

آموزش استفاده از ماژول های برنامه نویسی

(ماژول MPU6050)

آخرین تغییرات :

Wednesday, June 12, 2019

نگارش و گردآوری : مجید درهم بخش (Majid Derhambakhsh)

نگارنده کد و پشتیبان : مجید درهم بخش (Majid Derhambakhsh)

[E-mail](#)

فهرست

معرفی آی سی ۳

راهنمای استفاده از ماژول MPU6050 ۴

تنظیمات آی سی

Mpu6050_ConnectionTest تابع

Mpu6050_Init تابع

Mpu6050_AutoInit تابع

Mpu6050_GetRawAccelX-Y-Z توابع

Mpu6050_GetRawAccel تابع

Mpu6050_GetRawTemp تابع

Mpu6050_GetRawGyroX-Y-Z توابع

Mpu6050_GetRawGyro تابع

Mpu6050_GetAccelX-Y-Z توابع

Mpu6050_GetAccel تابع

Mpu6050_GetTemp تابع

Mpu6050_GetGyroX-Y-Z توابع

Mpu6050_GetGyro تابع

Mpu6050_GetAngleX-Y-Z توابع

نکات مهم ۱۱

نیازمندی ها ۱۲

مثال های استفاده از ماژول ۱۳

میکرکنترلر های پشتیبانی شده ۱۵

تغییرات نسخه ۱۶

معرفی آی سی

این آی سی شامل سنسورهای شتاب – زاویه و دما می باشد. مشخصات این سنسور به صورت زیر می باشد.

مشخصات سنسور ژيروسکوپ:

یک ژيروسکوپ سه محوره با قابلیت تنظیم رنج اندازه گیری ۲۵۰ یا ۵۰۰ یا ۱۰۰۰ یا ۲۰۰۰ درجه بر ثانیه.

جریان مصرفی در حالت کاری: ۳,۶ میلی آمپر

جریان مصرفی در حالت Standby برابر ۵ میکروآمپر است.

دارای سه ADC داخلی که داده های آنالوگ ژيروسکوپ را به دیجیتال تبدیل میکند .

مشخصات سنسور شتاب سنج:

شتاب سنج سه محوره با قابلیت رنج اندازه گیری ۲g یا ۴g یا ۸g یا ۱۶g

جریان در حالت کاری: ۵۰۰ میکروآمپر

قابلیت تشخیص تک ضربه (Tap detection)

دارای سه ADC داخلی که داده های آنالوگ شتاب سنج را به دیجیتال تبدیل میکند .

مشخصات کل سنسور :

رنج ولتاژ اعمالی از ۲,۳۷۵ تا ۳,۴۶ ولت

دارای سنسور دمای داخلی با قابلیت خروجی دیجیتال که از ۴۰- تا ۸۵+ درجه را میتواند اندازه گیری کند .

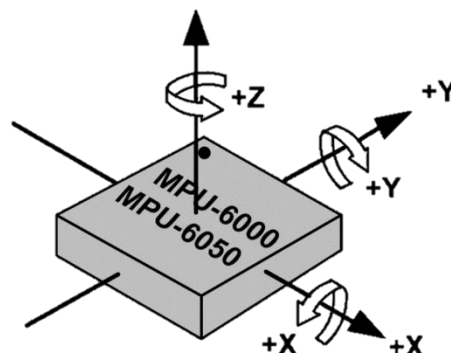
و وجود یک پردازنده داخلی (DMP) که از بار پردازنده میزبان (مانند AVR) کم میکند .

قابلیت ارتباط با میکروکنترلر از طریق پروتکل I2C

جریان مصرفی ۳,۹ میلی آمپر، در حالتی که سه محور شتاب سنج و سه محور ژيروسکوپ و پردازنده DMP فعال هستند .

استفاده از وقفه خارجی برای مواردی مانند تشخیص حرکت، تشخیص ضربه و...

دارای یک باس I2C کمکی برای ارتباط با سنسورهای خارجی (مثل قطب نما)



راهنمای استفاده از مژول MPU6050

در مرحله نخست برای استفاده از این مژول باید طبق دستور زیر آن را به برنامه اضافه کنید:

```
#include "location/MPU6050.h"
```

جهت استفاده از IC ابتدا باید آن را تنظیم نمایید. برای اینکار کافیسیت به روش های زیر عمل کنید:

تنظیمات اتوماتیک : جهت تنظیم اتوماتیک آی سی تنها کافیسیت تابع (`Mpu6050_AutoInit()`) را اجرا نمایید.

تنظیمات دلفواه : برای تنظیم آی سی کافیسیت به روش زیر عمل کنید:

مرحله اول:

Mpu6050_Config.x

x : ویژگی مورد نظر و دارای بخش های زیر است:

- o **SampleRateDivider** : جهت تنظیم نرخ نمونه برداری می باشد و دارای ورودی از 1-256 می باشد.
- o **ExtSync** : جهت تنظیم بین FSYNC و دارای آرگومان های زیر می باشد.

-----> غیر فعال سازی ورودی
_ES_INPUT_DISABLE
-----> تنظیم نمونه برداری توسط رجیستر دما
_ES_TEMP_OUT_L
-----> تنظیم نمونه برداری توسط رجیستر ژيروسکوپ محور x
_ES_GYRO_XOUT_L
-----> تنظیم نمونه برداری توسط رجیستر ژيروسکوپ محور y
_ES_GYRO_YOUT_L
-----> تنظیم نمونه برداری توسط رجیستر ژيروسکوپ محور z
_ES_GYRO_ZOUT_L
-----> تنظیم نمونه برداری توسط رجیستر شتاب سنج محور x
_ES_ACCEL_XOUT_L
-----> تنظیم نمونه برداری توسط رجیستر شتاب سنج محور y
_ES_ACCEL_YOUT_L
-----> تنظیم نمونه برداری توسط رجیستر شتاب سنج محور z
_ES_ACCEL_ZOUT_L

- o **DigitalLowPassFilter** : جهت تنظیم فیلتر پایین گذر دیجیتال داخلی و دارای آرگومان های زیر می باشد.

DLPF_CFG	Accelerometer (Fs = 1kHz)		Gyroscope		
	Bandwidth (Hz)	Delay (ms)	Bandwidth (Hz)	Delay (ms)	Fs (kHz)
_DLPF_CFG_0	260	0	256	0.98	8
_DLPF_CFG_1	184	2.0	188	1.9	1
_DLPF_CFG_2	94	3.0	98	2.8	1
_DLPF_CFG_3	44	4.9	42	4.8	1
_DLPF_CFG_4	21	8.5	20	8.3	1
_DLPF_CFG_5	10	13.8	10	13.4	1
_DLPF_CFG_6	5	19.0	5	18.6	1

○ **GyroFullScaleRange** : جهت تنظیم بازه اندازه گیری ژيروسکوپ و دارای آرگومان های زیر می باشد.

_GYRO_FULL_SCALE_RANGE_250	---->	بازه اندازه گیری ۲۵۰ درجه در ثانیه
_GYRO_FULL_SCALE_RANGE_500	---->	بازه اندازه گیری ۵۰۰ درجه در ثانیه
_GYRO_FULL_SCALE_RANGE_1000	---->	بازه اندازه گیری ۱۰۰۰ درجه در ثانیه
_GYRO_FULL_SCALE_RANGE_2000	---->	بازه اندازه گیری ۲۰۰۰ درجه در ثانیه

○ **AccelFullScaleRange** : جهت تنظیم بازه اندازه گیری شتاب سنج و دارای آرگومان های زیر می باشد.

_ACCEL_FULL_SCALE_RANGE_2G	---->	بازه اندازه گیری 2g
_ACCEL_FULL_SCALE_RANGE_4G	---->	بازه اندازه گیری 4g
_ACCEL_FULL_SCALE_RANGE_8G	---->	بازه اندازه گیری 8g
_ACCEL_FULL_SCALE_RANGE_16G	---->	بازه اندازه گیری 16g

○ **InterruptConfig** : جهت تنظیم بین وقفه خارجی می باشد و به صورت زیر استفاده می شود.

Mpu6050_Config.InterruptConfig.x

x : ویژگی مورد نظر و دارای بخش های زیر است:

- **IntLevel** : جهت تعیین سطح مروهی بین وقفه می باشد و دارای آرگومان های زیر است.

_____>	سطح منطقی یک
_____>	سطح منطقی صفر
- **IntOpen** : جهت تعیین حالت بین وقفه می باشد و دارای آرگومان های زیر است.

_____>	خروجی در حالت پوش پول
_____>	خروجی در حالت این درین
- **LatchIntEn** : جهت تعیین زمان وقفه می باشد و دارای آرگومان های زیر است.

_____>	پالس با عرض ۵۰ میکروثانیه
_____>	فعال بودن تا زمان پاک شدن پرچم وقفه

○ **InterruptEnable** : جهت فعال سازی بین وقفه خارجی و دارای آرگومان های زیر می باشد.

_INT_DISABLE	----->	غیرفعال سازی وقفه
_INT_DATA_RDY_EN	----->	فعال سازی وقفه آماده بودن داده ها
_INT_I2C_MST_INT_EN	----->	فعال سازی وقفه I2C
_INT_FIFO_OFLOW_EN	----->	فعال سازی وقفه سرریز شدن بافر

○ **ClockSelection** : جهت تنظیم منبع کلاک و دارای آرگومان های زیر می باشد.

_CLKSEL_INTERNAL_8MHZ_OSCILLATOR	----->	اسیلاتور داخلی ۸ مگاهرتز
_CLKSEL_X_AXIS_GYROSCOPE_REFERENCE	----->	ژيروسکوپ محور x
_CLKSEL_Y_AXIS_GYROSCOPE_REFERENCE	----->	ژيروسکوپ محور y
_CLKSEL_Z_AXIS_GYROSCOPE_REFERENCE	----->	ژيروسکوپ محور z
_CLKSEL_EXTERNAL_32768HZ_REFERENCE	----->	اسیلاتور خارجی ۳۲/۷۶۸ کیلوهرتز
_CLKSEL_EXTERNAL_1920KHZ_REFERENCE	----->	اسیلاتور ۱۹,۲ مگاهرتز
_CLKSEL_STOP	----->	غیر فعال سازی منبع کلاک

مرمله دوم: فراخوانی تابع Mpu6050_Init().

راهنمای خواندن مقادیر سنسورها

جهت خواندن مقادیر سنسور ها تنها کافیست توابع آن ها را اجرا نمایید. این توابع دارای ۳ دسته می باشد که شامل :

- (۱) توابع بررسی آی سی.
- (۲) توابع خواندن مقادیر خام.
- (۳) توابع خواندن مقادیر مماسیبه شده.

a. توابع بررسی آی سی : این توابع جهت بررسی وضعیت سنسور مانند وضعیت اتصال سنسور به پردازنده به کار می رود. و شامل موارد زیر است.

تابع `Mpu6050_ConnectionTest()` : این تابع جهت بررسی وضعیت اتصال سنسور به کار می رود و به صورت زیر مورد استفاده قرار می گیرد.

```
uint8_t cn;  
  
cn = Mpu6050_ConnectionTest();
```

مقادیر برگردانده شده :

`_TRUE`

`_FALSE`

b. توابع خواندن مقادیر خام : این توابع جهت دریافت مقادیر خام سنسور ها به کار می رود. و شامل موارد زیر است.

مقدار کل محور ها	مقدار محور z	مقدار محور y	مقادیر محور x	سنسور
<code>Mpu6050_GetRawAccel()</code>	<code>Mpu6050_GetRawAccelZ()</code>	<code>Mpu6050_GetRawAccelY()</code>	<code>Mpu6050_GetRawAccelX()</code>	توابع شتاب سنج
<code>Mpu6050_GetRawGyro()</code>	<code>Mpu6050_GetRawGyroZ()</code>	<code>Mpu6050_GetRawGyroY()</code>	<code>Mpu6050_GetRawGyroX()</code>	توابعژیروسکوپ
<code>Mpu6050_GetRawTemp()</code>				تابع دما

نکته : مقادیر ارائه شده کلیه توابع این بخش از نوع `int16_t` (int 16 bit) می باشد.

(ویرایش 0.2.1)

مثال Accel :

```
int16_t ax , ay , az;  
  
int16_t accel_value[3];  
  
Mpu6050_GetRawAccelX( &ax );  
  
Mpu6050_GetRawAccelY( &ay );  
  
Mpu6050_GetRawAccelZ( &az );  
  
Mpu6050_GetRawAccel( accel_value );
```

نکته: چیدمان داده ها در مقادیر کل محور ها (Mpu6050_GetRawAccel) به صورت زیر است.

accel_value[0] --> x	accel_value[1] --> y	accel_value[2] --> z
----------------------	----------------------	----------------------

مثال Gyro:

```
int16_t gx , gy , gz;

int16_t gyro_value[3];

Mpu6050_GetRawGyroX( &gx );

Mpu6050_GetRawGyroY( &gy );

Mpu6050_GetRawGyroZ( &gz );

Mpu6050_GetRawGyro( gyro_value );
```

نکته: چیدمان داده ها در مقادیر کل محور ها (Mpu6050_GetRawGyro) به صورت زیر است.

gyro_value[0] --> x	gyro_value[1] --> y	gyro_value[2] --> z
---------------------	---------------------	---------------------

مثال Temp:

```
int16_t tmp;

Mpu6050_GetRawTemp( &tmp );
```

(ویرایش 0.3.0 و جدیدتر)

مثال Accel:

```
uint8_t status;

int16_t ax , ay , az;

int16_t accel_value[3];

status = Mpu6050_GetRawAccelX( &ax );

status = Mpu6050_GetRawAccelY( &ay );

status = Mpu6050_GetRawAccelZ( &az );

status = Mpu6050_GetRawAccel( accel_value );
```

نکته: status شامل مقادیر _TRUE و _FALSE است.

نکته: چیدمان داده ها در مقادیر کل محور ها (Mpu6050_GetRawAccel) به صورت زیر است.

accel_value[0] --> x	accel_value[1] --> y	accel_value[2] --> z
----------------------	----------------------	----------------------

مثال Gyro:

```
uint8_t status;

int16_t gx , gy , gz;

int16_t gyro_value[3];

status = Mpu6050_GetRawGyroX( &gx );

status = Mpu6050_GetRawGyroY( &gy );

status = Mpu6050_GetRawGyroZ( &gz );

status = Mpu6050_GetRawGyro( gyro_value );
```



نکته : status شامل مقادیر `_TRUE` و `_FALSE` است.

نکته : چیدمان داده ها در مقادیر کل محور ها (`Mpu6050_GetRawGyro`) به صورت زیر است.

gyro_value[0] --> x	gyro_value[1] --> y	gyro_value[2] --> z
---------------------	---------------------	---------------------

مثال Temp :

```
uint8_t status;
```

```
int16_t tmp;
```

```
status = Mpu6050_GetRawTemp( &tmp );
```

نکته : status شامل مقادیر `_TRUE` و `_FALSE` است.

c. توابع خواندن مقادیر مناسبه شده : این توابع جهت دریافت مقادیر مناسبه شده سنسور ها به کار می رود. و شامل موارد زیر است.

مقدار کل محور ها	مقدار محور z	مقدار محور y	مقادیر محور x	سنسور
Mpu6050_GetAccel ()	Mpu6050_GetAccelZ()	Mpu6050_GetAccelY()	Mpu6050_GetAccelX()	توابع شتاب سنج
Mpu6050_GetGyro()	Mpu6050_GetGyroZ()	Mpu6050_GetGyroY()	Mpu6050_GetGyroX()	توابع ژيروسکوپ
Mpu6050_GetTemp()				تابع دما

نکته : مقادیر ارائه شده کلیه توابع این بخش از نوع `float` می باشد.

(ویرایش 0.2.1)

مثال Accel :

```
float ax , ay , az;
```

```
float accel_value[3];
```

```
Mpu6050_GetAccelX( &ax );
```

```
Mpu6050_GetAccelY( &ay );
```

```
Mpu6050_GetAccelZ( &az );
```

```
Mpu6050_GetAccel( accel_value );
```

نکته : چیدمان داده ها در مقادیر کل محور ها (`Mpu6050_GetAccel`) به صورت زیر است.

accel_value[0] --> x	accel_value[1] --> y	accel_value[2] --> z
----------------------	----------------------	----------------------

مثال Gyro :

```
float gx , gy , gz;
```

```
float gyro_value[3];
```

```
Mpu6050_GetGyroX( &gx );
```

```
Mpu6050_GetGyroY( &gy );
```




```
Mpu6050_GetGyroZ( &gz );
```

```
Mpu6050_GetGyro( gyro_value );
```

نکته : چیدمان داده ها در مقادیر کل محور ها (Mpu6050_GetGyro) به صورت زیر است.

gyro_value[0] --> x	gyro_value[1] --> y	gyro_value[2] --> z
---------------------	---------------------	---------------------

مثال Temp :

```
float tmp;
```

```
Mpu6050_GetTemp( &tmp );
```

(ویرایش 0.3.0 و مدیدتر)

مثال Accel :

```
uint8_t status;
```

```
float ax , ay , az;
```

```
float accel_value[3];
```

```
status = Mpu6050_GetAccelX( &ax );
```

```
status = Mpu6050_GetAccelY( &ay );
```

```
status = Mpu6050_GetAccelZ( &az );
```

```
status = Mpu6050_GetAccel( accel_value );
```

نکته : status شامل مقادیر TRUE و FALSE است.

نکته : چیدمان داده ها در مقادیر کل محور ها (Mpu6050_GetAccel) به صورت زیر است.

accel_value[0] --> x	accel_value[1] --> y	accel_value[2] --> z
----------------------	----------------------	----------------------

مثال Gyro :

```
uint8_t status;
```

```
float gx , gy , gz;
```

```
float gyro_value[3];
```

```
status = Mpu6050_GetGyroX( &gx );
```

```
status = Mpu6050_GetGyroY( &gy );
```

```
status = Mpu6050_GetGyroZ( &gz );
```

```
status = Mpu6050_GetGyro( gyro_value );
```

نکته : status شامل مقادیر TRUE و FALSE است.

نکته : چیدمان داده ها در مقادیر کل محور ها (Mpu6050_GetGyro) به صورت زیر است.

gyro_value[0] --> x	gyro_value[1] --> y	gyro_value[2] --> z
---------------------	---------------------	---------------------

مثال Temp :

```
uint8_t status;
```

```
float tmp;
```

```
status = Mpu6050_GetRawTemp( &tmp );
```

نکته : status شامل مقادیر `_TRUE` و `_FALSE` است.

(ویرایش 0.4.0 و جدیدتر)

مثال Angle :

```
uint8_t status;  
  
float angle_x, angle_y, angle_z;  
  
status = Mpu6050_GetAngleX( &angle_x );  
  
status = Mpu6050_GetAngleY( &angle_y );  
  
status = Mpu6050_GetAngleZ( &angle_z );
```

نکته : status شامل مقادیر `_TRUE` و `_FALSE` است.

نکات مهم

کلیه دستورات و تنظیمات با `_` به همراه مخفف آنها و مروف بزرگ شروع می شوند.

مثال:

```
Ext Sync -----> _ES_XXXXXX -----> _ES_INPUT_DISABLE
Digital Low Pass Filter Config -----> _DLPF_CFG_X -----> _DLPF_CFG_0
```

جهت تنظیم سطح پین ADO به فایل MPU6050_CONFIG.h مراجعه نمایید.

مثال:

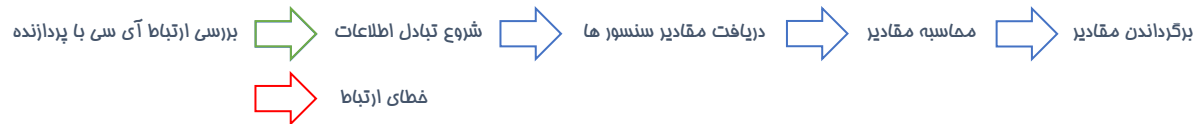
```
_ADO_LEVEL _ADO_LOW
_ADO_LEVEL _ADO_HIGH
```

کلیه توابع به صورت کامل کیس نوشته شده اند. (شروع هر کلمه با مروف بزرگ)

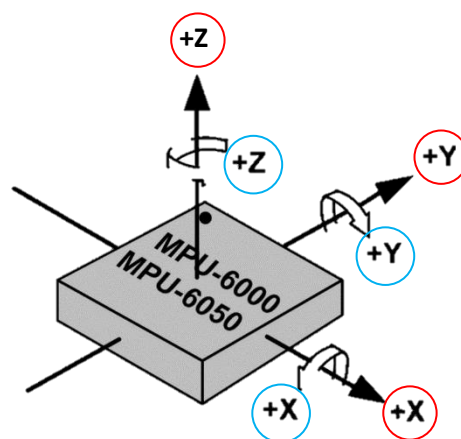
مثال:

```
Mpu6050_GetAccelX();
Mpu6050_GetTemp();
```

کلیه توابع خواندن مقادیر مرامل زیر را طی می کنند.



جهت محور ها در این آی سی به صورت زیر است.



زاویه



شتاب



نیازمندی ها

ماژول `I2C_UNIT.h` برای میکروکنترلر های AVR

Auto Config Mode

```
/*
 * MPU6050 Test Program.c
 *
 * Created: 2/15/2019 9:28:18 AM
 * Author : Majid Derhambakhsh
 */

/***** Includes *****/

#include <io.h>
#include "MPU6050.h"

/*****

#define _SDA 0x02
#define _SCL 0x01
#define _I2C_PORT_DIR DDRC /* I2C PORT */
#define _I2C_PORT_STATE PORTC /* I2C PORT */

*****/

int main(void)
{
    /* ----- */

    float temp = 0;
    float xg = 0; /* Variable for Gyroscope X */
    float yg = 0; /* Variable for Gyroscope Y */
    float zg = 0; /* Variable for Gyroscope Z */
    float xa = 0; /* Variable for Accelerometer X */
    float ya = 0; /* Variable for Accelerometer Y */
    float za = 0; /* Variable for Accelerometer Z */

    /* ----- */

    _I2C_PORT_DIR = _SCL | _SDA;
    _I2C_PORT_STATE = _SCL | _SDA;

    Mpu6050_AutoInit();

    /* ----- */

    while (1)
    {
        /* ----- Temp Value ----- */

        Mpu6050_GetTemp(&temp); /* Take temp value */

        /* ----- Gyro Value ----- */

        Mpu6050_GetGyroX(&xg); /* Take gyro x value */
        Mpu6050_GetGyroY(&yg); /* Take gyro y value */
        Mpu6050_GetGyroZ(&zg); /* Take gyro z value */

        /* ----- Accel Value ----- */

        Mpu6050_GetAccelX(&xa); /* Take accel x value */
        Mpu6050_GetAccelY(&ya); /* Take accel y value */
        Mpu6050_GetAccelZ(&za); /* Take accel z value */

    }
}
```

Manual Config Mode

```
/*
 * MPU6050 Test Program.c
 *
 * Created: 2/15/2019 9:28:18 AM
 * Author : Majid Derhambakhsh
 */

/***** Includes *****/

#include <io.h>
#include "MPU6050.h"

/*****

#define _SDA 0x02
#define _SCL 0x01
#define _I2C_PORT_DIR DDRC /* I2C PORT */
#define _I2C_PORT_STATE PORTC /* I2C PORT */

*****/

int main(void)
{
    /* ----- */

    float temp = 0;
    float xg = 0; /* Variable for Gyroscope X */
    float yg = 0; /* Variable for Gyroscope Y */
    float zg = 0; /* Variable for Gyroscope Z */
    float xa = 0; /* Variable for Accelerometer X */
    float ya = 0; /* Variable for Accelerometer Y */
    float za = 0; /* Variable for Accelerometer Z */

    /* ----- */

    _I2C_PORT_DIR = _SCL | _SDA;
    _I2C_PORT_STATE = _SCL | _SDA;

    I2C_Init();

    Mpu6050_Config.SampleRateDivider = 8;
    Mpu6050_Config.ExtSync = _ES_INPUT_DISABLE;
    Mpu6050_Config.DigitalLowPassFilter = _DLPF_CFG_0;
    Mpu6050_Config.InterruptEnable = _INT_DATA_RDY_EN;
    Mpu6050_Config.InterruptConfig.IntOpen = _INT_OPEN_PUSH_PULL;
    Mpu6050_Config.GyroFullScaleRange = _GYRO_FULL_SCALE_RANGE_500;
    Mpu6050_Config.InterruptConfig.IntLevel = _INT_LEVEL_ACTIVE_HIGH;
    Mpu6050_Config.ClockSelection = _CLKSEL_INTERNAL_8MHZ_OSCILLATOR;
    Mpu6050_Config.AccelFullScaleRange = _ACCEL_FULL_SCALE_RANGE_16G;

    Mpu6050_Init();

    /* ----- */

    while (1)
    {
        /* ----- Temp Value ----- */

        Mpu6050_GetTemp(&temp); /* Take temp value */

        /* ----- Gyro Value ----- */

        Mpu6050_GetGyroX(&xg); /* Take gyro x value */
        Mpu6050_GetGyroY(&yg); /* Take gyro y value */
        Mpu6050_GetGyroZ(&zg); /* Take gyro z value */

        /* ----- Accel Value ----- */

        Mpu6050_GetAccelX(&xa); /* Take accel x value */
        Mpu6050_GetAccelY(&ya); /* Take accel y value */
        Mpu6050_GetAccelZ(&za); /* Take accel z value */

    }
}
```

میکروکنترلر های پشتیبانی شده



میکروکنترلر های AVR سری mega – tiny توسط توابع I2C_UNIT

کامپایلر های Codevision و GNUC نظیر AtmelStudio

codevision



تغییرات نسخه

Version 0.4.0 :

- * Add : Function for calculate angle x,y,z
-

Version 0.3.0 :

- * Add : Instruction status
 - * Improve : code structure
-

Version 0.2.1 :

- * Add : Config section
 - * Add : Calculation function
 - * Add : Calculation function
 - * Improve : code structure
-

Version 0.0.0 :

- * Stable-Simple version
-

Support : Majid.do16@gmail.com