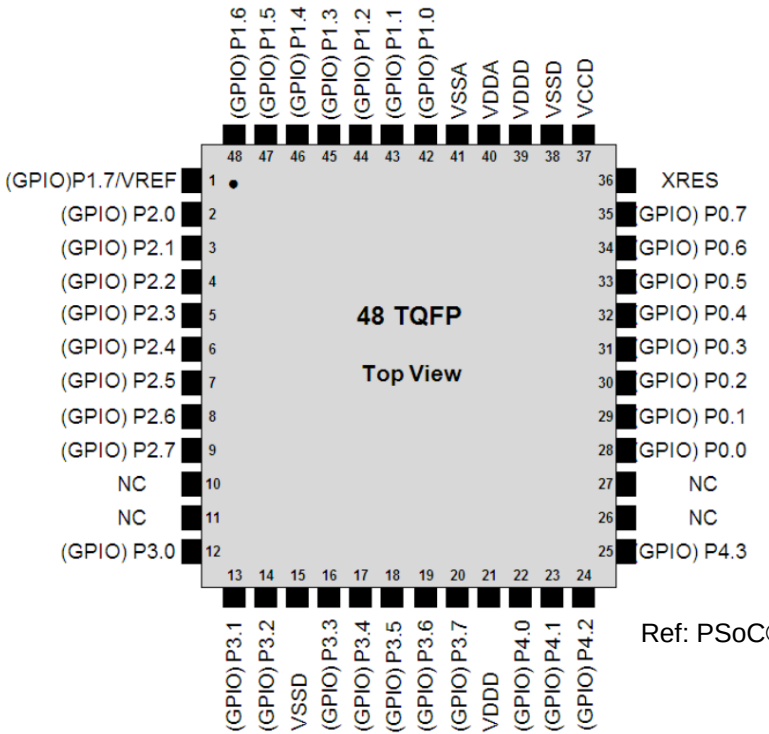
IO Pins, Drive Modes

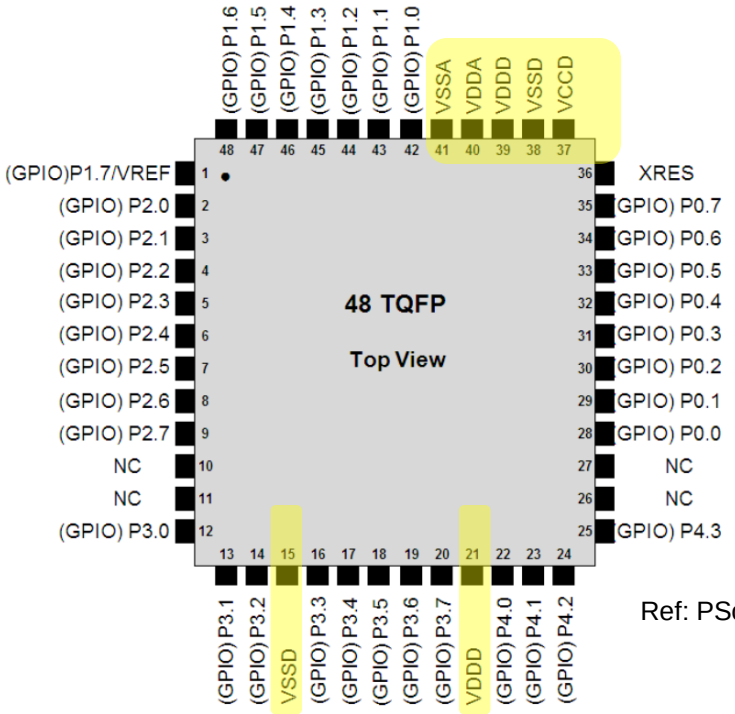
Types of Pins



- **Power Pins**
- **General Purpose IO Pins**
- **Special Function Pins**

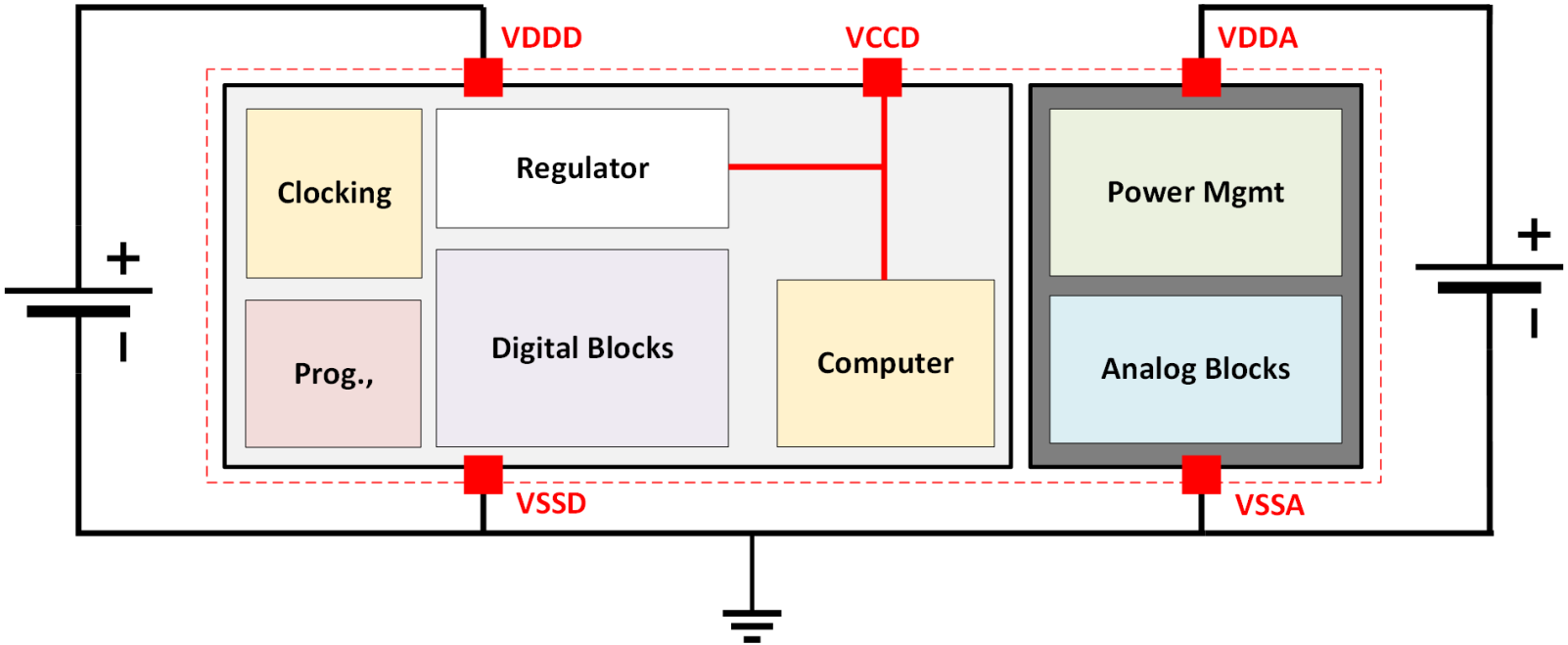
Ref: PSoC® 4: PSoC 4200 Family Datasheet

Power Supply Pins



Ref: PSoC® 4: PSoC 4200 Family Datasheet

Power System



Specification

Table 1. Absolute Maximum Ratings[1]

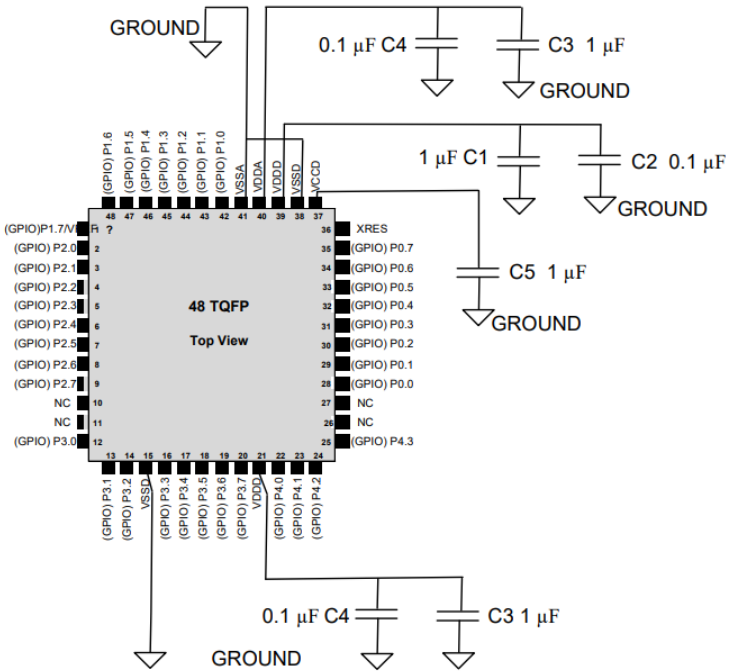
Spec ID#	Parameter	Description	Min	Typ	Max	Units	Details/ Conditions
SID1	V_{DDD_ABS}	Digital supply relative to V_{SSD}	–0.5	–	6	V	Absolute max
SID2	V_{CCD_ABS}	Direct digital core voltage input relative to Vssd	–0.5	–	1.95	V	Absolute max

Table 2. DC Specifications

Spec ID#	Parameter	Description	Min	Typ	Max	Units	Details/ Conditions
SID53	V_{DD}	Power Supply Input Voltage ($V_{DDA} = V_{DDD} = V_{DD}$)	1.8	–	5.5	V	With regulator enabled
SID255	V_{DDD}	Power Supply Input Voltage unregulated	1.71	1.8	1.89	V	Internally unregulated supply
SID54	V_{CCD}	Output voltage (for core logic)	–	1.8	–	V	

Ref: PSoC® 4: PSoC 4200 Family Datasheet

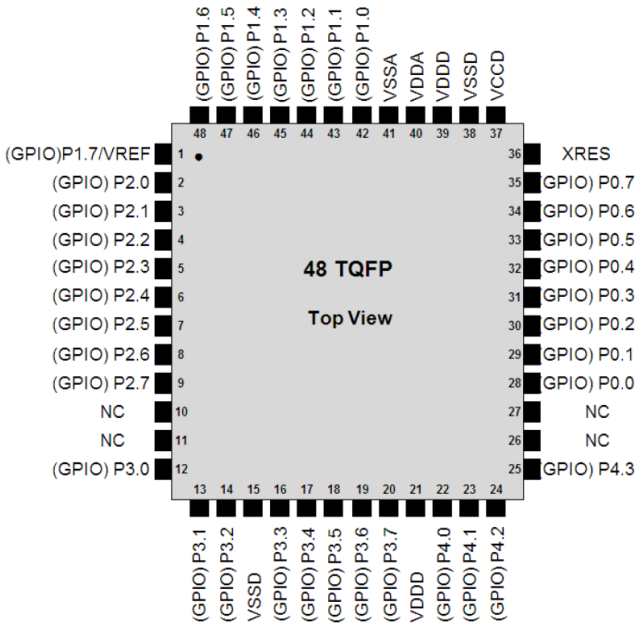
Bypass Capacitors



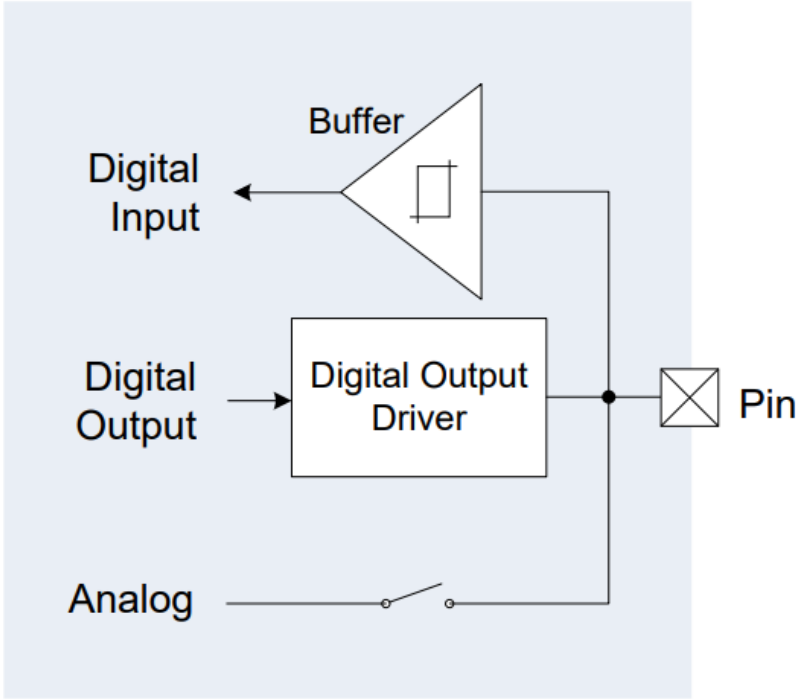
Power Supply	Bypass Capacitors
V_{DDD}–V_{SS}	0.1-µF ceramic at each pin (C2, C6) plus bulk capacitor 1 µF to 10 µF (C1). Total capacitance may be greater than 10 µF.
V_{DDA}–V_{SSA}	0.1-µF ceramic at pin (C4). Additional 1 µF to 10 µF (C3) bulk capacitor. Total capacitance may be greater than 10 µF.
V_{CCD}–V_{SS}	1-µF ceramic capacitor at the VCCD pin (C5)
V_{REF}–V_{SSA} (optional)	The internal bandgap may be bypassed with a 1-µF to 10-µF capacitor. Total capacitance may be greater than 10 µF.

Ref: PSoC® 4: PSoC 4200 Family Datasheet

GPIO Pins

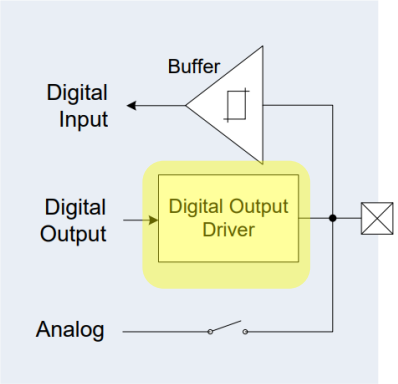


Organized into GPIO Ports (e.g., P0, P1, P2, P3, P4)

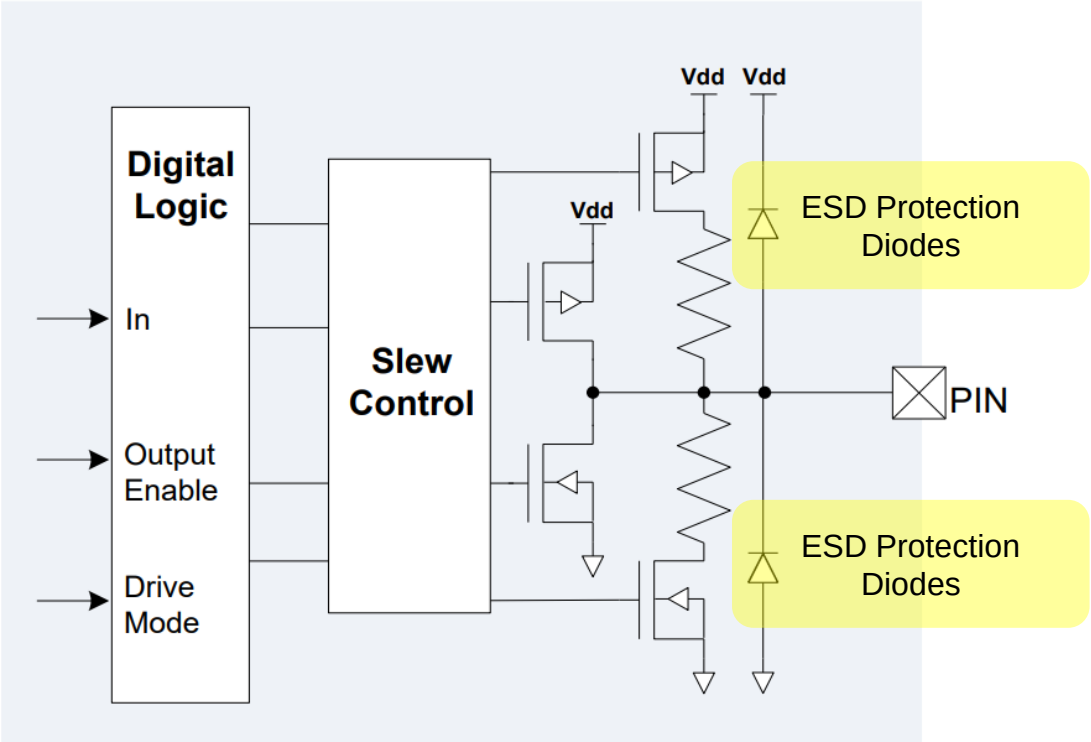


Ref: AN86439 – PSoC® 4 – Using GPIO Pins

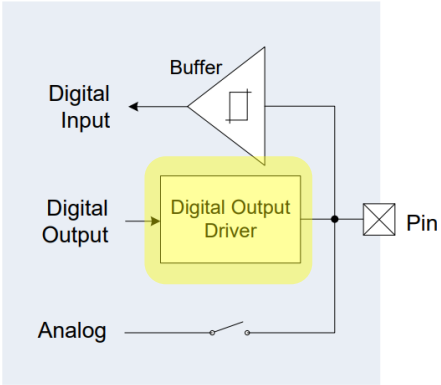
Output Driver

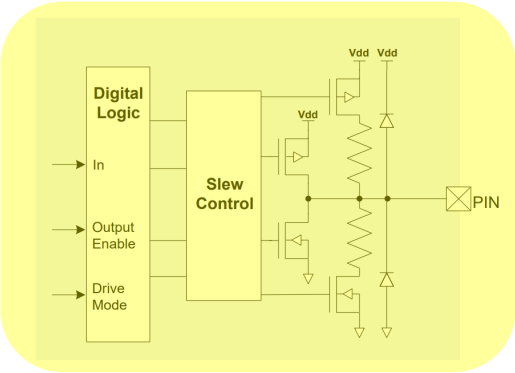


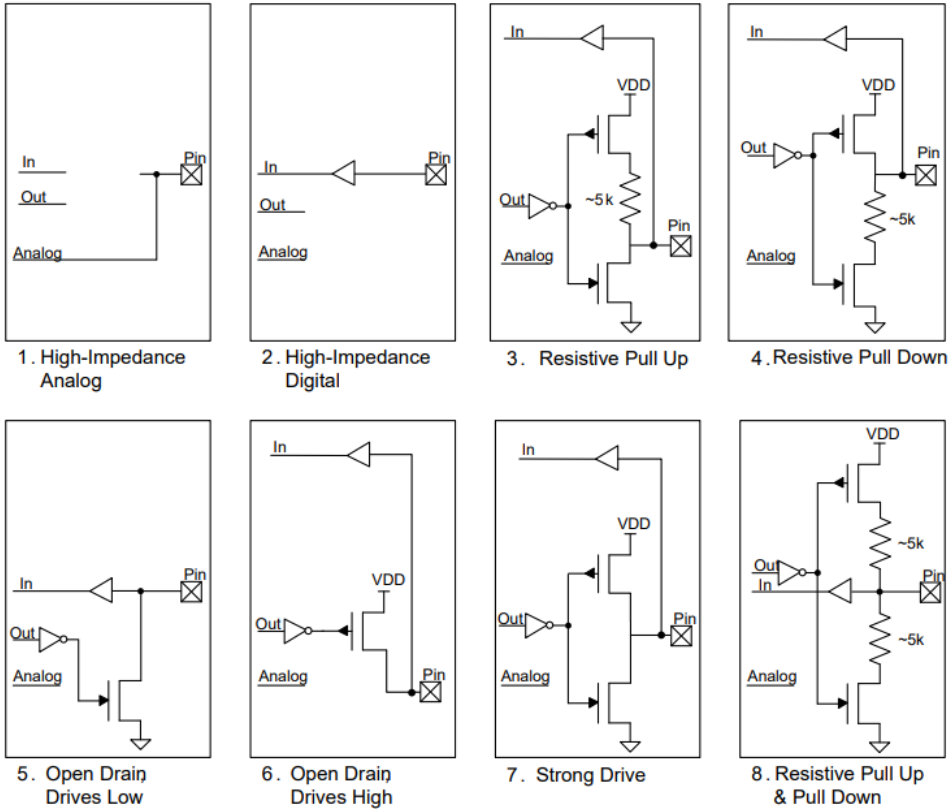
Ref: AN86439 – PSoC® 4 – Using GPIO Pins



Drive Modes

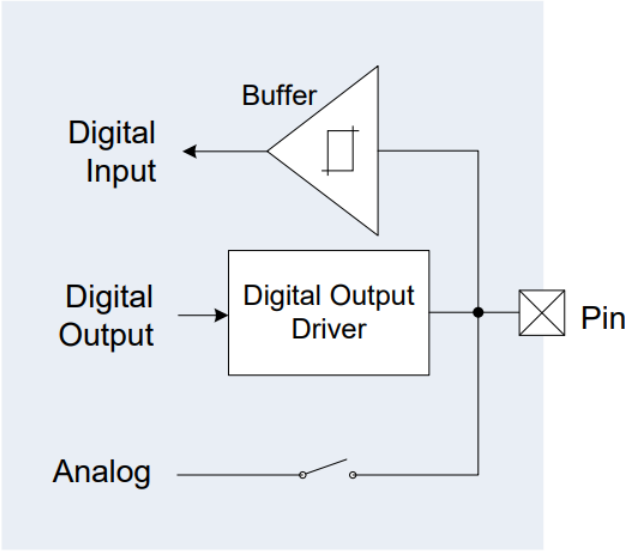




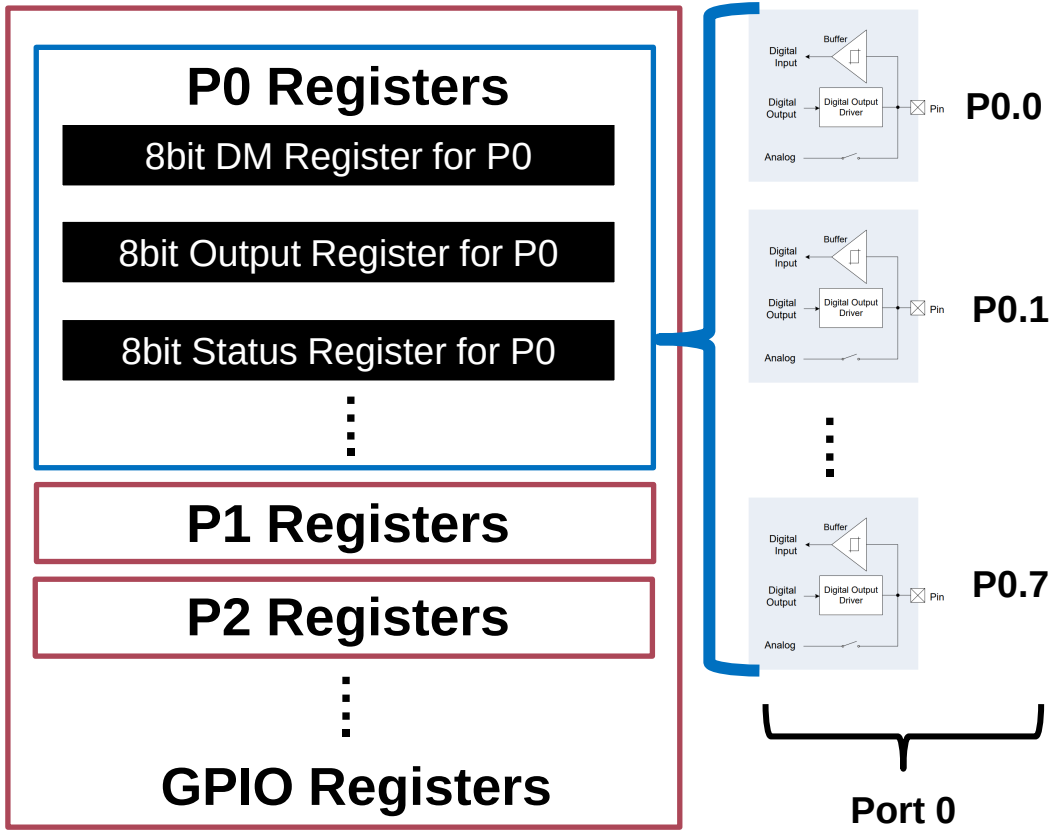


Ref: AN86439 – PSoC® 4 – Using GPIO Pins

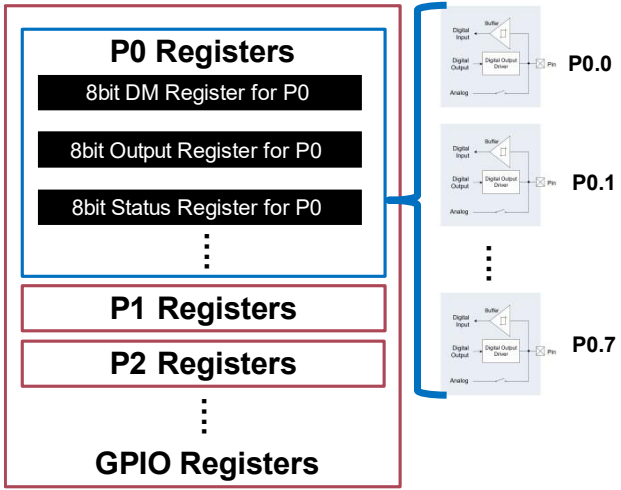
Configuring GPIOs



Ref: PSoC® 4: PSoC 4200 Family Datasheet



GPIO Register Control



Instructions → Processor → GPIO Registers → GPIO

↑

Analog or Digital Blocks

(e.g., SPI, I2C, ADC, DAC …)

Ref: PSoC® 4: PSoC 4200 Family Datasheet

Specifications

Table 1. Absolute Maximum Ratings[1]

Spec ID#	Parameter	Description	Min	Typ	Max	Units	Details/ Conditions
SID1	V_{DDD_ABS}	Digital supply relative to V_{SSD}	–0.5	–	6	V	Absolute max
SID2	V_{CCD_ABS}	Direct digital core voltage input relative to Vssd	–0.5	–	1.95	V	Absolute max
SID3	V_{GPIO_ABS}	GPIO voltage	–0.5	–	V_{DD}+0.5	V	Absolute max
SID4	I_{GPIO_ABS}	Maximum current per GPIO	–25	–	25	mA	Absolute max

Ref: PSoC® 4: PSoC 4200 Family Datasheet

Specifications (…)

Table 4. GPIO DC Specifications

Spec ID#	Parameter	Description	Min	Typ	Max	Units	Details/ Conditions
SID57	V_{IH}[2]	Input voltage high threshold	0.7 × V_{DDD}	–	–	V	CMOS Input
SID58	V_{IL}	Input voltage low threshold	–	–	0.3 × V_{DDD}	V	CMOS Input
SID59	V_{OH}	Output voltage high level	V_{DDD} –0.6	–	–	V	I_{OH} = 4 mA at 3-V V_{DDD}
SID60	V_{OH}	Output voltage high level	V_{DDD} –0.5	–	–	V	I_{OH} = 1 mA at 1.8-V V_{DDD}
SID61	V_{OL}	Output voltage low level	–	–	0.4	V	I_{OL} = 4 mA at 1.8-V V_{DDD}
SID62	V_{OL}	Output voltage low level	–	–	0.6	V	I_{OL} = 8 mA at 3-V V_{DDD}
SID62A	V_{OL}	Output voltage low level	–	–	0.4	V	I_{OL} = 3 mA at 3-V V_{DDD}
SID63	R_{PULLUP}	Pull-up resistor	3.5	5.6	8.5	kΩ	
SID64	$R_{PULLDOWN}$	Pull-down resistor	3.5	5.6	8.5	kΩ	

Ref: PSoC® 4: PSoC 4200 Family Datasheet

Specifications (…)

Table 4. GPIO DC Specifications

Spec ID#	Parameter	Description	Min	Typ	Max	Units	Details/ Conditions
SID69	I_{DIODE}	Current through protection diode to V_{DD}/Vss	–	–	100	µA	Guaranteed by characterization
SID69A	I_{TOT_GPIO}	Maximum Total Source or Sink Chip Current	–	–	200	mA	Guaranteed by characterization

Table 5. GPIO AC Specifications

(Guaranteed by Characterization)

Spec ID#	Parameter	Description	Min	Typ	Max	Units	Details/ Conditions
SID70	T_{RISEF}	Rise time in fast strong mode	2	–	12	ns	3.3-V V_{DDD}, Cload = 25 pF
SID71	T_{FALLF}	Fall time in fast strong mode	2	–	12	ns	3.3-V V_{DDD}, Cload = 25 pF
SID72	T_{RISES}	Rise time in slow strong mode	10	–	60	ns	3.3-V V_{DDD}, Cload = 25 pF
SID73	T_{FALLS}	Fall time in slow strong mode	10	–	60	ns	3.3-V V_{DDD}, Cload = 25 pF

Ref: PSoC® 4: PSoC 4200 Family Datasheet

Specifications (…)

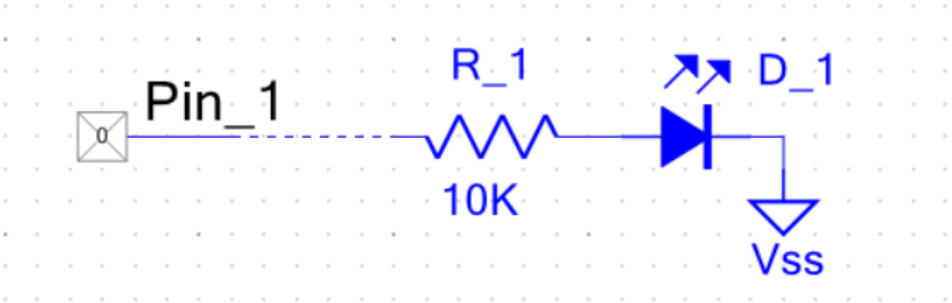
Table 5. GPIO AC Specifications

(Guaranteed by Characterization)

SID74	$F_{GPIOUT1}$	GPIO Fout;3.3 V ≤ V_{DDD} ≤ 5.5 V. Fast strong mode.	–	–	33	MHz	90/10%, 25-pF load, 60/40 duty cycle
SID75	$F_{GPIOUT2}$	GPIO Fout;1.7 V≤ V_{DDD}≤ 3.3 V. Fast strong mode.	–	–	16.7	MHz	90/10%, 25-pF load, 60/40 duty cycle
SID76	$F_{GPIOUT3}$	GPIO Fout;3.3 V ≤ V_{DDD} ≤ 5.5 V. Slow strong mode.	–	–	7	MHz	90/10%, 25-pF load, 60/40 duty cycle
SID245	$F_{GPIOUT4}$	GPIO Fout;1.7 V ≤ V_{DDD} ≤ 3.3 V. Slow strong mode.	–	–	3.5	MHz	90/10%, 25-pF load, 60/40 duty cycle
SID246	F_{GPIOIN}	GPIO input operating frequency; 1.71 V ≤ V_{DDD} ≤ 5.5 V	–	–	48	MHz	90/10% V_{IO}

Ref: PSoC® 4: PSoC 4200 Family Datasheet

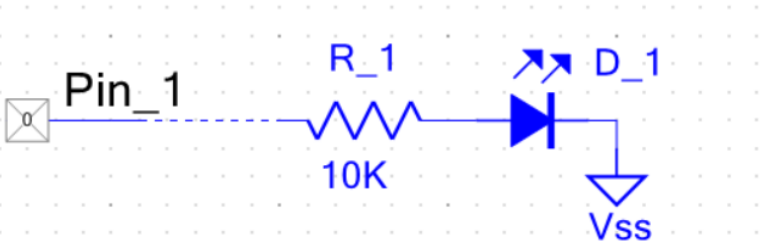
e.g., Application
LED Control

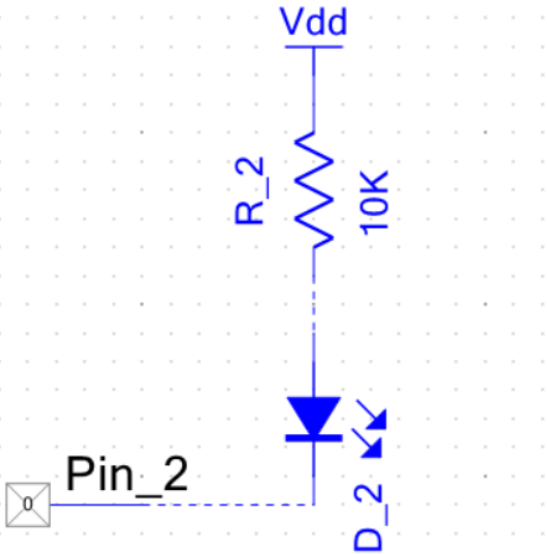


Pin_1 Drive Mode ?

R_1 Range ?

LED Drive Configs

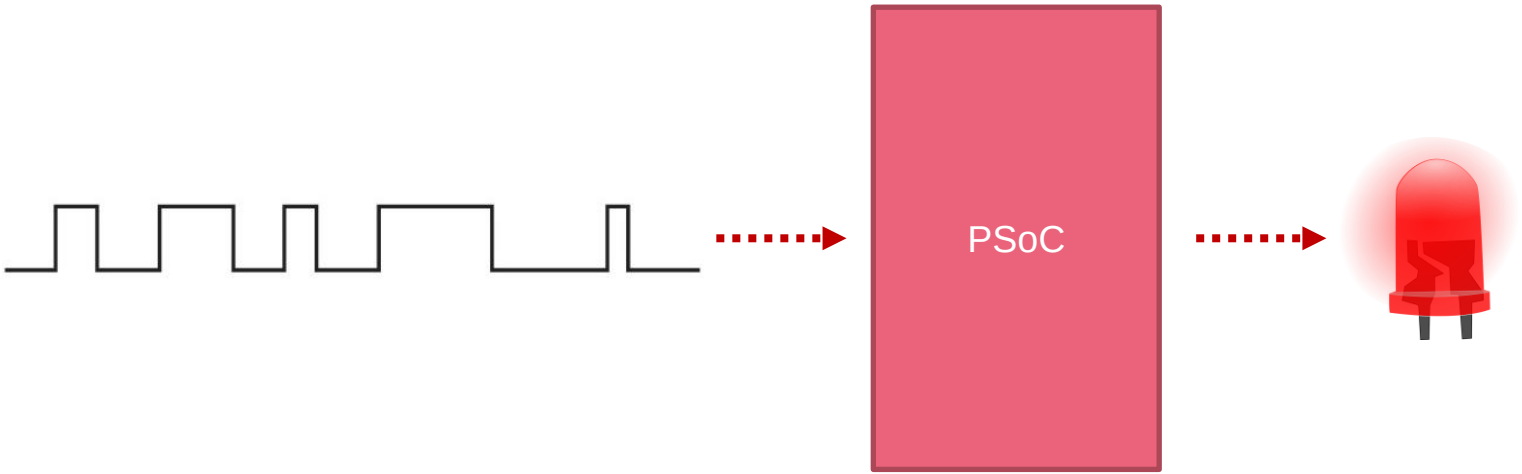




Drive Mode ?
Resistor Range ?

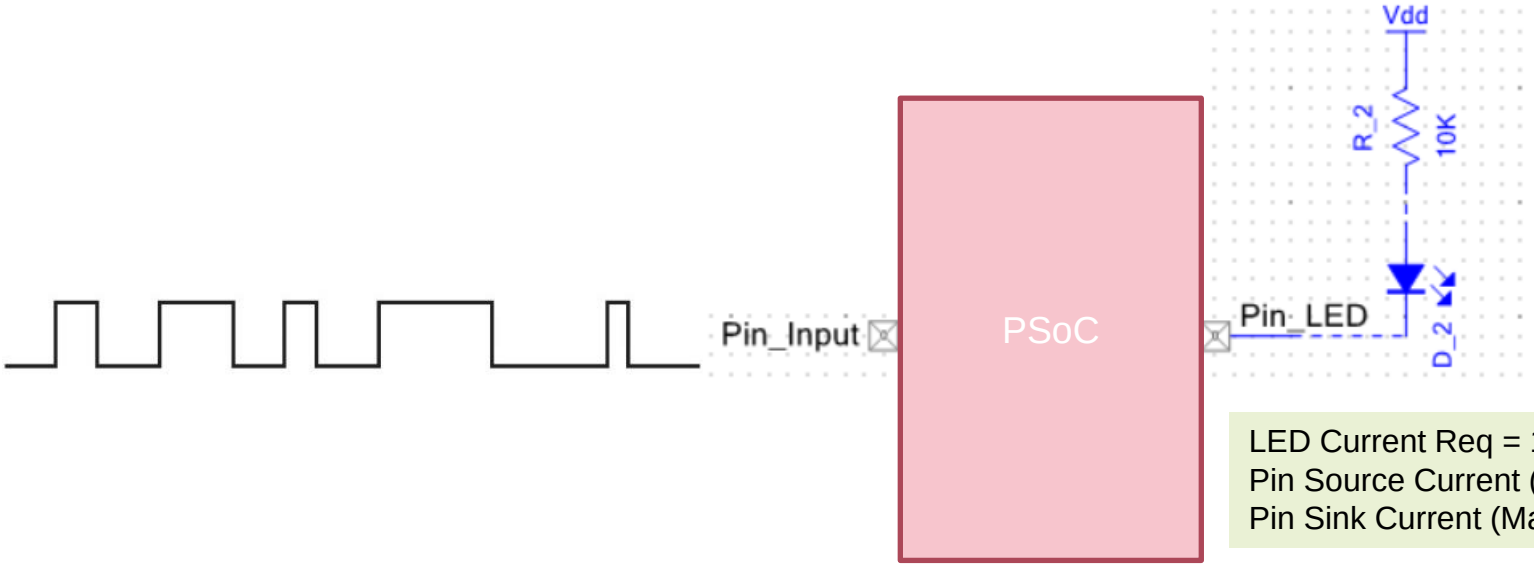
E.g., LED Ctrl

“Turn ON/OFF LED based on a Digital Signal”



LED Ctrl (…)

“Turn ON/OFF LED based on a Digital Signal”

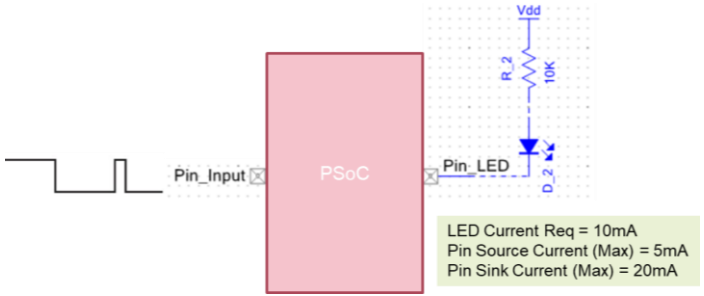


LED Current Req = 10mA
Pin Source Current (Max) = 5mA
Pin Sink Current (Max) = 20mA

Drive Mode of Pins ?, Firmware Logic ?

LED Ctrl (…)

“Turn ON/OFF LED based on a Digital Signal”

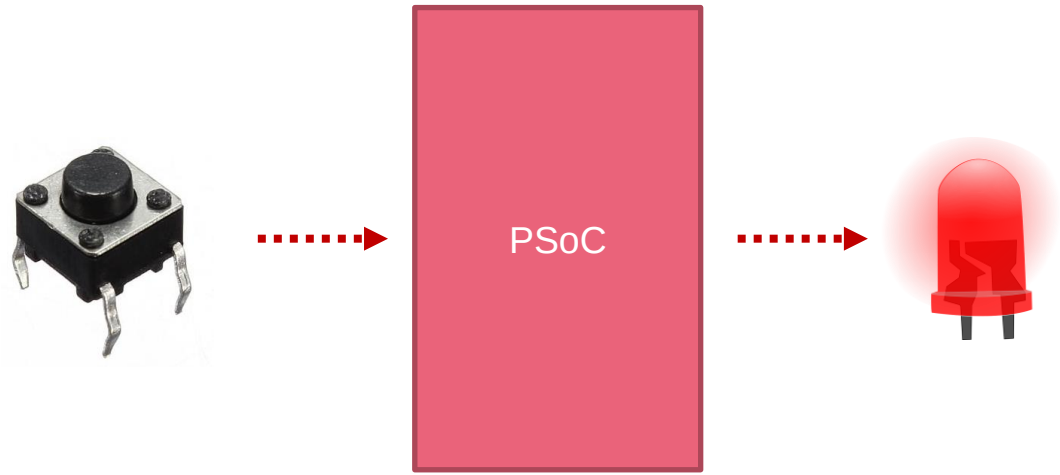



```c
#include <project.h>

int main()
{
    CyGlobalIntEnable; /* Enable global interrupts */

    /* Place your initialization/startup code here (e.g. MyInst_Start()) */
    /* Set Pin_LED drive mode to open-drain drives low */
    Pin_LED_SetDriveMode(Pin_LED_DM_OD_DRIVE_LOW);
    /* Set Pin_Input drive mode to Hi-Z digital input */
    Pin_Input_SetDriveMode(Pin_Input_DM_DIG_HIZ);

    for(;;)
    {
        /* Check the status of Pin_Input */
        if(Pin_Input_Read() == 0u)
        {
            /* If Pin_Input is low, set Pin_LED to high to turn ON LED*/
            Pin_LED_Write(1u);
        }
        else
        {
            Pin_LED_Write(0u);
        }
    }
}
```

Ref – creative commons

E.g., Switch ↔ LED

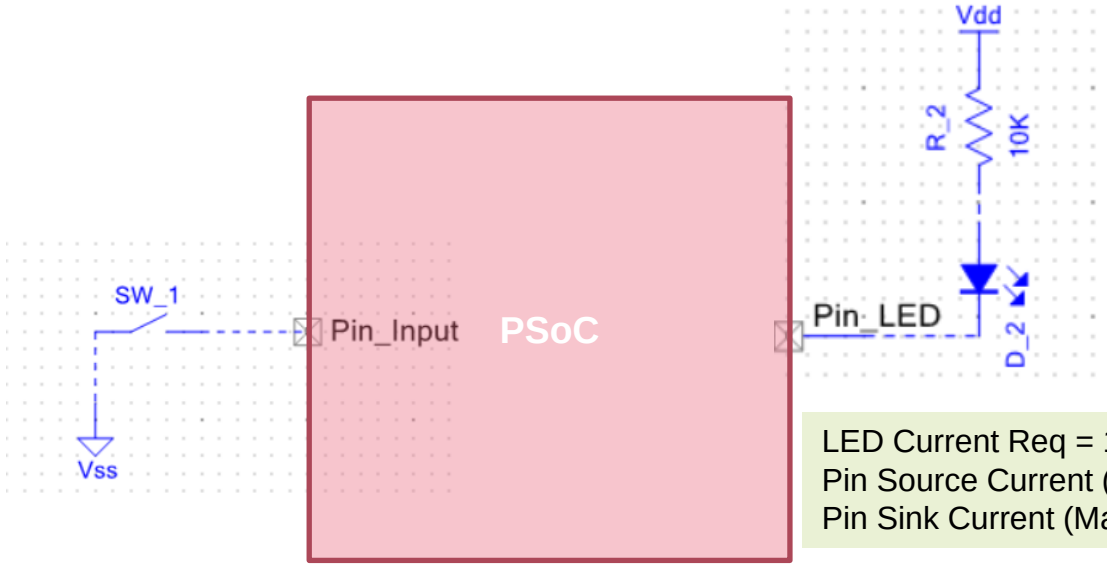
“Turn ON/OFF LED based on Switch Status”



Ref – creative commons -- https://robu.in/

Switch ↔ LED

“Turn ON/OFF LED based on Switch Status”

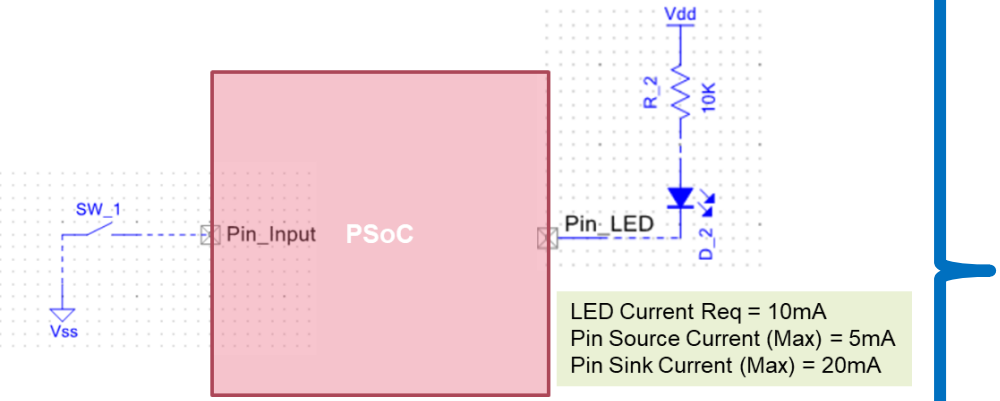


LED Current Req = 10mA
Pin Source Current (Max) = 5mA
Pin Sink Current (Max) = 20mA

Drive Mode of Pins ?, Firmware Logic ?

Switch ↔ LED

“Turn ON/OFF LED based on Switch Status”

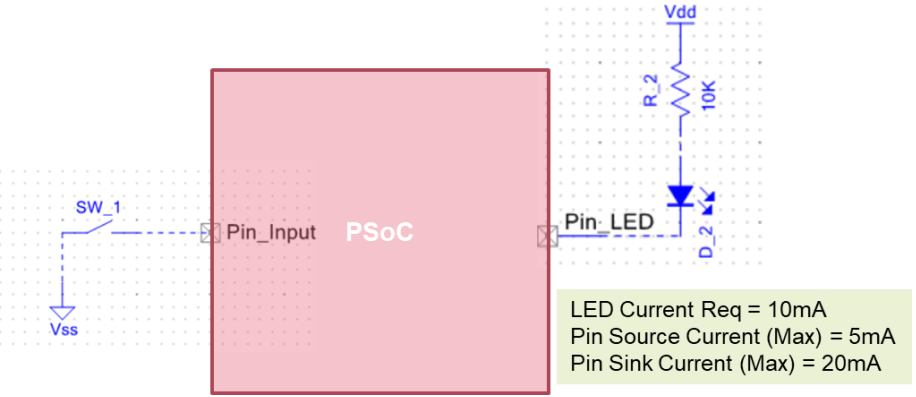



```c
#include <project.h>

int main()
{
    CyGlobalIntEnable;
    Pin_LED_SetDriveMode(Pin_LED_DM_OD_DRIVE_LOW);
    Pin_Input_SetDriveMode(Pin_Input_DM_RES_UP);
    Pin_Input_Write(0u);

    for(;;)
    {
        if(Pin_Input_Read() == 0u)
        {
            Pin_LED_Write(1u);
        }
        else
        {
            Pin_LED_Write(0u);
        }
    }
}
```

Ref – creative commons
-- https://robu.in/

Switch ↔ LED

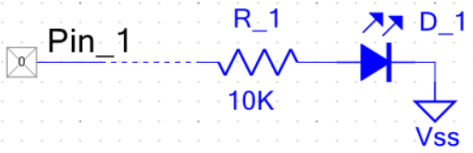
“Turn ON/OFF LED based on Switch Status”



Ref – creative commons

What if Resistive-Pull Up is not a Supported Drive Mode by the SoC ?

Reading – How to Design Resistor “R”



GPIO APIs

Functions

- uint8 Pin_Read(void)

  Reads the associated physical port (pin status register) and masks the required bits according to the width and bit position of the component instance.

- void Pin_Write(uint8 value)

  Writes the value to the physical port (data output register), masking and shifting the bits appropriately.

- uint8 Pin_ReadDataReg(void)

  Reads the associated physical port's data output register and masks the correct bits according to the width and bit position of the component instance.

- void Pin_SetDriveMode(uint8 mode)

  Sets the drive mode for each of the Pins component's pins.

- void Pin_SetInterruptMode(uint16 position, uint16 mode)

  Configures the interrupt mode for each of the Pins component's pins. Alternatively you may set the interrupt mode for all the pins specified in the Pins component.

- uint8 Pin_ClearInterrupt(void)

  Clears any active interrupts attached with the component and returns the value of the interrupt status register allowing determination of which pins generated an interrupt event.

Ref: PSoC® 4 GPIO Datasheet

Reading

CYPRESS PERFORM

AN86439

PSoC® 4 – Using GPIO Pins

10 GPIO Tips and Tricks

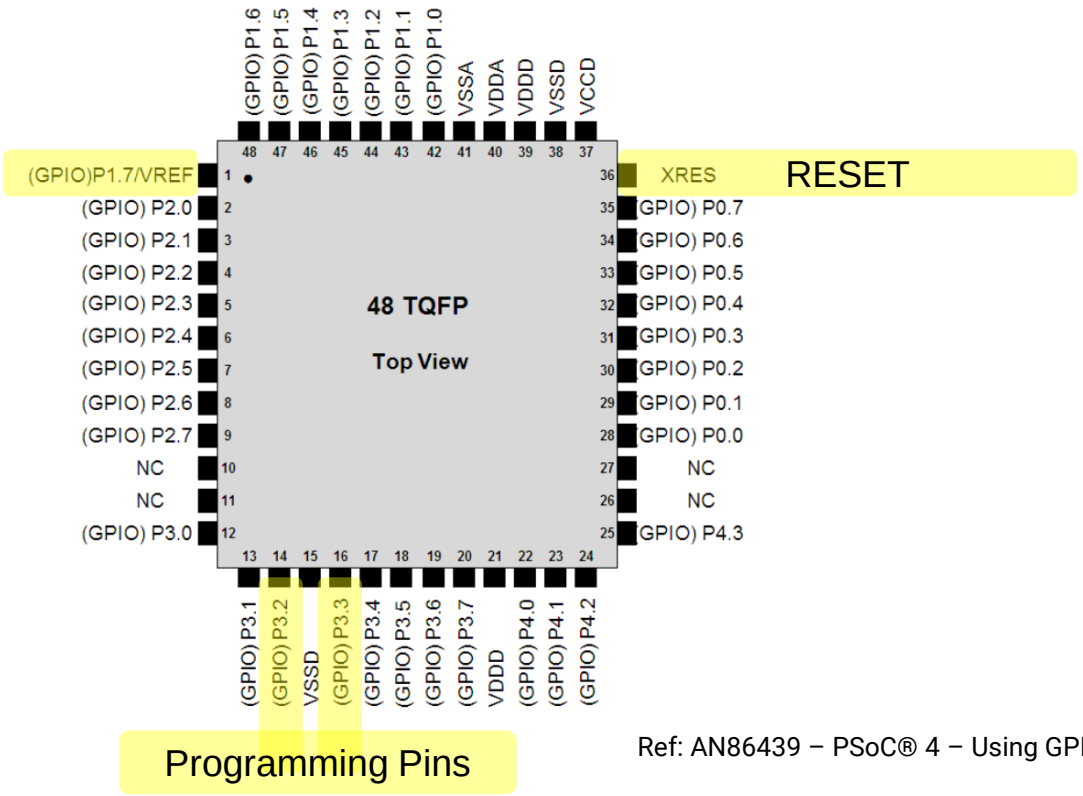
This section provides practical examples of how to use GPIO pins. All these examples are included in the PSoC Creator project provided with this application note.

Table 5. PSoC Creator Projects

#	Section	Project	PSoC 4000	PSoC 4100	PSoC 4200	PSoC 41xx-BL	PSoC 42xx-BL	PSoC 4100M	PSoC 4200M
1	Toggle an LED	Project1_ToggleLED	✓	✓	✓	✓	✓	✓	✓
2	Read an Input and Write to an Output	Project2_ReadingPin	✓	✓	✓	✓	✓	✓	✓

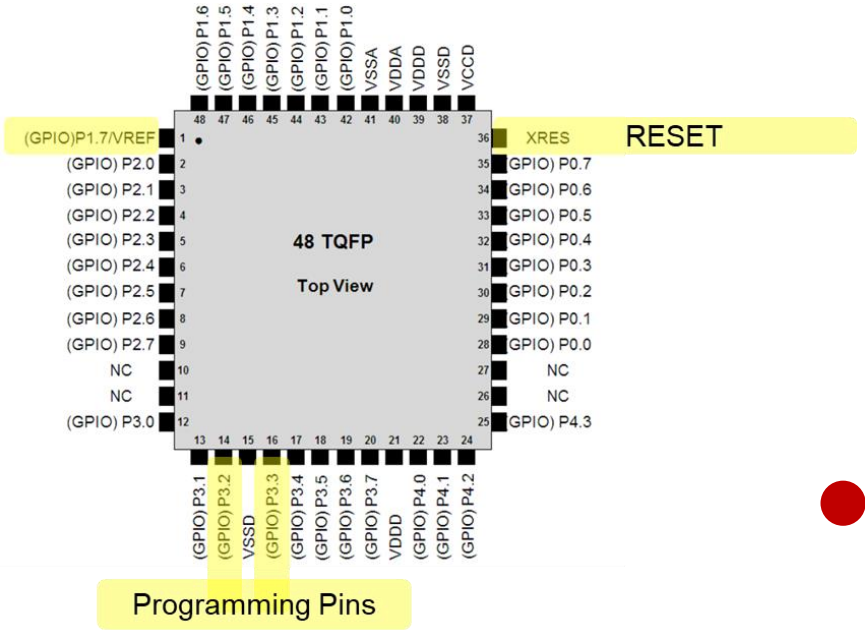
Ref: PSoC® 4 GPIO Datasheet

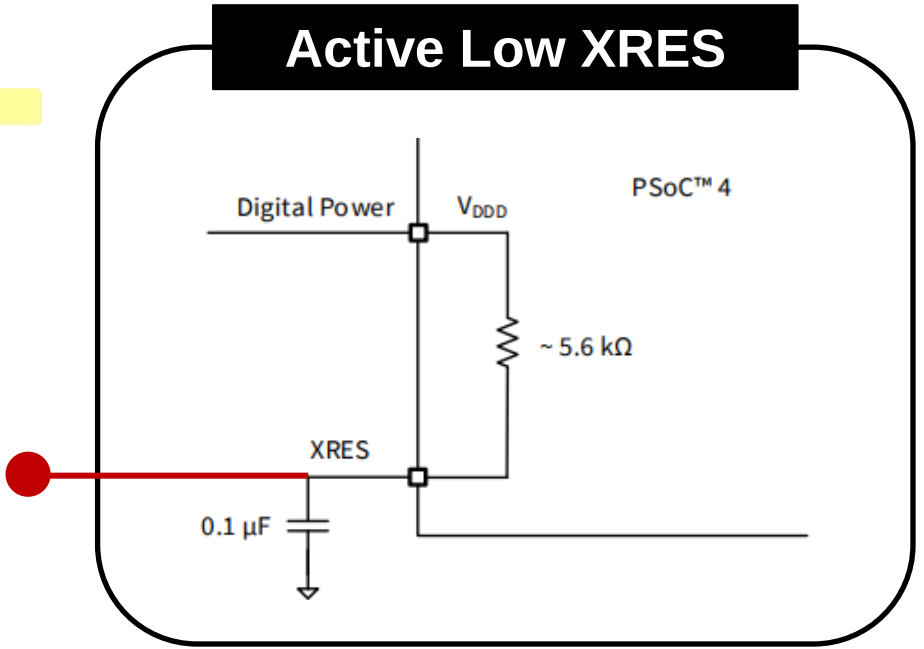
Special Function Pins



Ref: AN86439 – PSoC® 4 – Using GPIO Pins

RESET Pin





Ref: AN86439 – PSoC® 4 – Using GPIO Pins
AN88619 – PSoC Hardware Design Considerations

Specification

Table 6. XRES DC Specifications

Spec ID#	Parameter	Description	Min	Typ	Max	Units	Details/ Conditions
SID77	V_{IH}	Input voltage high threshold	0.7 × V_{DDD}	–	–	V	CMOS input
SID78	V_{IL}	Input voltage low threshold	–	–	0.3 × V_{DDD}	V	CMOS input
SID79	R_{PULLUP}	Pull-up resistor	3.5	5.6	8.5	kΩ	
SID80	C_{IN}	Input capacitance	–	3	–	pF	

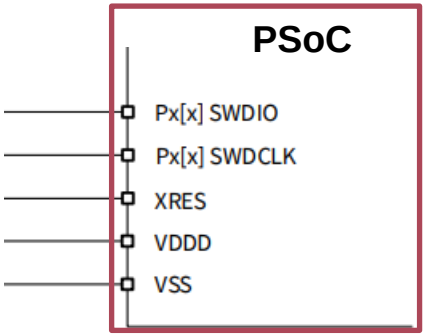
Table 7. XRES AC Specifications

Spec ID#	Parameter	Description	Min	Typ	Max	Units	Details/ Conditions
SID83	$T_{RESETWIDTH}$	Reset pulse width	1	–	–	µs	Guaranteed by characterization

Ref: PSoC® 4: PSoC 4200 Family Datasheet

SWD Prog. Pins

SWD Pins → **Programming PSoC (write FLASH with instructions)**



Ref: AN88619 – PSoC Hardware Design Considerations