

```

//incloem una llibreria per a cada sensor
#include <MPU6050_tockn0.h>
#include <MPU6050_tockn1.h>
#include <MPU6050_tockn2.h>
#include <MPU6050_tockn3.h>
#include <Wire.h>
#include <VarSpeedServo.h>

//enllacem cada dispositiu amb les comandes que utilitza la seva llibreria
MPU60500 mpu60500(Wire);
MPU60501 mpu60501(Wire);
MPU60502 mpu60502(Wire);
MPU60502 mpu60503(Wire);

//variable que utilitzarem per controlar cada quan executem el programa
long timer = 0;
//variable que utilitzarem perquè els sensors només mostrin els seus valors un
cop per cicle
bool sensor = false;
//variable que utilitzarem perquè els càlculs només s'executin un cop per cicle
bool calcul = false;

//adreça del multiplexor TCA9548A
#define TCAADDR 0x70

//subrutina per escollir de quin canal llegeix el multiplexor
void tcaselect(uint8_t i)
{
    if (i > 7) return;

    Wire.beginTransmission(TCAADDR);
    Wire.write(1 << i);
    Wire.endTransmission();
}

//definim els servomotors
VarSpeedServo espatlla;
VarSpeedServo colze;
VarSpeedServo canell;
VarSpeedServo pinca;
VarSpeedServo esquena;
VarSpeedServo ma;

//definim les variables que utilitzarà la subrutina de càlcul
const float pi = 3.14159265359;

```

```
float theta1;  
float theta2;  
float theta3;  
float theta5;
```

```
float X;  
float Y;  
float Z;
```

```
float R;  
float R12;  
float R1;  
float R2;  
float R3;  
float R4;  
long R5;
```

```
float S;  
float T;
```

```
int L1 = 310;  
int L2 = 80;  
int a2 = 240;  
int a3 = 300;  
int L5 = 240;
```

```
float teta1;  
float teta2;  
float teta3;  
float teta4;  
float teta5;
```

```
float Axy;  
float Az;
```

```
void setup()  
{  
    //enllacem cada servomotor a un PIN PWM de l'Arduino  
    pinca.attach(3);  
    canell.attach(5);  
    colze.attach(6);  
    espatlla.attach(9);  
    esquena.attach(10);  
    ma.attach(11);  
}
```

```

//declarem el pin 4 com a entrada
pinMode(4, INPUT);

//iniciem el port sèrie i l'I²C
Serial.begin(9600);
Wire.begin();

//calibrem cada giroscopi cada cop que iniciem el programa
tcselect(0);
mpu60500.begin();
mpu60500.calcGyroOffsets(true);

tcselect(1);
mpu60501.begin();
mpu60501.calcGyroOffsets(true);

tcselect(2);
mpu60502.begin();
mpu60502.calcGyroOffsets(true);

tcselect(3);
mpu60503.begin();
mpu60503.calcGyroOffsets(true);

//definim una posició inicial pel braç
esquena.write(0, 30, false);
espatlla.write(90, 30, false);
colze.write(90, 30, false);
canell.write(90, 30, false);
ma.write(0, 30, false);
pinca.write(90, 30, false);
}

void loop()
{
    //el programa executa contínuament la subrutina que llegeix els valors que
    ens arriben dels sensors
    adquirir_dades();

    //executem aquesta part del codi al inici de cada cicle
    if ((millis() - timer > 0) && (millis() - timer < 1000) && sensor == false)
    {
        //executa la subrutina que imprimeix els valors llegits anteriorment pel
        canal sèrie per ajudar a l'usuari a poder "debugar" amb més facilitat
        imprimir_sensors();
        //fem que la comanda anterior s'executi només 1 cop per cicle
    }
}

```

```

    sensor = true;
}
//executem aqueta part del codi quan hagi passat 1 segon
else if ((millis() - timer > 1000) && (millis() - timer < 4000) && calcul ==
false)
{
    //executa la subrutina que calcula els valors dels angles que s'han de
    transmetre a cada articulació
    calculs();
    //executa la subrutina que imprimeix els valors calculats a la subrutina
    void calculs() pel canal sèrie per ajudar a l'usuari a poder "debugar" amb
    més facilitat
    imprimir_valors();
    //fem que les comandes anteriors s'executin només 1 cop per cicle
    calcul = true;
    //executa la subrutina que transmet els angles adients als servomotors
    motors();
}
//executem aqueta part del codi quan hagin passat 4 segons
else if ((millis() - timer > 4000))
{
    //fem reset al comptador i a les variables booleans a fi de poder tornar
    a executar el programa.
    timer = millis();
    sensor = false;
    calcul = false;
}
}

void adquirir_dades()
{
    tcselect(0);
    mpu60500.update();
    tcselect(1);
    mpu60501.update();
    tcselect(2);
    mpu60502.update();
    tcselect(3);
    mpu60503.update();
}

void motors()
{
    //cada servo executa l'angle que li pertoca. Li sumem un offset que depèn de
    la posició inicial del servo.
    int angle1 = tetal + 0;

```

```

int angle2 = teta2 + 90;
int angle3 = teta3 + 90;
int angle4 = teta4 + 90;
int angle5 = teta5 + 0;

esquena.write(angle1, 30, false);
espatlla.write(angle2, 30, false);
colze.write(angle3, 30, false);
canell.write(angle4, 30, false);
ma.write(angle5, 30, false);

//obrim i tanquem la pinça en funció del polsador
if (digitalRead(4) == HIGH)
{
    pinca.write(90, 30, false);
}
else
{
    pinca.write(135, 30, false);
}
}

void imprimir_sensors()
{
    Serial.println(' ');
    Serial.print("Sensor 0: "); Serial.println(mpu60500.getAngleX());
    Serial.println(' ');
    Serial.print("Sensor 1: "); Serial.println(mpu60501.getAngleX());
    Serial.println(' ');
    Serial.print("Sensor 2: "); Serial.println(mpu60502.getAngleX());
    Serial.println(' ');
    Serial.print("Sensor 3: "); Serial.println(mpu60503.getAngleX());
    Serial.println(' ');
}

void calculs()
{
    //llegim els valors dels sensors i hi afegim un offset als angles perquè
    //s'adeqüin a la direcció original del model matricial
    //theta1 és la lectura del sensor de l'esquena
    int theta1_offset = 90;
    theta1 = (mpu60500.getAngleX() + theta1_offset) * pi / 180;
    //theta2 és la lectura del sensor de l'espatlla
    int theta2_offset = -0;
    theta2 = (-mpu60501.getAngleX() + theta2_offset) * pi / 180;
    //theta3 és la lectura del sensor del colze

```

```

int theta3_offset = -0;
theta3 = (mpu60501.getAngleX() - mpu60502.getAngleX() + theta3_offset) * pi
/ 180;
//theta5 és la lectura del sensor de la mà
int theta5_sensivility = 90 / 80;
theta5 = (mpu60503.getAngleX() * theta5_sensivility) * pi / 180;

//calcules la posició X, Y i Z de la punta de la pinça
X = +L2 * sin(theta1) - L5 * cos(theta3) * cos(-theta2 - theta3 - pi / 2) *
cos(theta1) * sin(theta2) - L5 * cos(theta3) * sin(-theta2 - theta3 - pi / 2)
* cos(theta1) * cos(theta2) + a3 * cos(theta3) * cos(theta1) * cos(theta2) -
L5 * sin(theta3) * cos(-theta2 - theta3 - pi / 2) * cos(theta1) * cos(theta2)
+ L5 * sin(theta3) * sin(-theta2 - theta3 - pi / 2) * cos(theta1) * sin(theta2)
- a3 * sin(theta3) * cos(theta1) * sin(theta2) + a2 * cos(theta1) *
cos(theta2);
Y = -L2 * cos(theta1) - L5 * cos(theta3) * cos(-theta2 - theta3 - pi / 2) *
sin(theta1) * sin(theta2) - L5 * cos(theta3) * sin(-theta2 - theta3 - pi / 2)
* sin(theta1) * cos(theta2) + a3 * cos(theta3) * sin(theta1) * cos(theta2) -
L5 * sin(theta3) * cos(-theta2 - theta3 - pi / 2) * sin(theta1) * cos(theta2)
+ L5 * sin(theta3) * sin(-theta2 - theta3 - pi / 2) * sin(theta1) * sin(theta2)
- a3 * sin(theta3) * sin(theta1) * sin(theta2) + a2 * sin(theta1) *
cos(theta2);
Z = L5 * cos(theta2) * cos(theta3) * cos(-theta2 - theta3 - pi / 2) - L5 *
cos(theta2) * sin(-theta2 - theta3 - pi / 2) * sin(theta3) + a3 * cos(theta2)
* sin(theta3) - L5 * cos(theta3) * sin(-theta2 - theta3 - pi / 2) * sin(theta2)
+ a3 * cos(theta3) * sin(theta2) - L5 * cos(-theta2 - theta3 - pi / 2) *
sin(theta2) * sin(theta3) + a2 * sin(theta2) + L1;

teta1 = theta1 * 180 / pi;
teta2 = theta2 * 180 / pi;
teta3 = theta3 * 180 / pi;

//si X és més gran que 350 establirem que X = 350 i recalcularem el valor
dels angles teta1, teta2 i teta 3
if (X > 350)
{
    X = 350;

    teta1 = atan(Y / 350) + asin(80 / sqrt( pow(350, 2) + pow(Y, 2)));
    teta1 = teta1 * 180 / pi;

    Axy = sqrt(pow(350, 2) + pow(Y, 2) - pow(L2, 2));
    Az = Z - 310;
}

```

```

teta2 = atan((( -pow(Axy, 2) * Az - pow(Az, 3) + 480 * Axy * Az + sqrt(-
pow(Axy , 6) - 2 * pow(Axy , 4) * pow(Az , 2) - pow(Axy , 2) * pow(Az , 4)
+ 1440 * pow(Axy , 5) + 1920 * pow(Axy , 3) * pow(Az , 2) + 480 * Axy *
pow(Az , 4) - 568800 * pow(Axy , 4) - 396000 * pow(Axy , 2) * pow(Az , 2)
- 57600 * pow(Az , 4) - 6912000 * pow(Axy , 3) - 31104000 * Axy * pow(Az ,
2) + 51204960000 * pow(Axy , 2) + 10368000000 * pow(Az , 2) - 11041920000000
* Axy + 7278336000000000) - 25200 * Az) / (pow(Axy , 2) + pow(Az , 2) - 480
* Axy + 57600)) / ((pow(Axy , 2) + pow(Az , 2) + (-pow(Axy , 2) * Az -
pow(Az , 3) + 480 * Axy * Az + sqrt(-pow(Axy , 6) - 2 * pow(Axy , 4) *
pow(Az , 2) - pow(Axy , 2) * pow(Az , 4) + 1440 * pow(Axy , 5) + 1920 *
pow(Axy , 3) * pow(Az , 2) + 480 * Axy * pow(Az , 4) - 568800 * pow(Axy ,
4) - 396000 * pow(Axy , 2) * pow(Az , 2) - 57600 * pow(Az , 4) - 6912000 *
pow(Axy , 3) - 31104000 * Axy * pow(Az , 2) + 51204960000 * pow(Axy , 2) +
103680000000 * pow(Az , 2) - 11041920000000 * Axy + 7278336000000000) - 25200
* Az) * Az / (pow(Axy , 2) + pow(Az , 2) - 480 * Axy + 57600) - 480 * Axy
+ 25200) / (Axy - 240)));
//      teta2 = teta2 * 180 / pi;

if (teta2 < 0)
{
    teta2 = -teta2 * 180 / pi - 180;
}
else
{
    teta2 = -teta2 * 180 / pi;
}

R12 = (pow(Axy, 2) + pow(Az, 2) - 480 * Axy - 720000 / 8);
R1 = R12 / 144000;

R2 = (pow(Axy, 2) * Az + pow(Az, 3) - 480 * Axy * Az + 25200 * Az);
R3 = ((144000 * Axy - 34560000) * (pow(Axy, 2) + pow(Az, 2) - 480 * Axy +
57600));
R4 = -pow(Axy, 6) - 2 * pow(Axy, 4) * pow(Az, 2) - pow(Axy, 2) * pow(Az,
4) + 1440 * pow(Axy, 5) + 1920 * pow(Axy, 3) * pow(Az, 2) + 480 * Axy *
pow(Az, 4) - 568800 * pow(Axy, 4) - 396000 * pow(Axy, 2) * pow(Az, 2) -
57600 * pow(Az, 4) - 6912000 * pow(Axy, 3) - 31104000 * Axy * pow(Az, 2) +
51204960000 * pow(Axy, 2) + 10368000000 * pow(Az, 2) - 11041920000000 *
Axy + 7278336000000000;
R5 = 480 * pow(Axy, 2) + 480 * pow(Az, 2) - 230400 * Axy + 27648000;
teta3 = atan(((1 / 480.0) * (R5) * (-pow(Axy, 2) * Az - pow(Az, 3) + 480 *
Axy * Az + sqrt(R4) - 25200.0 * Az) / R3 + R2 / (144000.0 * Axy -
34560000.0)) / R1);

if (teta3 < 0)
{

```

```

    teta3 = teta3 * 180 / pi + 180;
}
else
{
    teta3 = teta3 * 180 / pi;
}
}

//si X és més petita que -350 establirem que X = -350 i recalcularem el valor
dels angles teta1, teta2 i teta 3
else if (X < -350)
{
    X = -350;
    teta1 = atan(350 / Y) + asin(80 / (sqrt(pow(Y , 2) + pow(350 , 2))));
    teta1 = teta1 * 180 / pi + 90;

    Axy = sqrt(pow(350 , 2) + pow(Y , 2) - pow(L2 , 2));
    Az = Z - 310;

    teta2 = atan((( -pow(Axy, 2) * Az - pow(Az, 3) + 480 * Axy * Az + sqrt(-
pow(Axy , 6) - 2 * pow(Axy , 4) * pow(Az , 2) - pow(Axy , 2) * pow(Az , 4)
+ 1440 * pow(Axy , 5) + 1920 * pow(Axy , 3) * pow(Az , 2) + 480 * Axy *
pow(Az , 4) - 568800 * pow(Axy , 4) - 396000 * pow(Axy , 2) * pow(Az , 2)
- 57600 * pow(Az , 4) - 6912000 * pow(Axy , 3) - 31104000 * Axy * pow(Az ,
2) + 51204960000 * pow(Axy , 2) + 10368000000 * pow(Az , 2) - 11041920000000
* Axy + 7278336000000000) - 25200 * Az) / (pow(Axy , 2) + pow(Az , 2) - 480
* Axy + 57600)) / ((pow(Axy , 2) + pow(Az , 2) + (-pow(Axy , 2) * Az -
pow(Az , 3) + 480 * Axy * Az + sqrt(-pow(Axy , 6) - 2 * pow(Axy , 4) *
pow(Az , 2) - pow(Axy , 2) * pow(Az , 4) + 1440 * pow(Axy , 5) + 1920 *
pow(Axy , 3) * pow(Az , 2) + 480 * Axy * pow(Az , 4) - 568800 * pow(Axy ,
4) - 396000 * pow(Axy , 2) * pow(Az , 2) - 57600 * pow(Az , 4) - 6912000 *
pow(Axy , 3) - 31104000 * Axy * pow(Az , 2) + 51204960000 * pow(Axy , 2) +
10368000000 * pow(Az , 2) - 11041920000000 * Axy + 7278336000000000) - 25200
* Az) * Az / (pow(Axy , 2) + pow(Az , 2) - 480 * Axy + 57600) - 480 * Axy
+ 25200) / (Axy - 240)));

    if (teta2 < 0)
    {
        teta2 = -teta2 * 180 / pi - 180;
    }
    else
    {
        teta2 = -teta2 * 180 / pi;
    }

    R12 = (pow(Axy, 2) + pow(Az, 2) - 480 * Axy - 720000 / 8);

```



```

R1 = R12 / 144000;
R2 = (pow(Axy, 2) * Az + pow(Az, 3) - 480 * Axy * Az + 25200 * Az);
R3 = ((144000 * Axy - 34560000) * (pow(Axy, 2) + pow(Az, 2) - 480 * Axy +
57600));
R4 = -pow(Axy, 6) - 2 * pow(Axy, 4) * pow(Az, 2) - pow(Axy, 2) * pow(Az,
4) + 1440 * pow(Axy, 5) + 1920 * pow(Axy, 3) * pow(Az, 2) + 480 * Axy *
pow(Az, 4) - 568800 * pow(Axy, 4) - 396000 * pow(Axy, 2) * pow(Az, 2) -
57600 * pow(Az, 4) - 6912000 * pow(Axy, 3) - 31104000 * Axy * pow(Az, 2) +
51204960000 * pow(Axy, 2) + 10368000000 * pow(Az, 2) - 11041920000000 *
Axy + 7278336000000000;
R5 = 480 * pow(Axy, 2) + 480 * pow(Az, 2) - 230400 * Axy + 27648000;
teta3 = atan(((1 / 480.0) * (R5) * (-pow(Axy, 2) * Az - pow(Az, 3) + 480 *
Axy * Az + sqrt(R4) - 25200.0 * Az) / R3 + R2 / (144000.0 * Axy -
34560000.0)) / R1);

if (teta3 < 0)
{
    teta3 = teta3 * 180 / pi + 180;
}
else
{
    teta3 = teta3 * 180 / pi;
}

}
else
{
    teta1 = theta1;
    teta2 = theta2;
    teta3 = theta3;
    teta1 = teta1 / pi * 180;
    teta2 = teta2 / pi * 180;
    teta3 = teta3 / pi * 180;
}

}

teta1 = teta1 / 180 * pi;
teta2 = teta2 / 180 * pi;
teta3 = teta3 / 180 * pi;

R = +L2 * sin(teta1) - L5 * cos(teta3) * cos(-teta2 - teta3 - pi / 2) *
cos(teta1) * sin(teta2) - L5 * cos(teta3) * sin(-teta2 - teta3 - pi / 2) *
cos(teta1) * cos(teta2) + a3 * cos(teta3) * cos(teta1) * cos(teta2) - L5 *
sin(teta3) * cos(-teta2 - teta3 - pi / 2) * cos(teta1) * cos(teta2) + L5 *
sin(teta3) * sin(-teta2 - teta3 - pi / 2) * cos(teta1) * sin(teta2) - a3 *
sin(teta3) * cos(teta1) * sin(teta2) + a2 * cos(teta1) * cos(teta2);

```

```

    S = -L2 * cos(teta1) - L5 * cos(teta3) * cos(-teta2 - teta3 - pi / 2) *
    sin(teta1) * sin(teta2) - L5 * cos(teta3) * sin(-teta2 - teta3 - pi / 2) *
    sin(teta1) * cos(teta2) + a3 * cos(teta3) * sin(teta1) * cos(teta2) - L5 *
    sin(teta3) * cos(-teta2 - teta3 - pi / 2) * sin(teta1) * cos(teta2) + L5 *
    sin(teta3) * sin(-teta2 - teta3 - pi / 2) * sin(teta1) * sin(teta2) - a3 *
    sin(teta3) * sin(teta1) * sin(teta2) + a2 * sin(teta1) * cos(teta2);

    T = L5 * cos(teta2) * cos(teta3) * cos(-teta2 - teta3 - pi / 2) - L5 *
    cos(teta2) * sin(-teta2 - teta3 - pi / 2) * sin(teta3) + a3 * cos(teta2) *
    sin(teta3) - L5 * cos(teta3) * sin(-teta2 - teta3 - pi / 2) * sin(teta2) + a3
    * cos(teta3) * sin(teta2) - L5 * cos(-teta2 - teta3 - pi / 2) * sin(teta2) *
    sin(teta3) + a2 * sin(teta2) + L1;

    //finalment obtenim els valors de tots els angles on teta1 equival al servo
    de l'esquena, teta2 equival al servo de l'espatlla, teta3 equival al servo
    del colze, teta4 equival al servo del canell i teta5 equival al servo de la
    rotació de la mà

    teta1 = teta1 / pi * 180;
    teta2 = teta2 / pi * 180;
    teta3 = teta3 / pi * 180;
    teta4 = (teta2 + teta3);
    teta5 = theta5;

    theta1 = theta1 * 180 / pi;
    theta2 = theta2 * 180 / pi;
    theta3 = theta3 * 180 / pi;
}

void imprimir_valors()
{
    //X, Y i Z són els valors de la posició de la punta de la pinça calculats a
    partir dels valors dels angles directament llegits dels sensors
    Serial.print("X: ");
    Serial.println(X);
    Serial.print("Y: ");
    Serial.println(Y);
    Serial.print("Z: ");
    Serial.println(Z);

    //R, S i T són els valors de la posició de la punta de la pinça calculats a
    partir dels valors dels angles si aquests han hagut de ser re-calculats
    Serial.print("R: ");
    Serial.println(R);
    Serial.print("S: ");
    Serial.println(S);
    Serial.print("T: ");
    Serial.println(T);
}

```

```

Serial.println(" ");

Serial.print("Axy: ");
Serial.println(Axy);
Serial.print("Az: ");
Serial.println(Az);

Serial.println(" ");
Serial.print("R1: ");
Serial.println(R1);
Serial.print("R2: ");
Serial.println(R2);
Serial.print("R3: ");
Serial.println(R3);
Serial.print("R4: ");
Serial.println(R4);
Serial.print("R5: ");
Serial.println(R5);
Serial.println(" ");
//theta són els valors dels angles de cada una de les articulacions llegida
directament dels sensors
Serial.print("theta1: ");
Serial.println(theta1);
Serial.print("theta2: ");
Serial.println(theta2);
Serial.print("theta3: ");
Serial.println(theta3);
//teta són els valors dels angles de cada una de les articulacions després de
ser re-calculades
Serial.print("teta1: ");
Serial.println(teta1);
Serial.print("teta2: ");
Serial.println(teta2);
Serial.print("teta3: ");
Serial.println(teta3);
Serial.print("teta4: ");
Serial.println(teta4);
Serial.print("teta5: ");
Serial.println(teta5);
Serial.println(' ');
}

```