```
 1  /*
 2  Name:       L3G4200D.ino
 3  Created:    5/25/2017 12:53 PM
 4  Author: Michael Langford
 5  */
 6
 7  #include "L3G4200D.h"
 8
 9  #define CNTRL_REG_4                0x23
10  #define DPS500_4WIRESPI_CONTUPDT   0x10
11
12  #define CNTRL_REG_2                0x21
13  #define HP_FILTER_MODE30           0x20
14
15  #define CNTRL_REG_3                0x22
16  #define DATA_READY                 0x08
17
18  #define CNTRL_REG_1                0x20
19  #define SPS400_25CUTTOFF           0xff
20
21  #define READ                       0b10111111
22  #define WRITE                      0b00111111
23
24  float cal_x = 0.0f;
25  float cal_y = 0.0f;
26  float cal_z = 0.0f;
27
28  float x = 0;
29  float y = 0;
30  float z = 0;
31
32  float xr = 0.0f, yr = 0.0f, zr = 0.0f;
33
34  int readRegister(byte address);
35  void writeRegister(byte address, byte data);
36
37  float microseconds, lms;
38  float seconds;
39  float pre_update_seconds;
40
41  float GetYaw()
42  {
43      return z;
44  }
45
46  float GetPitch()
47  {
48      return x;
49  }
50
51  float GetRoll()
52  {
```

```
53      return y;
54  }
55
56  float GetYawRate()
57  {
58      return zr;
59  }
60
61  float GetPitchRate()
62  {
63      return xr;
64  }
65
66  float GetRollRate()
67  {
68      return yr;
69  }
70
71  float GetGyroElapsedTime()
72  {
73      return pre_update_seconds;
74  }
75
76  void ClearAngles(float yaw, float pitch, float roll)
77  {
78      x = pitch;
79      y = roll;
80      z = yaw;
81
82      seconds = 0.0f;
83      lms = micros();
84  }
85
86  void SetGyroAngles(float yaw, float pitch, float roll)
87  {
88      x = pitch;
89      y = roll;
90      z = yaw;
91  }
92
93  float get_cal_x()
94  {
95      return cal_x;
96  }
97
98  float get_cal_y()
99  {
100     return cal_y;
101 }
102
103 float get_cal_z()
104 {
```

```
105     return cal_z;
106 }
107
108 void init_L3G4200D() {
109     SPI.begin();
110
111     pinMode(10, OUTPUT);
112     digitalWrite(10, HIGH);
113
114     writeRegister(CNTRL_REG_1, SPS400_25CUTTOFF);
115     writeRegister(CNTRL_REG_4, DPS500_4WIRESPI_CONTUPDT);
116     writeRegister(CNTRL_REG_2, HP_FILTER_MODE30);
117     writeRegister(CNTRL_REG_3, DATA_READY);
118
119     Calibrate_Gyro();
120 }
121
122 void Calibrate_Gyro()
123 {
124     delay(100);
125     float cx = 0, cy = 0, cz = 0;
126     for (int i = 0; i < CALIBRATE_TIME; i++)
127     {
128         while (true)
129         {
130             if ((readRegister(0x27) & 8) == 8)
131                 break;
132         }
133         cx += (float)getX();
134         cy += (float)getY();
135         cz += (float)getZ();
136     }
137     cal_x = cx / (float)CALIBRATE_TIME;
138     cal_y = cy / (float)CALIBRATE_TIME;
139     cal_z = cz / (float)CALIBRATE_TIME;
140
141     x = y = z = 0.0f;
142     seconds = 0.0f;
143     lms = micros();
144 }
145
146 void update_L3G4200D() {
147
148     xr = ((float)getX() - cal_x) * 0.0175f;
149     yr = ((float)getY() - cal_y) * 0.0175f;
150     zr = ((float)getZ() - cal_z) * 0.0175f;
151
152     x += xr*seconds;
153     y += yr*seconds;
154     z += zr*seconds;
155
156     //iffy code :)
```

```
157     x += y*sinf(radians(zr*seconds));
158     y -= x*sinf(radians(zr*seconds));
159 
160     pre_update_seconds = seconds;
161     microseconds = (float)micros() - lms;
162     lms = micros();
163     seconds = microseconds / 1000000.0f;
164     if (seconds < 0)
165         seconds = -seconds;
166 }
167 
168 int16_t getX()
169 {
170     int16_t xval = (int16_t)(((readRegister(0x29) & 0xFF) << 8) | (readRegister
      (0x28) & 0xFF));
171     return xval;
172 }
173 
174 int16_t getY()
175 {
176     return (int16_t)(((readRegister(0x2B) & 0xFF) << 8) | (readRegister(0x2A) &
      0xFF));
177 }
178 
179 int16_t getZ()
180 {
181     return (int16_t)(((readRegister(0x2D) & 0xFF) << 8) | (readRegister(0x2C) &
      0xFF));
182 }
183 
184 int readRegister(byte address)
185 {
186     //SPI.setMOSI(PIN_SPI_MOSI);
187     //SPI.setMISO(PIN_SPI_MISO);
188     //SPI.setSCK(PIN_SPI_SCK);
189     //SPI.begin();
190 
191     SPI.beginTransaction(SPISettings(CLOCK_SPEED, MSBFIRST, SPI_MODE0));
192 
193     int toRead;
194 
195     address |= 0x80;
196 
197     digitalWrite(10, LOW);
198     SPI.transfer(address);
199     toRead = SPI.transfer(0);
200     digitalWrite(10, HIGH);
201     SPI.endTransaction();
202     return toRead;
203 }
204 
205 void writeRegister(byte address, byte data)
```

```
206 {
207     //SPI.setMOSI(PIN_SPI_MOSI);
208     //SPI.setMISO(PIN_SPI_MISO);
209     //SPI.setSCK(PIN_SPI_SCK);
210     //SPI.begin();
211 
212     SPI.beginTransaction(SPISettings(CLOCK_SPEED, MSBFIRST, SPI_MODE0));
213 
214     address &= 0x7F;
215 
216     digitalWrite(10, LOW);
217     SPI.transfer(address);
218     SPI.transfer(data);
219     digitalWrite(10, HIGH);
220 
221     SPI.endTransaction();
222 }
```