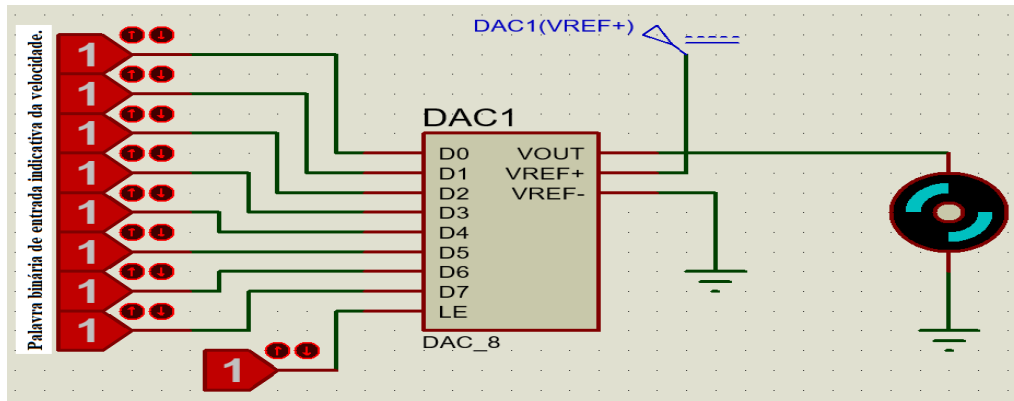


## BCC 265 – Atividade Aberta 02

Suponha a necessidade de atuar sobre 4 motores DC. Para tanto, apenas um motor deverá ser ativado por vez onde a sua seleção será feita por intermédio de uma palavra de seleção de dois *bits* (*bits*  $S_1$  e  $S_0$ ). Além da seleção do motor, o módulo também receberá uma palavra indicativa de velocidade composta por 4 *bits* (*bits*  $V_3$ ,  $V_2$ ,  $V_1$  e  $V_0$ ).

Externamente, cada motor terá a sua ligação feita através de um conversor digital-analógico (DAC) conforme ilustra a figura a seguir:



Na figura acima, tem-se:

- D0 a D7 → entradas da palavra binária a ser convertida para analógica (D7 = MSB)
- LE → sinal de habilitação (*enable*). Ligar no  $V_{cc}$ .
- Vout → saída convertida. Neste caso, pode-se ligar diretamente ao motor. O outro terminal do motor liga-se ao terra ( $GND$ ).
- VREF+ → tensão de referência para se realizar a conversão (“pólo positivo”). Pode-se ligar no  $V_{cc}$ .
- VREF- → tensão de referência para se realizar a conversão (“pólo negativo”). Pode-se ligar no terra ( $GND$ ).

Para controlar a velocidade do motor, o módulo de controle, a partir da palavra “V” (relativa à velocidade) fará o seguinte tratamento:

```
se V==0 então "Velocidade=0"
else se V < 3 então "Velocidade 1"
else se V < 8 então "Velocidade 2"
else "Velocidade 3".
```

Como mencionado, os bits resultantes do tratamento de “V” serão ligados, externamente, à um DAC – cuja saída atuará no motor.

Para conseguir as referidas velocidades, as entradas do DAC deverão ser ligadas, respectivamente, às seguintes palavras: “0000**1111**” (Velocidade 1), “00**11**00**11**” (Velocidade 2) e “**11**0000**11**” (Velocidade 3). A descrição “Velocidade=0” denota uma saída formada pela palavra “00000000”.

Implementar, em Verilog, o módulo de controle gerando-se apenas as palavras que serão atribuídas ao DAC externo a partir do valor de “V” (*bits*  $V_3$ ,  $V_2$ ,  $V_1$  e  $V_0$ ). O módulo de controle também deverá exportar os sinais de habilitação para os motores a partir da entrada “S” (*bits*  $S_1$  e  $S_0$ ). Para essa atividade, o módulo de simulação também deverá ser implementado.