SINGLE TRANSISTOR SWITCHING CIRCUIT DESIGN

Jeff Holtzman

■Hobbyists dealing with digital circuits are terrified by interfacing. But for computers to do useful things, they must have a way of talking with outside circuits: relays, motors, lightbulbs, LED's, etc. Things aren't so bad when dealing with standard TTL outputs, but what do you do when you have to connect a 5-horsepower motor to a CMOS inverter? Doing that is simple; we'll show you how to do the design calculations, and we'll present a short BASIC program that does the work. One feature of that program is that after making the calculations, it outputs the value of the closest standard resistor.

To design a switching circuit, before doing any calculations, make sure that a single transistor will do the job. There are no calculations involved, just a little common sense. If your driving device is a 6522 VIA, or a Z80 PIO, or similar, and you want to control an NC (Numerical Control) lathe with a 5-horsepower motor, you'll need a few transistors to build up enough drive to control a relay large enough.

If you need to turn an LED on and off, you may be able to use a transistorless driving circuit. You may get by using just a current-limiting resistor, although doing that may not allow the LED to glow at full brightness.

If a one-transistor circuit will do the job for you, choose a transistor. Double the voltage and current requirements of your load device, and then search your databooks and catalogs for a transistor that meets those specifications. Find several that will do, in case your first choice is unavailable, or expensive.

Every transistor has a DC-gain factor, called HFE. Unfortunately, HFE varies from device to device, even between devices of the same type. HFE also varies with load and temperature. Since many applications aren't critical, a few assumptions can be made to get things off the ground. Those will allow you to arrive at tentative values suitable for breadboarding; remember that slight adjustments may have to be made later.

With specifications handy, choose a value for HFE halfway between the minimum and maximum values given in the databook. If you don't have the specs, you can assume a value of 100 for a small-signal transistor, or 10 for a power transistor. That lets you estimate the value of the current-limiting resistor in the base circuit (R1 in Fig.1). As base current is defined as the collector current over HFE, you can get the base current by dividing the collector current by 100, or by 10, depending on the type of transistor. Before finding the value of R1, we have to find the value of R2.

Making the calculations

To calculate the values of the resistors in the switching circuit shown in Fig. 1, work backwards from the load. You have to know how much current the load will draw, and its resistance, to calculate the voltage drop, V(L), that will appear across it. According to Kirchoff's law,

V(CC) = V(L) + V(R2) + V(CE)

V(CE) is the voltage dropped across the collector and emitter leads of Q1; that voltage is often about 0.3 volt, so it may be ignored. By making that simplification and rearranging our equation we find that V(R2) = V(CC) - V(L). We need the resistance of R2, and by Ohm's law that is equal to the voltage across R2 divided by the current flowing through it, or, R2 = (V(CC) - V(L)) / I(L). If the load is an LED, we can assume that V(L) is 1.5 volt. If not, V(L) must be expressed as I(L) * R(L). In the latter case, the value of R2 would be expressed as R2 = (V(CC) - [I(L) * R(L)]) / (I(L))

To calculate the value of R1, we use the value of HFE we found above. Since HFE = I(C) / I(B), I(B) can be expressed as I(B) = I(C) / HFE.

By Thevenin's law we know that I(C) must equal I(L), so I(B) = I(L) / HFE. Again, by Kirchoff's law we know that V(IN) = V(R1) + V(BE). where V(BE) is the base-emitter voltage drop. For the sake of simplicity, we can assume that V(BE) is zero, so the voltage dropped across R1 must equal the input voltage. Therefore, the value of R1 can be simply expressed as R1 = V(IN) / I(B). Since I(B) can be expressed in terms of collector current, R1 = V(IN) / [I(C)/HFE].

The unknown quantities (R1 and R2) are expressed in terms of the known quantities, so it's time to plug the values into the equations and solve them. Rather than bang those values out your calculator, type the program shown in Listing 1 into your computer and save it for the next time you need to design a switching circuit.

The program provides features you may find convenient. For example, that program can be used to design a circuit with any driving voltage, but if you happen to be using TTL, the program assumes a value of 2.4 volts for V(IN). Likewise, if you will be driving an LED, the program assumes a load current of 15 mA, and a 1.5 volt drop across the LED.

The program was written in MBASIC-80, and it should be simple to translate into other dialects of BASIC. The main routines have been written as

```
10 REM switching transistor circuit
20 REM design  jh  10-22-85  for RE
30 FIRST = -1
40 PRINT CHR$(27);"*";
50 GOSUB 1000
60 PRINT
70 INPUT "Another? ", YESNO$
80 IF LEFT$(YESNO$,1) = "Y"
   OR LEFT$(YESNO$,1) = "y"
   THEN GOTO 50
90 END
1000 REM get and print data for
     switching circuit
1010 PRINT
1020 INPUT "Enter supply voltage
     (VCC) in volts: ", VCC
1030 INPUT "Enter HFE of
     transistor: ", HFE
1040 INPUT "Is load an LED
     (y/n) ? ", YESNO$
1050 YESNO$=LEFT$(YESNO$,1)
1060 IF YESNO$="Y"
     OR YESNO$="y"
     THEN VL=1.5: IL=15: GOTO 1100
1070 INPUT "Enter load
     resistance (RL): ", RL
1080 INPUT "Enter load
     current (IL) in mA: ", IL
1090 VL=IL*.001*RL
1100 IL=IL*.001
1110 INPUT "Does TTL drive this
     circuit (y/n) ? ", YESNO$
1120 YESNO$=LEFT$(YESNO$,1)
1130 IF YESNO$="Y"
     OR YESNO$="y"
     THEN VIN = 2.4: GOTO 1150
1140 INPUT "Enter driving
     voltage (VIN): ", VIN
1150 IB=IL/HFE
1160 R1=VIN/IB
1170 RIN=R1:GOSUB 10000:R1=ROUT
1180 R2=(VCC-VL)/IL
1190 RIN=R2:GOSUB 10000:R2=ROUT
1200 PRINT:PRINT "R1 =";R1;
     "ohms, R2 =";R2;
     "ohms, I(B) =";IB*1000;"ma"
1210 RETURN
10000 REM get closest standard
      resistor value
10010 REM input = rin
10020 REM output = rout
10030 REM sf = scale factor
10040 REM 25 standard values follow
10050 DATA 1.0, 1.1, 1.2, 1.3,
      1.5, 1.6, 1.8
10060 DATA 2.0, 2.2, 2.4, 2.7
10070 DATA 3.0, 3.3, 3.6, 3.9
10080 DATA 4.3, 4.7
10090 DATA 5.1, 5.6
10100 DATA 6.2, 6.8
10110 DATA 7.5, 8.2, 9.1, 10.0
10120 IF NOT FIRST THEN 10190
10130 FIRST = 0
10140 AMAX = 25 : REM must equal
      no of entries in table above
10150 DIM A (AMAX)
10160 FOR I = 1 TO AMAX
10170 READ A(I)
10180 NEXT I
10190 X = RIN
10200 SF = 0
10210 WHILE X > A(AMAX)
10220 X = X/10
10230 SF = SF + 1
10240 WEND
10250 I = 1
10260 WHILE X > A(I)
10270 I = I + 1
10280 WEND
10290 IF X-A(I-1) <= A(I)-X
      THEN X = A(I-1) ELSE X = A(I)
10300 ROUT = X * 10^SF
10310 RETURN
10320 END
```

subroutines for easy integration into your programs. Lines 1000-1210 get the data and calculate the resistor values; lines 10000-10320 select the closest standard resistor values. Lines 10-90 drive the main routine following line 1000. Line 30 sets a flag for the standard-value routine, so that it doesn't read the data contained in lines 10050-10110 in every time a set of calculations are made. Line 40 clears the screen of my CRT (an ADM 20); you'll probably have to use another method.

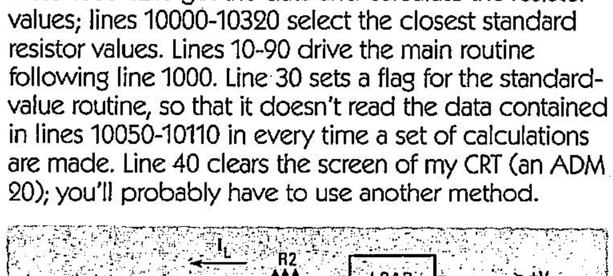

FIG. 1—R1 IN DIAGRAM ABOVE is used for current limiting and must be estimated. (See text.)

If your BASIC doesn't have the WHILE . . . WEND statements, you can simulate them by using an IF . . . THEN statement and a dummy loop. For example, the code below could be substituted for the loop running from line 10210 to line 10240: 10210/IF X < = A(MAX) then 10250

10220 X = X/10

10230 SF = SF + 1

10240 IF X > A(AMAX) THEN 10210

That's about all there is to calculating the values of the resistors in this simple switching circuit. Of course, you should verify that the calculated values actually do the job before building 10,000 duplicate circuits. Also, note that the program prints the value of I(B) the circuit will draw. That was done as a protective measure. If your driving circuit is CMOS, and I(B) is greater than a milliamp or so, you'll have to do one of two things: use a Darlington, or add another transistor that will provide a bit of "preamplification." If you opt for a Darlington, you can use our program as is. ◀CD▶