```
'--------Title--------

' File......robot_photo2.pbp
' Started....2/27/08
' Microcontroller used:  Microchip Technology PIC16F88-I/P
'                        microchip.com
' PicBasic Pro Code: micro-Engineering Labs, Inc.
'                    melabs.com

'--------Program Desciption--------

' The program drives two dc gearhead motors
' using an analog-to-digital converter,(AN4),
' to measure the voltage in a voltage divider
' with a CdS photocell,(an analog signal).
' It then converts the analog voltage into an 8-bit
' digital value (0 to 255) and displays it on an LCD.
' A flashlight is used to vary the brightness on
' the CdS photoresistor.

'------------Comments--------------

' The photoresistor,R4, is a Jameco #120299
' (Dark-300K, Light-8K)
' R3 was calculated to be 1K ohm.
' Refer to schematic robot_photo2.

'-----New PicBasic Pro Commands----

'---------PIC Connections---------

'       16F88 Pin              Wiring
'       ---------           ----------
'         RA0               LCD pin 11(DB4)
'         RA1               LCD pin 12(DB5)
'         RA2               LCD pin 13(DB6)
'         RA3               LCD pin 14(DB7)
'         RA4               Center Lead of Voltage Divider
'         RB0               To base of NPN controlling Motor 1
'         RB1               To base of NPN controlling Motor 2
'         RB3               LCD Enable(E)
'         RB4               LCD Register Select(RS)
'       See schematic for the other usual PIC connections

'---------LCD Connections,(Optional)---------

'        LCD Pin              Wiring
'       ---------           ----------
'          1               Ground(Vss)
'          2               + 5v(Vdd)
'          3               Center of 20K Pot(Contrast)
'          4               RB4(Register Select,RS)
'                          Note, RS is not connected to the
'                          default pin RA4.  See Constants/
'                          Defines below for explanation.
'          5               Ground(Read/Write,R/W)
'          6               RB3(Enable)
'          7               No Connection(DB0)
'          8               No Connection(DB1)
'          9               No Connection(DB2)
'         10               No Connection(DB3)
'         11               RA0(DB4)
'         12               RA1(DB5)
'         13               RA2(DB6)
'         14               RA3(DB7)

'---------Revision History--------

'---------Constants/Defines-------

' To free up AN4 (Pin RA4) for an analog input, the
' default LCD Register Select (RS) function must be
' removed from RA4. This is relocated to PORTB.4, (RB4),
' using the LCD DEFINE statements below.  All other
' default LCD pins and functions are left unchanged.
' See Curriculum Year 2, Lesson LCD3, POT Command and
' LCD DEFINES on this website for more details.

    DEFINE LCD_RSREG    PORTB   'PORTB - RS port
    DEFINE LCD_RSBIT    4       'Bit 4 - RS bit

'------------Variables------------

     p0              VAR BYTE       'Byte for voltage divider input
     temp_int        VAR WORD       'Word for integer
     temp_fract      VAR WORD       'Word for fraction

'----------Initialization---------

    ANSEL = %00010000  'Leaves AN4 in analog mode, but
                       'changes other analog bits to digital.
                       'See table below.

'    Analog Bit    Analog or Digital      PIC16F88 Pin
'   ------------  ------------------     --------------
'       AN0            Digital                RA0
'       AN1            Digital                RA1
'       AN2            Digital                RA2
'       AN3            Digital                RA3
'       AN4            Analog                 RA4
'       AN5            Digital                RB5
'       AN6            Digital                RB6

    OSCCON = $60                'Sets the internal oscillator in the
                                '16F88 to 4 MHz
    PORTB = %00000000           'Sets up pins RB0-RB7 of PORTB at LOW
    TRISB = %00000000           'Sets up all pins of PORTB as an outputs

'------------Main Code------------
```

```
    PAUSE 1000                 'Pause to allow LCD to setup

start:

    ADCIN 4, p0                'Read analog voltage on AN4 and
                               'convert to 10-bit digital value
                               'and store as p0.
    LCDOUT $FE,1,"Divider = ",#p0
                               'Clears LCD screen, displays
                               '"Divider = " and the 10-bit
                               'value of p0

' Choose to travel straight or turn:

    IF p0 >= 150 THEN        'If photoresistor value, p0, is
                             'greater than or equal to 500, the
    GOSUB straight           'program jumps to the subroutine straight.
                             'If p0 >= 500 is false, i.e. p0 < 500,
                             'the program advances to the ELSE
                             'program statement.

    ELSE

    GOSUB turn               'Program jumps to subroutine turn

    ENDIF

    GOTO start

    END

straight:

    HIGH 0 : HIGH 1          'Pins RB0 and RB1 are set at HIGH,
                             'turning on both NPN transistor switches,
                             'making the drive motors advance forward.
    PAUSE 100                'Pause 100 mS

    RETURN

turn:

    HIGH 0 : LOW 1           'Pin RB0 remains HIGH while pin RB1
                             'is set LOW, turning on only one NPN
                             'transistor switch, making the car turn.
    PAUSE 100                'Pause 100 mS

    RETURN
```