

```

1 //ESTE PROGRAMA GESTIONA UN IMU DE 6 DOF DE SALIDAS DIGITALES COMPATIBLE CON EL PROTOCOLO I2C
2
3 void main()
4 {
5     // DECLARACION DE LOS REGISTROS DEL ACELEROMETRO
6     #define READ_ADXL    0xA7
7     #define WRITE_ADXL   0xA6
8     #define DEVID        0x00
9     #define OFSX         0xIE
10    #define OFSY          0xIF
11    #define OFSZ          0x20
12    #define WINDOW        0x23
13    #define DATA_FORMAT  0x31
14    #define POWERCTL      0x2D
15    #define INT_ENABLE    0x2E
16    #define BWRATE        0x2C
17    #define DATA0        0x32
18    #define DATA1        0x33
19    #define DATAY0        0x34
20    #define DATAY1        0x35
21    #define DATAZ0       0x36
22    #define DATAZ1       0x37
23
24    // DECLARACION DE LOS REGISTROS DEL GIROSCOPO
25
26    #define READ_ITG       0xD3
27    #define WRITE_ITG      0xD2
28    #define SAMPLE_DIV    0x15
29    #define DLPF_FS       0x16
30    #define TEMP_HI        0x1B
31    #define TEMP_LO       0x1C
32    #define GYRO_X1        0x1D
33    #define GYRO_X0        0x1E
34    #define GYRO_Y1        0x1F
35    #define GYRO_Y0        0x20
36    #define GYRO_Z1        0x21
37    #define GYRO_Z0        0x22
38
39    int k,A1,B1,A2,B2;
40    signed int C2,C1;
41    char txt1[7],txt2[7];
42    // CONFIGURACION DEL PUERTO B Y C COMO SALIDA
43
44    DDRC = 1;
45    DDRB = 1;
46    PORTC = 1;
47    PORTB = 2;
48
49    TWI_Init(400000);           // initialize TWI communication
50
51    //----- INICIACLIZACION DEL ACELEROMETRO -----
52
53    TWI_Start();               // issue TWI start signal
54    TWI_Write(WRITE_ADXL);     // send byte via TWI (device address + W)
55    TWI_Write(DATA_FORMAT);
56    TWI_Write(0X08);
57    TWI_Stop();               // issue TWI stop signal
58    Delay_ms(1);
59
60    TWI_Start();               // issue TWI start signal
61    TWI_Write(WRITE_ADXL);     // send byte via TWI (device address + W)
62    TWI_Write(BWRATE);
63    TWI_Write(0X0F);
64    TWI_Stop();               // issue TWI stop signal
65    Delay_ms(1);
66
67    TWI_Start();               // issue TWI start signal
68    TWI_Write(WRITE_ADXL);     // send byte via TWI (device address + W)
69    TWI_Write(POWERCTL);
70    TWI_Write(0X08);
71    TWI_Stop();               // issue TWI stop signal
72    Delay_ms(1);
73
74    TWI_Start();               // issue TWI start signal
75    TWI_Write(WRITE_ADXL);     // send byte via TWI (device address + W)
76    TWI_Write(INT_ENABLE);
77    TWI_Write(0X80);

```

```

78     TWI_Stop();                // issue TWI stop signal
79     Delay_ms(1);
80
81     //-----O-----//
82
83     TWI_Start();                // INICIALIZACION DEL GYROSCOPIO
84     TWI_Write(WRITE_ITG);
85     TWI_Write(DLPF_FS);
86     TWI_Write(0X18);
87     TWI_Stop();
88     Delay_ms(1);
89
90     TWI_Start();                // INICIALIZACION DEL GYROSCOPIO
91     TWI_Write(WRITE_ITG);
92     TWI_Write(SAMPLE_DIV);
93     TWI_Write(0X00);
94     TWI_Stop();
95     Delay_ms(1);
96
97     UART1_Init(115200);         // Initialize UART module at 11520000 bps
98     Delay_ms(100);
99     while(1)
100     {
101         if(UART1_Data_Ready()==1)
102         {
103             {
104                 if(UART1_Read() == '@')
105
106                 //----- DATOS DEL ACELEROMETRO -----
107                 {
108                     TWI_Start();                // issue TWI start signal
109                     TWI_Write(WRITE_ADXL);       // send byte via TWI (device address + W)
110                     TWI_Write(DATA_X0);
111                     TWI_Start();
112                     TWI_Write(READ_ADXL);
113
114                     for(k=0;k<3;k++)
115                     {
116                         A1=TWI_Read(1);
117                         B1=TWI_Read(1);
118                         //UART1_Write(B1);
119                         //UART1_Write(A1);
120                         C1=B1*0x100+A1;
121                         IntToStr(C1, txt1);
122                         UART1_Write_Text(txt1);
123                         //UART1_Write(32);
124                         A1=B1=C1=0;
125                         Delay_ms(1);
126                     }
127                     TWI_Stop();
128                     Delay_ms(1);
129
130                     //----- DATOS DEL GYROSCOPIO -----
131                     TWI_Start();                // issue TWI start signal
132                     TWI_Write(WRITE_ITG);       // send byte via TWI (device address + W)
133                     TWI_Write(GYRO_X1);
134                     TWI_Start();                // issue TWI start signal
135                     TWI_Write(READ_ITG);       // send byte via TWI (device address + W)
136
137                     for(k=0;k<3;k++)
138                     {
139                         A2=TWI_Read(1);
140                         B2=TWI_Read(1);
141                         //UART1_Write(A2);
142                         //UART1_Write(B2);
143                         C2=A2*0x100+B2;
144                         IntToStr(C2, txt2);
145                         UART1_Write_Text(txt2);
146                         //UART1_Write(32);      //espacio ascii
147                         A2=B2=C2=0;
148                         Delay_ms(1);
149                     }
150                     TWI_Stop();
151                     TWI_Close();                // issue TWI stop signal
152                 }
153             }
154         }
155     }

```