

ADXL345 정리

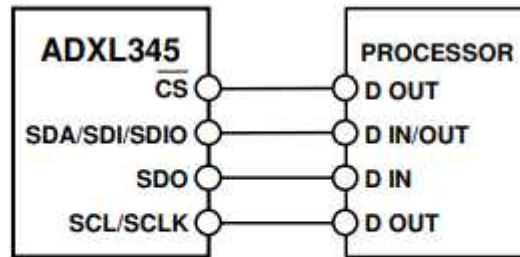


Figure 3. 4-Wire SPI connection

- 4선 모드 사용.

- * SCK = serial clock (직렬 클럭)
- * MOSI = Master out Slave in 슬레이브에 데이터를 전송하는데 사용
- * MISO = Master In Slave out 슬레이브에서 데이터를 수신하는데 사용
- * CS = chip Select 슬레이브를 선택하는데 사용

여기서 슬레이브 장치를 활성화하려면 CS 핀을 로우로 당겨야 하고 읽기 또는 쓰기가 완료된 후 핀을 다시 하이로 당겨야 한다. 그러면 슬레이브 장치가 비활성화 된다.

Clearing the $\overline{\text{SPI}}$ bit in the **DATA_FORMAT** register selects 4-wire mode while setting the SPI bit selects 3-wire mode. The maximum SPI clock speed is 5 MHz, with 12 pF maximum loading and the timing scheme follows CPOL = 1, CPHA = 1.

데이터시트 확인하면 clock speed가 5Mhz 미만인 spi와 cpol = 1 및 cpha = 1로 설정

- 쓰기 기능

```

void adxl_write (uint8_t address, uint8_t value)
{
    uint8_t data[2];
    data[0] = address|0x40; // multibyte write enabled
    data[1] = value;
    HAL_GPIO_WritePin (GPIOA, GPIO_PIN_0, GPIO_PIN_RESET);
    HAL_SPI_Transmit (&hspi1, data, 2, 100);
    HAL_GPIO_WritePin (GPIOA, GPIO_PIN_0, GPIO_PIN_SET);
}
  
```

- * 슬레이브를 활성화 하려면 CS를 RESET
- * 데이터를 쓰고자 하는 주소를 전송
- * 데이터 전송
- * 슬레이브를 비활성화하려면 CS pin을 HIGH

* data[0] 에서 address 는 0x40 과 OR 입니다 . 이것은 멀티바이트 쓰기를 위한 것입니다. 단일 전송에서 하나 이상의 바이트를 전송하고 싶다고 ADXL에 알립니다. ADXL 데이터시트에 따르면 그렇게 하려면 이 바이트가 높아야 합니다.

To read or write multiple bytes in a single transmission, the Multi-Byte bit, located after the R/W bit in the first byte transfer, must be set. After the register addressing and the first byte of data, continued clock pulses will cause the ADXL345 to point to the next register for read or write. Continued clock pulses will continue to shift the register that is pointed to until the clock pulses are ceased and CS is de-asserted. To perform reads or writes on different, non-sequential registers, CS must be de-asserted between transmissions and the new register must be addressed separately.

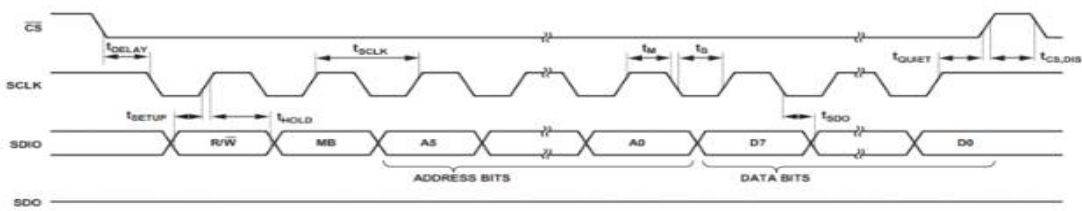
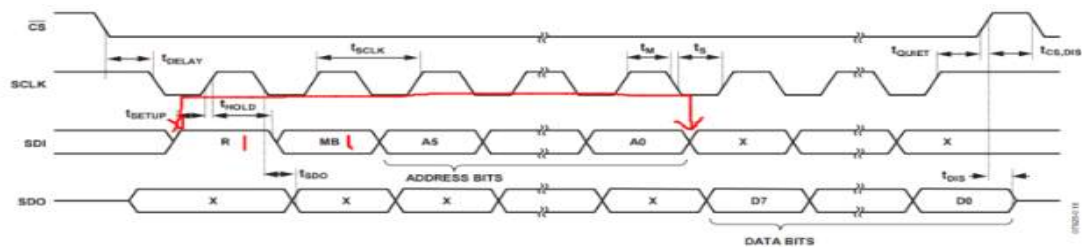
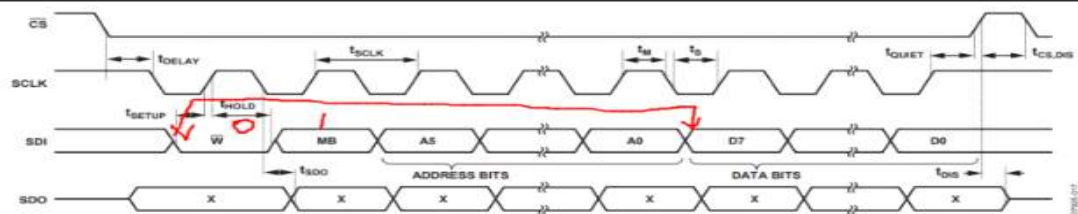
- 읽기 기능

`void adxl_read (uint8_t address)`

```
{
    address |= 0x80; // read operation
    address |= 0x40; // multibyte read
    HAL_GPIO_WritePin (GPIOA, GPIO_PIN_0, GPIO_PIN_RESET);
    HAL_SPI_Transmit (&hspi1, &address, 1, 100);
    HAL_SPI_Receive (&hspi1, data_rec, 6, 100);
    HAL_GPIO_WritePin (GPIOA, GPIO_PIN_0, GPIO_PIN_SET);
}
```

- * 슬레이브를 활성화하려면 CS를 RESET
- * 데이터를 읽고자 하는 주소를 전송
- * 데이터를 수신하고 이 경우 6바이트
- * 슬레이브를 비활성화하려면 CS pin을 high

* 주소는 0x80 및 0x40과 OR입니다 . ADXL 데이터시트에 따르면 데이터를 읽으려면 마지막 비트를 HIGH로 설정해야 하고 멀티바이트 읽기/쓰기의 경우 6번째 비트를 HIGH로 설정해야 하기 때문입니다.



- ADXL 초기화 기능

```
void adxl_init (void)
{
    adxl_write (0x31, 0x01); // data_format range= +- 4g
    adxl_write (0x2d, 0x00); // reset all bits
    adxl_write (0x2d, 0x08); // power_cntl measure and wake up 8hz
}
```

* adxl write 0x31

D7	D6	D5	D4	D3	D2	D1	D0
SELF_TEST	SPI	INT_INVERT	X	FULL_RES	JUSTIFY	RANGE	

* adxl_write (0x31, 0x01); // data_format range= +- 4g

D1	D0	g-Range
0	0	±2 g
0	1	±4 g
1	0	±8 g
1	1	±16 g

* adxl write 0x2d

0x2D POWER_CTL (read/write)

D7	D6	D5	D4	D3	D2	D1	D0
X	X	LINK	AUTO_SLEEP	MEASURE	SLEEP	WAKEUP	

adxl_write (0x2d, 0x08); // power_cntl measure and wake up 8hz

Table 13. WAKEUP Rates

D1	D0	Frequency (Hz)
0	0	8
0	1	4
1	0	2
1	1	1

D3 = 1 D1, D0 = 0 설정