

1) Configuring the SPI signal pins

The MMC/SD-Cards expect all signals to be active-low, i.e. at “1” during idle, and “0” when asserted.

The SPI Clock should be set to shift out the data on the **falling edge**, with idle polarity **low**. For Motorola HCS12 or DPS processors, this means the CPOL and CPHA configuration bits in the control registers should both be set to 0.

All data is transmitted MSB first.

The empty character is set to 0xFF, and empty characters should **not** be ignored.

The Slave Select pin should **not** be under SPI control but instead asserted and de-asserted under software control (it should not go on and off with each character transmitted, but remain asserted “=0” during each complete access operation).

The shift clock rate can be anything up to 25 MHz. (For MMC cards up to 400 KHz.)

Initialization code produced with Codewarrior for HCS12:

```
SPI0CR1 = 4;          /* Reset the device register */
SPI0SR;             /* Read the status register */
SPI0DR;            /* Read the device register */
/* SPI0BR: ??=0, SPPR2=0, SPPR1=0, SPPR0=0, ??=0, SPR2=0, SPR1=0, SPR0=1 */
SPI0BR = 1;         /* Set the baud rate register */
/* SPI0CR2: ??=0, ??=0, ??=0, MODFEN=0, BIDIROE=0, ??=0, SPISWAI=0, SPC0=0 */
SPI0CR2 = 0;        /* Set control register 2 */
/* SPI0CR1: SPIE=0, SPE=0, SPTIE=0, MSTR=1, CPOL=0, CPHA=0, SSOE=0, LSBFE=0 */
SPI0CR1 = 16;       /* Set control register 1 */
```

A useful method for testing the correct configuration is to output an uppercase “U” in an endless loop and observe the MOSI and CLK pins with an oscilloscope. The “U” outputs alternate 0 and 1 bits. Both signals should be high when the SPI is idle.

