## Introduction

This application note describes how to configure your Amped RF Wi-Fi module to transfer data using a Serial Peripheral Interface (SPI). Modules can act as master or slave. The default mode is slave.

SPI Signal Lines

The SPI protocol is a Master-Slave protocol where the speed of transfer is determined by the clock (SCLK) sent from the master. The data from the master is sent over the Master Out Slave (MOSI) line, and the data from the slave is sent over using the Master In Slave Out (MISO). The Slave Select line (SS) is used by the master to select the slave device.
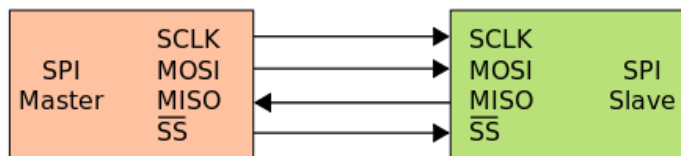

Data Transmission

Communication between the master and slave can only occur when the slave is selected. After the slave is selected data is then transferred over the MISO and MOSI lines using the clock pulses. When data is clocked to be sent, data is transferred to AND from the master at the same time. This kind of message exchange is called full duplex.
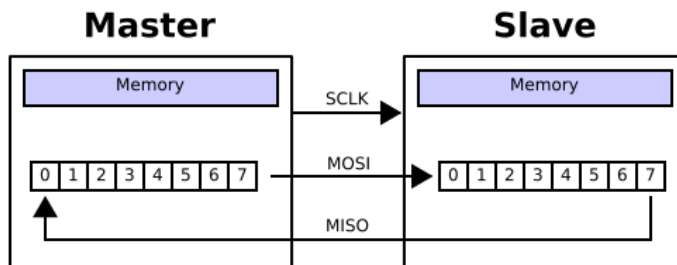
Clock Polarity and Phase

In addition to the clock frequency, the master must configure the clock polarity (CPOL) and phase (CPHA). The default settings we use are CPOL=1, and CPHA=1.
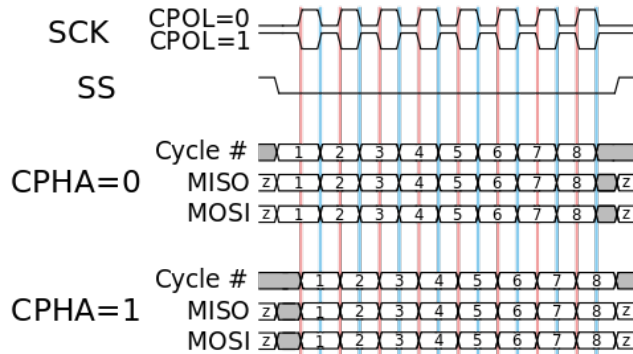
Slave Request (SREQ) for Enhanced SPI

In a standard SPI protocol, the master controls both SCLK and SS lines, so there is no way for the slave to indicate that it has data to send. We've enhanced SPI to solve this problem by defining an SPI Slave Request line (SREQ). Enabling this line by the slave will indicate to the master that the slave has data to send. The master will then use the SS line to select the slave and pulse the clock (SCLK) until all the data sent by the slave is received by the master.

Note: gpio[5] is designated as the SREQ line.

SS and SREQ line already pull down in default, they will become valid when pull up these two lines.

SS and SREQ as Interrupts

These lines may also be configured to trigger interrupts to speed responses. The SS line may trigger an interrupt for the slave device, while the SREQ line may trigger an interrupt in the master.

AT Commands

By default, the abSerial AT commands will always be exchanged using the UART instead of SPI. Once a net link has been established and the module is in BypassMode, the SPI will be in effect. This is the default behaviour when SPI is enabled in the configuration:

```
var29 OutMtuSize = 1000
var36 SPIEnable  = true
var37 SPIMode    = master
```

The OutMtusize is the data length that SPI will transfer; a fixed value.

Upon closing the net link, the UART will once again become the serial interface.

Typical Application

Command mode on the module is tuned best for UART communication, whereas SPI may be ideal for high speed data transfer. An application can take advantage of this by having the interface be in a UART mode when not connected, and switching to SPI mode when connected to transfer data. This allows easy access for configuration and control with a standard UART interface, while still able to take advantage of the high speed SPI.

Most applications using SPI to exchange data have specific formats. For example, if a message frame is defined with a size of 1000 bytes, a 1000 byte message can be sent and another received at the same time by taking advantage of the full duplex nature of SPI.