

```

' {$STAMP BS2}
'*****
'* Program:  Bot.BS2                      Author:  Brian & Charlie          *
'* Date:      3/26/2010                  Revision: 14.0                    *
'*****

'-----[ Program Description ]-----
'This program is the control program for the Rover-Bot Tracked Vehicle.
'-----[ I/O Definitions ]-----
LfIrOut    PIN 0      'Left IR LED, Output
LfIrIn     PIN 1      'Left IR Sensor, Input
RtIrOut    PIN 2      'Right IR LED, Output
RtIrIn     PIN 3      'Right IR Sensor, Input
TxEnable   PIN 4      'Transmitter(27982)TR PIN
Tx         PIN 5      'Transmitter(27982)DATA pin
Ping       PIN 6      'Ping Sensor
ALeft      PIN 7      'Activate Left Anode
CRight     PIN 8      'Activate Right/ Activate Left Cathode
ARight     PIN 9      'Activate Right Anode
Lights     PIN 10     'Lights, Front
Speaker    PIN 11     'Buzzer, Output
Servo_AB   PIN 12     'Left + Right Servos
Servo_C    PIN 14     'Up/Down Servo
Servo_D    PIN 15     'Left/Right Claw Servo
'-----[ Constants ]-----
Zone       CON      4000      'Buzzer Frequency
T2400      CON      396      'Baud Rate 2400
T9600      CON      84       'Baud Rate 9600
T19K2      CON      32       'Baud Rate 12000
Trigger    CON      5        'trigger pulse = 10 uS
Scale      CON      $200     'raw x 2.00 = uS
LcdBaud    CON      T19K2    'baud rate for LCD
LcdBkSpc   CON      $08      'move cursor left
LcdRt      CON      $09      'move cursor right
LcdLF      CON      $0A      'move cursor down 1 line
LcdCls     CON      $0C      'clear LCD (use PAUSE 5 after)
LcdCR      CON      $0D      'move pos 0 of next line
LcdOff     CON      $15      'LCD off
LcdLine1   CON      $80      'move to line 1, column 0
LcdLine2   CON      $94      'move to line 2, column 0
inscale    CON      872      'scale factor for inches at 2000 feet
'-----[ Variables ]-----
PB1        VAR      Bit      'Enables Light Control
irDetectLeft VAR      Bit      'Temporary Storage for IR Left
irDetectRight VAR      Bit      'Temporary Storage for IR Right
distanceLeft VAR      Nib      'Right Distance, Value
distanceRight VAR      Nib      'Left Distance, Value
freqSelect  VAR      Nib      'Value of Frequency
Pos         VAR      Byte      'Position of Joystick Value
irFrequency VAR      Word      'Temporary Storage for Frequency
DispNote    VAR      Word      'Count for FREQOUT
Distance    VAR      Word      'Ping Distance Measurment
ServoCountA VAR      Word      'Servo Counter for Servo C
ServoCountB VAR      Word      'Servo Counter for Servo D
'-----[ Initialization for Servos and Ping ]-----
FREQOUT Speaker, 500, Zone      'Send a Test Note on Powerup
INPUT Servo_AB                  'Make Sure Servos are Not Moving.
DO : LOOP UNTIL Servo_AB        'Wait for ServoPAL to Power Up.
LOW Servo_AB                    'Set Pin, and Hold
PAUSE 100                      '100mS Reset, ServoPAL.
HIGH Servo_AB                   'Raise PIN
PAUSE 200                      'Wait for PIN, Active.
ServoCountA = 670               'Set Value to Servo C
ServoCountB = 700               'Set Value to Servo D
'-----[ Main Program ]-----
Main:
  SEROUT Servo_C, 84, ["C"]      'Send "C" to Hold Servo
  GOSUB IR                      'Read IR Leds, Store
  GOSUB DataTx                  'Tranmit Data
  GOSUB DataRx                  'Receive Data

```

```

GOSUB PingOut                                'Read Ping Value, Store
GOSUB Move                                    'Move Robot
IF PB1 = 1 THEN GOSUB Light                  'Activate Lights PB1 = 1
IF PB1 = 0 THEN GOSUB NoLight                'Deactivate Lights PB1 = 0
GOTO Main
' -----[Subroutines ]-----
DataTx:
HIGH TxEnable                                'Enable Transmitter
PULSOUT Tx,1200                              'Send Sync Pulse to Radio
SEROUT Tx, 16572, ["DataTx",DEC3 Distance, 'Sends Ping
DEC1 distanceLeft,DEC1 distanceRight]       'Sends Left + Right IR
PAUSE 10                                     'Wait
RETURN

DataRx:
LOW TxEnable                                'Enable Receiver
SERIN Tx, 16572, [WAIT("DataRx"),DEC1 Pos,DEC PB1] 'Receives Data
PAUSE 10                                     'Wait
RETURN

IR:
distanceLeft = 0                            'Set IR Zone 5
distanceRight = 0                           'Set IR Zone 5
FOR freqSelect = 0 TO 4                     'Select Frequency
LOOKUP freqSelect,[37500,38250,39500,40500,41500], irFrequency 'Table Lookup
FREQOUT LfIrOut,1,irFrequency               'IR Left, Transmit
irDetectLeft = IN1                          'Store Distance
FREQOUT RtIrOut,1,irFrequency               'IR Right, Transmit
irDetectRight = IN3                         'Store Distance
distanceLeft = distanceLeft + irDetectLeft  'Sum Values, Left
distanceRight = distanceRight + irDetectRight 'Sum Values, Right
NEXT
RETURN

PingOut:
PULSOUT Ping, 5                             'Set PIN 15 for 10uS
PULSIN Ping, 5, Distance                     'Wait for Pin to Low
Distance = Distance ** inscale               'Scale Inches Down
PAUSE 100                                    'Wait
RETURN

Move:
IF Pos = 0 THEN GOSUB Neutral                'Robot: Neutral
IF Pos = 1 THEN GOSUB Forward                'Robot: Forward
IF Pos = 2 THEN GOSUB Backward               'Robot: Backward
IF Pos = 3 THEN GOSUB Right                 'Robot: Right
IF Pos = 4 THEN GOSUB Left                  'Robot: Left
IF Pos = 5 THEN GOSUB Up                    'Robot: Claw Up
IF Pos = 6 THEN GOSUB Down                  'Robot: Claw Down
IF Pos = 7 THEN GOSUB Open                  'Robot: Claw Open
IF Pos = 8 THEN GOSUB Close                  'Robot: Claw Close
RETURN

Neutral:
PULSOUT Servo_AB, 2000                      'Turn Right Servo Off.
PULSOUT Servo_AB, 2000                      'Turn Left Servo Off.
RETURN

Forward:
DO
PULSOUT Servo_AB, 1000                      'Right Servo: Forward
PULSOUT Servo_AB, 500                      'Left Servo: Forward
GOSUB DataTx                               'Sends DATA
GOSUB DataRx                               'Receives DATA
LOOP WHILE Pos = 1                          'Loop Position = 1
RETURN

Backward:
DO
IF distanceRight = 0 THEN GOSUB Buzz         'IR Right Zone 0, Buzz

```

IF distanceRight = 1 THEN GOSUB Buzz	'IR Right Zone 1, Buzz
IF distanceLeft = 0 THEN GOSUB Buzz	'IR Left Zone 0, Buzz
IF distanceLeft = 1 THEN GOSUB Buzz	'IR Left Zone 1, Buzz
PULSOUT Servo_AB, 500	'Right Servo: Reverse
PULSOUT Servo_AB, 1000	'Left Servo: Reverse
GOSUB IR	'Read IR Leds, Store
GOSUB DataTx	'Sends DATA
GOSUB DataRx	'Receives DATA
LOOP WHILE Pos = 2	'Loop Position = 2
RETURN	
Right:	
DO	
PULSOUT Servo_AB, 500	'Right Servo: Reverse
PULSOUT Servo_AB, 500	'Left Servo: Forward
LOW ALeft	'Turn off Left Blinker
HIGH ARight	'Turn on Right Blinker
LOW CRight	'Common Ground
PAUSE 10	'Wait
LOW ALeft	'Turn off Left Blinker
LOW ARight	'Turn off Right Blinker
LOW CRight	'Common Ground
PAUSE 10	'Wait
GOSUB DataTx	'Sends DATA
GOSUB DataRx	'Receives DATA
LOOP WHILE Pos = 3	'Loop Position = 3
RETURN	
Left:	
DO	
PULSOUT Servo_AB, 1000	'Right Servo: Forward
PULSOUT Servo_AB, 1000	'Left Servo: Reverse
HIGH ALeft	'Turn on Left Blinker
LOW ARight	'Turn off Right Blinker
LOW CRight	'Common Ground
PAUSE 10	'Wait
LOW ALeft	'Turn off Left Blinker
LOW ARight	'Turn off Right Blinker
LOW CRight	'Common Ground
PAUSE 10	'Wait
GOSUB DataTx	'Sends DATA
GOSUB DataRx	'Receives DATA
LOOP WHILE Pos = 4	'Loop Position = 4
RETURN	
Up:	
DO	
SEROUT Servo_C, 84, ["0"]	'Add Count, MAX 4
GOSUB DataTx	'Sends DATA
GOSUB DataRx	'Receives DATA
LOOP WHILE Pos = 5	'Loop Position = 5
RETURN	
Down:	
DO	
SEROUT Servo_C, 84, ["5"]	'Subtract Count, MIN 0
GOSUB DataTx	'Sends DATA
GOSUB DataRx	'Receives DATA
LOOP WHILE Pos = 6	'Loop Position = 6
RETURN	
Open:	
DO	
IF POS = 7 THEN ServoCountB = ServoCountB + 10 'Increments servo	
ServoCountB = ServoCountB MAX 830	'Sets MAX 830
PULSOUT Servo_D , ServoCountB	'Moves Servo
PAUSE 5	'Pause
GOSUB DataTx	'Sends DATA
GOSUB DataRx	'Receives DATA
LOOP WHILE Pos = 7	'Loop Position = 7

RETURN

Close:

```
DO
  IF POS = 8 THEN ServoCountB = ServoCountB - 10 'De-Increments servo
  ServoCountB = ServoCountB MIN 600 'Sets MIN 600
  PULSOUT Servo_D , ServoCountB 'Moves Servo
  PAUSE 5 'Pause
  GOSUB DataTx 'Sends DATA
  GOSUB DataRx 'Receives DATA
  LOOP WHILE Pos = 8 'Loop Position = 8
```

RETURN

Light:

```
  HIGH Lights 'Turn on Front Lights
RETURN
```

NoLight:

```
  LOW Lights 'Turn off Front Lights
RETURN
```

Buzz:

```
  FREQOUT Speaker, 500, Zone 'Send Alarm Signal
  PULSOUT Servo_AB, 2000 'Turn Right Servo Off.
  PULSOUT Servo_AB, 2000 'Turn Left Servo Off.
GOTO Main 'Return to Top
```