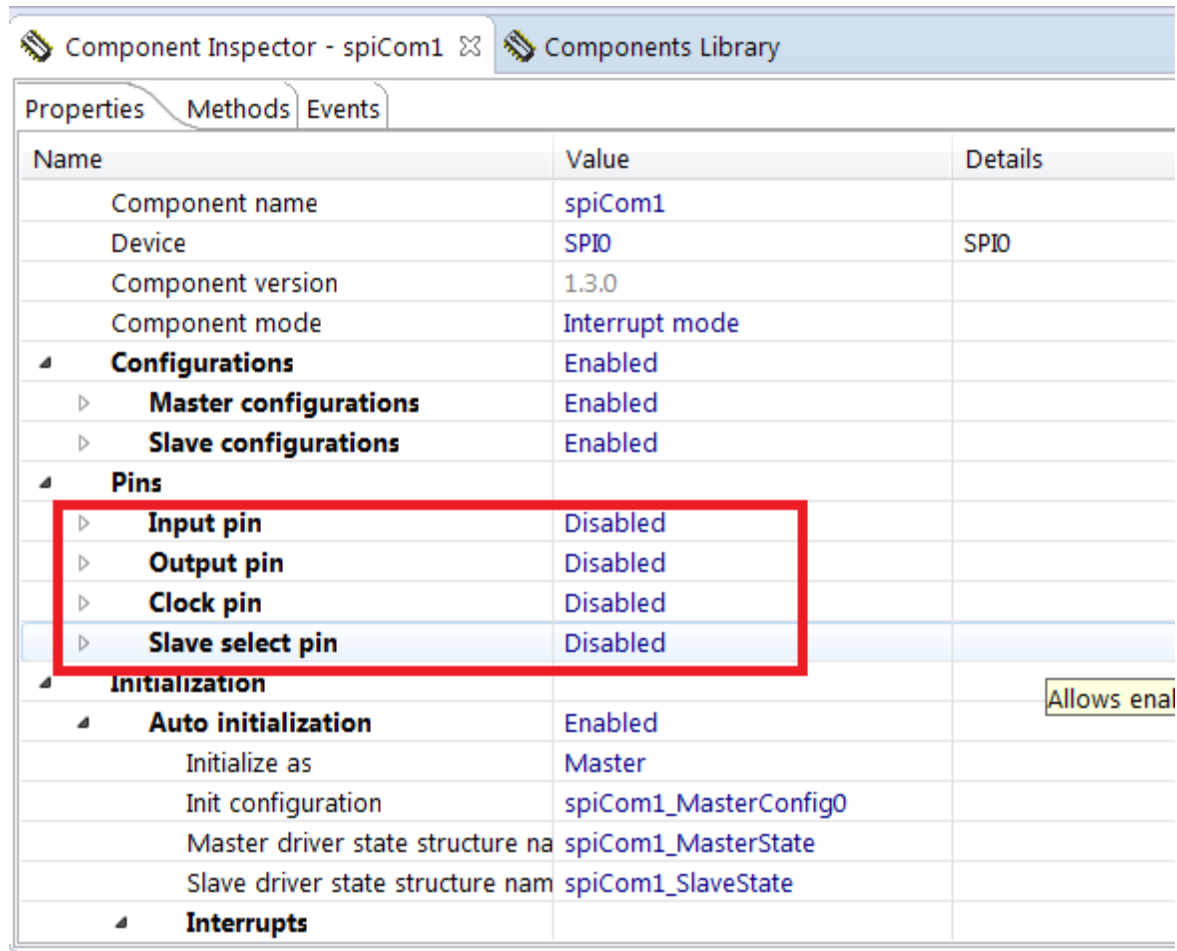


When you use the KDS 3.0.0 and project is created with KSDK 1.3.0 + ProcessorExpert, the PinSettings component is available for KL27Z. In the case you need share pins between two devices, for example SPI (fsl\_spi component) and GPIO (fsl\_gpio component), you can do it by the following way:

Add the fsl\_spi component in the project and disable all pins:



The screenshot shows the Component Inspector for the spiCom1 component. The Properties tab is active, displaying a table of configuration parameters. The Pins section is highlighted with a red box, indicating that all pins are disabled.

Name	Value	Details
Component name	spiCom1	
Device	SPI0	SPI0
Component version	1.3.0	
Component mode	Interrupt mode	
<b>Configurations</b>	Enabled	
<b>Master configurations</b>	Enabled	
<b>Slave configurations</b>	Enabled	
<b>Pins</b>		
Input pin	Disabled	
Output pin	Disabled	
Clock pin	Disabled	
Slave select pin	Disabled	
<b>Initialization</b>		
<b>Auto initialization</b>	Enabled	Allows enal
Initialize as	Master	
Init configuration	spiCom1_MasterConfig0	
Master driver state structure nam	spiCom1_MasterState	
Slave driver state structure nam	spiCom1_SlaveState	
<b>Interrupts</b>		

Add fsl\_gpio component in the project and select the pin(s) you want to use, e.g. PTD1 pin that will be shared with SPI:

The screenshot shows the 'Component Inspector - gpio1' window with the 'Properties' tab selected. The 'Input pins' section is expanded, and 'Pin 0' is highlighted with a red box. The 'Pin 0' row is also highlighted in blue. The 'Pin' column shows 'J1\_3' and the 'Details' column shows 'ADC0\_SE5b/PTD1/SPI0\_SCK/TPM0\_CH1/FXIO0\_D1'. A yellow tooltip with a 'P' icon is visible over the 'Electrical features' section, displaying the text 'Electrical features for selected pin.'.

Name	Value	Details
Component name	gpio1	
Component version	1.3.0	
Input pins	Enabled	
Input configurations	1	
Input configuration 0	Enabled	
Configuration name	gpio1_InpConfig0	
Input pins number	1	
Pin 0	Enabled	
Pin	J1_3	ADC0_SE5b/PTD1/SPI0_SCK/TPM0_CH1/FXIO0_D1
Pin name	J1_3	
Interrupt/DMA	Interrupt/DMA request is disab...	
Electrical features		
Pull enable	Enabled	
Pull select	Down	Electrical features for selected pin.
Passive filter	Enabled	
Output pins	Enabled	
Output configurations	1	
Output configuration 0	Enabled	
Configuration name	gpio1_OutConfig0	

Open the PinSettings component in the Configuration Inspector and do the following:

- open Routing tab and select View Mode – Collapsed
- select SPI tab
- Uncheck the Init checkbox of the SPI0 device (disable the SPI pins routing in the initialization code)
- Select pins that will be used for the SPI device (select the PTD1 – J1\_3 name that will be shared). The shared pins will contain information about ignored error. There is not reported any error that block generating of Processor Expert code.

Project Explorer

- Test FRDM-KL27Z
  - Includes
  - Documentation

Components - Test FRDM-KL27Z

- Generator\_Configurations
- Oss
  - osa1:fsl\_os\_abstraction
- Processors
  - Cpu:MKL27Z64VLH4
- Components
  - clockMan1:fsl\_clock\_manager
  - pin\_init:PinSettings**
  - intMan1:fsl\_interrupt\_manager
  - spiCom1:fsl\_spi
  - gpio1:fsl\_gpio

Component Inspector - pin\_init

Routing | Functional Properties | Methods | Settings

View Mode:  Collapsed  Pins

Options:  Show Only Configurable Signals

Generate Report: [HTML Report](#)

ADC CMP CoreDebug FLEXIO GPIO I2C LLWU LPTMR LPUART OSC RCM RTC SCB SIM **SPI** SUPPLY TPM UART USB VREF

Signals	Init	Pin/Signal Selection	Direction	Selected Pin/Signal Name
<b>SPIO</b>	<input type="checkbox"/>			
MISO - Master data in, slave data out		J1_5	Not Specified	PTD2/SPIO_MOSI/UART2_RX/TPMO_CH2/SPIO_M...
MOSI - Master data out, slave data in		J1_7	Not Specified	PTD3/SPIO_MISO/UART2_TX/TPMO_CH3/SPIO_M...
SCK - Serial clock		J1_3	Not Specified	Ignored error: Selected value is in conflict with ...
SS - Slave select		No pin routed	No pin routed	
<b>SPI1</b>	<input checked="" type="checkbox"/>			
MISO - Master data in, slave data out		No pin routed	No pin routed	
MOSI - Master data out, slave data in		No pin routed	No pin routed	
SCK - Serial clock		No pin routed	No pin routed	
SS - Slave select		No pin routed	No pin routed	

When you generate the Processor Expert code that you can see that there are available `init_gpio_pins()` and `init_spi_pins()` methods of the `pin_init:PinSettings` component in the project. You can use these function for switching of shared pin(s), see the routing of shared PTD1 pin below:

```
/*FUNCTION*****  
*  
* Function Name : init_gpio_pins  
* Description   : GPIO method sets registers according routing settings.  
* Call this method code to route desired pins.  
*END*****/  
void init_gpio_pins(uint32_t instance)  
{  
    switch(instance) {  
        case GPIOD_IDX:                /* GPIOD_IDX */  
            /* Affects PORTD_PCR1 register */  
            PORT_HAL_SetPullMode(PORTD,1UL,kPortPuLLDown);  
            PORT_HAL_SetMuxMode(PORTD,1UL,kPortMuxAsGpio);  
            PORT_HAL_SetPullCmd(PORTD,1UL,true);  
            break;  
        case GPIOE_IDX:                /* GPIOE_IDX */  
            /* Affects PORTE_PCR24 register */  
            PORT_HAL_SetMuxMode(PORTE,24UL,kPortMuxAsGpio);  
            PORT_HAL_SetSlewRateMode(PORTE,24UL,kPortSlowSlewRate);  
            break;  
        default:  
            break;  
    }  
}
```

```

/*FUNCTION*****
**
*
* Function Name : init_spi_pins
* Description  : SPI method sets registers according routing settings.
* Call this method code to route desired pins.
*END*****
/
void init_spi_pins(uint32_t instance)
{
    switch(instance) {
        case SPI0_IDX:          /* SPI0_IDX */
            /* Affects PORTD_PCR2 register */
            PORT_HAL_SetMuxMode(PORTD,2UL,kPortMuxAlt5);
            /* Affects PORTD_PCR3 register */
            PORT_HAL_SetMuxMode(PORTD,3UL,kPortMuxAlt5);
            /* Affects PORTD_PCR1 register */
            PORT_HAL_SetPullMode(PORTD,1UL,kPortPullDown);
            PORT_HAL_SetMuxMode(PORTD,1UL,kPortMuxAlt2);
            PORT_HAL_SetPullCmd(PORTD,1UL,true);
            break;
        default:
            break;
    }
}

```

Please note, that the shared pin(s) will be initialized as GPIO. The hardware\_init() initialization function will call the init\_gpio\_pins() function only, see below:

```
void hardware_init(void) {  
  
    /* Enable clock for PORTs */  
    SIM_HAL_EnableClock(SIM,kSimClockGatePortA);  
    SIM_HAL_EnableClock(SIM,kSimClockGatePortD);  
    SIM_HAL_EnableClock(SIM,kSimClockGatePortE);  
  
    /* Setup board clock source. */  
    g_xtal0ClkFreq = 32768U;          /* Value of the external crystal or oscillator clock frequency of the system  
oscillator (OSC) in Hz */  
  
    init_coredebug_pins(CoreDebug_IDX);  
    init_gpio_pins(GPIOD_IDX);  
    init_gpio_pins(GPIOE_IDX);  
    init_osc_pins(OSC0_IDX);  
    init_rcm_pins(RCM_IDX);  
}
```