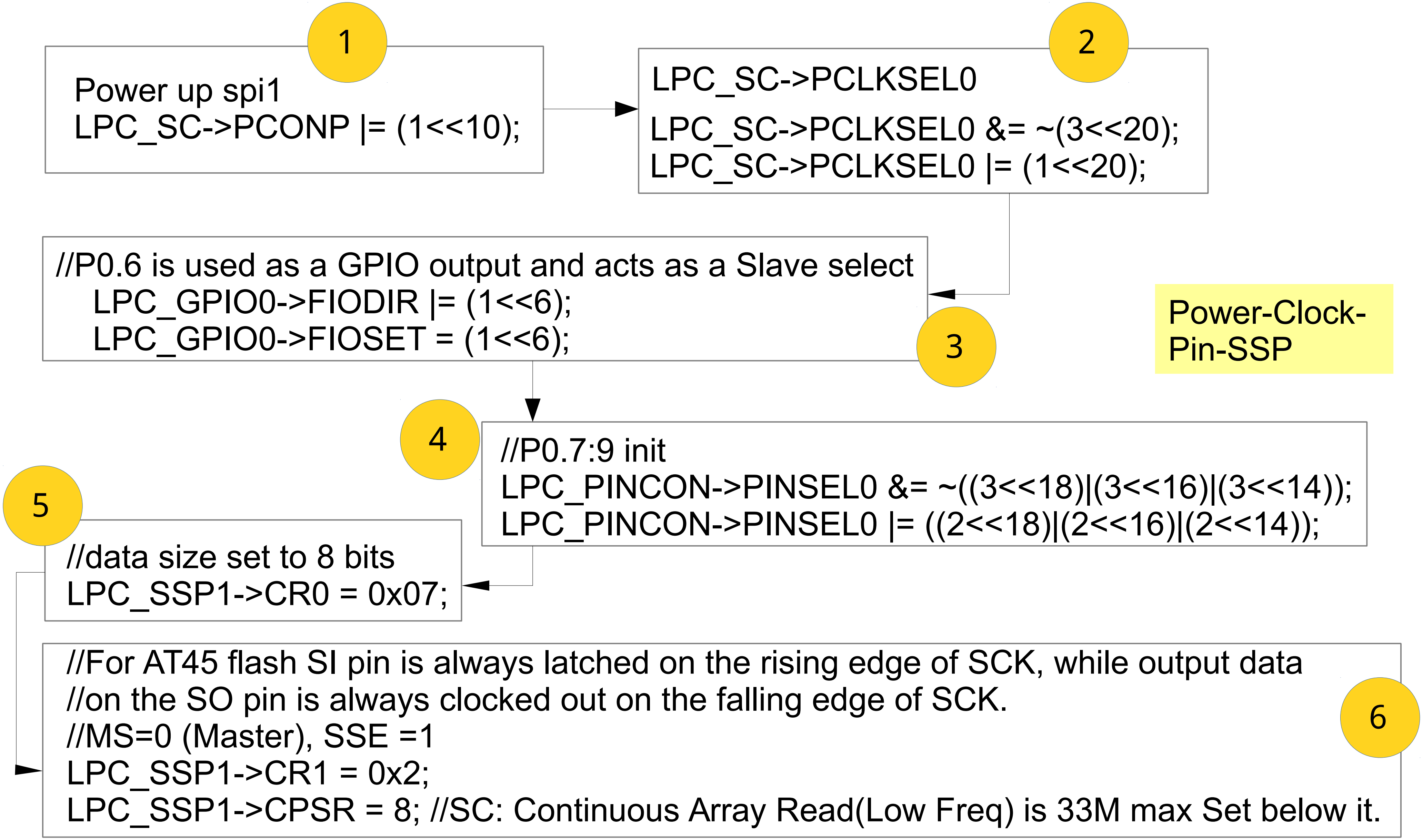
SSP_Test.c SPI Sample Code

```
void initSSP1(void){
    //power up spi1
    LPC_SC->PCONP |= (1<<10);
    //01 PCLK_peripheral = CCLK.. 01, since it's CCLK/1
    LPC_SC->PCLKSEL0 &= ~(3<<20);
    LPC_SC->PCLKSEL0 |= (1<<20);

    //P0.6 is used as a GPIO output and acts as a Slave select
    LPC_GPIO0->FIODIR |= (1<<6);
    LPC_GPIO0->FIOSET = (1<<6);
    //P0.7:9 init
    LPC_PINCON->PINSEL0 &= ~((3<<18)|(3<<16)|(3<<14));
    LPC_PINCON->PINSEL0 |= ((2<<18)|(2<<16)|(2<<14));
    //data size set to 8 bits
    LPC_SSP1->CR0 = 0x07;
    //For AT45 flash SI pin is always latched on the rising edge of SCK, while output data
    //on the SO pin is always clocked out on the falling edge of SCK.
    //MS=0 (Master), SSE =1
    LPC_SSP1->CR1 = 0x2;
    LPC_SSP1->CPSR = 8; //SC for Continuous Array Read(Low Frequency)
                        // is 33Mhx max. We are setting it below it.
}
```

Init & Config for SPRs



1

```
Power up spi1
LPC_SC->PCONP |= (1<<10);
```

2

```
LPC_SC->PCLKSEL0
LPC_SC->PCLKSEL0 &= ~(3<<20);
LPC_SC->PCLKSEL0 |= (1<<20);
```

3

```
//P0.6 is used as a GPIO output and acts as a Slave select
   LPC_GPIO0->FIODIR |= (1<<6);
   LPC_GPIO0->FIOSET = (1<<6);
```

4

```
//P0.7:9 init
LPC_PINCON->PINSEL0 &= ~((3<<18)|(3<<16)|(3<<14));
LPC_PINCON->PINSEL0 |= ((2<<18)|(2<<16)|(2<<14));
```

5

```
//data size set to 8 bits
LPC_SSP1->CR0 = 0x07;
```

6

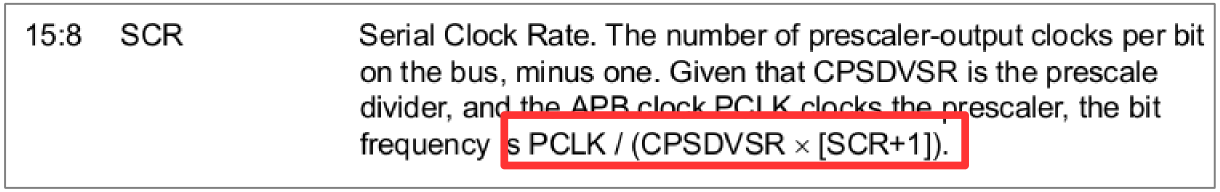
```
//For AT45 flash SI pin is always latched on the rising edge of SCK, while output data
//on the SO pin is always clocked out on the falling edge of SCK.
//MS=0 (Master), SSE =1
LPC_SSP1->CR1 = 0x2;
LPC_SSP1->CPSR = 8; //SC: Continuous Array Read(Low Freq) is 33M max Set below it.
```

Power-Clock-Pin-SSP

SPR: CR0

5

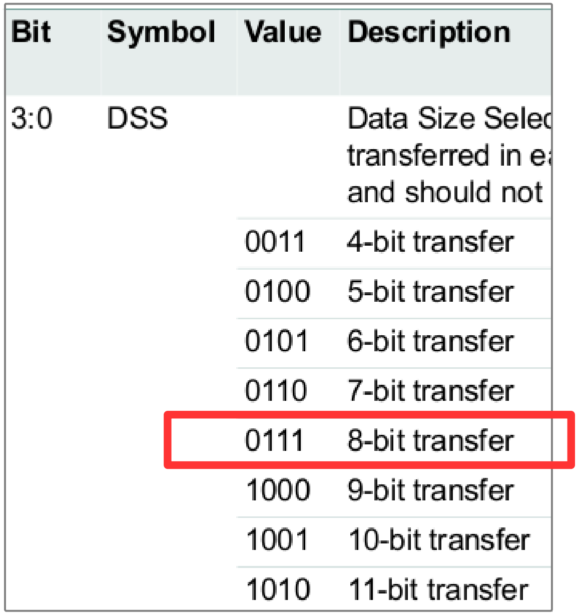
```
//data size set to 8 bits
LPC_SSP1->CR0 = 0x07;
```



15:8	SCR	Serial Clock Rate. The number of prescaler-output clocks per bit on the bus, minus one. Given that CPSDVSR is the prescale divider, and the APB clock PCLK clocks the prescaler, the bit frequency is PCLK / (CPSDVSR × [SCR+1]).

Table 371: SSPn Control Register 0
pp. 431

$$f_{SPI} = PCLK / (CPSDVSR * [SCR+1]) \quad .. (1)$$



Bit	Symbol	Value	Description
3:0	DSS		Data Size Selec transferred in ea and should not
		0011	4-bit transfer
		0100	5-bit transfer
		0101	6-bit transfer
		0110	7-bit transfer
		0111	8-bit transfer
		1000	9-bit transfer
		1001	10-bit transfer
		1010	11-bit transfer

Note:
(1) PCLK peripheral clock
(2) Special purpose register: CPSDVSR, range from 2 to 254
(3) SCR from CR0[15:8] field.

Example: Design SPI Flash interface at 10 Khz clock rate, suppose PCLK = 50 Mhz.
Sol: From equation (1), we have

$$10X10^3 = PCLK/(CPSDVSR * (SCR+1))$$

There are 2 parameters to be fixed in this design, CPSDVSR and SCR

Define SPI Interface Clock

Continued from the previous

$$\boxed{f\_{SPI} = PCLK / (CPSDVSR * [SCR+1]) \quad .. (1)}$$

substitute PCLK = 50 Mhz in to the equation, we have

10X10^3 = 50 x 10^6 /(CPSDVSR * (SCR+1))

5x10^3 = CPSDVSR * (SCR+1)

Note CPSDVSR is in the range of 2 to 254 and SCR is 8 bits, so it is in the range of 0 to 255, let's try to make SCR=64, by SCR[7:0]=0100 0000 = 0x40, so

CPSDVSR = 5x10^3 / 65 = 76.923

So, make CPSDVR = 77

1001101 so the hex is 0x4d.

SSP Configuration

18.1 Basic configuration

The two SSP interfaces, SSP0 and SSP1 are configured using the following registers:
1. Power: In the PCONP register (Table 46), set bit PCSSP0 to enable SSP0 and bit PCSSP1 to enable SSP1.

2. Clock: In PCLKSEL0 select PCLK_SSP1; in PCLKSEL1 select PCLK_SSP0 (see Section 4.7.3. In master mode, the clock must be scaled down (see Section 18.6.5).

3. Pins: Select the SSP pins through the PINSEL registers (Section 8.5) and pin modes through the PINMODE registers (Section 8.4).

4. Interrupts: Interrupts are enabled in the SSP0IMSC register for SSP0 and SSP1IMSC register for SSP1 Table 376. Interrupts are enabled in the NVIC using the appropriate Interrupt Set Enable register, see Table 50.

5. Initialization: There are two control registers for each of the SSP ports to be configured: SSP0CR0 and SSP0CR1 for SSP0, SSP1CR0 and SSP1CR1 for SSP1. See Section 18.6.1 and Section 18.6.2.

6. DMA: The Rx and Tx FIFOs of the SSP interfaces can be connected to the GPDMA controller (see Section 18.6.10). For GPDMA system connections, see Table 544.
Remark: SSP0 is intended to be used as an alternative for the SPI interface, which is included as a legacy peripheral. Only one of these peripherals can be used at the any one time.