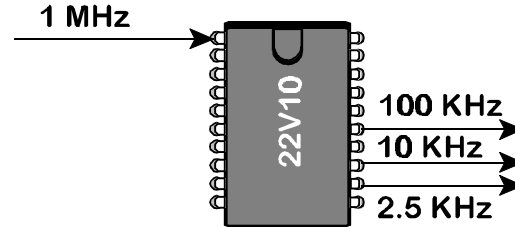


A LABORATORY CLOCK GENERATOR

In some experiments, the 1MHz clock provided may be too fast. Using one 22V10 PAL and a minimal knowledge of VHDL, one can fabricate a clock generator as shown below.

The 22V10 may be programmed with two divide by 10 counters (Decade Counters) and a divide by 4 counter. With a 1MHz input, this will produce a 100KHz, a 10KHz and a 2.5 KHz clock source.



The actual pinout that might be obtained is shown below.

c22V10	
clk_in = 1	24 * not used
not used * 2	23 = (cntr2_IBV_2)
not used * 3	22 = (cntr3_IBV_0)
not used * 4	21 = (cntr2_IBV_0)
not used * 5	20 = (cntr1_IBV_2)
not used * 6	19 = (cntr1_IBV_0)
not used * 7	18 = (cntr1_IBV_1)
not used * 8	17 = c100
not used * 9	16 = (cntr2_IBV_1)
not used * 10	15 = c2
not used * 11	14 = c10
not used * 12	13 * not used

The entity description is given below (it is so simple, there is no need to make you figure it out!)

```
entity clock_gen is port(
    clk_in: in std_logic;
    c100, c10, c2: out std_logic);
end clock_gen;
```

You will need, in the architecture part, three integer signals:

```
signal cntr1, cntr2: integer (0 to 15);
signal cntr3: integer (0 to 3);
```

From here, with a few if..then..else statements, you should be able to devise this clock generator. It will be a lot easier to use than wiring up several SN74163's to do the job!