



X・Y・Z 3軸加速度センサー

±2g・±4g・±8g・±16g

LIS3DH

MEMS digital output motion sensor ultra low-power high performance 3-axes “nano” accelerometer

★STマイクロ社の超小型3軸加速度センサーを使いやすい14ピンDIPモジュール化にしました。

★I2Cインターフェース、SPIインターフェース両対応です。（どちらかを選択）

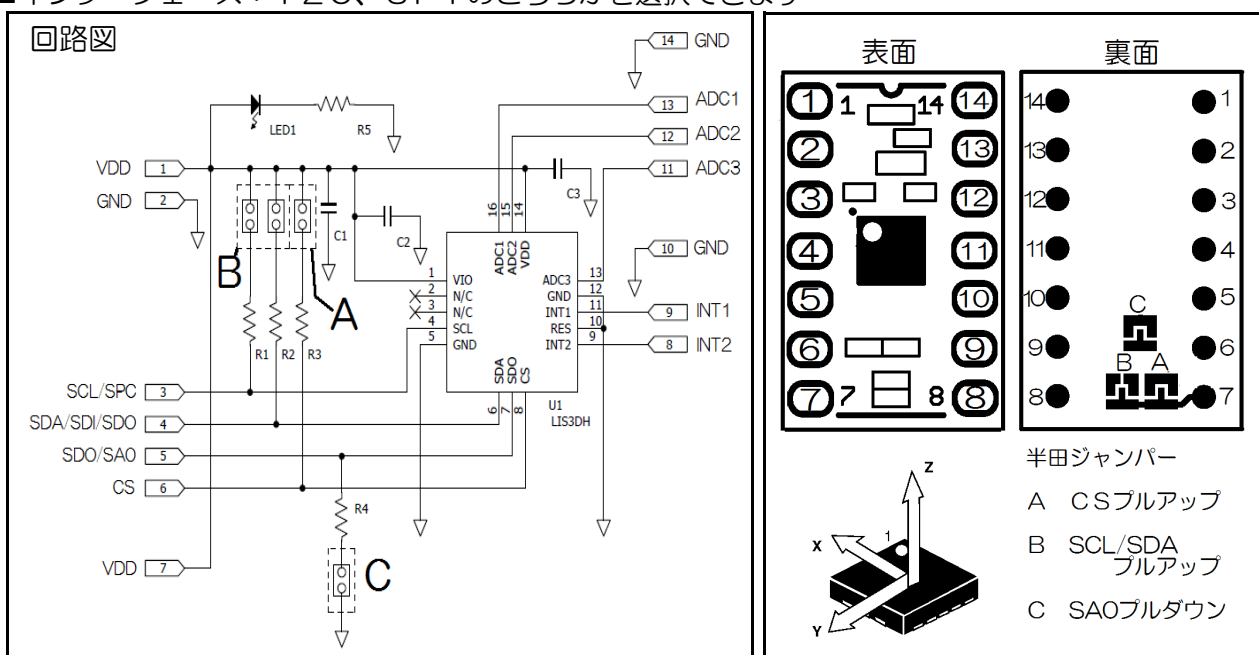
★X、Y、Z3軸の加速度gが12ビットデータで読み取れます。

★マイコンを使った測定、制御などに最適です。 ★A/Dコンバータ入力（3CH）もあります。

■電源電圧：DC3.3V（動作範囲：DC1.71V～DC3.6V）

■消費電流：1mA（Vdd 3V）

■インターフェース：I2C、SPIのどちらかを選択できます



■ピン説明■

番号	名称	I2Cの場合の機能	SPIの場合の機能
1, 7	VDD	電源入力 DC1.71V～3.6V	電源入力 DC1.71V～3.6V
2, 10, 14	GND	GND	GND
3	SCL/SPC	SCL シリアルクロック	SPC シリアルポートクロック
4	SDA/SDI	SDA シリアルデータ	SDI シリアルデータインプット
5	SDO/SAO	SAO スレーブアドレスの下位ビット選択	SDO シリアルデータアウトプット
6	CS	半田ジャンパーAを半田付するとI2Cになる	CS チップセレクト
8	INT2	インターラプト2	インターラプト2
9	INT1	インターラプト1	インターラプト1
11	ADC3	A/Dコンバータ入力1	A/Dコンバータ入力1
12	ADC2	A/Dコンバータ入力2	A/Dコンバータ入力2
13	ADC1	A/Dコンバータ入力3	A/Dコンバータ入力3

■半田ジャンパー■

基板裏面に3箇所の半田ジャンパーがあります。それぞれ必要に応じて半田付けしてください。

A : CSプルアップ	I 2C・SPIを選択します。半田付けをするとI 2C、半田付けしないとSPIになります。
B : SCL/SDA プルアップ	I 2C時、半田付けをすると、SCL信号とSDA信号が抵抗でプルアップされます。SPI時は、半田付けしないしなください。
C : SA0プルダウン	I 2C時SA0（スレーブアドレスのLSB）は、必ず「1」または「0」にする必要があります。半田付けすると「0」になります。「1」にするには、ジャンパーは半田付けしないで、5番ピン（SDA/SA0）をVDDに接続します。SPI時は半田付けしないしなください。

■I 2Cインターフェイスでの接続■

- 1、通常は半田ジャンパー「A」「B」「C」を半田付けします。その場合は「A」でI 2Cが選択され、「B」でI 2CのSCLとSDAがプルアップされます。「C」でスレーブアドレスが、WRITEアドレス0x30（00110000）、READアドレス0x31（00110001）になります。
- 2、マイコンとの接続は、3番ピン（SCL）、4番ピン（SDA）の2線で接続します。

■SPIインターフェイスでの接続■

- 1、半田ジャンパー「A」「B」「C」は、すべて半田付けしません。
- 2、SPC（3番ピン）、SDI（4番ピン）、SDO（5番ピン）、CS（6番ピン）の4線で接続します。
- 3、CS=Lでアクティブ、クロックアイドル時=H、クロック立ち上がりでデータ読み込みです。

■レジスタ表■

Name	Type	Register address		Default	Comment
		Hex	Binary		
Reserved (do not modify)		00 - 06			Reserved
STATUS_REG_AUX	r	07	000 0111		
OUT_ADC1_L	r	08	000 1000	output	
OUT_ADC1_H	r	09	000 1001	output	
OUT_ADC2_L	r	0A	000 1010	output	
OUT_ADC2_H	r	0B	000 1011	output	
OUT_ADC3_L	r	0C	000 1100	output	
OUT_ADC3_H	r	0D	000 1101	output	
INT_COUNTER_REG	r	0E	000 1110		
WHO_AM_I	r	0F	000 1111	00110011	Dummy register
Reserved (do not modify)		10 - 1E			Reserved
TEMP_CFG_REG	rw	1F	001 1111		
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	
CTRL_REG6	rw	25	010 0101	00000000	
REFERENCE	rw	26	010 0110	00000000	
STATUS_REG2	r	27	010 0111	00000000	
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		
INT1_CFG	rw	30	011 0000	00000000	
INT1_SOURCE	r	31	011 0001	00000000	
INT1_THS	rw	32	011 0010	00000000	
INT1_DURATION	rw	33	011 0011	00000000	
Reserved	rw	34-37		00000000	
CLICK_CFG	rw	38	011 1000	00000000	
CLICK_SRC	r	39	011 1001	00000000	
CLICK_THS	rw	3A	011 1010	00000000	
TIME_LIMIT	rw	3B	011 1011	00000000	
TIME_LATENCY	rw	3C	011 1100	00000000	
TIME_WINDOW	rw	3D	011 1101	00000000	

はじめに「WHO_AM_I」（0x0F）を読みます。

「WHO_AM_I」は、LIS3DHと正しく通信出来ているか、確認する為のレジスタです。正しく通信出来ている場合、0x33が読み取れます。

パワーダウンをOFFにして、X軸Y軸Z軸をイネーブルにする為、「CTRL_REG1」（0x20）に0x7Fを書き込みます。

「OUT_X_L」（0x28）、
「OUT_X_H」（0x29）、
「OUT_Y_L」（0x2A）、
「OUT_Y_H」（0x2B）、
「OUT_Z_L」（0x2C）、
「OUT_Z_H」（0x2D）で各軸の値を読み込みます。

詳しくは弊社ホームページのメーカー資料をごらんください。