

# Ersatz der Kress-Spindel durch eine HF500 Hochfrequenzspindel (an LinuxCNC)

jean-claude.feltes@education.lu

Die Kress-Spindel lief mit 230V, geschaltet über einen Relaiskontakt der Motorsteuerung. Da diese sehr viel Lärm gemacht hat, wurde sie durch eine HF-Spindel ersetzt:  
<https://www.sorotec.de/shop/STEPCRAFT-HF-500-Spindel-mit-2m-Anschlusskabel.html>

Diese kommt mit einem Frequenzumrichter.

Die Spindel kann manuell eingeschaltet und in der Drehzahl über ein internes Poti des Frequenzumrichters gesteuert werden.

Für den CNC-Betrieb ist dies nicht praktikabel, denn die Spindel muss vom CNC-Programm ein- und ausgeschaltet werden können.

Die Drehzahl könnte ebenfalls vom Programm gesteuert werden, denn LinuxCNC bietet ein entsprechendes PWM-Signal. Dies erschien mir zur Zeit aber weder notwendig noch wünschenswert, die Möglichkeit die Drehzahl manuell über ein Poti einzustellen, erschien mir praktischer. Leider kann dazu nicht das Poti des Frequenzumrichters benutzt werden, denn bei externer Steuerung wird neben dem Einschaltsignal auch ein externes PWM-Signal erwartet.

Ein erster Versuch nutzte einfach das Spindle\_ON-Signal des Parallelports, statt des PWM-Signals wurde eine analoge Spannung 0...5V von einem Potentiometer benutzt.

Dies funktionierte prinzipiell, aber beim Hochfahren und vor allem beim Herunterfahren des PCs ergaben sich unkontrollierte Zustände am Parallelport, so dass die Spindel unkontrolliert eingeschaltet wurde. Ein unhaltbarer und gefährlicher Zustand.

Zunächst erstaunte mich dieses Verhalten. Warum hatte denn die alte Spindel korrekt funktioniert?

Bis ich auf die Idee kam: bei dieser wurde (in der Motorsteuerung) das „Charge pump“-Signal ausgewertet! Dieses Signal ist ein Rechtecksignal mit einer Frequenz im kHz-Bereich, welches nur dann anliegt, wenn die CNC-Steuerung läuft. Beim Hoch- und Runterfahren des PCs liegt es nicht an.

Die Auswertung des „Charge pump“- Signals kann leicht mit einem (retriggerbaren) Monoflop erfolgen. Der 74HC123 enthält 2 davon in einem IC, die Eingangsbeschaltung des Monoflops kann benutzt werden um „Charge pump“ und „Spindle ON“- Signal logisch zu verknüpfen.

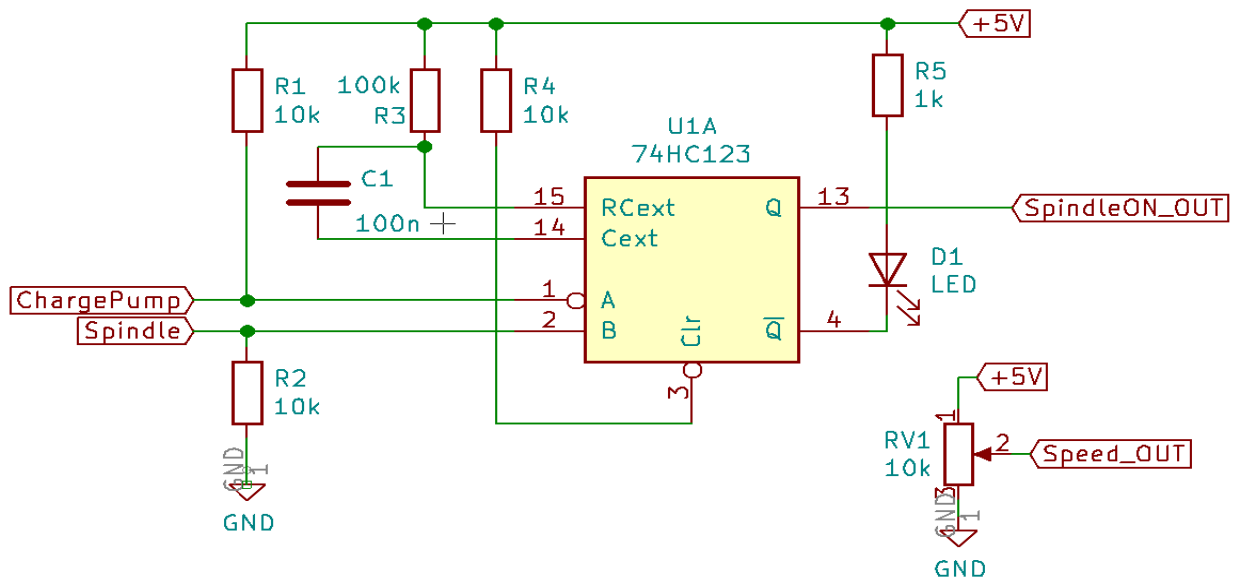
Die Eingänge A und B sind intern UND-verknüpft.

Am B-Eingang liegt das Spindle\_ON-Signal als Enable.

Am (invertierten) A-Eingang liegt das Charge\_Pump-Signal. Da dieses ein Rechteck-Signal ist, wird das Monoflop fortwährend getriggert und am Ausgang ergibt sich dauerhaft logisch 1, wenn beide Signale gleichzeitig anliegen.

Die Zeitkonstante liegt in der Größenordnung  $100\text{k}\Omega * 0.1\mu\text{F} = 10\text{ms}$ , so dass die Spindel schnell genug stoppt, und die Impulsdauer gross genug ist gegenüber der Periodendauer des Rechtecksignals.

Mit diesen Überlegungen ergibt sich die folgende Schaltung:



Die definitive Schaltung hat noch zwei kleine Erweiterungen:

- Ein 1nF-Kondensator parallel zu R2 beseitigt Störimpulse. Ohne diesen lief die Spindel öfter für Sekundenbruchteile, da schon sehr kurze Impulse am Monoflopeingang ausreichen um diesen zu triggern.
- Am Speed OUT Ausgang liegt ein 47 $\mu$ F-Elko um die Spannung zu glätten.