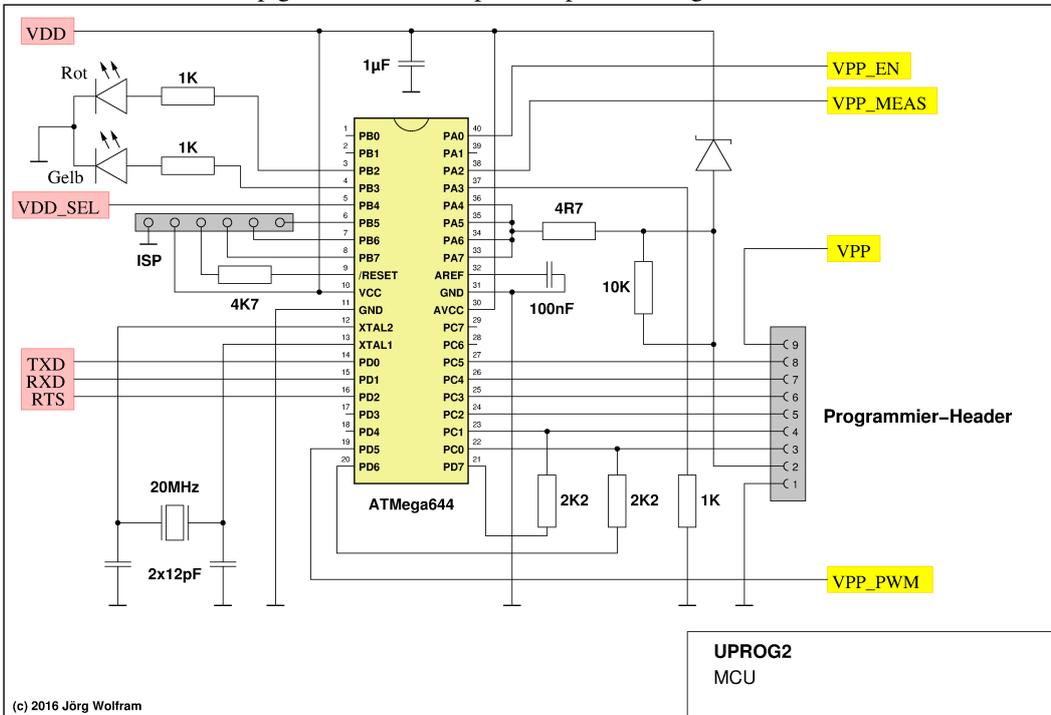


Der Typ des 3,3V Spannungsreglers ist unkritisch, Er sollte ab ca. 4,5V funktionieren und dabei 500mA bei 3,3V liefern. In meinem Programmer habe ich einen LM1117MPX-3.3 eingesetzt. Der FET sollte eine möglichst kleine U_{gs} haben (Logic-FET).

3 Die Zentraleinheit

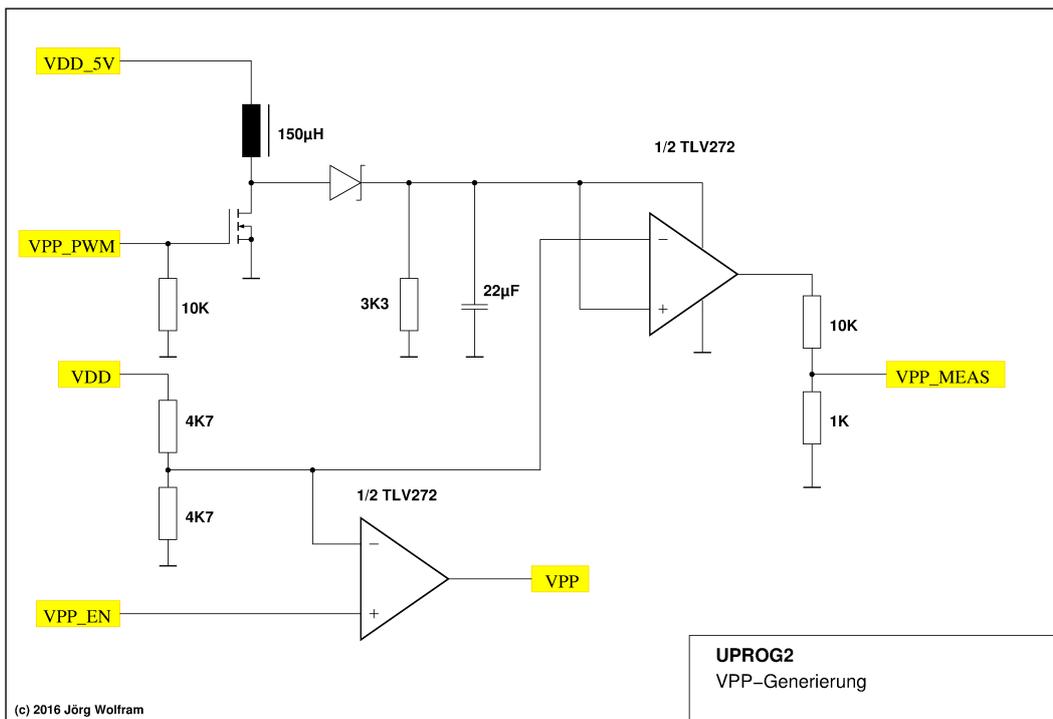
Kern ist ein ATmega644, der mit 20MHz getaktet wird. Die notwendige Firmware muss vor der ersten Verwendung via ISP-Header auf den Chip geflasht werden. Spätere Updates erfolgen dann über USB.



Insgesamt sind drei LEDs verbaut, eine grüne für die Betriebsspannung, eine gelbe für den 5V-Betrieb und eine rote als Aktivitätsanzeige. Die Festlegung der Anzahl der Programmierpins resultiert übrigens aus der DSUB9-Verbindung von älteren Varianten.

4 Programmierspannungs-Erzeugung

Die Programmierspannung (hauptsächlich für PIC-Controller) besteht aus einem PWM-gesteuerten Aufwärtswandler sowie einem OPV zum Ein-/Ausschalten. Der FET sollte wieder eine möglichst kleine U_{gs} haben (Logic-FET).



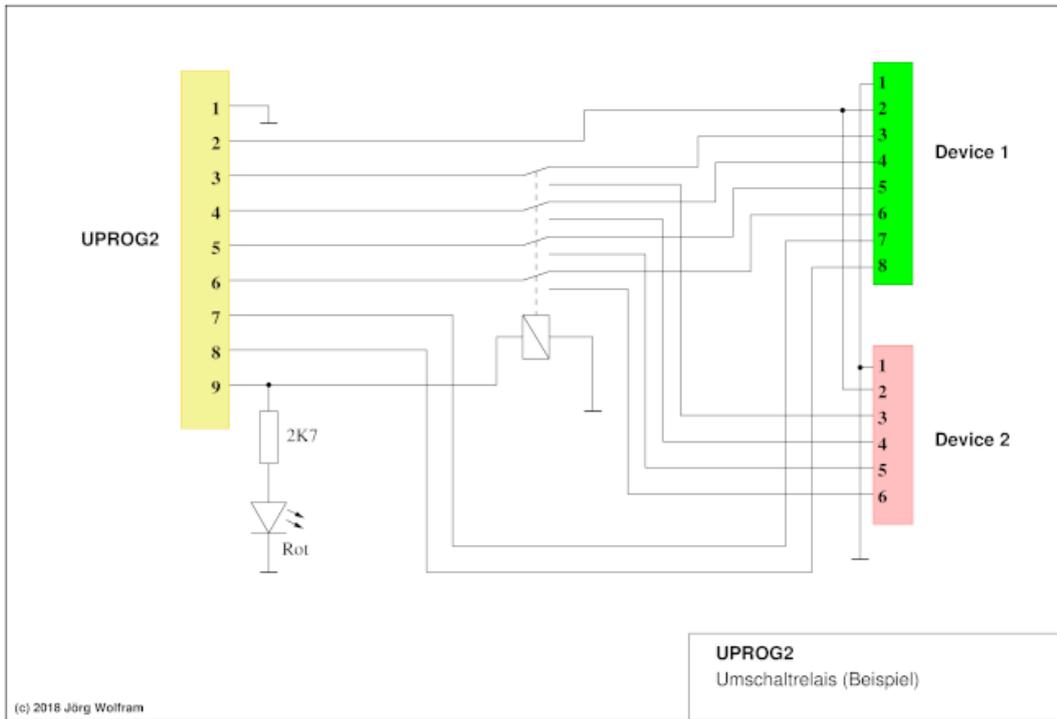
Über den zweiten OPV wird die Programmiervspannung gemessen und ggf. nachgeregelt.

5 Variante mit Bluetooth

Die Variante mit Bluetooth ist zur Zeit nicht dokumentiert, da sie nur als Lochrasteraufbau vorliegt. Als Bluetooth-Modul wird ein BTM112 oder BTM222 verwendet, die Power-Led habe ich dabei an den LINK-Anschluss des BT-Moduls angeschlossen. Dazu kommt noch ein LIPO-Akku nebst Step-Up auf 5V und Ladeelektronik, das könnte man aber durch eine kleine PowerBank ersetzen. Die Baudrate habe ich auf 230,4K festgelegt, außerdem muss PD4 für die Bluetooth-Variante auf Masse gelegt werden.

6 Device-Umschaltung

Die Device-Umschaltung (ab Version 1.30) ist nicht sonderlich kompliziert und auch nicht dargestellt, da ich sie „mal schnell“ auf Lochraster aufgebaut habe. Die folgende Schaltung ist nur ein Beispiel:



Die letzten beiden Steuersignale gehen hier nur an Device 1. Die meisten Devices kommen mit den ersten 4 Steuersignalen (nach GND/VCC) aus, so dass das kein Problem ist.