
There's Probably A Better Way To Do It

Dec 29, 2019

Contents

1	Electronics	3
1.1	SKR 1.3 and Fysetc TMC2209 V3	3
1.2	References	5

This is a software developer's attempt to keep his learnings about 3D printing in one place as he finds himself re-inventing the wheel over and over again sometimes.

English is not my first language so please be patient with my writings.

Any issues you see you can report here: <https://github.com/pkucmus/3dprinting-docs/issues>

I am not affiliated with any entity mentioned - just sharing my notes.

1.1 SKR 1.3 and Fysetc TMC2209 V3

I assume this also applies to SKR 1.4.

Fysetc made a stepstick with the TMC2209 stepper motor driver. The pinout of a stepstick is not a standard which leads to some compatibility issues one in particular is when you try to use it on the popular SKR 1.3 board. Big Tree Tech produced the SKR 1.3 board with a 32-bit chip on it and a pinout that allows to leverage the SPI/UART modes of the TMC chips without the need to run cables from the stick to the board.

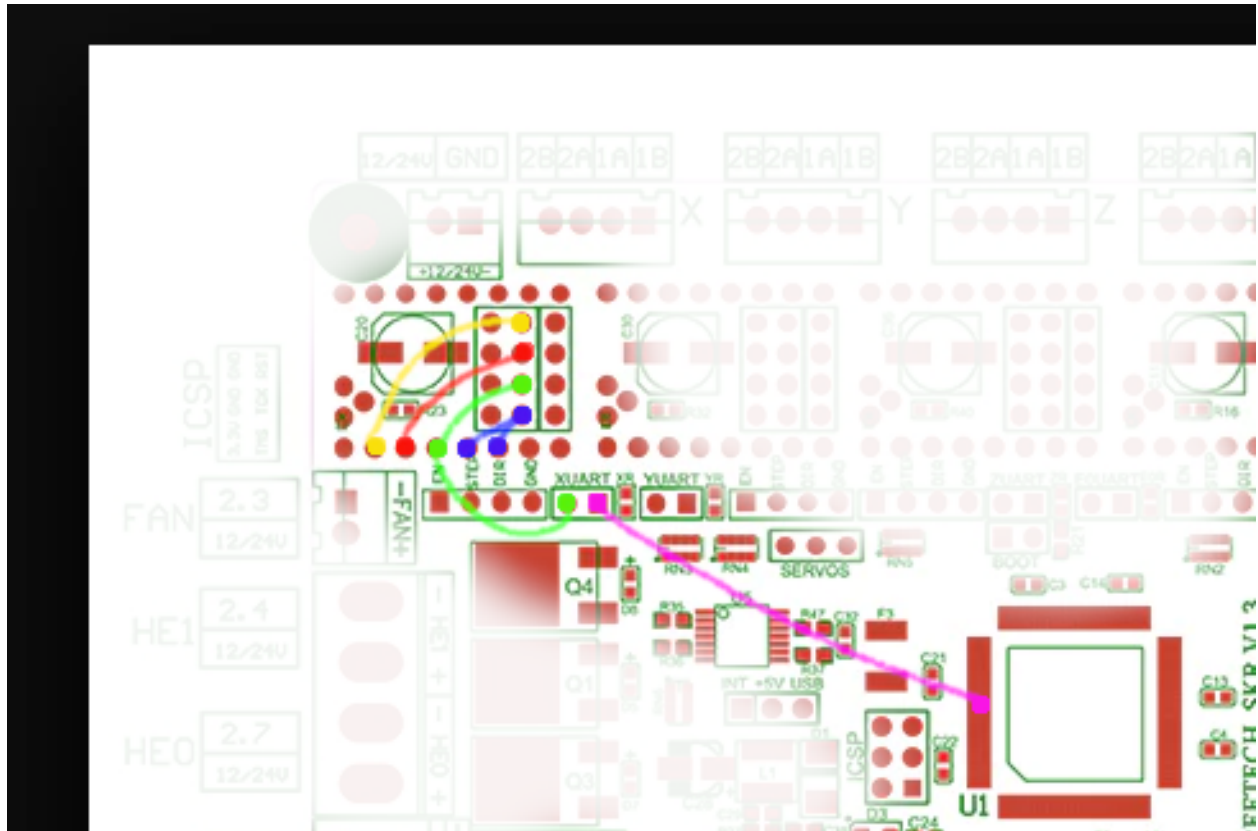
For the most part this works, if you make sure that the pinout of your stick matches the pinout of your board you can match parts to make a successful build - but due to the lack of (proper) documentation you may run into a situation like the one I ran into recently - had to marry Fysetc's TMC2209 V3 with an SKR 1.3 board.

1.1.1 The solder mask issue

Fysetc made an oopsie and used an inappropriate solder mask on the sticks which mark the TX and SPREAD pins incorrectly - this is not a huge deal as long as you notice/know about it ([ref](#))

1.1.2 UART, PDN, TX, RX

I don't fully understand what PDN stands for but [Wikipedia's definition](#) matches so sticking with that let's say it's just an interface allowing us to put the TMC's on a "communication network" using UART. UART is a communication protocol which at minimum requires receive and transmit pins but they figured out how to use UART with only one pin which is labeled as PDN. To unwind TX and RX from a PDN pin you just connect two cables (paths) to the PDN pin with a 1K ohm resistor on the TX line - that's it and that's exactly what is being done on the SKR 1.3 board - the M2 pins are lead to the so called UART pins (the two red ones below the steppers are to be) to the one on the left, the one on the right goes to the LPC1768 CPU of the board. X taken as an example goes to the 4.29 pin through a 1K ohm resistor and straight to the 1.17 pin (it's a software serial [leveraged as such in Marlin 2.0](#)).



Fysetc made exactly the same thing on the TMC2209 V3 stepsticks. The Trinamic 2209 chip has a PDN pin which they “mapped” to the RX pin through a 0 ohm resistor and a 1k ohm resistor to the TX pin - this means that the RX pin is effectively still a “full” PDN pin - this is important.

1.1.3 Discrepancies in pinouts

This is the gist of the problem - Fysetc imagined differently where the UART should be than Big Tree Tech so we ended up with the following pin mapping (the bottom row of the stepstick “socket”)

BTT SKR 1.3	BTT TMC2209 V1.2	Fysetc TMC2209 V3	Fysetc TMC2209 (before V3)
EN	EN	EN	EN
M0	MS1	MS1	MS1
M1	MS2	MS2	MS2
M2 (UART pin)	PDN	SPREAD	TX
RST	PDN	RX	RX
SLP	CLK	TX	CLK
STP	STEP	STEP	STEP
DIR	DIR	DIR	DIR

The M2 pin on the board for each of the stepsticks is the desired pin for the UART to use, at least by BTT’s “standard”. But our TMC2209 V3 is connected to the SLP and RST pins which additionally (by my multimeter checks) are shorted together. The conclusion is that **there is no way of using the out-of-the-box UART solution** of SKR 1.3 with the Fysetc TMC2209 V3 stepsticks.

1.1.4 The solution

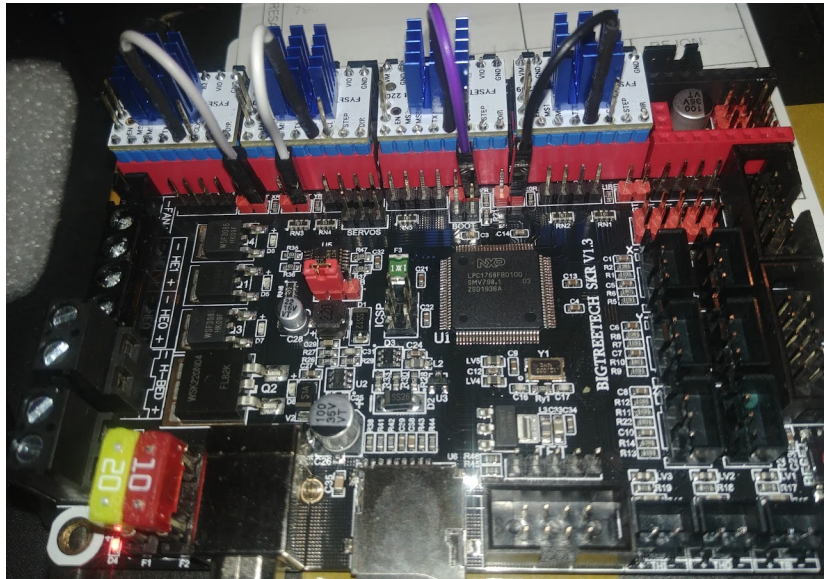
There are probably two solutions but only the first one is tested.

Jumper wires

I know this is not what you signed for when buying the SKR 1.3 but it is what it is :[

The solution that I can confirm working is to run jumper wires from the RX pin of the stepstick to the right side of the red UART pins on the board. This will effectively connect the PDN of the TMC chip to the TX and RX pins configured as the software serial in the boards CPU.

The picture below shows an example of how I dealt with it.



Re-mapping the software UART in Marlin

With a high chance of success one could try to find to which CPU pins are the RST and SLP pins in the stepstick headers connected to and set those [in the software](#) accordingly. That would leverage the 1K ohm resistor on the stepstick and left us with no wires. But I have no idea if the mentioned pins can work as UART (they don't necessarily have to) or if that solution is possible as I did not test it.

1.2 References

[LPC1768 pinout](#)