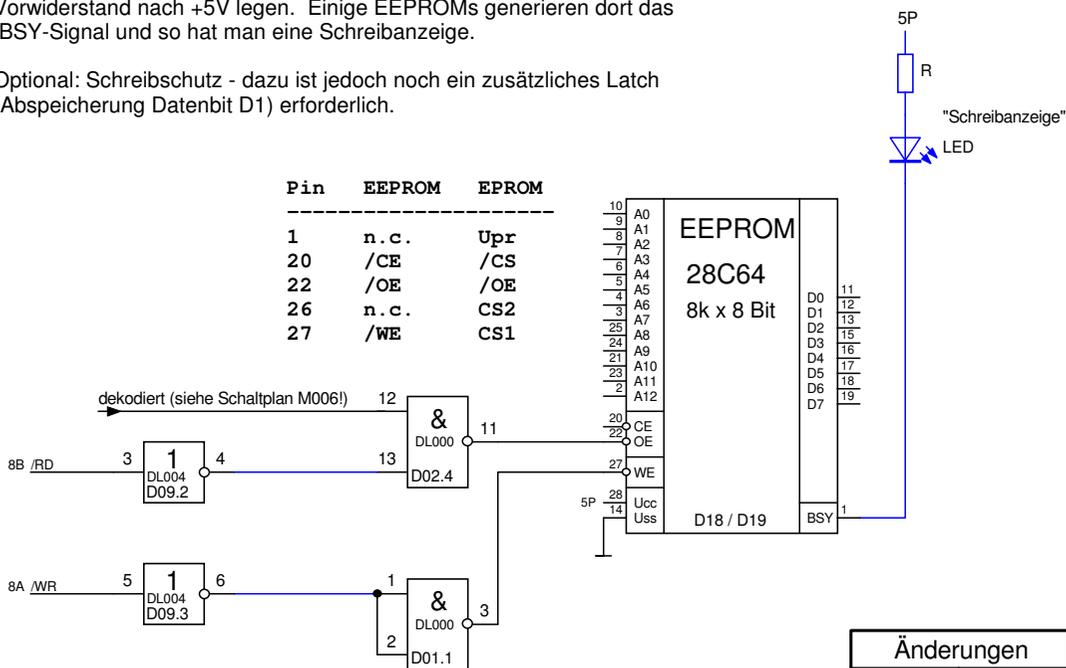


## erforderliche Änderungen am Modul:

- Die /CS-Dekodierung funktioniert schon bei Lese- und Schreibzugriffen, da weder /WR noch /RD berücksichtigt wurden.
- von Pin 13 des D02.4 das invertierte PRSEL abtrennen und das Signal RD (von Pin 4 des D09.2) anlegen. Damit wird /OE nur bei Lesezugriffen in das Modul aktiv.
- WR vom Pin 6 des D09.4 anstelle von PRSEL an die Pins 1 und 2 von D01.1 legen. Damit bekommen die (E)EPROMs ein undekodiertes Schreibsignal (also bei allen Schreibzugriffen), aber da das /CS-Signal vollständig dekodiert wird, ist das ausreichend. Man könnte also auch das /RD-Signal direkt an den /OE-Eingang legen.
- Upr von den Pins 1 der (E)EPROM's abtrennen, diese aber zusammenschaltet gelassen. Man kann das unbeschaltet lassen, oder zur Sicherheit mit einem PullUp-Widerstand nach +5V schalten. Oder optional eine LED mit Vorwiderstand nach +5V legen. Einige EEPROMs generieren dort das /BSY-Signal und so hat man eine Schreibanzeige.

Optional: Schreibschutz - dazu ist jedoch noch ein zusätzliches Latch (Abspeicherung Datenbit D1) erforderlich.

Pin	EEPROM	EPROM
1	n. c.	Upr
20	/CE	/CS
22	/OE	/OE
26	n. c.	CS2
27	/WE	CS1



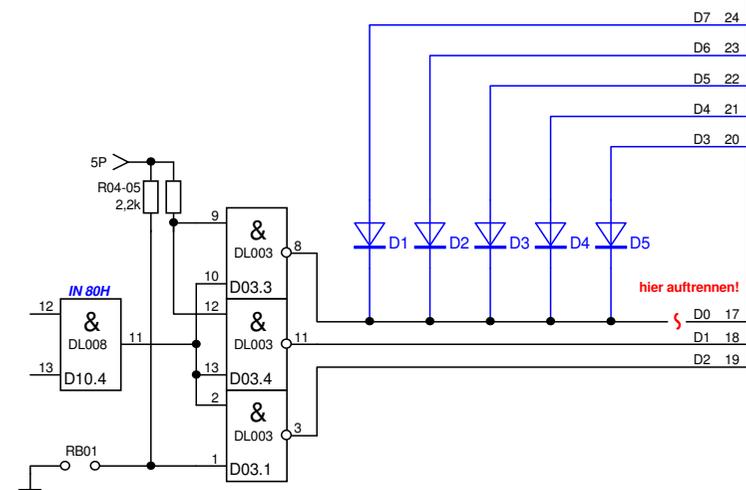
## Erzeugung Strukturbyte 01H:

**F8H = M028 (RB01 offen)**

**FCH = M006 (RB01 geschlossen)**

**01H = Autostart-ROM**

- Brücke RB01 öffnen
- Ausgang Pin 8 von D03.3 vom Datenbus DB0 abtrennen
- an diesen Ausgang fünf Schottkydioden mit den Anoden an den Datenbus DB3...DB7 schalten.



Änderungen		Datum	Name	Bezeichnung	Blattzahl: 1
Datum	Name	gez.: 31.12.2005	M. Leubner	Umbau M006/M028 auf 16K EEPROM	Blatt-Nr.: 1
		gepr.:			
		neu = blau!		Zeichnungs-Nr.:	