

## What is PFMate

PFMate is a PF (Power Functions) Motor controller from NXT. It is designed to be a compact fit on PF IR (Infra Red) Receiver, and provide reliable IR communication to control PF motors.

## Connections

Can be connected to any of the four sensor ports of NXT by using standard Cables from NXT set.

You can attach PFMate on top the IR receiver with standard '1 x 2 Bricks' as spacers (Red bricks shown in adjacent picture).



For spacers between IR Receiver and PFMate on each side,

You can use 1 x 2 Bricks as shown:  or

Two of 1 x 2 plates as shown: . You can place a small piece of scotch-tape on the brick to get a tight fit.

## Programming Techniques for reading in I2C mode

### NXT-G:

Download the NXT-G block available in the NXT-G Blocks Repository at Mindsensor's website, at following location:

[http://www.mindsensors.com/index.php?module=documents&JAS\\_DocumentManager\\_op=viewDocument&JAS\\_Document\\_id=95](http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=95)

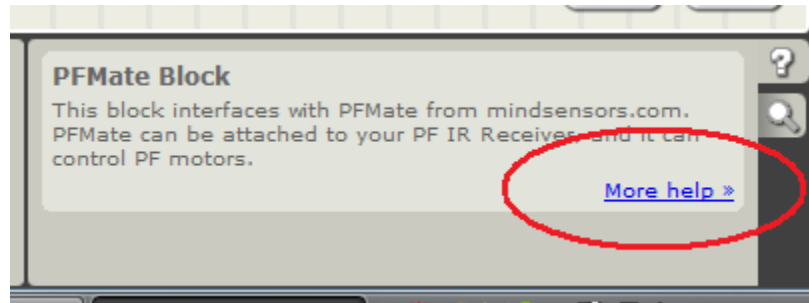


Also download sample program from following location, and modify to suit your needs.

[http://www.mindsensors.com/index.php?module=documents&JAS\\_DocumentManager\\_op=viewDocument&JAS\\_Document\\_id=96](http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=96)

Further help is available in block.

When you use the block in your program, select any PFMate block and in the bottom right corner, you will see url for help.



#### RobotC:

Download library file and sample program from following location, and modify the sample program to suit your needs. Or include the library file in your own program by include directive as:

```
#include "PFMate-lib.c"
```

[http://www.mindsensors.com/index.php?module=documents&JAS\\_DocumentManager\\_op=viewDocument&JAS\\_Document\\_id=97](http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=97)

Library API for programming:

```
void PFMate_controlBothMotors(tSensors port,  
                             byte addr, byte channel,  
                             byte operationA, byte speedA,  
                             byte operationB, byte speedB)
```

This function controls both motors at the same time. It can run, brake or float the motor. The run operation can be forward or reverse with speed ranging from 1 to 7

#### Parameters:

**port:** the sensor port on NXT to which PFMate is attached.

**addr:** I2C address of PFMate.

**channel:** The channel number to which your IR tower is set to.

**operationA:** float/forward/reverse or brake

**speedA:** value between 1-7 (this value is used when operation is forward/reverse).

**operationB:** float/forward/reverse or brake

**speedB:** value between 1-7 (this value is used by forward/reverse operations).

```
void PFMate_controlMotorA(tSensors port,  
                           byte addr, byte channel, byte operation, byte speed)
```

This function controls motor A only (the motor B state is unchanged).

**Parameters:**

**port:** the sensor port on NXT to which PFMate is attached.

**addr:** I2C address of PFMate.

**channel:** The channel number to which your IR tower is set to.

**operation:** float/forward/reverse or brake

**speed:** value between 1-7 (this value is used by forward/reverse operations).

```
void PFMate_controlMotorB(tSensors port,  
                           byte addr, byte channel, byte operation, byte speed)
```

This function controls motor B only (the motor A state is unchanged).

**Parameters:**

**port:** the sensor port on NXT to which PFMate is attached.

**addr:** I2C address of PFMate.

**channel:** The channel number to which your IR tower is set to.

**operation:** float/forward/reverse or brake

**speed:** value between 1-7 (this value is used by forward/reverse operations).

**NXC:**

Download the sample programs and library file available at following location, and include the library file it in your program by #include directive.

[http://www.mindsensors.com/index.php?module=documents&JAS\\_DocumentManager\\_op=viewDocument&JAS\\_Document\\_id=98](http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=98)

The API provided by this library file is similar to the one explained above.



**NOTE:** While using with NXT-G or NXC, ensure to use firmware version 1.29.

You can download latest firmware here:

[http://www.mindsensors.com/index.php?module=pagemaster&PAGE\\_user\\_op=view\\_page&PAGE\\_id=137](http://www.mindsensors.com/index.php?module=pagemaster&PAGE_user_op=view_page&PAGE_id=137)



**Troubleshooting Tip:** If you notice that the PF IR receiver is missing some commands, slow down the communication by introducing 300ms or more delay before issuing command. (The PF IR Tower does tend to miss commands if it receives too rapidly. Commands sent by hand from IR remote are slow enough for it to not miss anything).

## I2C Operations

PFMate is operated by loading the registers with appropriate values (described below), and then issuing a Go command to send an IR signal to receiver.

### Supported Command(s)

Commands		Action
ASCII	Hex	
G	0x47	Send IR signal to PF IR receiver

## I2C Registers:

The PFMate appears as a set of registers as follows:

Register	Read	Write
0x00-0x07	Software version - <i>V1.00</i>	-
0x08-0x0f	Vendor Id - <i>mndsnsrs</i>	-
0x10-0x17	Device ID - PFMate	-
0x41	-	Command
0x42	Channel for IR Receiver	Channel for IR Receiver (valid values 1, 2, 3 or 4)
0x43	Motor(s) to operate on	Motor(s) to operate on - Both: 0 Motor A: 1 Motor B: 2
0x44	Motor A Operation	Motor A Operation - Float: 0

		Forward: 1 Reverse: 2 Brake: 3
0x45	Motor A Speed	Motor A Speed - (Valid values between 1 -7. This value is used if the motor operation is Forward or Reverse, and ignored for Float/Brake)
0x46	Motor B Operation	Motor B Operation - Float: 0 Forward: 1 Reverse: 2 Brake: 3
0x47	Motor B Speed	Motor B Speed - (Valid values between 1 -7. This value is used if the motor operation is Forward or Reverse, and ignored for Float/Brake)

## Current Consumption

Average measured current profile is as follows:

Current Consumption	Duration
15mA	Continuous

## I2C Bus address

**Factory Default Address: 0x48**

### Changing the I2C Bus Address:

The I2C bus address of PFMate can be changed. To set an address different from default address, send sequence of following commands on the command register:

0xA0, 0xAA, 0xA5, <new I2C address>

Note: Send these commands with no break/read operation in between. This new address is effective immediately. Please note down your address carefully for future reference.

You can download the address change and scan functions from our website at [www.mindsensors.com](http://www.mindsensors.com). These functions are written in RobotC.

Alternately, you can download NXT executable programs from following location: [http://www.mindsensors.com/index.php?module=documents&JAS\\_DocumentManager\\_op=viewDocument&JAS\\_Document\\_id=91](http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=91)