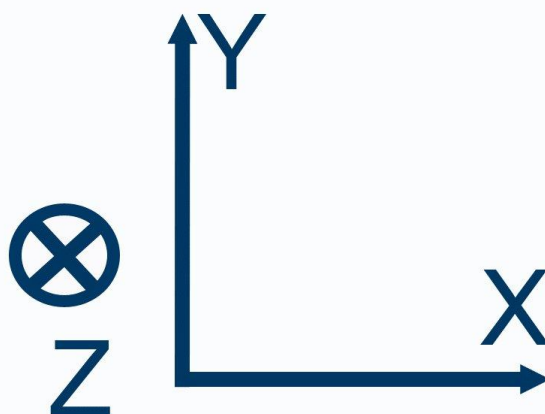
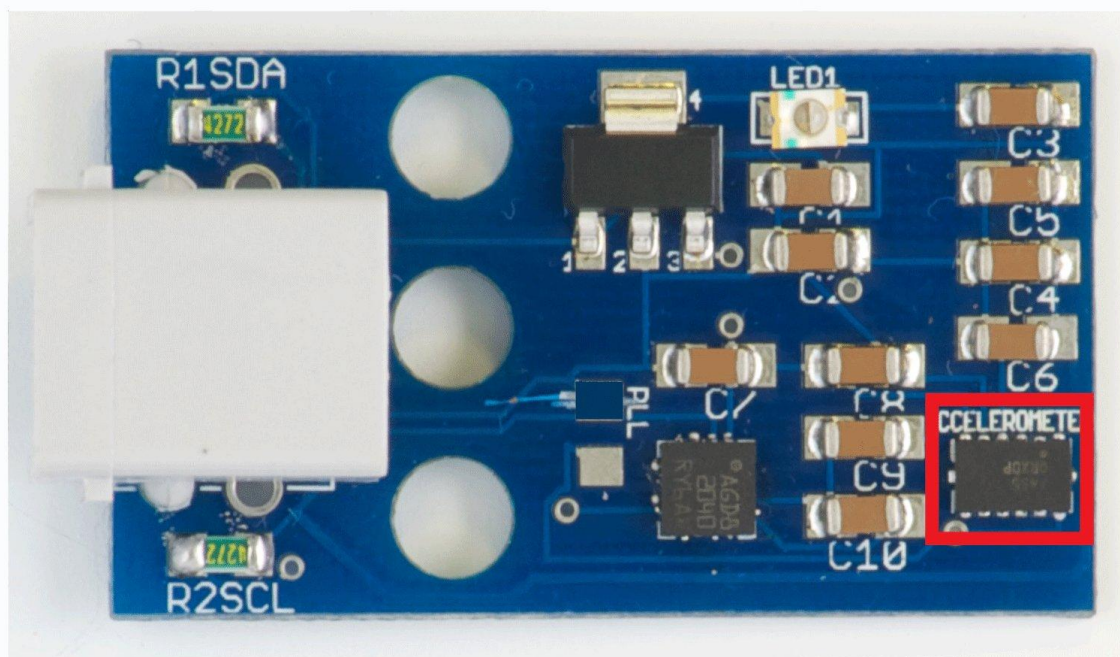


Руководства по использованию и примеры кода можно найти по ссылке:  
<http://www.dexterindustries.com/download.html#dLight>

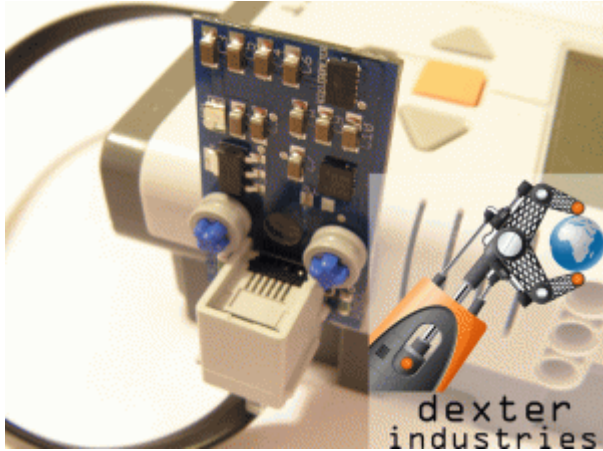
## dIMU Inertial Motion Sensor

# dIMU Accelerometer Axes



- An accelerometer and gyroscope for the LEGO® MINDSTORMS® NXT and EV3.
- EV3 Ready.
- Measures acceleration, tilt, and rotation on all 3 axes.
- Build LEGO® robots that can balance, measure acceleration, and know which way is up.
- Measure static and dynamic acceleration.
- **Accelerometer:** Measure acceleration up to 8 g's on all three axes. Measure between  $\pm 2g$ ,  $\pm 4g$ , and  $\pm 8g$ .
- **Gyroscope:** Measures rate of rotation. The sensor can measure  $\pm 250$ ,  $\pm 500$ , and  $\pm 2,000$  degrees per second.
- See what we've done with the dIMU [here](#).

## dIMU Sensor



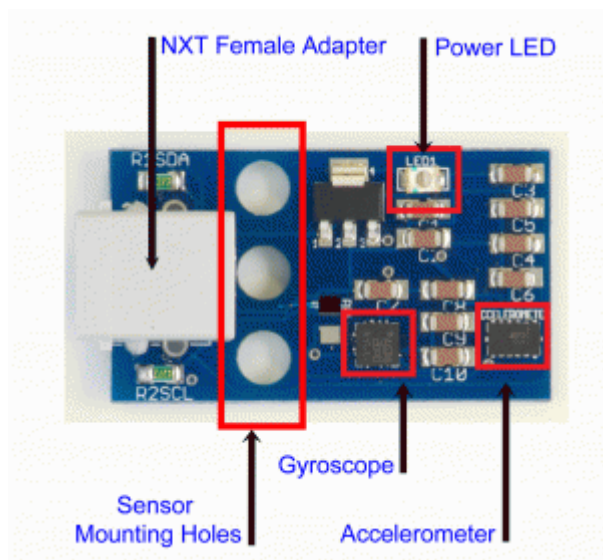
### Introduction

The Dexter Industries [Inertial Motion Unit \(IMU\) Sensor](#) measures acceleration, tilt, and rotation on all three axes. The sensor combines an accelerometer and a gyroscope into one sensor for the LEGO Mindstorms system. Build robots that know which way is up, can measure tilt, balance themselves, and measure acceleration and rotation on all axes all at once.

**Accelerometer:** Measure up to 8 g's on all three axes. Measure between  $\pm 2g$ ,  $\pm 4g$ , and  $\pm 8g$ .

**Gyroscope:** Measures rate of rotation. The sensor can measure  $\pm 250$ ,  $\pm 500$ , and  $\pm 2,000$  degrees per second.

### Setup



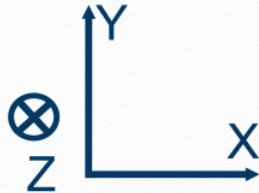
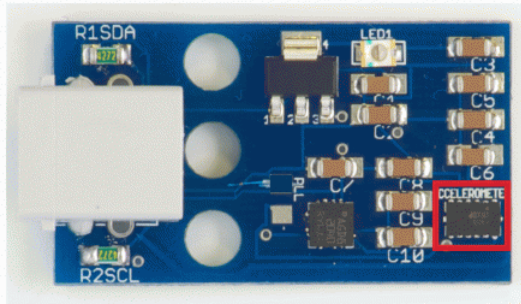
### Layout

- **Accelerometer:** measures acceleration on the x, y, and z axes.
- **Gyroscope:** measures rotation on the x, y, and z axes.
- **NXT Female Adapter:** connects to the LEGO Mindstorms NXT.
- **Power LED:** This LED will light when the power is turned on.
- **Sensor Mounting Holes:** Use these to mount the sensor to your robot.

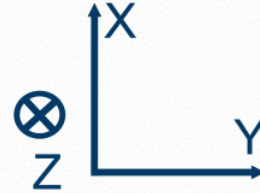
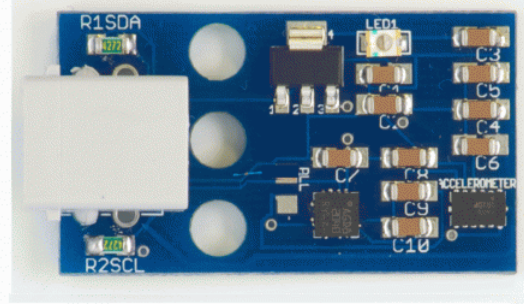
### Axes

Below are diagrams of the axes of the gyroscope sensor and the accelerometer sensor with directions of the angular axes.

## dIMU Accelerometer Axes



## dIMU Gyroscope Axes



### Hints

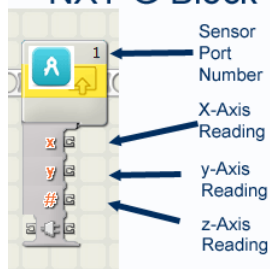
- **Prevent stress damage to the board:** be sure to use at least two of NXT holes to mount to LEGO contraptions.
- **Calibration:** The accelerometer is particularly sensitive and can be calibrated on a flat surface.
- **Mounting Holes:** LEGO crosses can be used to mount the [dIMU](#) to your robot.

## Software

### NXT-G

NXT-G Blocks are provided for the [dIMU](#). Separate blocks are provided for the gyroscope and for the accelerometer. For instructions on [how to install the NXT-G blocks into NXT-G](#), please see our video. The easiest way to get started with the [dIMU](#) is to use the Hello World program found in the NXT-G Block download in our [downloads section](#).

### dIMU Gyroscope NXT-G Block



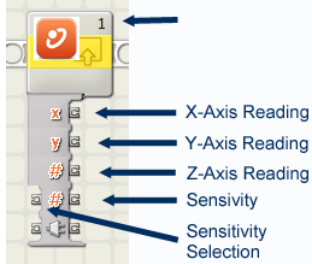
**Gyroscope Block** - The gyroscope sensor can be read with the Gyroscope Block.

The gyroscope block outputs degrees per second (dps) for the axis you select. The below program shows the Gyroscope reading the x-axis. All three axes can be read at once.

**Fast Gyroscope Block** – The Fast Gyroscope Block is similar to the Gyroscope block. There are two things to be aware of: at the beginning of your program, you *must* run a normal Gyroscope Block. This will set up the sensor for operation. Use the block to designate the degrees-per-second you would like to read. You must also specify which axis you want the robot to read. The more axes you check, the slower the sensor will read. To get the fastest response, you should only read one axis. Examples are provided in the Fast Gyroscope Block for NXT-G in our [downloads section](#).



## dimu Accelerometer NXT-G Block

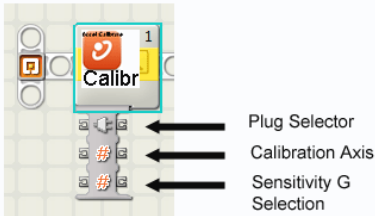


**Accelerometer Block** – The accelerometer sensor can be read with the Accelerometer Block.

The accelerometer block returns milli-G's (1000th of a G, a unit of gravity) for the axis you select. For example, if the sensor is placed perpendicular to the ground and not moving, it will read "1000" or 1 G. The below program shows the Accelerometer reading the x-axis. All three axes can be read at once.

The Sensitivity Selection input allows you to select the maximum number of G's to be read by the accelerometer. At times you may want to read very small changes in gravity. For this, the default maximum acceleration reading is 2 G. At other times, you may want to read very large changes in gravity. For this you can input "1" to read a maximum of 4 G or "2" to read a maximum of 8 G.

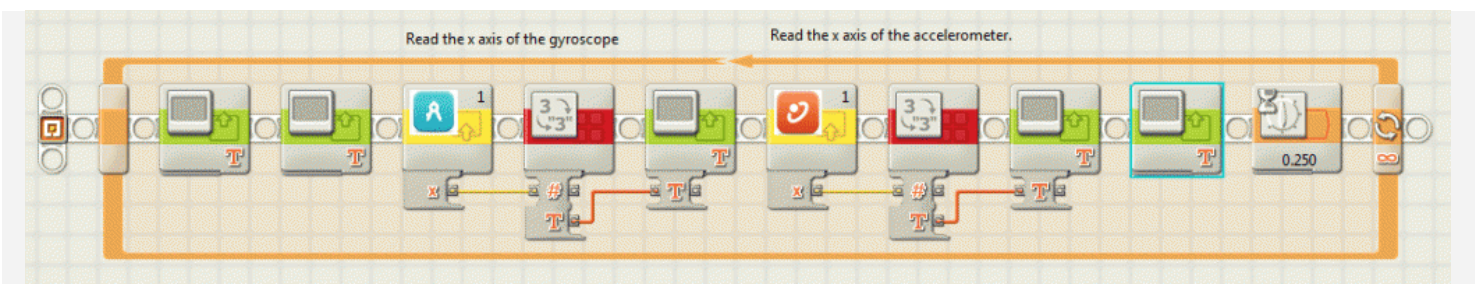
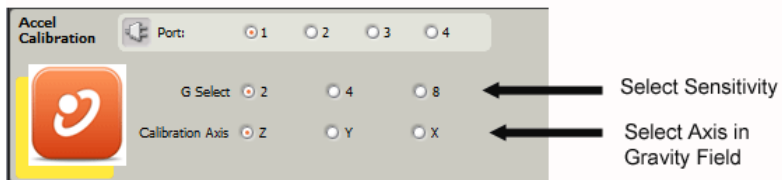
## dimu Accelerometer Calibration NXT-G Block



**Accelerometer Calibration Block** – The accelerometer sensor can be calibrated with the

Accelerometer Calibration Block. To calibrate the block, you must select the Gravity Sensitivity (G Select) and the axis of the accelerometer that gravity will be passing through. For example, if you lay the sensor flat on the ground, you must calibrate to the "Z" axis. If you lay the sensor on any other side, you must calibrate to that axis.

## dimu Accelerometer Calibration NXT-G Block



An example program of the dimu using NXT-G to read an accelerometer and gyroscope for the LEGO Mindstorms NXT.

## RobotC



[RobotC](#) code is provided in the [Downloads](#) section of our website. RobotC drivers have been directly incorporated into [the latest 3rd Party Drivers Suite](#), which can be found on the [BotBench website](#).

A basic example for both the gyroscope and the accelerometer are provided in our [Downloads](#) section, with detailed commenting to better your understanding of the sensor and its operation. The example provides an in-depth explanation of the I2C calls for communicating with the [dIMU](#).

## NXC

Example code is provided for [NXC](#) in our [Downloads section](#). These examples were developed by Matthew and John Hanson (thanks!).

## Labview for Lego Mindstorms

VI's for Labview for Lego Mindstorms are provided by [Tufts CEEQ](#). You can find them on our [downloads page](#).

## Lejos

You can find Lejos software in our [downloads section](#) and a great explanation and [tutorial on Aswin's site here](#).

## Calibration of the Accelerometer

Calibration of the Accelerometer is done by writing values to the sensor. The accelerometer stores these values into memory and offsets readings by this value. The calibration will be lost however when power is lost to the sensor.

## Technical Specifications

The [dIMU](#) is both an accelerometer and a gyroscope. For each part, we've outlined the technical specifications below. Both sensors are digital and use unique I2C addresses. This allows them to be used alongside other I2C sensors on the same port.

### Accelerometer Technical Specifications

- Axes: Reads acceleration on 3 axes (x, y, and z).
- Multi-sensitivity: Sensitivity can be chosen with the software. The sensor can be set to  $\pm 2g$ ,  $\pm 4g$ , and  $\pm 8g$  sensitivity.
- Measures static and dynamic acceleration.
- Minimum Acceleration: 0.015g.
- Maximum Acceleration:  $\pm 8g$ .
- Features low power consumption, self-test for z-axis, chip-level calibration, and 10-bit value data output.

### Gyroscope Technical Specifications

- Axes: Reads rotational speed on 3 axes (x, y, and z).
- Multi-sensitivity: Sensitivity can be chosen with the software. The sensor can be set to  $\pm 250$ ,  $\pm 500$ , and  $\pm 2,000$  degrees per second (dps).
- Minimum Rotational Speed:  $\pm 1$  dps.
- Maximum Rotational Speed:  $\pm 2,000$  dps.
- Features low power consumption, 16-bit value data output, sleep mode, and embedded temperature sensor.

## Data Sheets

The gyroscope is manufactured by ST and the accelerometer is manufactured by Freescale. Datasheets and application notes for both the [accelerometer and gyroscope can be downloaded here](#).

## Accelerometer Registers

- Read Address: 0x3A.
- See Page 21 of the Accelerometer datasheet for more details.
- Below is a Register Summary.

Address	Name	Definition	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
\$00	XOUTL	10 bits output value X LSB	XOUT[7]	XOUT[6]	XOUT[5]	XOUT[4]	XOUT[3]	XOUT[2]	XOUT[1]	XOUT[0]
\$01	XOUTH	10 bits output value X MSB	—	—	—	—	—	—	XOUT[9]	XOUT[8]
\$02	YOUTL	10 bits output value Y LSB	YOUT[7]	YOUT[6]	YOUT[5]	YOUT[4]	YOUT[3]	YOUT[2]	YOUT[1]	YOUT[0]
\$03	YOUTH	10 bits output value Y MSB	—	—	—	—	—	—	YOUT[9]	YOUT[8]
\$04	ZOUTL	10 bits output value Z LSB	ZOUT[7]	ZOUT[6]	ZOUT[5]	ZOUT[4]	ZOUT[3]	ZOUT[2]	ZOUT[1]	ZOUT[0]
\$05	ZOUTH	10 bits output value Z MSB	—	—	—	—	—	—	ZOUT[9]	ZOUT[8]
\$06	XOUT8	8 bits output value X	XOUT[7]	XOUT[6]	XOUT[5]	XOUT[4]	XOUT[3]	XOUT[2]	XOUT[1]	XOUT[0]
\$07	YOUT8	8 bits output value Y	YOUT[7]	YOUT[6]	YOUT[5]	YOUT[4]	YOUT[3]	YOUT[2]	YOUT[1]	YOUT[0]
\$08	ZOUT8	8 bits output value Z	ZOUT[7]	ZOUT[6]	ZOUT[5]	ZOUT[4]	ZOUT[3]	ZOUT[2]	ZOUT[1]	ZOUT[0]
\$09	STATUS	Status registers	—	—	—	—	—	PERR	DOVR	DRDY
\$0A	DETSRC	Detection source registers	LDX	LDY	LDZ	PDX	PDY	PDZ	INT2	INT1
\$0B	TOUT	"Temperature output value" (Optional)	TMP[7]	TMP[6]	TMP[5]	TMP[4]	TMP[3]	TMP[2]	TMP[1]	TMP[0]
\$0C		(Reserved)	—	—	—	—	—	—	—	—
\$0D	I2CAD	I <sup>2</sup> C device address	I2CDIS	DAD[6]	DAD[5]	DAD[4]	DAD[3]	DAD[2]	DAD[1]	DAD[0]
\$0E	USRINF	User Information (Optional)	UI[7]	UI[6]	UI[5]	UI[4]	UI[3]	UI[2]	UI[1]	UI[0]
\$0F	WHOAMI	"Who am I" value (Optional)	ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]
\$10	XOFFL	Offset drift X value (LSB)	XOFF[7]	XOFF[6]	XOFF[5]	XOFF[4]	XOFF[3]	XOFF[2]	XOFF[1]	XOFF[0]
\$11	XOFFH	Offset drift X value (MSB)	—	—	—	—	—	XOFF[10]	XOFF[9]	XOFF[8]
\$12	YOFFL	Offset drift Y value (LSB)	YOFF[7]	YOFF[6]	YOFF[5]	YOFF[4]	YOFF[3]	YOFF[2]	YOFF[1]	YOFF[0]
\$13	YOFFH	Offset drift Y value (MSB)	—	—	—	—	—	YOFF[10]	YOFF[9]	YOFF[8]
\$14	ZOFFL	Offset drift Z value (LSB)	ZOFF[7]	ZOFF[6]	ZOFF[5]	ZOFF[4]	ZOFF[3]	ZOFF[2]	ZOFF[1]	ZOFF[0]
\$15	ZOFFH	Offset drift Z value (MSB)	—	—	—	—	—	ZOFF[10]	ZOFF[9]	ZOFF[8]
\$16	MCTL	Mode control	—	DRPD	SPI3W	STON	GLVL[1]	GLVL[0]	MOD[1]	MOD[0]
\$17	INTRST	Interrupt latch reset	—	—	—	—	—	—	CLRINT2	CLRINT1
\$18	CTL1	Control 1	DFBW	THOPT	ZDA	YDA	XDA	INTRG[1]	INTRG[0]	INTPIN
\$19	CTL2	Control 2	—	—	—	—	—	DRVO	PDPL	LDPL
\$1A	LDTH	Level detection threshold limit value	LDTH[7]	LDTH[6]	LDTH[5]	LDTH[4]	LDTH[3]	LDTH[2]	LDTH[1]	LDTH[0]
\$1B	PDTH	Pulse detection threshold limit value	PDTH[7]	PDTH[6]	PDTH[5]	PDTH[4]	PDTH[3]	PDTH[2]	PDTH[1]	PDTH[0]
\$1C	PW	Pulse duration value	PD[7]	PD[6]	PD[5]	PD[4]	PD[3]	PD[2]	PD[1]	PD[0]
\$1D	LT	Latency time value	LT[7]	LT[6]	LT[5]	LT[4]	LT[3]	LT[2]	LT[1]	LT[0]
\$1E	TW	Time window for 2 <sup>nd</sup> pulse value	TW[7]	TW[6]	TW[5]	TW[4]	TW[3]	TW[2]	TW[1]	TW[0]
\$1F		(Reserved)	—	—	—	—	—	—	—	—

Accelerometer Register Summary. See Page 21 of the supplied datasheet for more information.

## Gyroscope Registers

- Read Address: 0xD2.
- See Page 27 of the Gyroscope datasheet for more details.
- Below is a Register Summary.

**Table 18. Register address map**

Name	Type	Register address		Default	Comment
		Hex	Binary		
Reserved	-	00-0E	-	-	
WHO_AM_I	r	0F	000 1111	11010011	
Reserved	-	10-1F	-	-	
CTRL_REG1	rw	20	010 0000	00000111	
CTRL_REG2	rw	21	010 0001	00000000	
CTRL_REG3	rw	22	010 0010	00000000	
CTRL_REG4	rw	23	010 0011	00000000	
CTRL_REG5	rw	24	010 0100	00000000	
REFERENCE	rw	25	010 0101	00000000	
OUT_TEMP	r	26	010 0110	output	
STATUS_REG	r	27	010 0111	output	
OUT_X_L	r	28	010 1000	output	
OUT_X_H	r	29	010 1001	output	
OUT_Y_L	r	2A	010 1010	output	
OUT_Y_H	r	2B	010 1011	output	
OUT_Z_L	r	2C	010 1100	output	
OUT_Z_H	r	2D	010 1101	output	
FIFO_CTRL_REG	rw	2E	010 1110	00000000	
FIFO_SRC_REG	r	2F	010 1111	output	
INT1_CFG	rw	30	011 0000	00000000	
INT1_SRC	r	31	011 0001	output	
INT1_TSH_XH	rw	32	011 0010	00000000	
INT1_TSH_XL	rw	33	011 0011	00000000	
INT1_TSH_YH	rw	34	011 0100	00000000	
INT1_TSH_YL	rw	35	011 0101	00000000	
INT1_TSH_ZH	rw	36	011 0110	00000000	
INT1_TSH_ZL	rw	37	011 0111	00000000	
INT1_DURATION	rw	38	011 1000	00000000	

Gyroscope Register Summary. See Page 27 of the supplied datasheet for more information.

## Helpful Hints

- For both sensors, a sequence of commands must be issued to initialize both the gyroscope and the accelerometer. These commands are illustrated in our RobotC example code.

## Other Links and Examples:

Project: [Unicycle Balancing Robot](#)

- Downloads: [Fast Gyroscope NXT-G Blocks](#)
- Project: [Segway in RobotC](#)
- Example: [Lejos!](#)
- Project: [AnyWay + NXTBee = Remote Controlled Segway](#)
- Project: [AnyWay Building Instructions and Program](#)
- Examples: [Labview for the dIMU](#)
- Project: [dIMU Line Follower](#)