


```

    }

    else
    {
        Console.WriteLine("COMANDO 0");
        Comando = 9;
    }
}

static void TTF_EEG(double[,] EEG_data, int n)
{
    System.Numerics.Complex[] complejo = new
System.Numerics.Complex[muestra];

    for (int i = 0; i < muestra; i++)
    {
        complejo[i] = new System.Numerics.Complex(EEG_data[0, i], 0);
    }

    MathNet.Numerics.IntegralTransforms.Fourier.Forward(complejo);

    EEG_data[n,0] = 1;

    for (int i = 1; i < muestra; i++)
    {
        EEG_data[n,i] = System.Numerics.Complex.Abs(complejo[i]);
    }

}

static void brazo_robot(int comando, Kinova.API.Jaco.CJacoArm Jaco_m)
{
    Jaco_m.ControlManager.StartControlAPI();

    //Declare and initialize a CJoystickValue that will emulate a
GeneralJoystick

    Kinova.DLL.Data.Jaco.CJoystickValue value = new
Kinova.DLL.Data.Jaco.CJoystickValue();

    if (Jaco_m.JacoIsReady())
    {

        // int mensaje = serial_comm();
        int mensaje = comando;

        Console.WriteLine(mensaje);
    }
}

```

```
if (mensaje != 255)
{

    Jaco_m.ControlManager.StartControlAPI();
    // X UP
    // if (((Button)m_Grid.Children[4]).IsPressed)
    if (mensaje == 4)
    {
        value.InclineLR = -1f;
    }

    // Y UP
    //if (((Button)m_Grid.Children[5]).IsPressed)
    if (mensaje == 3)
    {
        value.InclineFB = -1f;
    }

    // X DOWN
    //if (((Button)m_Grid.Children[7]).IsPressed)
    if (mensaje == 2)
    {
        value.InclineLR = 1f;
    }

    // Y DOWN
    //if (((Button)m_Grid.Children[8]).IsPressed)
    if (mensaje == 1)
    {
        value.InclineFB = 1f;
    }

    // Z DOWN
    if (mensaje == 9)
    {
        value.Rotate = 1f;
    }

    // Z UP
    if (mensaje == 8)
    {
        value.Rotate = -1f;
    }

    //abre grip
    // if (mensaje == 6)

    //abre grip
    // if (mensaje == 7)

    // Store Position
    if (mensaje == 5)
    {
        value.ButtonValue[2] = 1;
    }
    /*
    // Go To Position
```



```

        if (((Button)m_Grid.Children[11]).IsPressed)
        {
            value.ButtonValue[7] = 1;
        }*/

        //We update JACO with the new data.

////////////////////////////////////
        Jaco_m.ControlManager.SendJoystickFunctionality(value);

        if (mensaje == 15)
        {

            value.InclineFB = 0;
            value.InclineLR = 0;
            value.Rotate = 0;

            Thread.Sleep(50);

            Jaco_m.ControlManager.SendJoystickFunctionality(value);
        }
    }
}

static void Comunicar(int comando)
{
    //while (true)
    // {
        Socket cliente = new Socket(AddressFamily.InterNetwork,
        SocketType.Stream, ProtocolType.Tcp);
        IPEndPoint MiDireccion = new
        IPEndPoint(IPAddress.Parse("192.168.1.33"), 1234);

        // try

        cliente.Connect(MiDireccion);
        Console.WriteLine("Conectado con éxito,ingrese mensaje:");
        string msg = comando.ToString();
        //string msg = Console.ReadLine();
        byte[] msgBuffer = Encoding.Default.GetBytes(msg);
        cliente.Send(msgBuffer, 0, msgBuffer.Length, 0);
        Thread.Sleep(100);

        byte[] buffer = new byte[255];
        int rec = cliente.Receive(buffer, 0, buffer.Length, 0);
        Array.Resize(ref buffer, rec);

        Console.WriteLine("Recibido: {0}",
        Encoding.Default.GetString(buffer));
        cliente.Close();

        // }

    }

}
}
}

```

Programa Arduino

```
void setup() {
  // put your setup code here, to run once:
  pinMode(13,OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(7, OUTPUT);
  #define periodo1 1000/31
  #define periodo2 1000/30
  #define periodo3 1000/29
  #define periodo4 1000/28
}

void loop() {
  // put your main code here, to run repeated

  int cont1=0;
  int cont2=0;
  int cont3=0;
  int cont4=0;

  while(true)
  {
    delay(1);
    cont1=cont1+1;
    cont2=cont2+1;
    cont3=cont3+1;
    cont4=cont4+1;
    // LED 1 //
    if(cont1==(periodo1/2))
    {
      digitalWrite(13, HIGH);
    }
    if(cont1==periodo1)
    {
      digitalWrite(13, LOW);
      cont1=0;
    }

    // LED 2 //

    if(cont2==(periodo2/2))
    {
      digitalWrite(12, HIGH);
    }
    if(cont2==periodo2)
    {
      digitalWrite(12, LOW);
      cont2=0;
    }

    // LED 3 //
```

```
if(cont3==(periodo3/2))
{
    digitalWrite(8, HIGH);
}
if(cont3==periodo3)
{
    digitalWrite(8, LOW);
    cont3=0;
}

// LED 4 //

if(cont4==(periodo4/2))
{
    digitalWrite(7, HIGH);
}
if(cont4==periodo4)
{
    digitalWrite(7, LOW);
    cont4=0;
}
}
}
```

