

SPI send 8 Bit erklärt

In diesem Code wird eine Funktion `SPISend8Bit` definiert, die einen 8-Bit-Wert über die serielle Peripherie-Schnittstelle (SPI) sendet. Hier ist, was in den einzelnen Schritten passiert:

1. `PORTB &= ~(1 << SS);` setzt den Chip Select (CS) Pin auf niedrig, indem das entsprechende Bit im PORTB-Register gelöscht wird. Dies signalisiert dem SPI-Slave-Gerät, dass eine Übertragung beginnt.
2. `SPDR = data;` lädt den zu sendenden 8-Bit-Wert in das SPI-Datenregister (SPDR). Dies startet die Übertragung der Daten über die SPI-Schnittstelle.
3. `while (!(SPSR & (1 << SPIF)));` ist eine Warteschleife, die darauf wartet, dass das SPIF-Bit (SPI Interrupt Flag) im SPSR-Register (SPI Status Register) gesetzt wird. Dieses Bit wird gesetzt, wenn die Übertragung abgeschlossen ist.
4. `PORTB |= (1 << SS);` setzt den Chip Select (CS) Pin wieder auf hoch, indem das entsprechende Bit im PORTB-Register gesetzt wird. Dies signalisiert dem SPI-Slave-Gerät, dass die Übertragung beendet ist.

Zusammenfassend sendet diese Funktion einen 8-Bit-Wert über die SPI-Schnittstelle, indem sie den CS-Pin auf niedrig setzt, den Wert in das SPDR-Register lädt, auf das Ende der Übertragung wartet und dann den CS-Pin wieder auf hoch setzt.

Dieser Code ist typisch für die Kommunikation mit SPI-Geräten wie Sensor-ICs, Displays oder anderen Peripheriegeräten, die über eine SPI-Schnittstelle angebunden sind.