RQ-5 Paper Robot
Supplementary Info

Document Rev 1.1

June 1, 2011

Passion for innovation

Trademark

Innovati®,  , and BASIC Commander® are registered trademarks of Innovati, Inc.

InnoBASIC™, cmdBUS™ , EV8™ and EV-Writer™ are trademarks of Innovati, Inc.

Paper Planet and RQ-5 are trademarks of IdeasRiver, Inc.

Due to continual product improvements, Innovati reserves the right to make modifications to its products without prior notice. Innovati does not recommend the use of its products for application that may present a risk to human life due to malfunction or otherwise.

Disclaimer

Full responsibility for any applications using Innovati products rests firmly with the user and as such Innovati will not be held responsible for any damages that may occur when using Innovati products. This includes damage to equipment or property, personal damage to life or health, damage caused by loss of profits, goodwill or otherwise. Innovati products should not be used for any life saving applications as Innovati's products are designed for experimental or prototyping purposes only. Innovati is not responsible for any safety, communication or other related regulations. It is advised that children under the age of 14 should only conduct experiments under parental or adult supervision.

Errata

We hope that our users will find this user's guide a useful, easy to use and interesting publication, as our efforts to do this have been considerable. Additionally, a substantial amount of effort has been put into this user's guide to ensure accuracy and complete and error free content, however it is almost inevitable that certain errors may have remained undetected. As Innovati will continue to improve the accuracy of its user's guide, any detected errors will be published on its website. If you find any errors in the user's guide, please contact us via email service@innovati.com.tw. For the most up-to-date information, please visit our web site at http://www.innovati.com.tw.

Table of Contents

RQ-5 Paper Robot Supplementary Info

Introduction

Build your own Paper Robot! Easy to build, no glue, no scissors, simple plug-and-play electronics. The RQ-5 paper robot is an excellent kit to learn the basics of robotics. In addition, the RQ-5 paper robot is also expandable for further robotics learning and exploration. In this document, you can find the circuit diagram, program code and example codes to get you started.

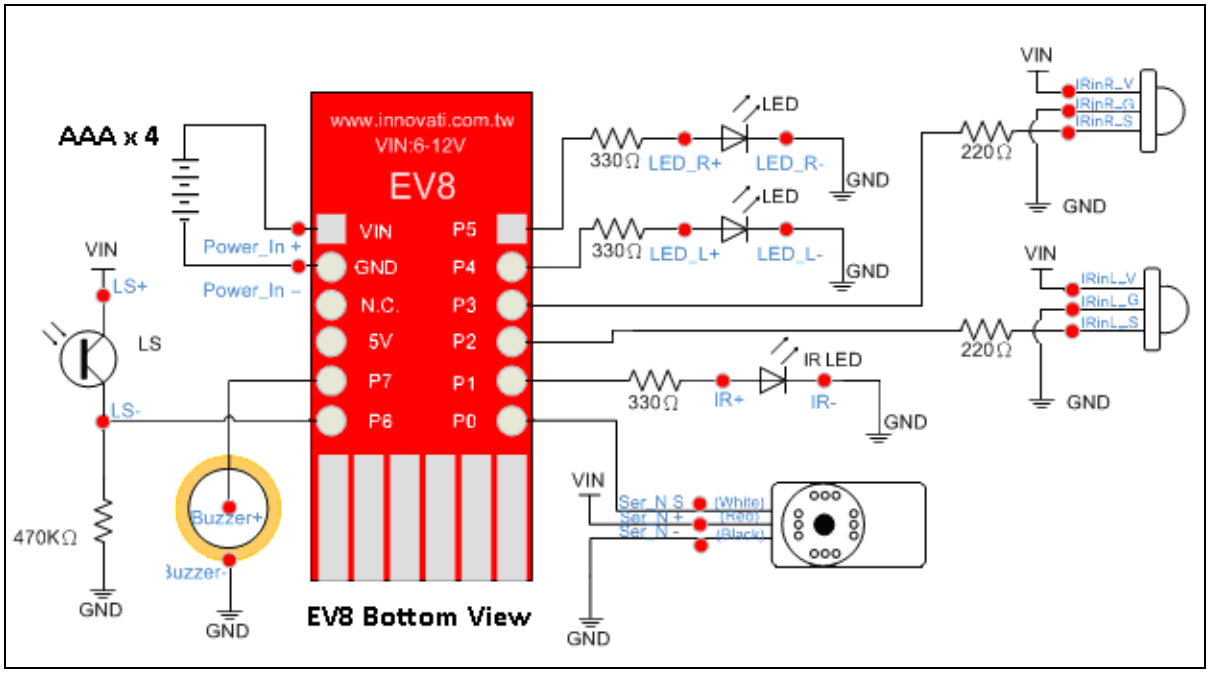
Functions

The RQ-5 Paper Robot is pre-downloaded with the basic functions. After power on, the LEDs of both eyes will be turned on and then the servo will be set to its neutral position, which makes the robot facing front for about 5 seconds. If your robot doesn't face front after power on due to assembly or twisted by external force, take this period of time to twist gently to slide the head to its right position. After 5 seconds, the IR obstacle detection function will be activated. If obstacle in front is detected, the robot will turn its head until the obstacle is removed. Meanwhile, if no obstacle is detected, the robot will detect the ambient light is dark or cover the light sensor on the top by the hand, the robot will play a short melody with eyes blinking.

Anyway, you can try to put in your own featured functions into RQ-5. Simply purchase an EV-Writer, install the development environment, write your program and download via the EV-Writer to the EV8 control board inside the robot, then your robot has new life and is at your command now! Just kick off and explore the various possibilities.

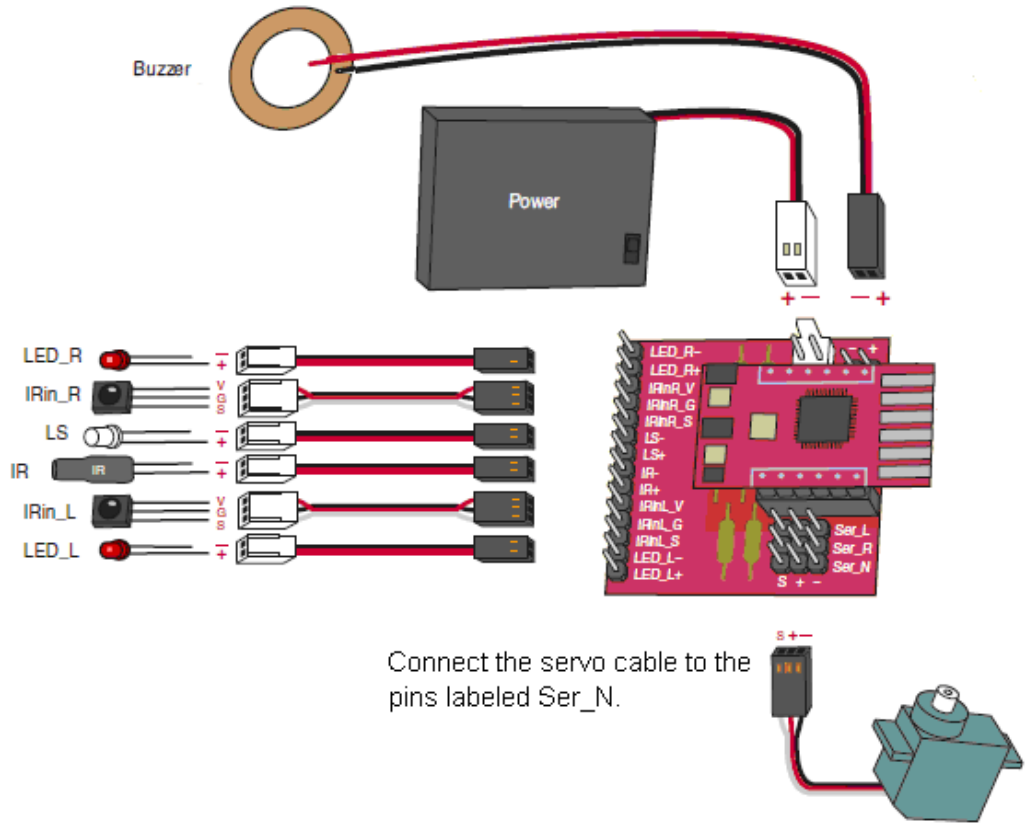
Circuit Diagram

In this paragraph, you can see the circuit that we used inside of the RQ-5 robot. First of all, you can see the EV8 module, which is the brain of the robot and all the intelligence comes from. In this diagram, except the EV8 module and devices that attach onto the robot body, shown with red dots connections, all the resistors and relating power connections are placed on the load board (LB), see connection diagram below.



Connection Diagram

In this paragraph, you can see the connection diagram, which shows you how to connect the circuit intuitively. Please follow the instruction carefully and mind the polarity shown in the diagram. Mind the sharp edges of the components pins during processing. You may trim the extra part of the pins after inserting into the connectors.



Program Code

Here shows the source code of the RQ-5 paper robot, which has been pre-downloaded into the EV8 controller module before shipping. This program is provided as learning materials for those who would like to learn the basics of robotics.

```
'================================================================================
' RQ-5 Demo Program Code Version 1.1
' Code Provided by James Luo, Innovati, inc.
' Date: June 1, 2011
' Function:
'    1. Move head and LEDs flash when obstacle in front is detected
'    2. Play music when light sensor is covered (dark)
'================================================================================

Const tempo_SPD  As Byte = 50    'tempo
Const NOTE_NUM   As Byte = 10    '11 notes (size of array), adjust size for more notes
Const Servo      As Byte = 0     'Servo I/O pin number
Const IR         As Byte = 1     'IR LED I/O pin number
Const IRin_L     As Byte = 2     'Left IR sensor I/O pin number
Const IRin_R     As Byte = 3     'Right IR sensor I/O pin number
Const LED_L      As Byte = 4     'Left-eye LED I/O pin number
Const LED_R      As Byte = 5     'Right-eye LED I/O pin number
Const LS         As Byte = 6     'Light Sensor (LS) I/O pin number
Const Buzzer     As Byte = 7     'Buzzer I/O pin number

'array of notes, increase the size of array for more notes
Const Rhythm(NOTE_NUM) As Byte = {10,0,10,0,10,0,8,0,10,0,12}

'array of note length, increase the size of array for more notes
Const Duration(NOTE_NUM) As Byte = {2,1,2,4,2,4,2,4,2,1,5}

Sub Main()
    Dim IR_L,IR_R,IRSelect,i As Byte

'--------------------------------------------------------------------------------
' Turn on eyes
'--------------------------------------------------------------------------------

    High LED_L                          'turn on left-eye LED
    High LED_R                          'turn on right-eye LED

'--------------------------------------------------------------------------------
' Standstill 5 seconds for servo position adjustment
'--------------------------------------------------------------------------------

    For i = 0 To 250                'keep sending pulses for about 5 seconds
        Pulseout Servo, 300         'move the servo to neutral position
        Pause 18                    'form an approximate 20 ms cycle
    Next i

'--------------------------------------------------------------------------------
```

```
' main Loop
'--------------------------------------------------------------------------

    Do
        Sound(IR, 5, 38400)                 'sound command for 38.4kHz IR carrier
        IR_R = In(IRin_R)                   'get right side IR sensor status
        IR_L = In(IRin_L)                   'get left side IR sensor status

        If IR_R = 0 And IR_L = 0 Then
            High LED_L                      'turn on left-eye LED
            High LED_R                      'turn on right-eye LED
            Move_Head()                     'call Move_Head routine
        Elseif IR_R = 1 And IR_L = 1 Then
            High LED_L                      'turn on left-eye LED
            High LED_R                      'turn on right-eye LED
            If In(LS) = 0 Then              'check if ambient light is dark
                Play_Notes()                'call buzzer Play_Notes routine
                High LED_L                  'turn on left-eye LED
                High LED_R                  'turn on right-eye LED
            End If
        End If
    Loop
End Sub

'--------------------------------------------------------------------------
' Routine to move the head
'--------------------------------------------------------------------------

Sub Move_Head()
    Dim i As Word               'variable for counter index
    Dim Pos As Word             'variable for servo position

    For i = 0 To 20             'turn head from neutral to right in 21 steps
        Pos = 300+(i*4)         'position from 1.5ms to 1.9ms step 0.02ms
        Pulseout Servo, Pos     'generate pulse to move the servo
        Pause 18                'form an approximate 20 ms cycle
    Next i

    For i = 0 To 40             'turn head from right to left in 41 steps
        Pos = 380-(i*4)         'position from 1.9ms to 1.1ms step 0.02ms
        Pulseout Servo, Pos     'generate pulse to move the servo
        Pause 18                'form an approximate 20 ms cycle
    Next i

    For i = 0 To 20             'turn head from left to neutral in 21 steps
        Pos = 220+(i*4)         'position from 1.1ms to 1.5ms step 0.02ms
        Pulseout Servo, Pos     'generate pulse to move the servo
        Pause 18                'form an approximate 20 ms cycle
    Next i
End Sub

'--------------------------------------------------------------------------
' Routine to use the buzzer to play the notes
```

```
' The Select Case table contains frequency of notes for 2 octaves.
' Add more notes in the table, if needed.
'-------------------------------------------------------------------------

Sub Play_Notes()
    Dim i As Byte
    Dim tempo As Word

    For i=0 To NOTE_NUM             'number of notes to play
        tempo = Duration(i)         'load the note length
        Toggle LED_L                'toggle the left-eye LED on/off status
        Toggle LED_R                'toggle the right-eye LED on/off status

        Select Case Rhythm(i)       'play notes through Select Case table
            Case 0
                Pause tempo_SPD*tempo                   'rest
            Case 1
                Sound(Buzzer, tempo_SPD*tempo, 523)     'Do (C5)
            Case 2
                Sound(Buzzer, tempo_SPD*tempo, 587)     'Re (D5)
            Case 3
                Sound(Buzzer, tempo_SPD*tempo, 659)     'Mi (E5)
            Case 4
                Sound(Buzzer, tempo_SPD*tempo, 698)     'Fa (F5)
            Case 5
                Sound(Buzzer, tempo_SPD*tempo, 784)     'Sol (G5)
            Case 6
                Sound(Buzzer, tempo_SPD*tempo, 880)     'La (A5)
            Case 7
                Sound(Buzzer, tempo_SPD*tempo, 987)     'Ti (B5)
            Case 8
                Sound(Buzzer, tempo_SPD*tempo, 1047)    'Do (C6)
            Case 9
                Sound(Buzzer, tempo_SPD*tempo, 1175)    'Re (D6)
            Case 10
                Sound(Buzzer, tempo_SPD*tempo, 1319)    'Mi (E6)
            Case 11
                Sound(Buzzer, tempo_SPD*tempo, 1397)    'Fa (F6)
            Case 12
                Sound(Buzzer, tempo_SPD*tempo, 1568)    'Sol (G6)
            Case 13
                Sound(Buzzer, tempo_SPD*tempo, 1760)    'La (A6)
            Case 14
                Sound(Buzzer, tempo_SPD*tempo, 1976)    'Ti (B6)
            Case 15
                Sound(Buzzer, tempo_SPD*tempo, 2093)    'Do (C7)
        End Select
    Next i
End Sub
```

Write Your Own Program

We encourage you to try to write your own code, make your own featured robots. Just follow the step shown below and you'll start to explore the world of robotics. We are more than happy to receive your source code and act as a platform to share your awesome programs with other robot hobbyists.

Step 1:
Install the software and download the zip file to check the zip file tutorial. The tutorial programs show the basics of programming language and how to access the sensors and react through light, sound and actuator.

Step 2:
Refer to the "BASIC Commander and innoBASIC Workshop User's Manual", which can be found in our website at www.innovati.com.tw.

Step 3:
After reading the User's Manual, install the "innoBASIC Workshop 2" integrated development environment, which can be found in our website.

Step 4:
Start to write your own program code and download via EV-Writer. Please note that you need to purchase an EV-Writer, which help download your finished program code into the EV8 module. The "EV8 and EV-Writer User's Guide" can be found in our website.

More questions to visit our forum. Your paper robot is now at your command. If any question in programming your own robot, visit our forum for answers.

FAQs

Q1: There is no obstacle in front of the RQ-5 robot, but why the RQ-5 keeps on moving its head after power on?
A1: Please check the IR emitter (IR LED) black tube. If the IR LED is not inserted into the tube completely. The IR light leakage might be detected by the IR receiver just next to it, which would be interpreted as an obstacle in front.

Q2: The RQ-5 robot is not facing in front, I re-open the head and screw again, but just cannot get it right. Can I just twist the head?
A2: Yes, you may twist the head gentle after power on. The paper part is clapped by two pieces of servo horns. Gentle twist on the head is OK for the robot .

Q3: When I fasten the screw, the servo (with the body) moves, but the head is not fastened yet, what can I do? And how can I align the head to the front?

A3: You can hold the robot and twist the screw tighter until the paper head is pinched. After power on, the robot keeps its head in facing front gesture for 5 second. If the head is not facing the front, twist the head gently to slide the paper between the servo horns to the right position.

Q4: Can I add servos to the arms?

A4: Yes, you may purchase extra 2 servos for the arms. Due to the limited I/Os, the 2 servos share the I/Os with the eyes LEDs.