

What is SensorMUX

SensorMUX is a multiplexer to connect 4 NXT sensors to single NXT port. SensorMUX is powered by external battery. This multiplexer can connect any sensor designed for Mindstorms NXT (LEGO and/or third party).



Connections

Connect SensorMUX to Sensor Port of your NXT.
Connect your sensors to SensorMUX ports.

Supplying power to your SensorMUX

The SensorMUX has Green terminals to connect external battery. SensorMUX needs external battery voltage between 7 and 9.6 volts to operate.

We recommend using LiPO battery (which has the best power to weight and power to price ratio for this application). However, several other batteries will work, few battery options are listed below.



NOTE

SensorMUX is rated for external power supply of 9.6 volts DC max. Ensure to not exceed this value. While connecting external battery, ensure correct polarity.

If the voltage of your battery drops below operating level, the LED on the selected channel of SensorMUX will blink continuously.

Battery Options:

Lithium-Polymer Battery (also known as Li-Po or LiPo batteries)

With SensorMUX, use 7.4 Volts battery. Select the capacity based on your load and anticipated duration of use. (For nominal load, we recommend 7.4Volts, 800 mAh battery.)



You can purchase these batteries in electronic hobby stores, or on-line RC toys stores. An example url is here:

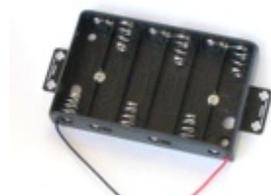
To connect these batteries to your SensorMUX, you can purchase connector pins from [mindsensors.com](http://www.mindsensors.com) website.



http://www.mindsensors.com/index.php?module=pagemaster&PAGE_user_op=view_page&PAGE_id=167

6AA batteries (rechargeable or non-rechargeable).

To hold and mount these batteries, you can purchase a holder with NXT mounts as shown here from [Mindsensors.com](http://www.mindsensors.com) website.



http://www.mindsensors.com/index.php?module=pagemaster&PAGE_user_op=view_page&PAGE_id=146

RC rechargeable battery
- 7.2Volts or 9.6Volts Ni-CD or Ni-MH RC rechargeable battery.
You can buy such battery and it's charger in local toy store.



To connect these batteries to SensorMUX, you can purchase following connector cable from [mindsensors.com](http://www.mindsensors.com) website:



http://www.mindsensors.com/index.php?module=pagemaster&PAGE_user_op=view_page&PAGE_id=145

Programming Techniques for SensorMUX

NXT-G:

You can download the SensorMUX block from following location:



http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=230

Install this block in your NXT-G using following instructions:

http://www.mindsensors.com/index.php?module=pagemaster&PAGE_user_op=view_page&PAGE_id=81

Download sample program from following location, and modify to suit your needs.

http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=233

NXC:

Download the sample programs and library file available at following location, and include the library file it in your program by #include directive. Or Modify the Sample program to suit your needs.

http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=232

RobotC:

To use this SensorMUX with RobotC, you need RobotC firmware 3.59 or higher.

I2C Registers:

When Channel 0 is selected SensorMUX appears as a set of registers as follows:

Register	Read	Write
0x41	-	Command Register
0x42	Channel selected (or to be selected)	-
0x43	Battery voltage	-

I2C Bus address

Factory Default Address: 0x24.

Current Consumption

Average measured current profile is as follows:

Current Consumption	Duration
3mA	Continuous

Advanced Information

Channels Selectors:

SensorMUX has 4 real channels (and 5th virtual channel).

The channels are selected as follows:

Character	Action
'0'	Select virtual

	channel 0
'1'	Select Channel 1
'2'	Select Channel 2
'3'	Select Channel 3
'4'	Select Channel 4

How to Select Channel in your program:

To select a channel following character sequence is sent to the NXT sensor port where SensorMUX is connected->

0x55 0xAA 'c'

where 'c' is channel as defined in 'Channel Selectors' above.

To send each bit of this sequence call SetSensorType() for each bit of this sequence as follows; followed by 1 millisecond wait.

0 => SENSOR_TYPE_LIGHT_ACTIVE

1 => SENSOR_TYPE_LIGHT_INACTIVE