

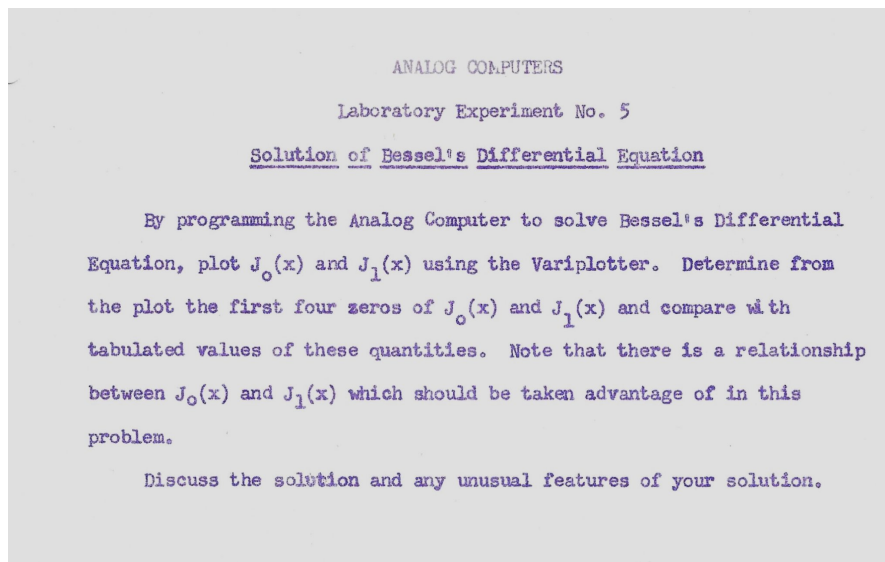


Figure 1: Classic Exercise

Generating BESSEL functions

In June 2022, fellow analog computer enthusiast Dr. CHRIS GILES sent me the classic exercise shown in figure 1. It is the basis of this application note.

BESSEL *functions* were first described by DANIEL BERNOULLI¹ and later generalised by FRIEDRICH BESSEL.² BESSEL functions of the first kind are usually denoted by $J_n(t)$ and are solutions of the BESSEL differential equation

$$t^2\ddot{y} + t\dot{y} + (t^2 - n^2)y = 0. \quad (1)$$

¹01/27/1700–03/27/1782

²07/22/1784–03/17/1846

Sometimes these are called *cylindrical harmonics*. The parameter n in the equation above defines the *order*. In the following, $n = 0$ and $n = 1$ are assumed.

For $n = 0$ equation (1) can be written as

$$\ddot{y} = -\frac{1}{t}\dot{y} - y$$

after dividing by t^2 and solving for $\ddot{y}$. This can be readily transformed into an analog computer program by applying KELVIN's feedback technique. The only thing to take into account is the term $\frac{1}{t}$ which is not well suited for an analog computer due to the pole at $t = 0$. It is far more easy to directly generate the quotient $\frac{\dot{y}}{t}$ as $\dot{y} \rightarrow 0$ with $t \rightarrow 0$.

The resulting program is shown in figure 2. Time t has been substituted by machine time τ which is generated using an integrator. The parameter $\dot{\tau}$ determines how fast τ rises and should be set so that $0 \leq \tau \leq 1$ during one computer run.

The relationship between $J_0(t)$ and $J_1(t)$ mentioned in the original exercise is an interesting one and can be found in [BRONSTEIN et al. 1989, p. 442] or any other standard textbook. In general

$$\frac{d}{dt} (t^{-n} J_n(t)) = -t^{-n} J_{n+1}(t)$$

holds, which implies

$$J_1(t) = -\dot{J}_0(t)$$

for the case $n = 0$. Accordingly $J_1(\tau)$ is readily available in the program as it is just $-\dot{y}$.

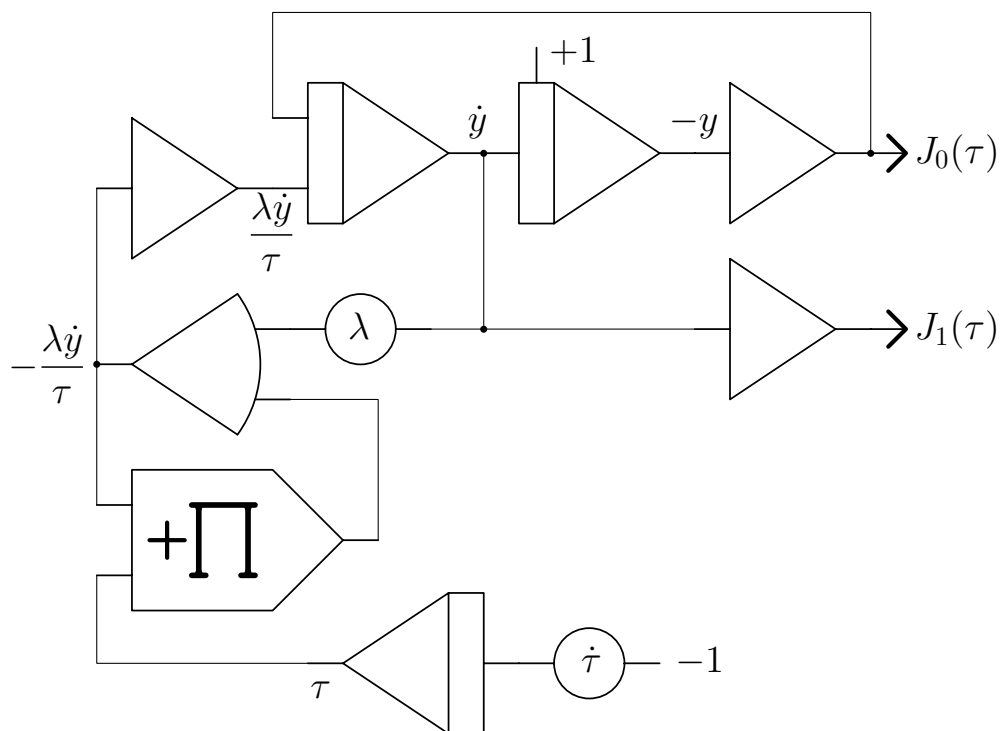
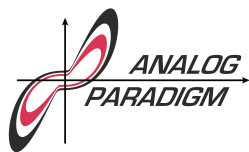


Figure 2: Analog computer program for generating BESSEL functions $J_0(\tau)$ and $J_1(\tau)$



Analog Computer Applications

Figure 3 shows the overall program setup on *THE ANALOG THING*.³ A typical result is shown in figure 4. Note that the program has not been properly scaled. $\dot{\tau}$ was set according to the operate-time of the machine running in repetitive mode. The central scaling factor λ was set to get the desired result. ☺

References

- [BRONSTEIN et al. 1989] I. N. BRONSTEIN, K. A. SEMENDJAJEW, *Taschenbuch der Mathematik*, 24. Auflage, Verlag Harri Deutsch, Thun und Frankfurt/Main, 1989

³See <http://the-analog-thing.org>.

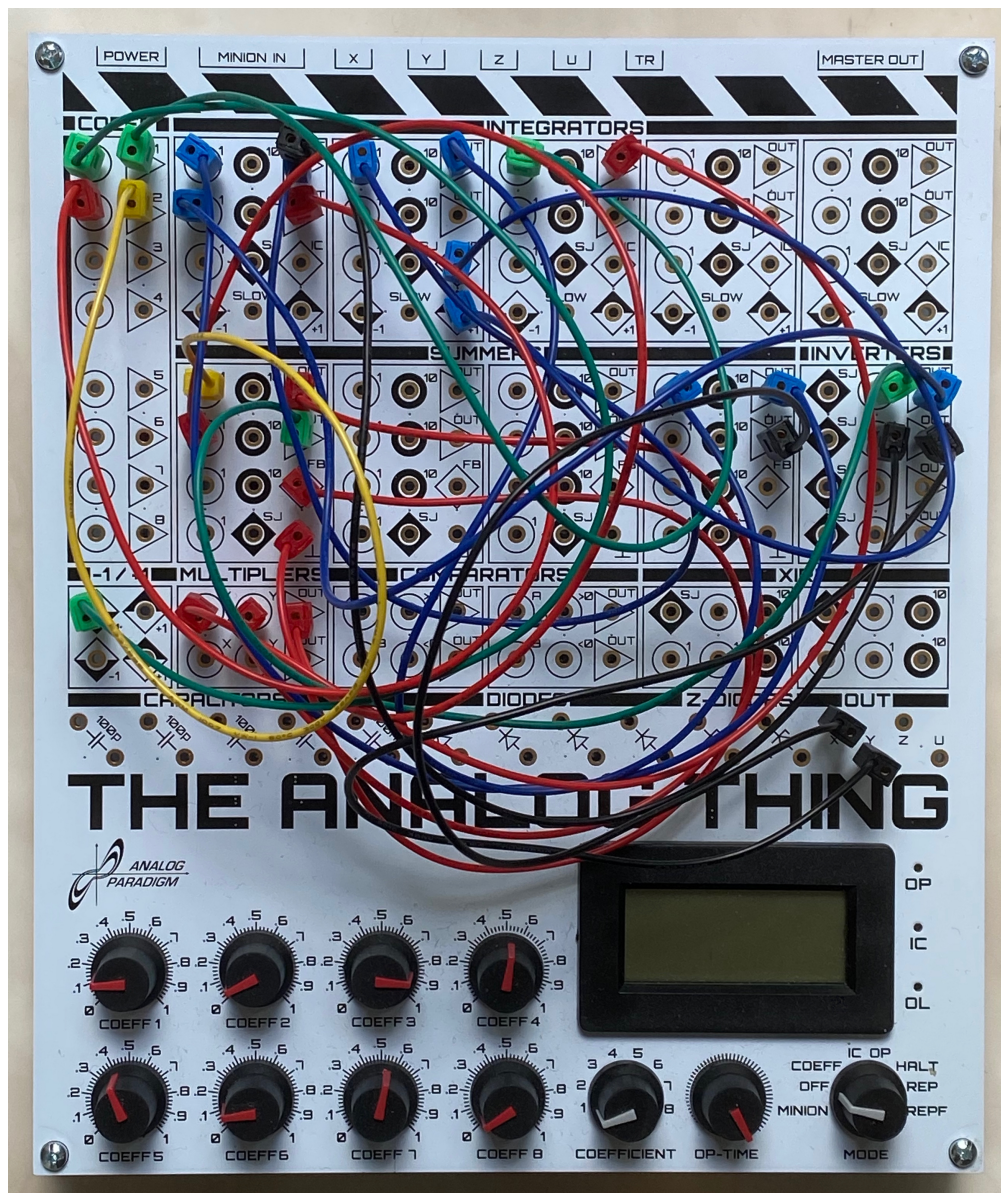


Figure 3: Setup for generating $J_0(\tau)$ and $J_1(\tau)$

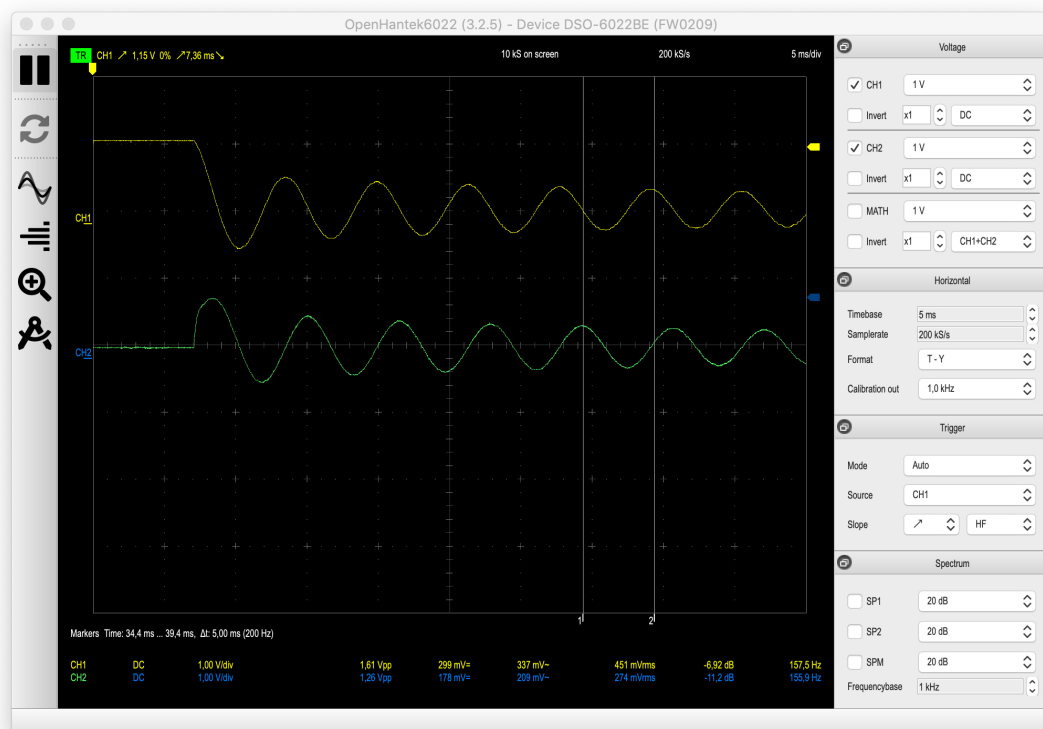


Figure 4: Typical output for $J_0(\tau)$ and $J_1(\tau)$