# Circuits and Transforms

Gautam Singh

Contents

**1 Bilinear Transform** 1

***Abstract*—This manual provides a simple introduction to Transforms**

## 1 Bilinear Transform

1. Formulate the differential equation for Fig. 1.1.

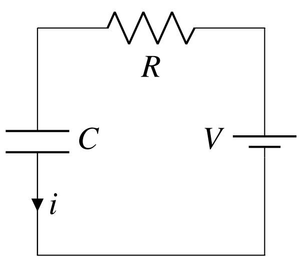


Fig. 1.1

**Solution:** Applying KVL on the loop,

$$V - iR - \frac{1}{C}\int_0^t i dt = 0 \tag{1.1}$$

where $i(0) = 0$, $V_C(0) = 0$. Denote by $V_C$ the voltage at the capacitor. Then,

$$i = C\frac{dV_C}{dt} \tag{1.2}$$

and therefore using (1.2) in (1.1), we get the differential equation

$$V - \tau\frac{dV_C}{dt} - V_C = 0 \tag{1.3}$$

where $\tau \triangleq RC$ is the time constant of the circuit.

2. Find $H(s)$ considering the output voltage at the capacitor.
**Solution:** Transforming Fig. 1.1 to the $s$-domain,

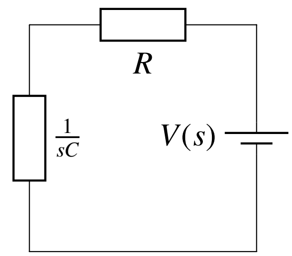


Fig. 1.2

Thus, using the voltage division formula, the voltage across the capacitor is given by

$$V_C(s) = V(s)\frac{\frac{1}{sC}}{\frac{1}{sC} + R} \tag{1.4}$$

$$= V(s)\frac{1}{1 + sCR} \tag{1.5}$$

Therefore, the transfer function is given by

$$H(s) = \frac{V_C(s)}{V(s)} = \frac{1}{1 + sCR} \tag{1.6}$$

3. Plot $H(s)$. What kind of filter is it?
**Solution:** The Python code `codes/1_3.py` plots $H(s)$. Clearly, $H(s)$ is a low-pass filter.

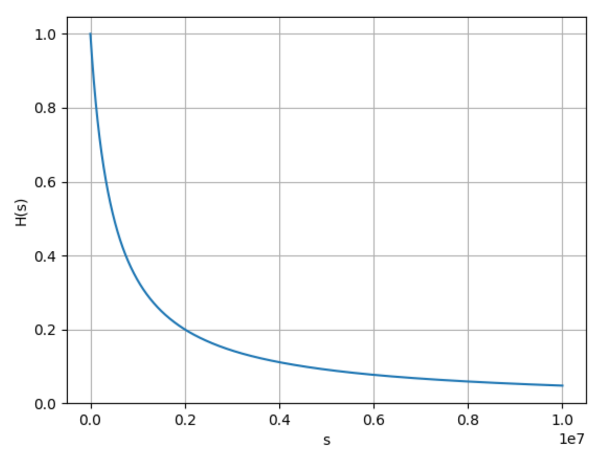


Fig. 1.3: Plot of $H(s)$.

4. Using trapezoidal rule for integration, formu-

late the difference equation by considering

$$y(n) = y(t)|_{t=n} \tag{1.7}$$

**Solution:** Integrating (1.3) between limits $n-1$ to $n$ and applying the trapezoidal formula,

$$\frac{v_C(n) + v_C(n-1)}{2} + \tau\left(v_C(n) - v_C(n-1)\right) = \frac{V(n) + V(n-1)}{2} \tag{1.8}$$

for $n > 0$, where $v(0) = 0$.

5. Find $H(z)$.
**Solution:** Applying the Z-transform on both sides of (1.8),

$$V_C(z)\left[(2\tau + 1) - z^{-1}(2\tau - 1)\right] = V(z)\left(1 + z^{-1}\right) \tag{1.9}$$

Hence,

$$H(z) = \frac{1 + z^{-1}}{(2\tau + 1) - (2\tau - 1)z^{-1}} \tag{1.10}$$

since $\left|\frac{2\tau-1}{2\tau+1}\right| < 1$, the ROC is $|z| > 1$.

6. How can you obtain $H(z)$ from $H(s)$?
**Solution:** We use the bilinear transformation. Setting

$$s \triangleq \frac{2}{T}\frac{1 - z^{-1}}{1 + z^{-1}} \tag{1.11}$$

we get

$$H(z) = \frac{1}{1 + \frac{2\tau}{T}\frac{1-z^{-1}}{1+z^{-1}}} \tag{1.12}$$

Setting $T = 1$ gives (1.10).

7. Find $v_C(n)$. Verify using ngspice and the differential equation.
**Solution:** Note that $v(n) = Vu(n)$. Thus,

$$V(z) = \frac{V}{1 - z^{-1}} \tag{1.13}$$

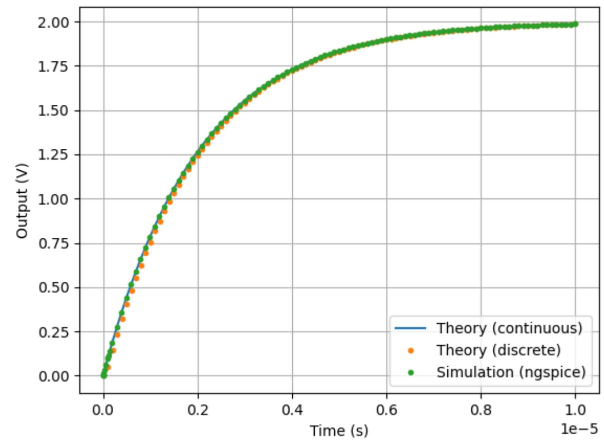


Fig. 1.7: Representation of output across $C$.

Therefore,

$$V_C(z) = H(z)V(z) \tag{1.14}$$

$$= \frac{TV\left(1 + z^{-1}\right)}{\left(1 - z^{-1}\right)\left((2\tau + T) - (2\tau - T)z^{-1}\right)} \tag{1.15}$$

$$= \frac{V\left(1 + z^{-1}\right)}{2} \sum_{k=-\infty}^{\infty} \left(1 - p^k\right) u(k) z^{-k} \tag{1.16}$$

where $p \triangleq \frac{2\tau-T}{2\tau+T}$. Thus,

$$v_C(n) = \begin{cases} \frac{V}{2}\left[u(n)\left(1 - p^n\right) + u(n-1)\left(1 - p^{n-1}\right)\right] & n > 0 \\ 0 & n \le 0 \end{cases} \tag{1.17}$$

We take $T = 10^{-7}$ as the sampling interval. The python code `codes/1_7.py` verifies these equalities.