

SPU-001 TC2 SCK Protection

When managing the SPU-001 TC2 SPI bus, it is important that the SCK clock signal generated by the host MCU is free of irregular pulses or noise that cross the logic threshold, *even when the CS signal is not asserted*. Regardless of the state of the CS signals, the only behavior that SPU-001 TC2 permits on the SCK signal is pulses in multiples of 8.

This becomes a problem if another chip sharing the SPI bus communicates with the host MCU using a protocol where bits are sent in non-multiples of 8, or if the SCK signal floats when chips are not selected. An example of the latter is given below. The three red arrows show where the host allows the SCK signal to float while the CS signal is not asserted (logic high). This behavior causes the SPU to lose synchronization in a way that is difficult to recover from without a hard reset.

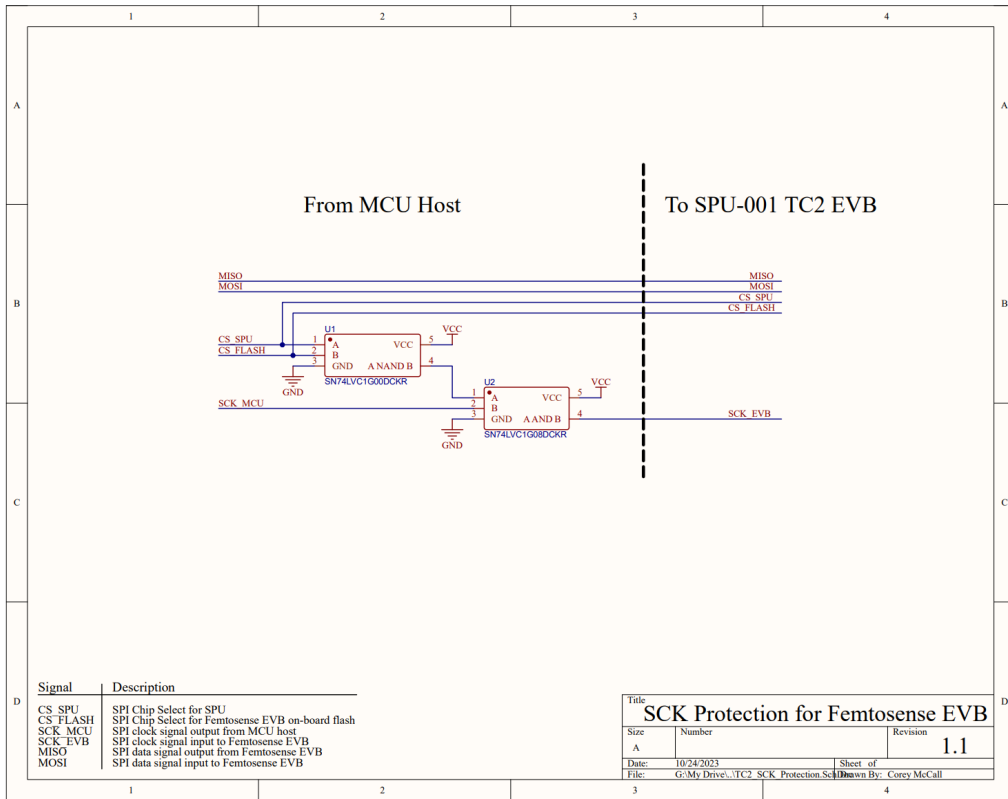


Figure 1. SPI bus traces contaminated by a floating SCK signal when no chips are selected. The three red arrows highlight areas where SPU-001 TC2 is affected.

If this behavior is expected, one solution is to intercept the SCK signal between the host MCU and SPU-001 TC2 in order to protect the SCK that SPU-001 TC2 sees. For example, using a Femtosense SPU-001 TC2 EVB (containing SPU-001 TC2 as well as a flash chip on the same SPI bus), a logic equation can be applied to the intercepted signal such that the SCK that the EVB sees is set to logic low unless one of the CS signals are asserted:

$$SCK_{EVB} = \overline{(CS_{SPU} \cdot CS_{FLASH})} \cdot SCK_{MCU}$$

This can be implemented with discrete logic gates such as the TI SN74LVC1G00DCKR (NAND) and TI SN74LVC1G08DCKR (AND) as shown below. Note that only the SPI signals are shown.



Future versions of the Femtosense SPU chip and EVB will not require this protection of the SCK signal.

Change Log

Version	Release Date	Description
1.0	2023-04-06	Initial release
1.1	2023-10-24	Clarification around EVK/EVB terminology