6 Channel 2.4GHz XBee Remote Control

6 Channel 2.4GHz XBee Remote Control is designed to be used with any custom designed robots such as robot used in robowars, quadcopter, surveillance robot and industrial applications. It contains user selectable XBee or XBee Pro wireless module with line of site communication range of 90m or 1200m.

Remote control has two 2 axis precision analog joysticks with zero adjustment and two user programmable switches. 4 more switches can be added on request. Remote also has built-in 2axis analog accelerometer. It can be used to add innovative control methods in your robot such as tilt based robot control etc.

Remote control is based on ATMEGA8 microcontroller. Microcontroller's ISP cable is made available outside so that user can load his custom code on the remote control.

Remote requires minimum of 6V supply. It has battery bay for 8 AA size pensile cells. You can use any ordinary cells or NiMH batteries. It is possible to open remote and add connector for custom made battery packs.

Features:

- Operating Voltage : 7V to 12V
- Microcontroller: ATMEGA8
- Two 2 axis precision Joysticks with horizontal and vertical movement and zero calibration
- Two switches (upgradeable to 6)
- Accelerometer MMA7361L : 3 axis with high sensitivity of 800mV/g at 1.5g (Only X and Y axis are connected to the microcontroller's ADC.
- User selectable XBee or XBee Pro wireless module with line of site communication range of 90m or 1200m.
- In-System Programming port available for customized code loading
- Battery low indicator
- Buzzer for user application
- Data transaction LED indicator

6 channel 2.4GHz XBee Remote Control configuration

Figure 1: Remote Control Front side peripherals

Figure 2: Battery bay and ISP cable

Remote control basic operation

All the systems on the remote control operates on 3.3V. ATMEGA8 is clocked on 7.3728MHz by using external crystal oscillator. Two 2 axis precision analog joysticks, X and Y axis of the accelerometer are connected to the ADC pins of the ATMEGA8. Two switches are connected to the GPIO pins of the microcontroller. You can add 4 more switches to add custom functions. 10 pin FRC wire compatible to ATMEL's standard 10 pin ISP connector is taken out for In System Programming (ISP). You can add XBee or XBee Pro wireless module with line of site communication range of 90m or 1200m as per your requirement. In default condition communication happens at 115200bps. The battery level is sensed by the resister divider circuit and the buzzer will beeps when battery is discharged below threshold level set at 8V.

All the codes for the remote are written in AVR Studio IDE.

ATMEGA8 microcontroller pin Connections

PIN NO	Pin name	USED FOR
3	GND	Ground
4	VCC	5V
5	GND	Ground
6	VCC	5V
7	XTAL1/TOSC1/PB6	External Crystal 7.3728MHz
8	XTAL2/TOSC1/PB7	
12	(ICP) PB0	Right Switch
13	(OC1A) PB1	Left Switch
14	(SS/OC1B) PB2	ISP (In System Programming)
15	(MOSI/OC2) PB3	
16	(MISO) PB4	
17	PB5 (SCK)	
18	AVCC	5V
19	ADC6	ADC input for Battery Voltage
20	AREF	5V
21	GND	Ground
22	ADC7	Reserved for future use
23	PC0 (ADC0)	ADC input for right joystick analog horizontal position
24	PC1 (ADC1)	ADC input for right joystick analog vertical position
25	PC2 (ADC2)	ADC input for left joystick analog vertical position
26	PC3 (ADC3)	ADC input for left joystick analog horizontal position
27	PC4 (ADC4/SDA)	ADC input for Accelerometer Y axis analog position
28	PC5 (ADC5/SCL)	ADC input for Accelerometer X axis analog position
29	PC6 (RESET)	Power on Reset circuit
30	PD0 (RXD)	UART receive for XBee wireless module
31	PD1 (TXD)	UART transmit for XBee wireless module
32	PD2 (INT0)	Buzzer

Table 1: ATMEGA8 microcontroller pin connections

Remote Control Firmware:

Remote control firmware is located in the “Accessories / Remote control / firmware” folder in the Accessories folder. Same firmware is factory loaded in the remote control.

In the application example remote control is acts as slave and the device to be controlled acts as a master. Whenever the master want the remote control’s input data, it sends string of 4 characters as "NEXR" as command packet, at 115200 bps to the remote. Upon receiving the string, remote control sends back the string containing 2 header bytes followed by 8 byte data containing instantaneous positions of the Two 2 axis precision analog joysticks, X and y axis of the accelerometer, Remote’s battery voltage and logic state of the switches. All the communication takes place at 115200 bps.

Receiving packet:
N - 1st byte of packet
E - 2nd byte of packet
X - 3rd byte of packet
R - 4th byte of packet

Transmitting packet:
F - 1st byte of packet (1st header byte)
B - 2nd byte of packet (2nd header byte)
Byte1 - 3rd byte of packet (Vertical position of left joystick)
Byte2 - 4th byte of packet (horizontal position of left joystick)
Byte3 - 5th byte of packet (Vertical position of right joystick)
Byte4 - 6th byte of packet (horizontal position of right joystick)
Byte5 - 7th byte of packet (Accelerometer x axis position)
Byte6 - 8th byte of packet (Accelerometer y axis position)
Byte7 - 9th byte of packet (Battery Voltage)
Byte8 - 10th byte of packet(Right and left switch positions in bitwise format in 10 byte as bit0 shows right switch and bit1 shows left switch position)

In the example code, command packet is received in the UART ISR function **“SIGNAL (SIG_UART_RECV)”.** If valid packet is received, the analog to digital conversion is done for all sensors by calling the function **“ADC_conversion ()”** and the result is stored in respective variables. Data packet is send in the **“SIGNAL (SIG_UART_RECV)”**

After successfully transaction of 10 such packets between remote control and host, remote’s buzzer beeps twice indicating successful establishment of the communication between host and the remote.

Battery Voltage Calculation:

The Battery voltage is derived from the voltage divider circuit formed with 2.2K and 4.4K resistor and connected to ADC6 of the ATMEGA8 Microcontroller.

Analog value is calculated using the following formula

A = (Digital value * K) + 0.7V

Where
A = Equivalent value to Battery Voltage
K = (3.3/255) * ((2.2+4.7)/2.2) = 0.042
0.7V = Voltage drop due to diode connected in series with the battery voltage applied.

UART is configured at baud rate is 115200bps.

Note:
1. Make sure that in the configuration options following settings are done for proper operation of the code while writing the program using the AVR Studio IDE.
    Microcontroller: atmega8
    Frequency: 7372800Hz
    Optimization: -O0
    (For more information read section: Selecting proper optimization options below figure 2.22 in the software manual)

Package contains:

1. 6 Channel 2.4GHz XBee Remote Control : Qty. 1
2. Documentation CD