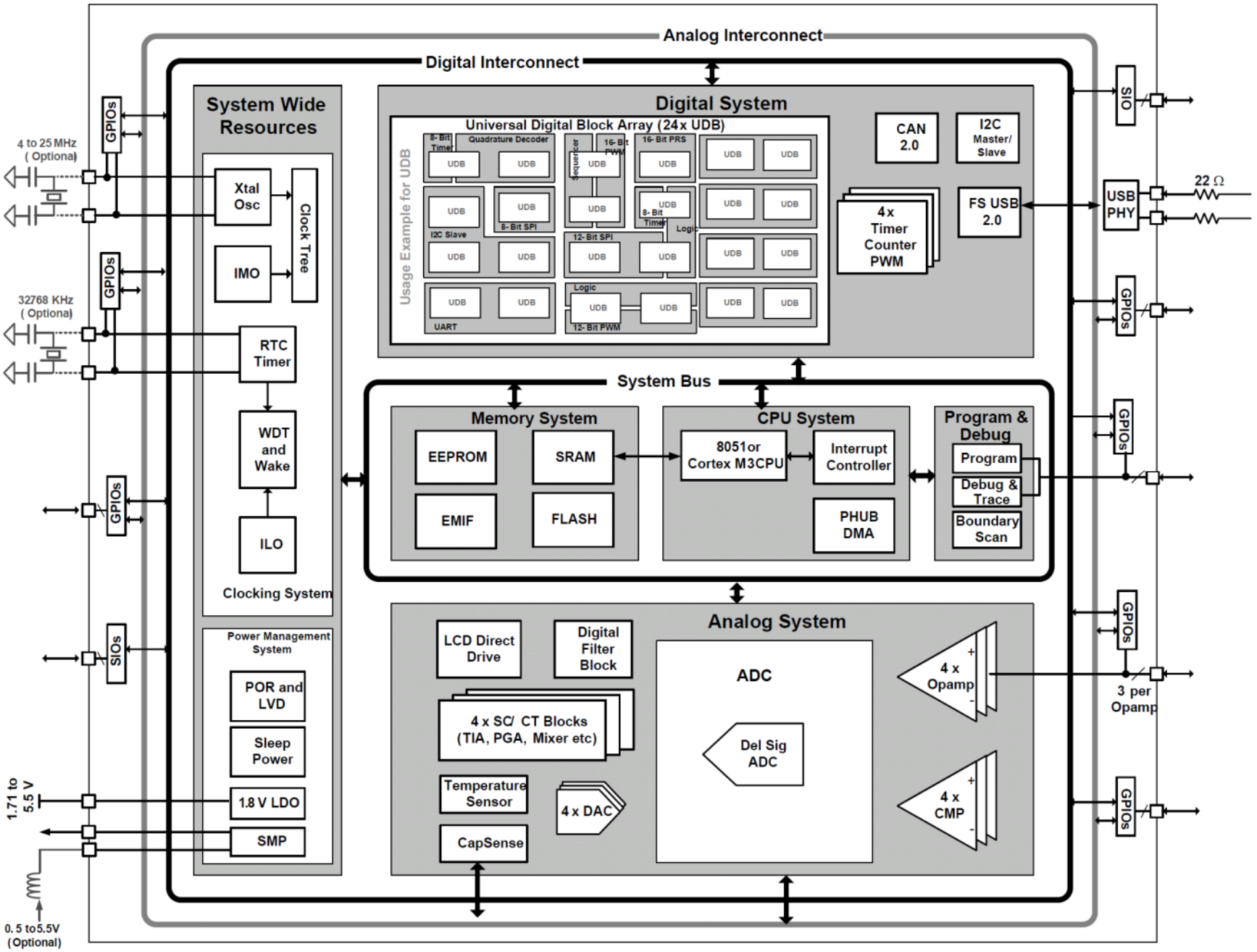
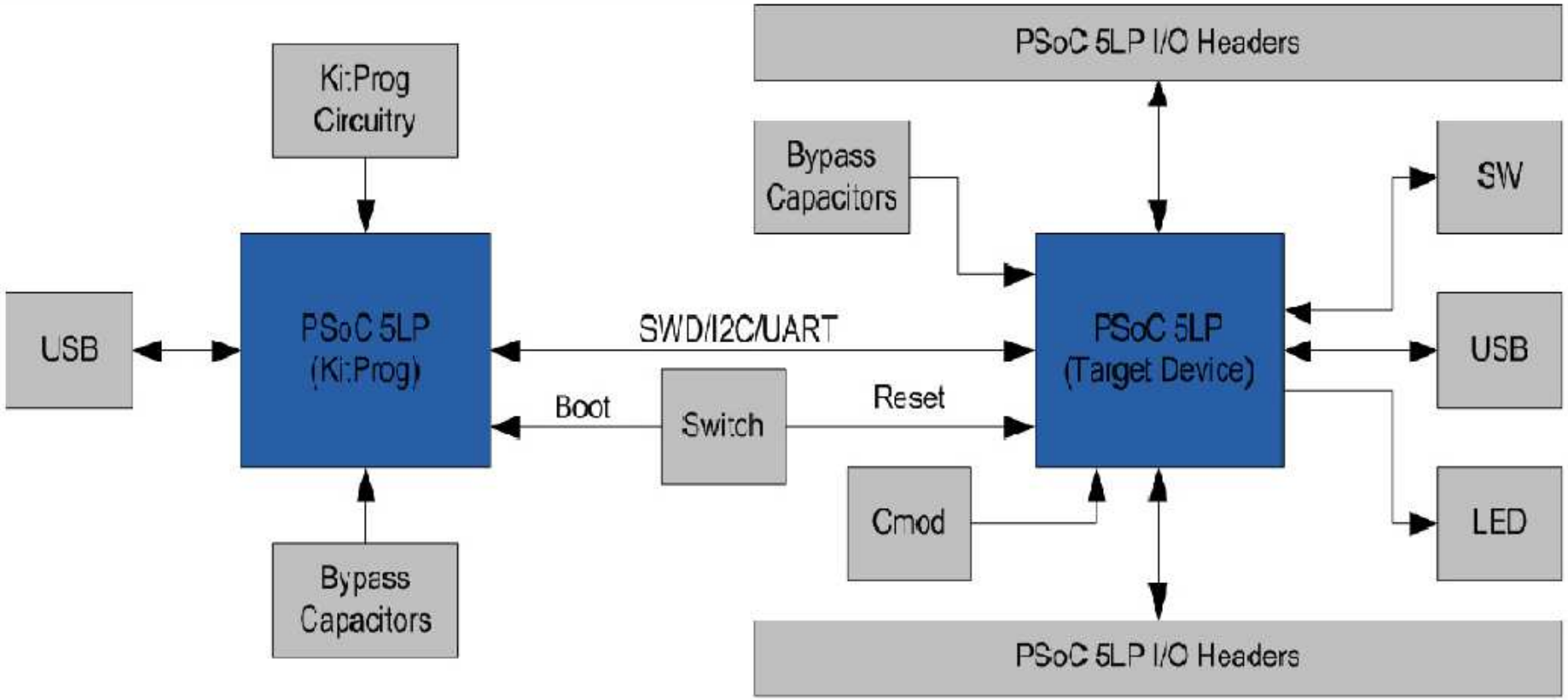
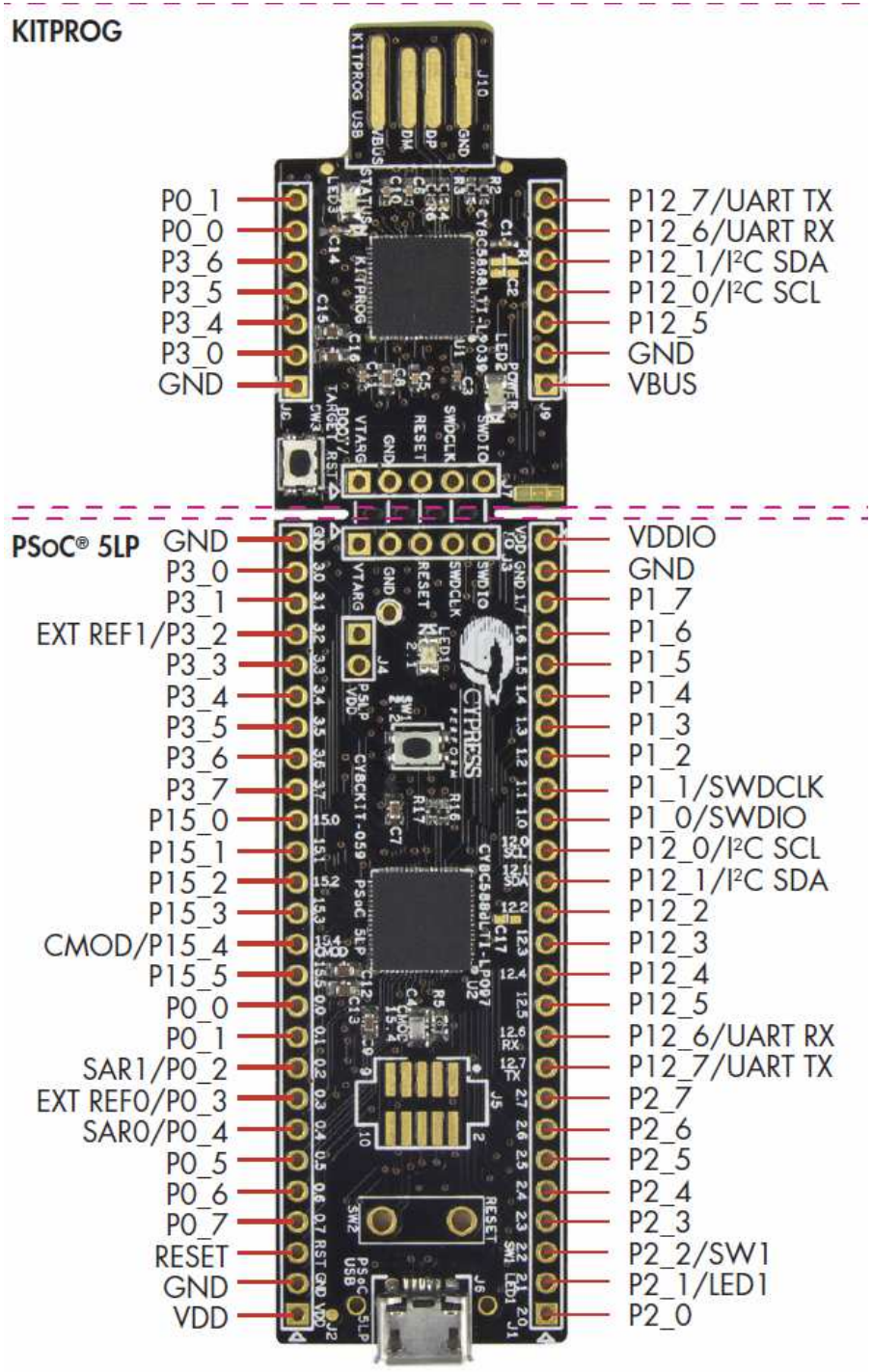
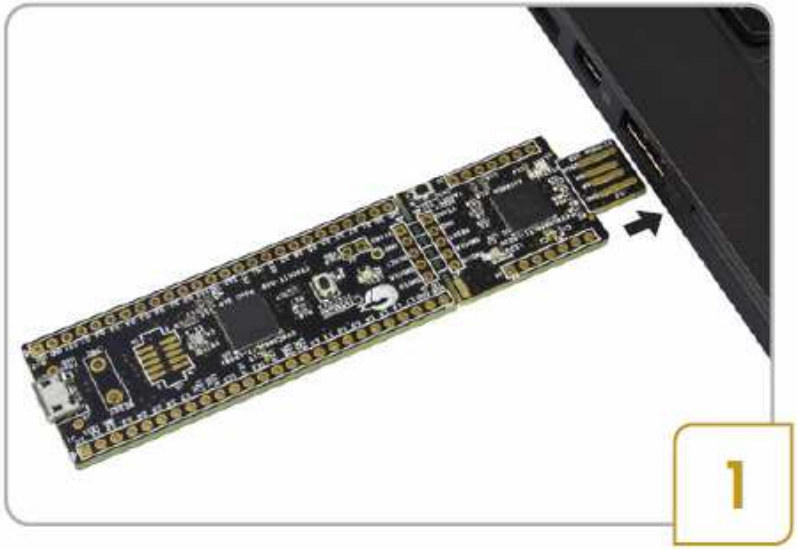
Welcome!

Cypress PSoC:



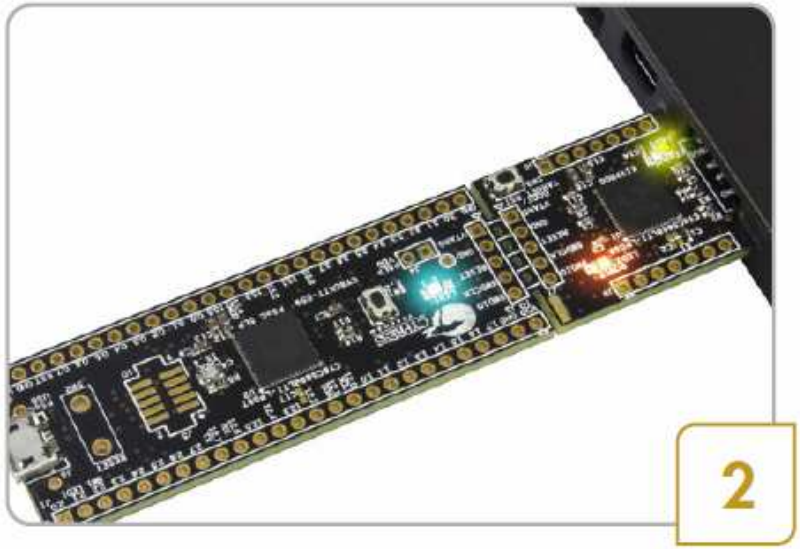

PSoC 5LP I/O Headers
Ki:Prog Circuitry
Bypass Capacitors
SW
USB
PSoC 5LP (Ki:Prog)
SWD/I2C/UART
PSoC 5LP (Target Device)
USB
Boot
Switch
Reset
Cmod
LED
Bypass Capacitors
PSoC 5LP I/O Headers


KITPROG
P0_1
P0_0
P3_6
P3_5
P3_4
P3_0
GND
P12_7/UART TX
P12_6/UART RX
P12_1/I²C SDA
P12_0/I²C SCL
P12_5
GND
VBUS
PSoC® 5LP
GND
P3_0
P3_1
EXT REF1/P3_2
P3_3
P3_4
P3_5
P3_6
P3_7
P15_0
P15_1
P15_2
P15_3
CMOD/P15_4
P15_5
P0_0
P0_1
SAR1/P0_2
EXT REF0/P0_3
SAR0/P0_4
P0_5
P0_6
P0_7
RESET
GND
VDD
VDDIO
GND
P1_7
P1_6
P1_5
P1_4
P1_3
P1_2
P1_1/SWDCLK
P1_0/SWDIO
P12_0/I²C SCL
P12_1/I²C SDA
P12_2
P12_3
P12_4
P12_5
P12_6/UART RX
P12_7/UART TX
P2_7
P2_6
P2_5
P2_4
P2_3
P2_2/SW1
P2_1/LED1
P2_0



1

- Connect the board to your computer using the USB connector



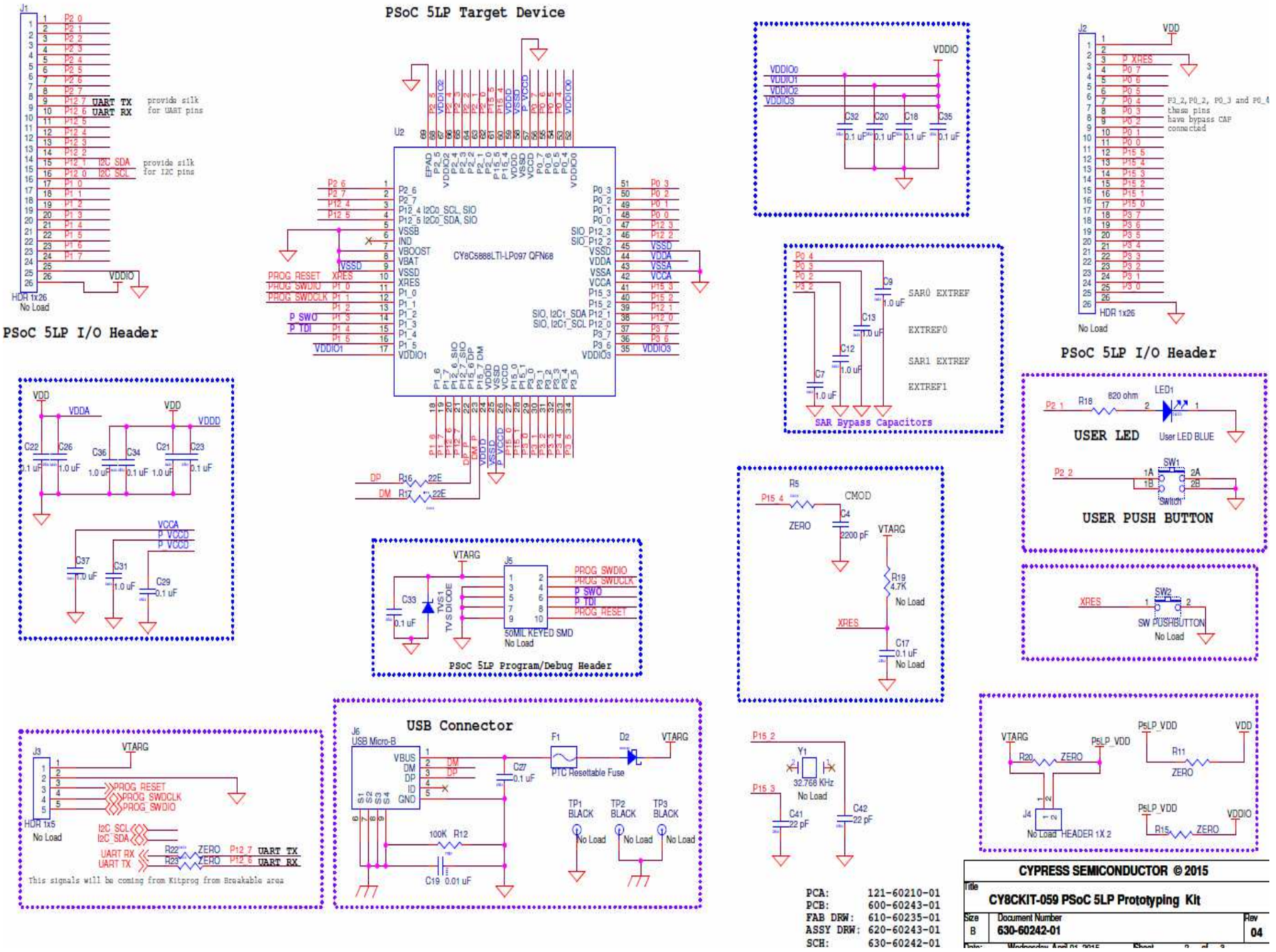
2

- Amber LED indicates power on
- Green LED indicates status
- Blue LED on the board blinks

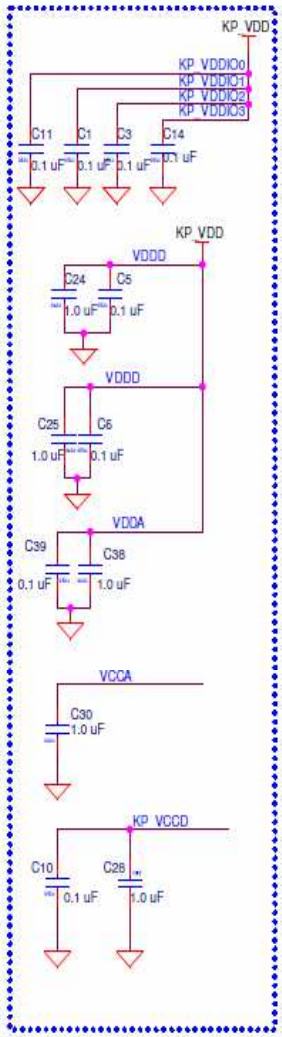
3

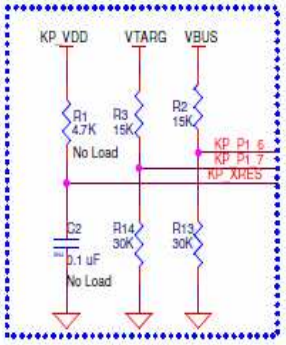
For more information on the kit, please go to the following web page:

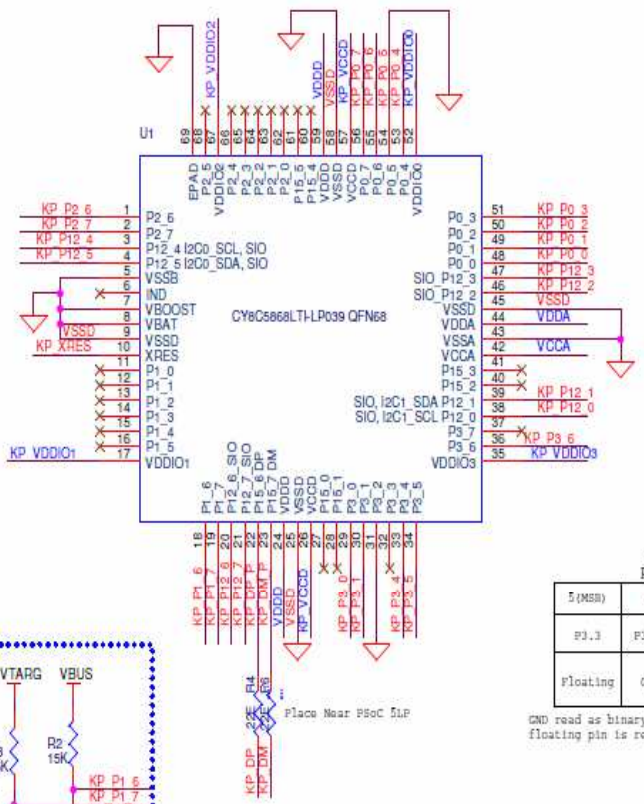
www.cypress.com/CY8CKIT-059

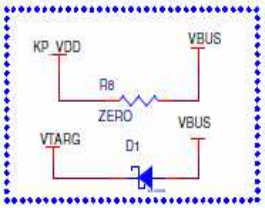

PSoC 5LP Target Device
PSoC 5LP I/O Header
PSoC 5LP I/O Header
PSoC 5LP Program/Debug Header
USB Connector
SAR Bypass Capacitors
USER LED
USER PUSH BUTTON
CY8C5888LTI-LP097 QFN68
HDR 1x26
No Load
PCA: 121-60210-01
PCB: 600-60243-01
FAB DRW: 610-60235-01
ASSY DRW: 620-60243-01
SCH: 630-60242-01
CYPRESS SEMICONDUCTOR © 2015
CY8CKIT-059 PSoC 5LP Prototyping Kit
Document Number
630-60242-01
Rev
04

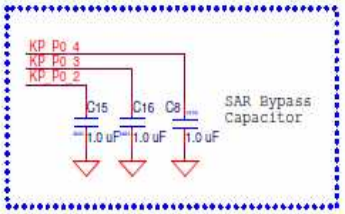
Kitprog on-board Programmer / Debugger







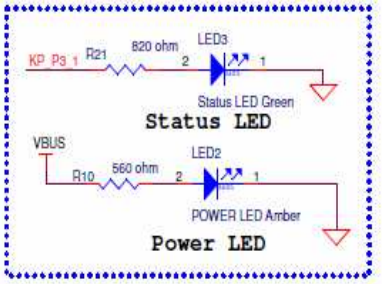


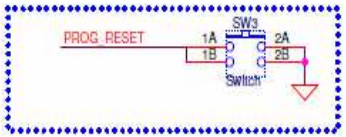


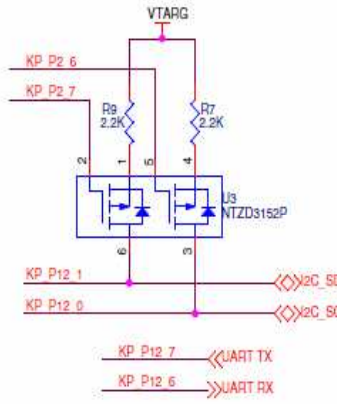
P3_3, P3_2, P0_7, P0_6, P0_5, P0_4 pins are reserved for HW REV ID

5(MSB)	4	3	2	1	0(LSB)
P3.3	P3.2	P0.7	P0.6	P0.5	P0.4
Floating	GND	Floating	Floating	GND	Floating

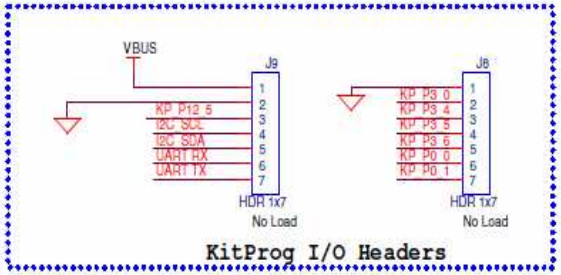
GND read as binary "1"
floating pin is read as binary "0"



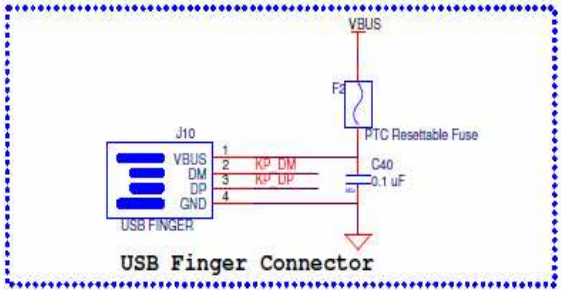




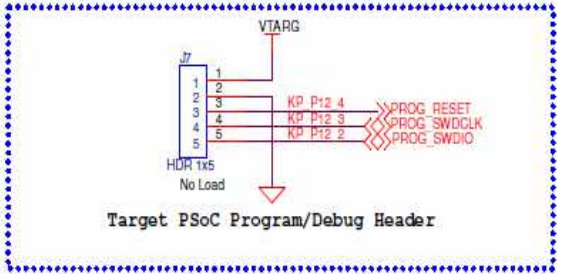
I2C & UART Connection



KitProg I/O Headers



USB Finger Connector

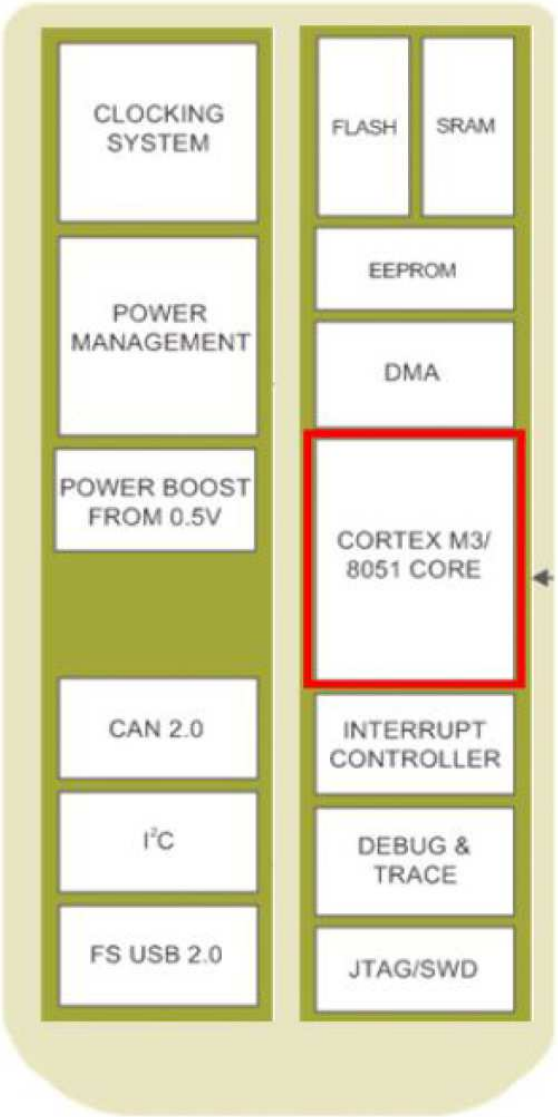


Target PSoC Program/Debug Header

PCA:	121-60210-01
PCB:	600-60243-01
FAB DRW:	610-60235-01
ASSY DRW:	620-60243-01
SCH:	630-60242-01

Title: CY8CKIT-059 PSoC 5LP Prototyping Kit

Size: B | Document Number: 630-60242-01



ARM Cortex-M3

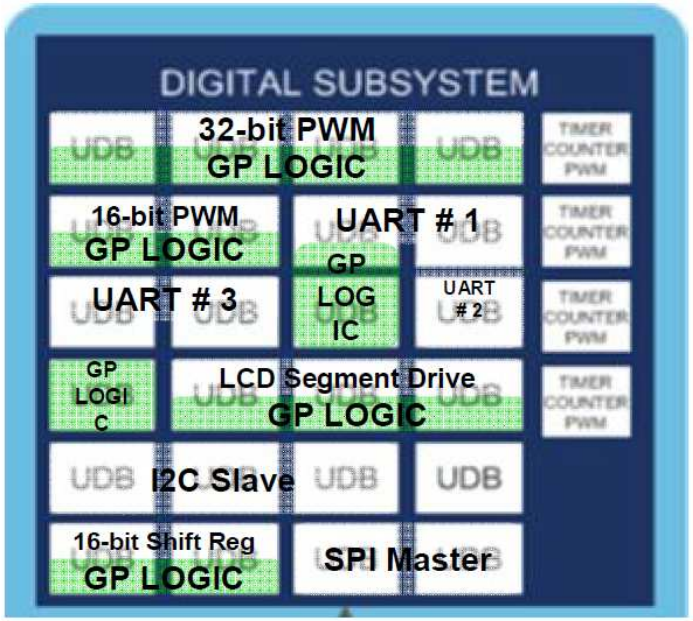
- Industry's leading embedded CPU company
- Broad support for middleware and applications
- Upto 67 MHz; 83 DMIPS
- Enhanced v7 ARM architecture
- Thumb2 Instruction Set
- 16- and 32-bit Instructions (no mode switching)
- 32-bit ALU; Hardware multiply and divide
- Single cycle 3-stage pipeline; Harvard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

8051

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

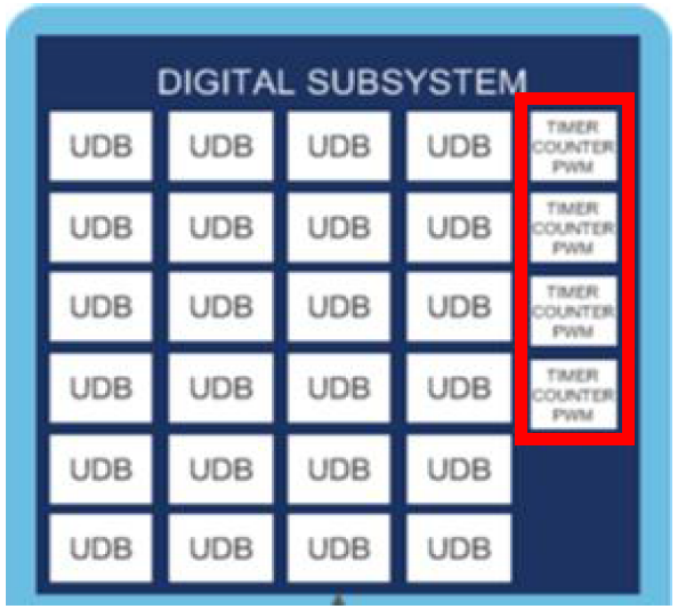
Universal Digital Block Arrays (UDBs)

- 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



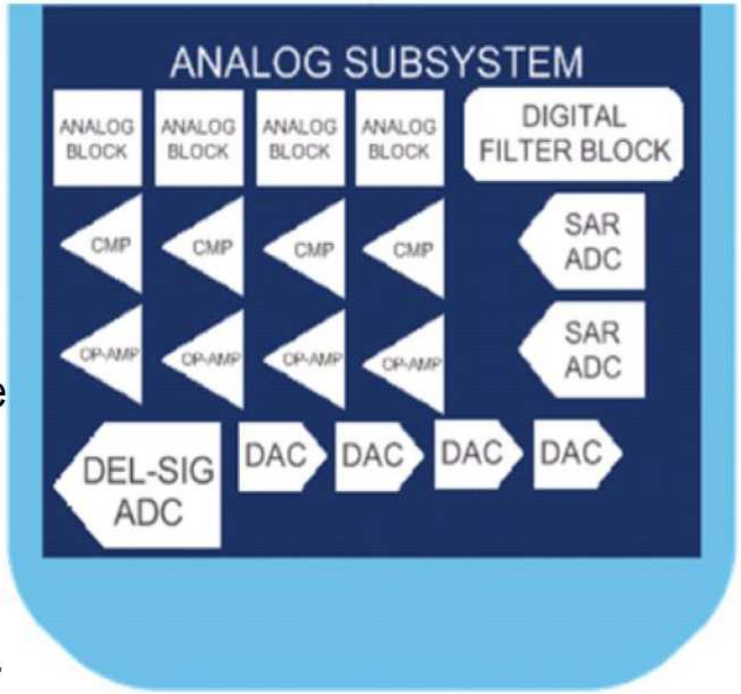
Organized 8/16-bit Timer/Counter/PWM Blocks

- 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



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
- DAC's: 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

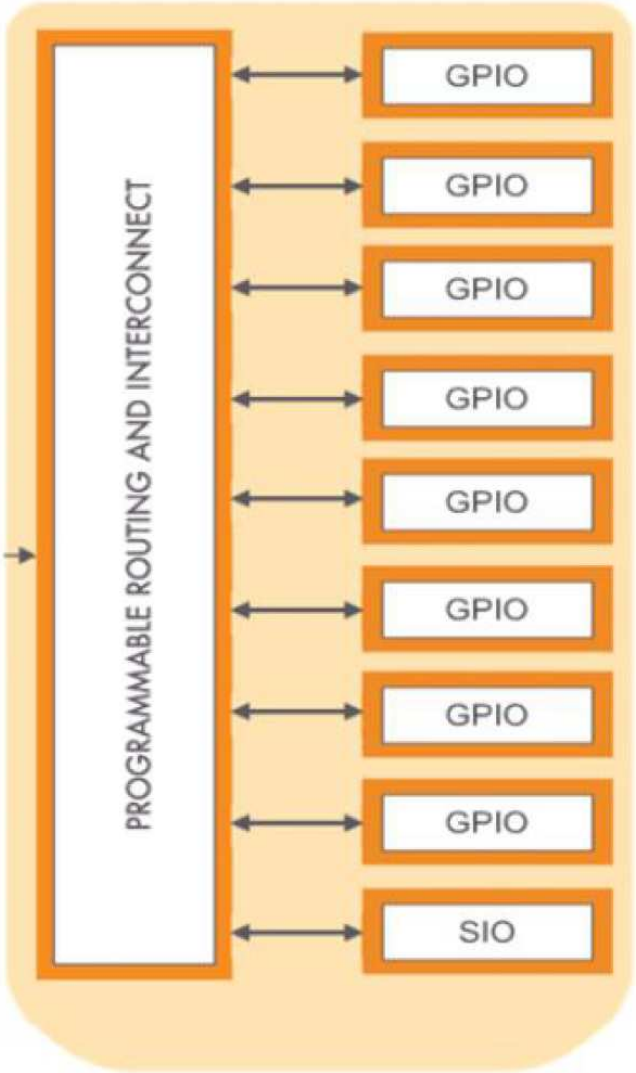


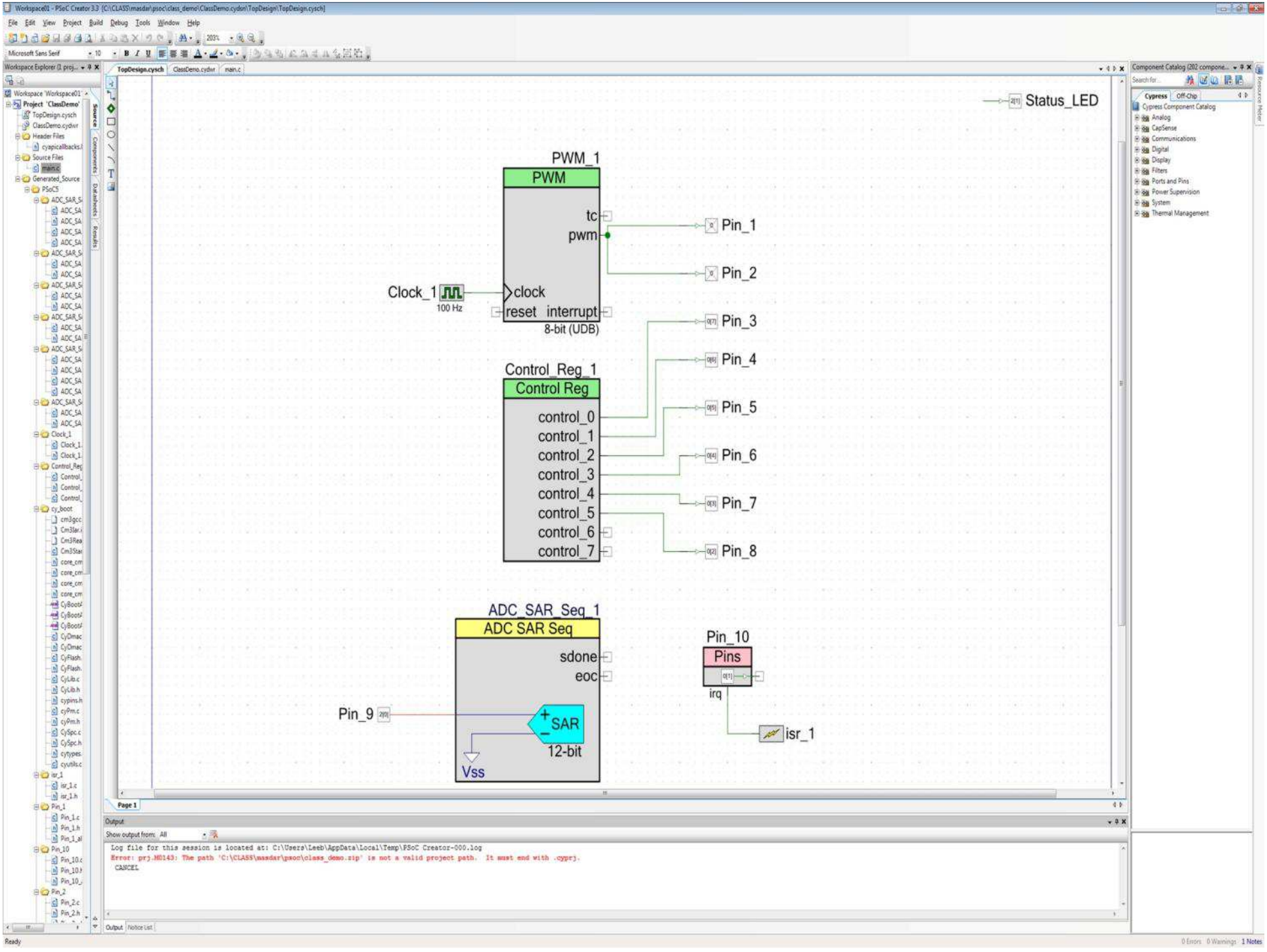
Input / Output System

- 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

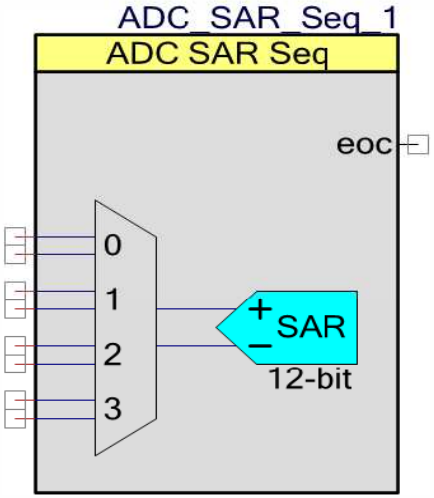



Status_LED
PWM_1
PWM
tc
pwm
clock
reset
interrupt
8-bit (UDB)
Clock_1
100 Hz
Pin_1
Pin_2
Pin_3
Pin_4
Pin_5
Pin_6
Pin_7
Pin_8
Control_Reg_1
Control Reg
control_0
control_1
control_2
control_3
control_4
control_5
control_6
control_7
ADC_SAR_Seq_1
ADC SAR Seq
sdone
eoc
Pin_9
SAR
12-bit
Vss
Pin_10
Pins
irq
isr_1
Workspace01 - PSoC Creator 3.3
File Edit View Project Build Debug Tools Window Help
Workspace Explorer
TopDesign.cysch
ClassDemo.cydwr
main.c
Component Catalog
Cypress Component Catalog
Analog
CapSense
Communications
Digital
Display
Filters
Ports and Pins
Power Supervision
System
Thermal Management
Output
Log file for this session is located at: C:\Users\Leeb\AppData\Local\Temp\PSoC Creator-000.log
Error: prj.M0143: The path 'C:\CLASS\masdar\psoc\class_demo.zip' is not a valid project path. It must end with .cyprj.
CANCEL
Page 1
Ready

Sequencing Successive Approximation ADC (ADC_SAR_Seq)

2.0

Features



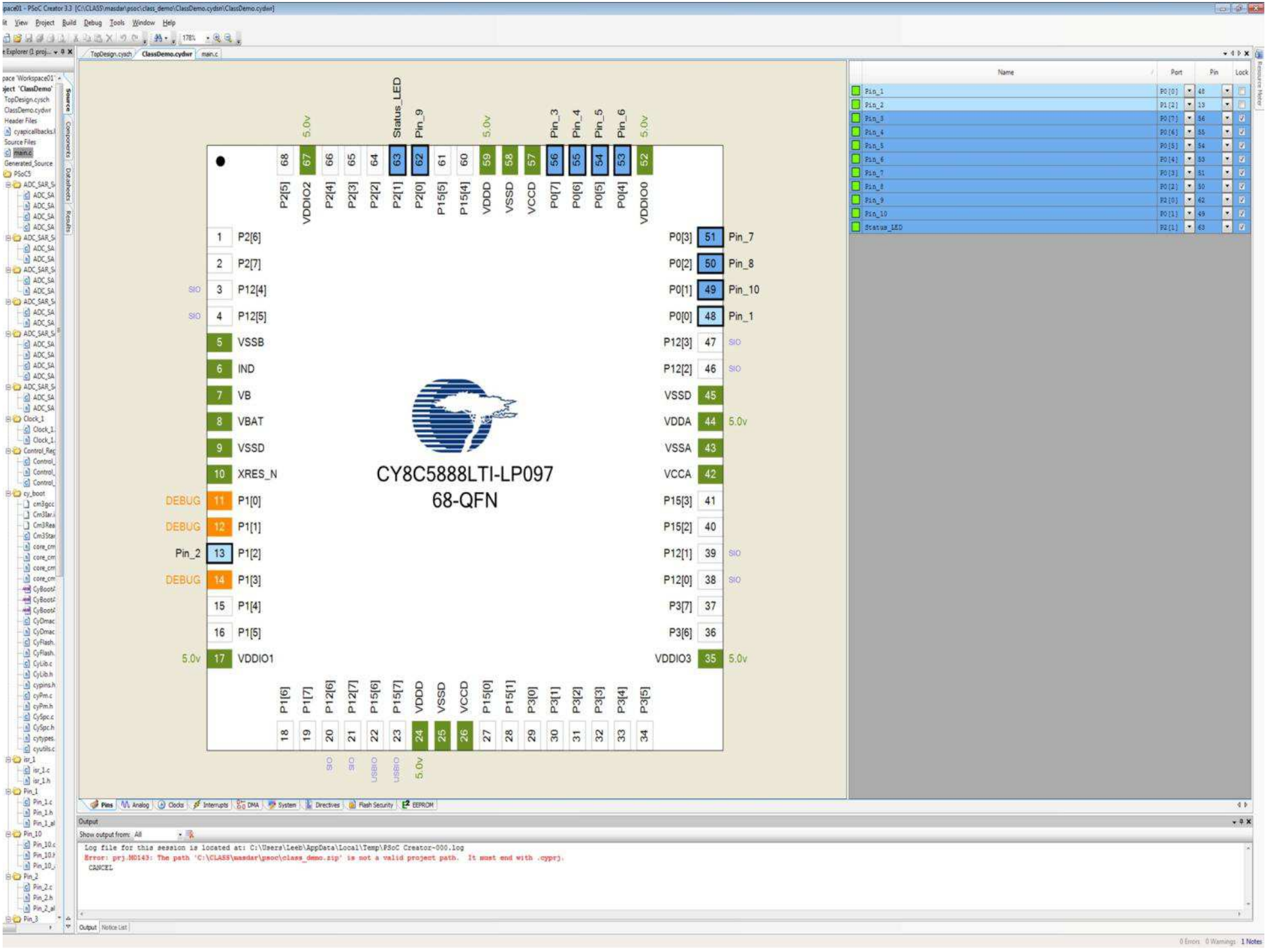
- Supports PSoC 5LP devices
- Selectable resolution (8, 10 or 12 bit) and sample rate (up to 1 Msps)
- Scans up to 64 single ended or 32 differential channels automatically, or just a single input

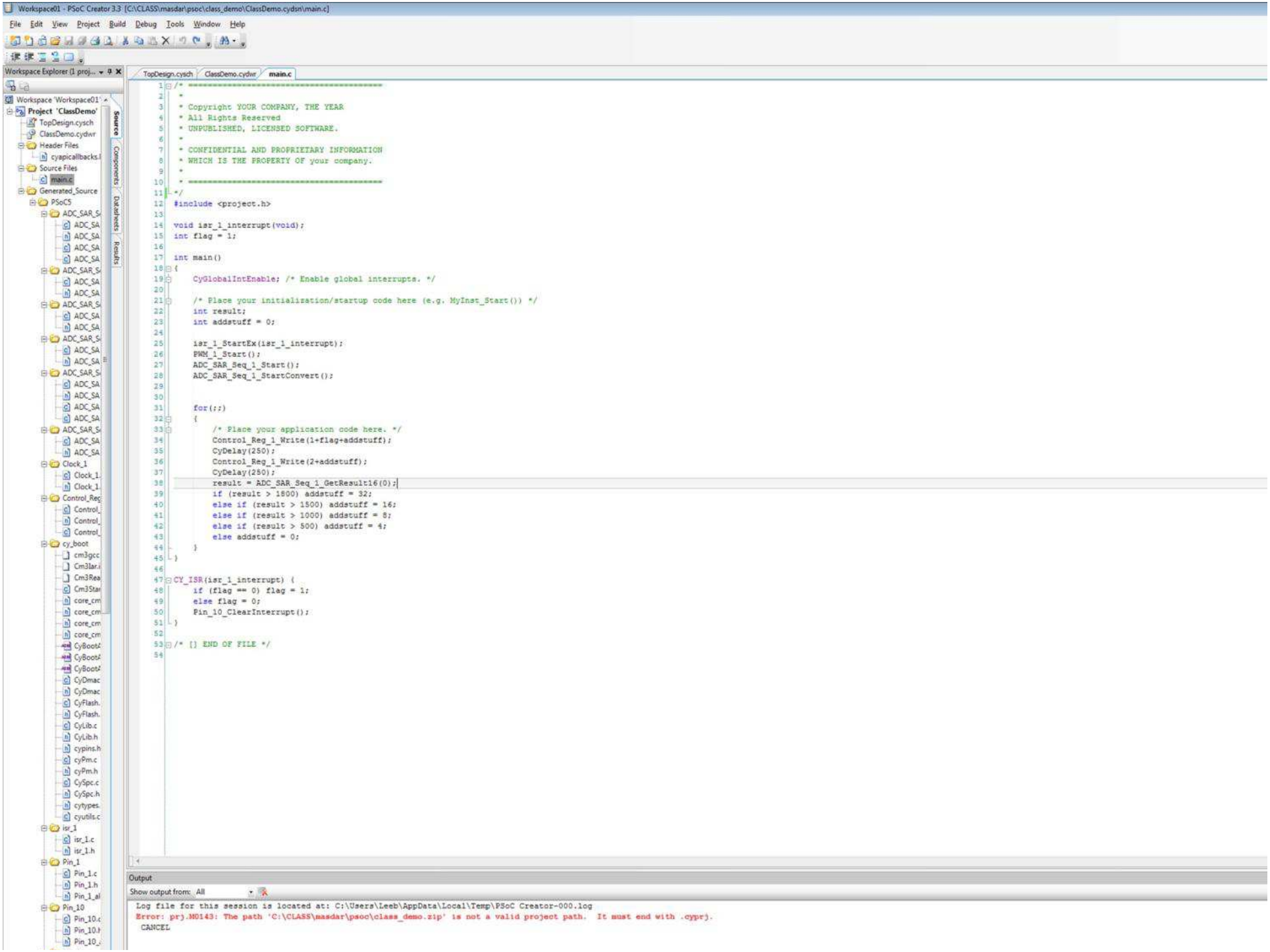
Note Only the GPIOs can be connected to the channels inputs. The actual maximum number of input channels depends on the number of routable analog GPIOs that are available on a specific PSoC part and package.

General Description

The Sequencing SAR ADC component enables makes it possible for you to configure and then use the different operational modes of the SAR ADC on PSoC 5LP. You also have schematic level and firmware level support for seamless use of the Sequencing SAR ADC in PSoC Creator designs and projects. You are able to configure multiple analog channels that are automatically scanned with the results placed in individual SRAM locations.

When to Use the ADC_SAR_Seq


Status_LED
Pin_9
Pin_3
Pin_4
Pin_5
Pin_6
5.0v
P2[5]
VDDIO2
P2[4]
P2[3]
P2[2]
P2[1]
P2[0]
P15[5]
P15[4]
VDDD
VSSD
VCCD
P0[7]
P0[6]
P0[5]
P0[4]
VDDIO0
1 P2[6]
2 P2[7]
SIO 3 P12[4]
SIO 4 P12[5]
5 VSSB
6 IND
7 VB
8 VBAT
9 VSSD
10 XRES_N
DEBUG 11 P1[0]
DEBUG 12 P1[1]
Pin_2 13 P1[2]
DEBUG 14 P1[3]
15 P1[4]
16 P1[5]
5.0v 17 VDDIO1
P0[3] 51 Pin_7
P0[2] 50 Pin_8
P0[1] 49 Pin_10
P0[0] 48 Pin_1
P12[3] 47 SIO
P12[2] 46 SIO
VSSD 45
VDDA 44 5.0v
VSSA 43
VCCA 42
P15[3] 41
P15[2] 40
P12[1] 39 SIO
P12[0] 38 SIO
P3[7] 37
P3[6] 36
VDDIO3 35 5.0v
CY8C5888LTI-LP097
68-QFN
P1[6]
P1[7]
P12[6]
P12[7]
P15[6]
P15[7]
VDDD
VSSD
VCCD
P15[0]
P15[1]
P3[0]
P3[1]
P3[2]
P3[3]
P3[4]
P3[5]
SIO
SIO
USBIO
USBIO
5.0v
Name
Port
Pin
Lock
Pin_1 P0[0] 48
Pin_2 P1[2] 13
Pin_3 P0[7] 56
Pin_4 P0[6] 55
Pin_5 P0[5] 54
Pin_6 P0[4] 53
Pin_7 P0[3] 51
Pin_8 P0[2] 50
Pin_9 P2[0] 62
Pin_10 P0[1] 49
Status_LED P2[1] 63
Output
Log file for this session is located at: C:\Users\Leeb\AppData\Local\Temp\PSoC Creator-000.log
Error: prj.MD143: The path 'C:\CLASS\masdar\psoc\class_demo.zip' is not a valid project path. It must end with .cyprj.
CANCEL


Workspace01 - PSoC Creator 3.3 [C:\CLASS\masdar\psoc\class_demo\ClassDemo.cydsn\main.c]
File Edit View Project Build Debug Tools Window Help
Workspace Explorer (1 proj...
Workspace 'Workspace01'
Project 'ClassDemo'
TopDesign.cysch
ClassDemo.cydwr
Header Files
cyapicallbacks.h
Source Files
main.c
Generated_Source
PSoC5
Clock_1
Control_Reg
cy_boot
isr_1
Pin_1
Pin_10
TopDesign.cysch
ClassDemo.cydwr
main.c
/* ==
*
* Copyright YOUR COMPANY, THE YEAR
* All Rights Reserved
* UNPUBLISHED, LICENSED SOFTWARE.
*
* CONFIDENTIAL AND PROPRIETARY INFORMATION
* WHICH IS THE PROPERTY OF your company.
*
* ==
*/
#include <project.h>
void isr_1_interrupt(void);
int flag = 1;
int main()
{
CyGlobalIntEnable; /* Enable global interrupts. */
/* Place your initialization/startup code here (e.g. MyInst_Start()) */
int result;
int addstuff = 0;
isr_1_StartEx(isr_1_interrupt);
PWM_1_Start();
ADC_SAR_Seq_1_Start();
ADC_SAR_Seq_1_StartConvert();
for(;;)
{
/* Place your application code here. */
Control_Reg_1_Write(1+flag+addstuff);
CyDelay(250);
Control_Reg_1_Write(2+addstuff);
CyDelay(250);
result = ADC_SAR_Seq_1_GetResult16(0);
if (result > 1800) addstuff = 32;
else if (result > 1500) addstuff = 16;
else if (result > 1000) addstuff = 8;
else if (result > 500) addstuff = 4;
else addstuff = 0;
}
}
CY_ISR(isr_1_interrupt) {
if (flag == 0) flag = 1;
else flag = 0;
Pin_10_ClearInterrupt();
}
/* [] END OF FILE */
Output
Show output from: All
Log file for this session is located at: C:\Users\Leeb\AppData\Local\Temp\PSoC Creator-000.log
Error: prj.M0143: The path 'C:\CLASS\masdar\psoc\class_demo.zip' is not a valid project path. It must end with .cyprj.
CANCEL

```
#include <project.h>
void isr_1_interrupt(void);
int flag = 1;
int main()
{
    CyGlobalIntEnable; /* Enable global interrupts. */
    int result;
    int addstuff = 0;

    isr_1_StartEx(isr_1_interrupt);
    PWM_1_Start();
    ADC_SAR_Seq_1_Start();
    ADC_SAR_Seq_1_StartConvert();
    for(;;)
    {
        Control_Reg_1_Write(1+flag+addstuff);
        CyDelay(250);
        Control_Reg_1_Write(2+addstuff);
        CyDelay(250);
        result = ADC_SAR_Seq_1_GetResult16(0);
        if (result > 1800) addstuff = 32;
        else if (result > 1500) addstuff = 16;
        else if (result > 1000) addstuff = 8;
        else if (result > 500) addstuff = 4;
        else addstuff = 0;
    }
}
CY_ISR(isr_1_interrupt) {
    if (flag == 0) flag = 1;
    else flag = 0;
    Pin_10_ClearInterrupt();
}
```