```
'--------Title--------
'
' File......robot1.pbp
' Started....3/14/06
' Microcontroller used:  Microchip Technology 16F84A
'                        microchip.com
' PBPro Code, micro-Engineering Labs, Inc.
'             melabs.com

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

' Program takes elements of previous programs and
' applies to a simple robotic application.
' When the switch in front of the robot is hit, the
' robot will backup and turn before moving forward again.
' Note, for the servos to act as motors, they
' must be modified or "hacked".  See the book
' Amphibionics by Karl Williams which gives an
' in depth treatment on how to modify servos.
' Even if you don't construct a robot, the components
' can be wired to a breadboard to test the electronics
' and programming.

'-----------Connections-----------

'       16F84A Pin          Wiring
'       ---------           ----------
'       RB0                 LED to signal right servos action
'       RB1                 LED to signal left servos action
'       RB2                 Control wire for right servo
'       RB3                 Control wire for left servo
'       RA0                 Momentary switch

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


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


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

    c0      VAR     BYTE        'Byte for counter
    switch  VAR     PORTA.0     'Pin A0 is assigned the name switch
                                'using the VAR command.

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

    TRISA = %11111111           'All A pins are setup as inputs.
                                'This can also be written TRISA = 255
    TRISB = %11110000           'Pins B0, B1, B2, and B3 are set as
outputs,
                                'all other PORTB pins are set as inputs

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

```
start:

    IF switch = 0 THEN      'If the momentary switch is pressed, program
                            'goes to backupturn label

    GOTO backupturn

    ELSE                    'If the momentary switch is not pressed, the
                            'program goes to the forward label

    GOTO forward

     'Robot moves in forward direction

forward:                    'Forward loop label

    FOR c0 = 1 TO 10        'Send forward signal 10 times

    HIGH 0                  'Turns on LED connected to B0

    HIGH 1                  'Turns on LED connected to B1

    PULSOUT 2,100           'Sends counter-clockwise pulse signal to
                            'hacked servo connected to B2

    PULSOUT 3,200           'Sends clockwise pulse signal to hacked
                            'servo connected to B3


    PAUSE 20                'Pulse Interval:
                            'Wait 20 msec before sending next pulse

    NEXT                    'Goes to next value of c0

    GOTO start              'Go to see if momentary switch is pressed

       'Backup and turn sequence

backupturn:                 'Backup and turn label

       'Backup sequence

    FOR c0 = 1 TO 200       'Send backup signal 200 times

    LOW 0                   'Turns off LED connected to B0

    LOW 1                   'Turns off LED connected to B1

    PULSOUT 2,200           'Sends clockwise pulse signal to
                            'hacked servo connected to B2

    PULSOUT 3,100           'Sends counter-clockwise pulse signal
                            'to hacked servo connected to B3
```

```
PAUSE 20                'Pulse Interval:
                        'Wait 20 msec before sending next pulse

NEXT                    'Goes to next value of c0

 'Turn sequence

FOR c0 = 1 TO 100       'Send turn signal 100 times

HIGH 0                  'Turns on LED connected to B0

LOW 1                   'Turns off LED connected to B1

PULSOUT 2,100           'Sends counter-clockwise pulse signal
                        'to hacked servo connected to B2

PULSOUT 3,100           'Sends counter-clockwise pulse signal
                        'to hacked servo connected to B3


PAUSE 20                'Pulse Interval:
                        'Wait 20 msec before sending next pulse

NEXT                    'Goes to next value of c0

GOTO start              'Go to see if momentary switch is pressed


ENDIF

END
```