

# Vorgehensweise:

Analyse des gegebenen Schaltbilds

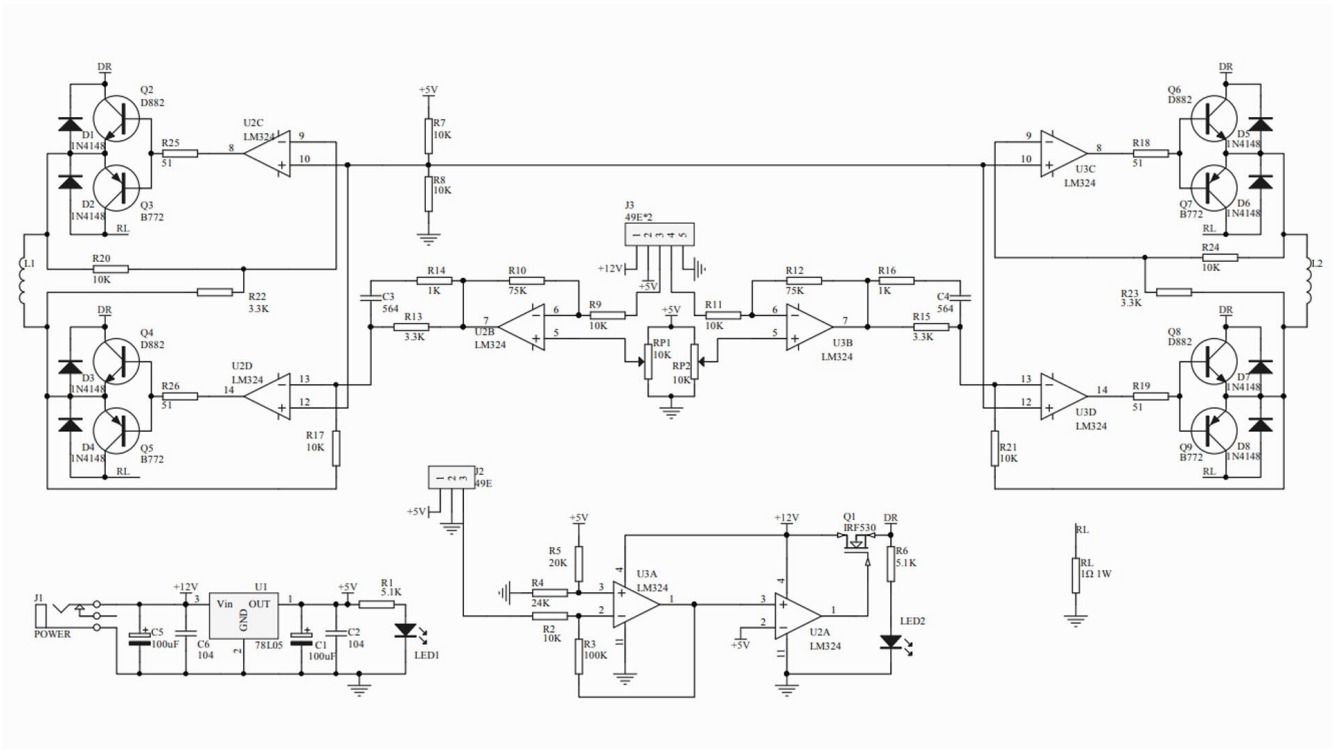
Übertragungsfunktion der Schaltung

Diskretisierung der Übertragungsfunktion

Aufstellen der Differenzengleichung

Simulation der Sprungantwort als C – Programm

Portierung des Reglers auf  $\mu\text{C}$



Analyse der Beschaltung um U2D:

$$Z_{in} = R_{13} \parallel (R_{14} + C_3)$$

$$Z_{out} = R_{17}$$

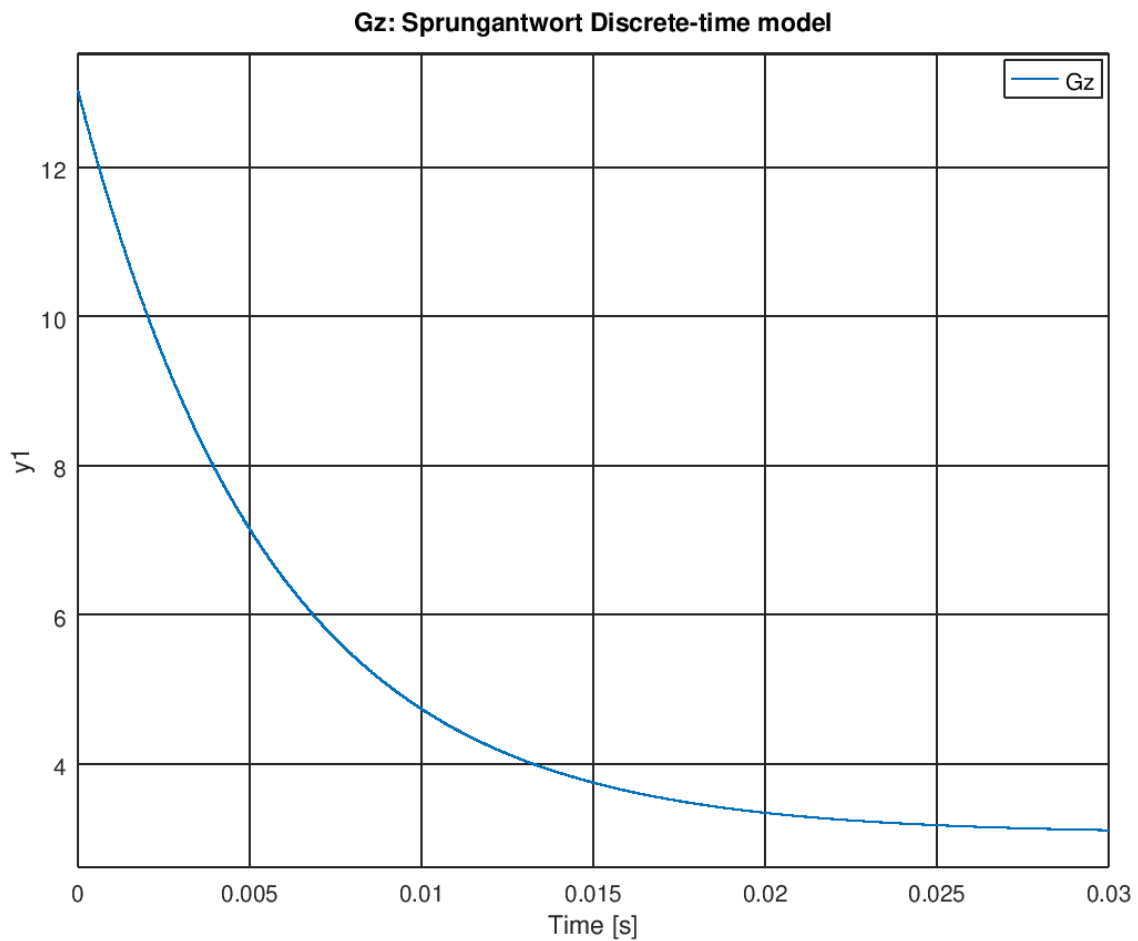
Übertragungsfunktion  $G(s) = Z_{out} / Z_{in}$

$$G(s) = \frac{0.07297 s + 3.03}{0.0056 s + 1}$$

Hilfsmittel:

Skript Regelungstechnik

octave ( matlab clone)



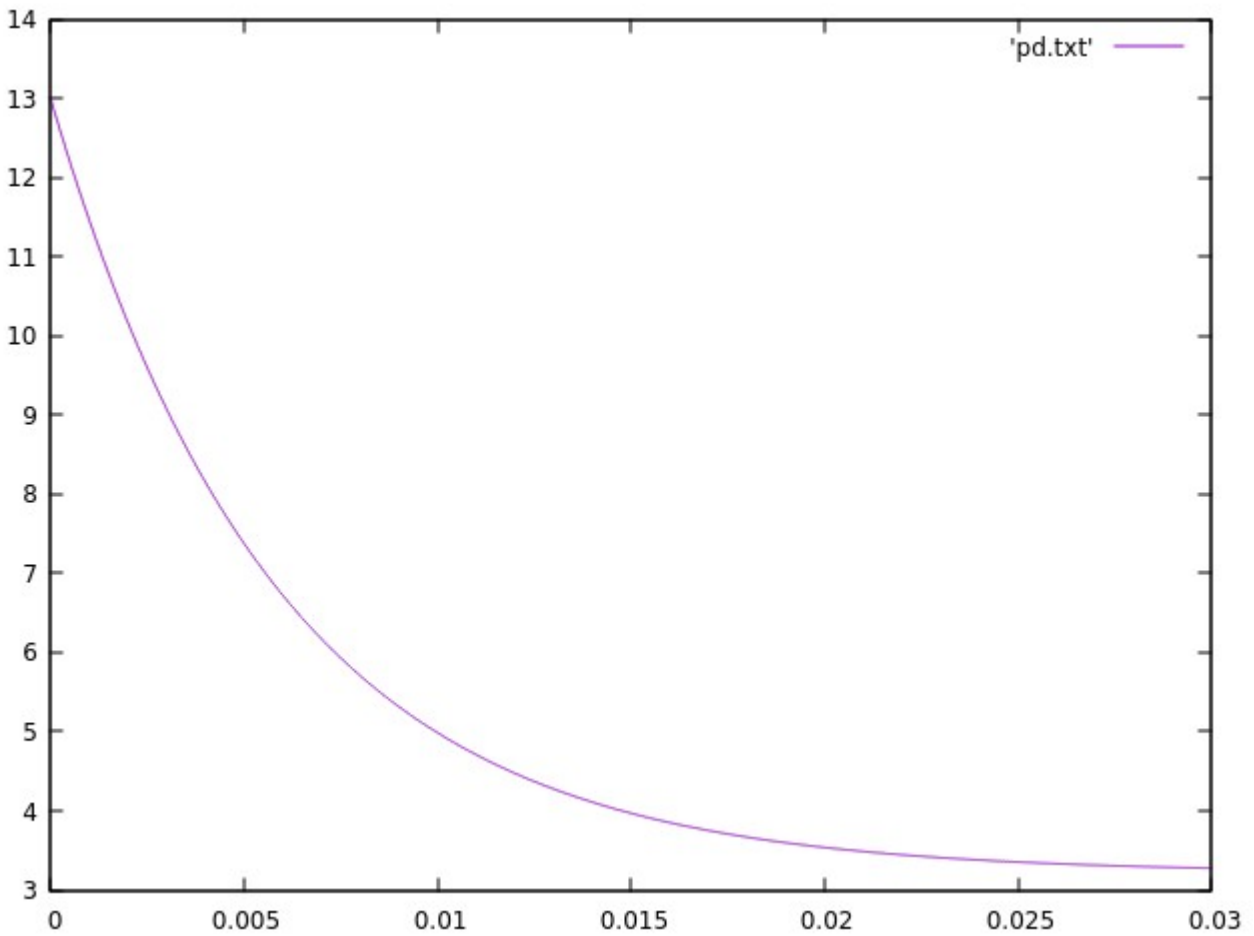
Diskretisierung  $G(s) \rightarrow G(z)$

$$G(z) = \frac{z - 0.9996}{0.0768z - 0.07666}$$

Schaltung zeigt PD - Verhalten

Hilsmittel:

Tustinformel



$$G(z) = Y(z) / X(z)$$

Umstellen nach  $Y(z)$

und Differenzgleichung aufstellen

$$y(k) = 13.0225 * x(k) - 13.0170 * x(k-1) + 0.99829 * y(k-1)$$

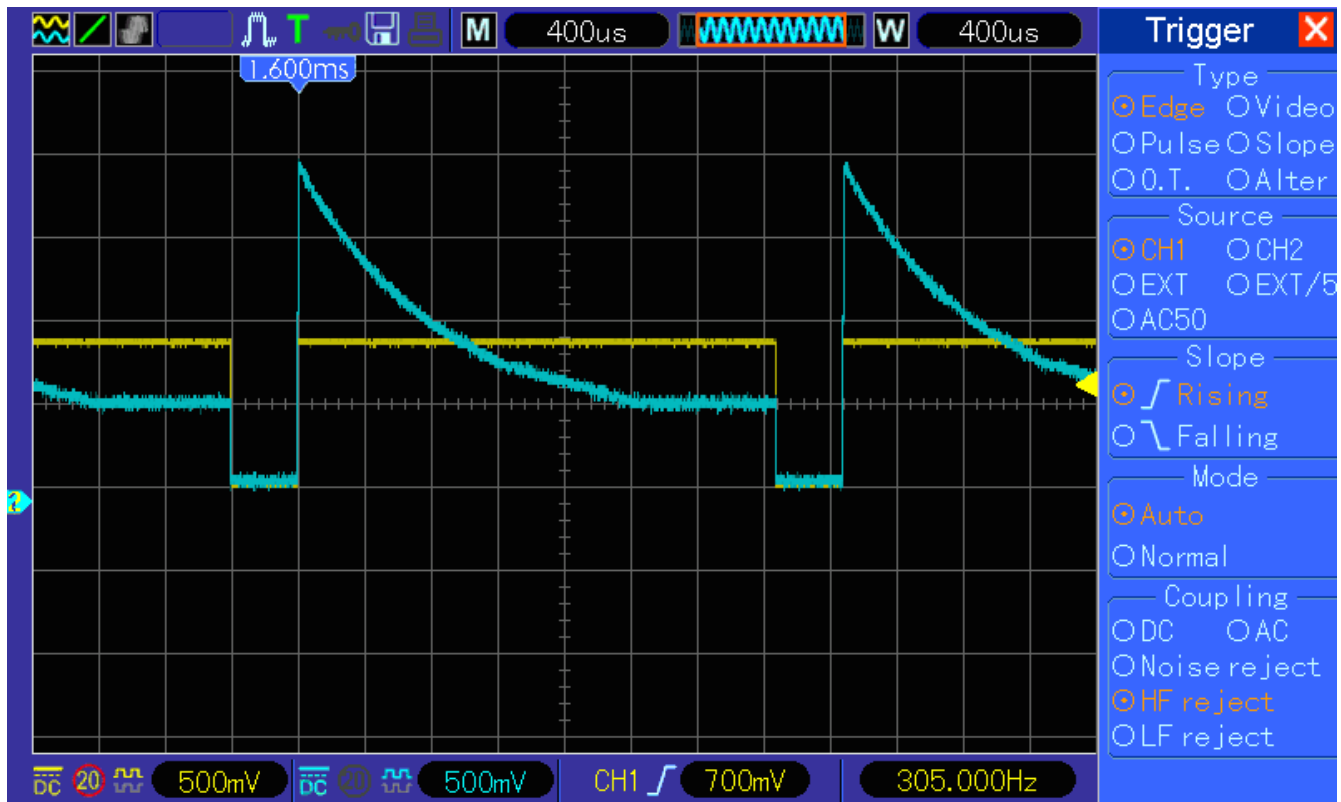
Simulation der Sprungantwort mit C-Programm:

$$y = 13.0225 * x - 13.0170 * x_{-1} + 0.99829 * y_{-1};$$

Hilfsmittel:

Viel Rechnerei und Brüche vereinfachen auf Papier.

gcc



Portierung des Reglers auf einen  $\mu\text{C}$ .

Messen der Sprungantwort.

Hilfsmittel:

Oszi ( das printf für Elektroniker)

$\mu\text{C}$  mit DAC und ADC

CH1 (gelb) : Reglerinput (Wiederholung der Sprungfunktion für besseres Triggern)

CH2 (blau) : Reglerausgang (In den Pulspausen ( $400\mu\text{s}$ ) wird der Regler ausgeschalten)