

```
1  /* ESTA FUNCIÓN DEVUELVE LOS DATOS QUE SE QUIEREN OBTENER DEL MPU6050 (PITCH, YAW Y
2  ROLL) UNA VEZ SE LE HAN DADO EL VALOR DEL CUATERNIÓN DE ACTITUD*/
3  void QuatToEuler(Quaternion q, float &yaw, float &pitch, float &roll) {
4      //El proceso que seguidamente se muestra es completamente algebraico, en la
5      memoria del trabajo se encuentra explicado el proceso.
6      test = q.x * q.z - q.w * q.y;
7      sqx = q.x * q.x;
8      sqy = q.y * q.y;
9      sqz = q.z * q.z;
10     sqw = q.w * q.w;
11     roll = (180 / pi) * atan2(2 * q.y * q.z + 2 * q.w * q.x, sqz - sqy - sqx + sqw);
12     pitch = -(180 / pi) * asin(2 * test);
13     yaw = (180 / pi) * atan2(2 * q.x * q.y + 2 * q.w * q.z, sqx + sqw - sqz - sqy);
14     //Se asegura que en caso de gimbal lock, el resultado esté definido
15     if (pitch >= 89.5) {
16         pitch = 90;
17         yaw = (180 / pi) * 2 * atan2(q.z, q.w);
18         roll = 0;
19     } else if (pitch <= -89.5) {
20         pitch = -90;
21         yaw = (180 / pi) * 2 * atan2(q.z, q.w);
22         roll = 0;
23     }
24 }
```