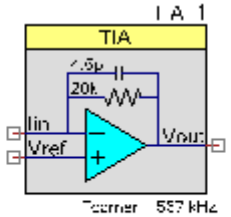
Accuracy:

- Gain: +/- 5%
- Vos: 10 mV

Analog Peripheral: TIA

Specs:

- Trans-Impedance Amplifier
- Conversion gain is a ñresistorò
  - Current IN -> Volts OUT
  - Vout = Vref ï Iin*Rfb
  - Adjustable Rfb = Programmable gain (20k to 1M)
  - Adjustable Cfb = Programmable bandwidth (up to 4.7 pF)
  - Calibrated with on-chip IDAC (internal resistors +/- 30%)
- Applicable to current output sensors
  - Glucose meters
  - Photo-diodes ï light meters, medium speed IR receiver

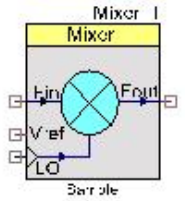


Double click to configure TIA

Analog Peripheral: Mixer

Specs:

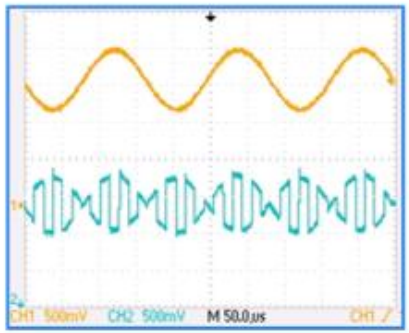
- LO_Freq = Local Oscillator



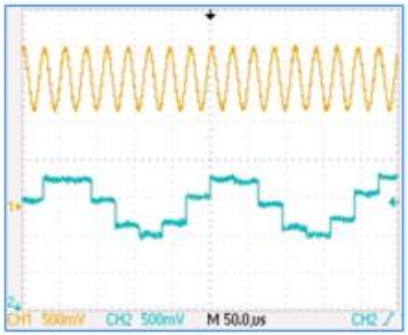
Double click to configure Mixer

Mixer Modes:

- Up-Mixer: Multiplier
  - Clock up to 1.0 MHz
  - Output at LO_Freq +/- Fin
  - Example 200 kHz input clocked at 255 kHz to narrowband filter at 455 kHZ



- Down-Mixer Sampler
  - Clock up to 4.0 MHz
  - Output at Fin ï LO_Freq
  - Example 455 clocked at 435 kHz to low-pass filter at 25 kHz



Analog Peripheral: CapSense®

Specs:

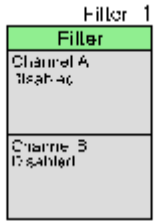
- CapSense peripheral uses configuration of existing system resources
- Two simultaneous CapSense systems possible
- CapSense Configurations
  - All have similar configuration patterns
  - Buttons: Basic CapSense sensor with On/Off detection
  - Sliders: Linear and radial with interpolated positioning (also supports diplexing to reduce pin count)
  - Touch Pads: X, Y interpolated positioning
  - Matrix Buttons
  - Proximity Sensors
  - Generic Sensors

Double click to configure Capsense

Analog Peripheral: DFB

Specs:

- Digital Filter Block
- What it does:
  - Works on digitized data from ADC or any other source
  - Sequentially calculate multiple filters
  - Removes noise and unwanted frequencies from signals
  - Replaces analog filters requiring external components
- How it works:
  - No coding or coefficients! All handled by GUI
  - 24-bit filter co-processor
  - 128 pairs of data/coefficients
- Setting for filters can be made for 2 channels
- Select type:
  - Low Pass, Band Pass, High Pass, Notch
- Select implementation:
  - FIR or IIR Biquad
- Specify sample rate
- Specify number of taps
- Select frequency parameters



Double click to configure DFB

Review

You should now be able to:

- Understand analog in PSoC 3 / PSoC 5
- Use and understand analog peripherals in PSoC Creator

INTRODUCTION TO PSOC 3 AND PSOC 5

ANALOG PERIPHERALS LAB

Analog Peripherals Lab

Lab Objective

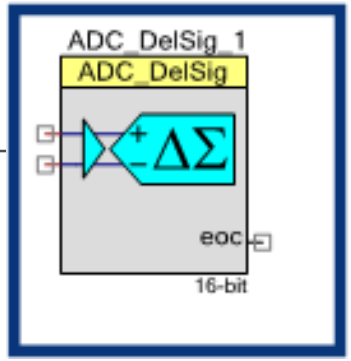
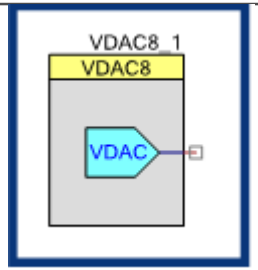
- To use a VDAC to convert a clock signal into an analog signal then convert it back into digital signal using an ADC DelSig.
- To output the signal on the LCD Screen using the vertical bar graph option using your PSoC 3 Development kit for this.
- To be able to see a sine wave on the screen when the lab is complete.

Analog Peripherals Lab

Instructions:

1. Open LAB 4 from the flash drive and find the following components from the Component catalog on the right side of the screen and place them in the boxes given on the ñTop Design.cyschòschematic file.

2. These are shown below:

Clock 	ADC Del-Sig 
VADC 	
Interrupt 	

Analog Peripherals Lab

3. Place the components inside the right boxes shown on the schematic file
4. Double click on the clock component to open it in configuration mode and set it to 256 Hz
5. Double click on the VDAC component to open it in configuration mode and make the following changes to its properties

Set Properties to:

Output Range: 0 ï 1.02V	StrobeMode: External
Speed: Slow	Data Source: CPU or DMA
Value: 100mV; 19 bytes	

6. Double click on the ADC Del-Sig and set properties as below

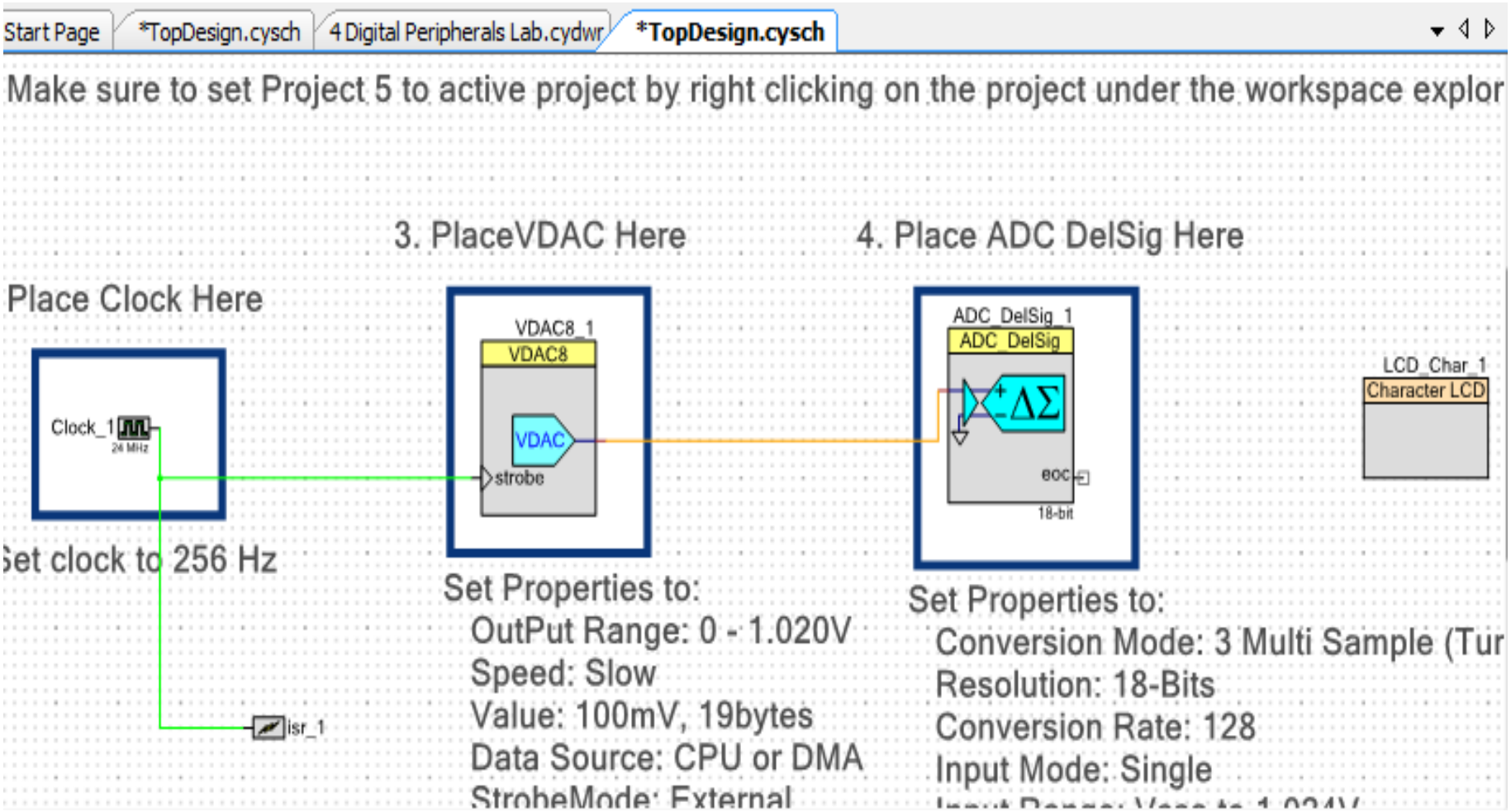
Conversion Mode: 3 Multi Sample (Turbo)	Input Mode: Single	Buffer Gain: 1
Resolution: 18-Bits	Buffer Mode: Rail to Rail	Input Range: Vssa to 1.024V
Conversion Rate: 128	Vref: Internal 1.024V	

Analog Peripherals Lab

7. Use the wire tool to make the following connection between the components:
   - Connect Clock to the strobe input of the VDAC
   - Connect the Clock to the Interrupt
   - Connect VDAC output to the input of the Single ended ADC DelSig

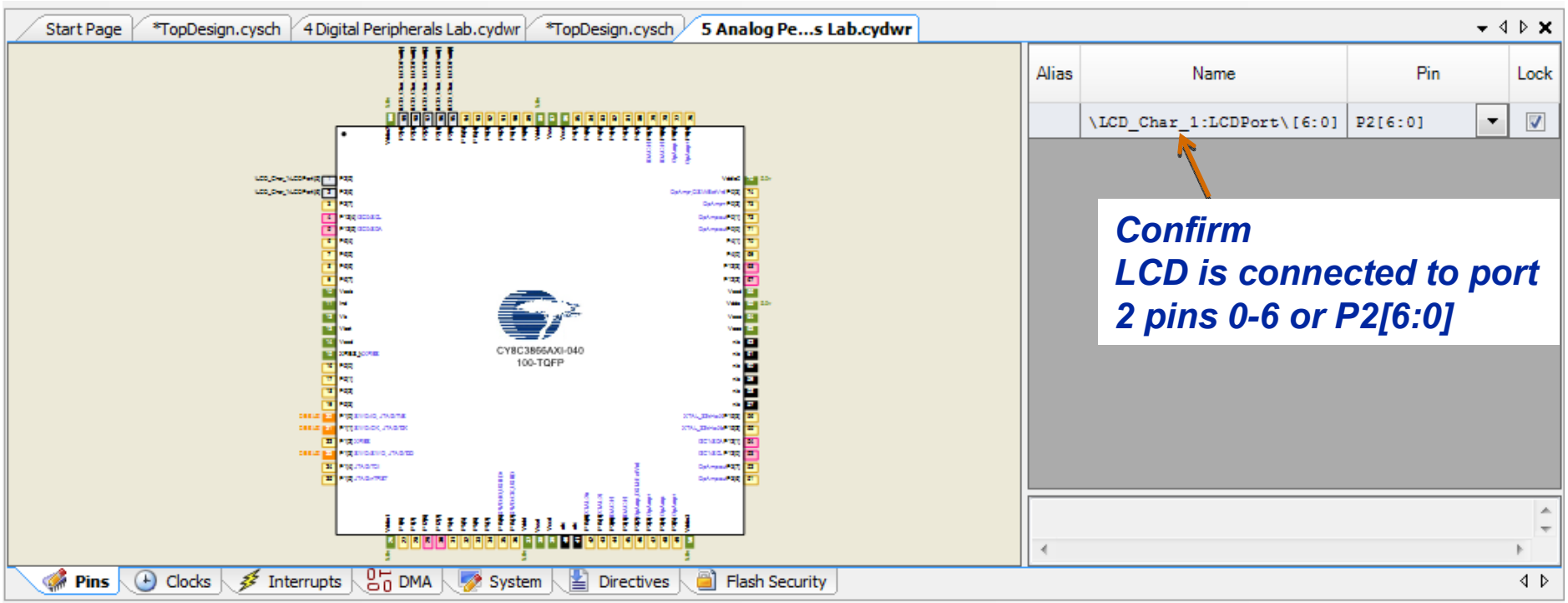
Analog Peripherals Lab

8. Your final schematic should look like this when complete



Analog Peripherals Lab

9. In the Workspace Explorer double click on the .cydwr file to open Design wide Resources (as explained in Overview lab)
10. In the Design wide Resources tab locate the section for pins on the right
11. Configure the I/O so that the LCD is connected to port 2 pins 0-6 or P2(6:0).



Analog Peripherals Lab

12. Build the project by going to the Build menu selecting Build Analog Peripherals Lab or pressing Shift + F6. This will take some time to build the project.
13. Program the board by going to the Debug menu and in the drop down click Program
14. Push the Reset button on your board located near Port D

    Verify: You should see a sine wave moving across the LCD screen.

Analog Peripherals Lab