***Ventajas y Desventajas del PROTOCOLO SPI***

Tanto el bus SPI como el I2C son llamados buses de tarjeta, es decir están pensados para trabajar a distancias pequeñas del entorno de una tarjeta, en caso de necesitar un bus serie a larga distancia hay que ir a buses de campo como RS485, CAN, etc…

El canal SPI fue diseñado para aplicaciones de transmisión de datos a velocidades altas (10 Mbps) y distancias cortas, del orden de 10 a 20 cms, ó bien dentro de un mismo PCB (circuito impreso), entre 2 circuitos integrados como podrían ser un  microcontrolador y otro dispositivo.

**Ventajas de SPI sobre I2C.**

* I2C No es Full-Duplex por lo que no permite envíos y recepciones al mismo tiempo.
* I2C un poco más complejo que SPI.
* I2C no tiene control de errores, por ejemplo mediante paridad etc. Aunque se puede realizar por Software.
* Velocidades de comunicación relativamente elevadas. En el caso de Arduino de hasta 8 Mhz.
* Completo control sobre la trama de bits al no exigir direccionamiento ni ACK.
* Se requiere un hardware sencillo (Barato)
* Requiere un menor consumo y menor electrónica de conexión que el I2C
* Como el Clock lo proporciona el master, los esclavos no necesitan osciladores (más barato)

**Desventajas de SPI:**

* No hay control del flujo por hardware.
* Las comunicaciones tiene que estar perfectamente establecidas de antemano. No puedes enviar mensajes de diferentes longitudes cuando convenga.
* No hay confirmación de la recepción como ocurre en I2C con el ACK. Es decir no sabemos si el mensaje a llegado al destino.
* Usa más pines que otros buses, ya que necesita uno por cada esclavo. Eso implica que no hay direccionamiento en la propia trama. A menos que se diseñe por software.
* Funcionamiento a distancias cortas
* Master único y casi sin posibilidad de master múltiple

