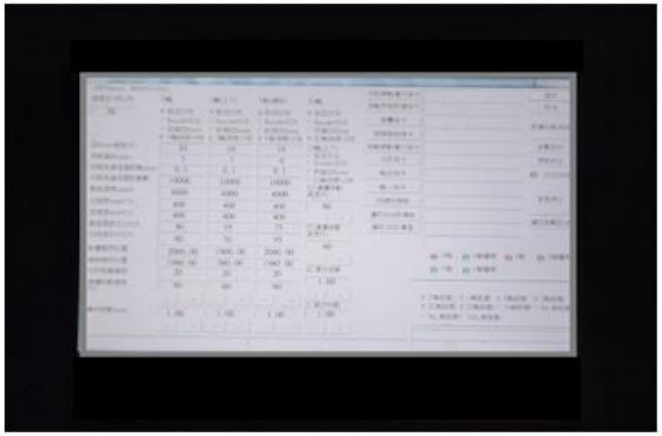
Android 二軸伺服控制器

二軸伺服馬達運動控制器



10.1 吋 Android 觸控型 HMI 操作螢幕

以手機做遠端監控及操作

接頭說明


ADC
AI1
Agnd
AI2
Agnd
AI3
Agnd
AI4
Agnd
DAC
AO4
Agnd
AO3
Agnd
AO2
AO
AO
com
O11
O10
O09
O08
O16
O15
O14
O13
com
O03
O02
O01
O00
I15
I14
I13
I12
I11
I10
I09
I08
I07
I06
I05
I04
I03
I02
I01
I00
DB2
DB1
TN2
TN1
GND
HMY
GND
LY2
GND
LY1
1234.25 mm
X軸啟動
GND
HMX
GND
LX2
GND
LX1
2345.25 mm
Y軸啟動
RS232-2
RS232-1
TX2-
TX2+
RX2-
RX2+
GND
TX1-
TX1+
RX1-
RX1+
GND
TN10
DB4
TN9
DB3

DI 範例程式

```
public boolean DI0, DI1, DI2, DI3, DI4, DI5, DI6, DI7,DI8, DI9, DI10, DI11, DI12, DI13, DI14, DI15;
private void GetInputStatus()
{

    if((RxByteData[0] & 0x01) == 0x01)  DI0 = true; else  DI0 = false;
    if(((RxByteData[0] >> 1) & 0x01) == 0x01) DI1 = true; else DI1 = false;
    if(((RxByteData[0] >> 2) & 0x01) == 0x01) DI2 = true; else DI2 = false;
    if(((RxByteData[0] >> 3) & 0x01) == 0x01) DI3 = true; else DI3 = false;
    if(((RxByteData[0] >> 4) & 0x01) == 0x01) DI4 = true; else DI4 = false;
    if(((RxByteData[0] >> 5) & 0x01) == 0x01) DI5 = true; else DI5 = false;
    if(((RxByteData[0] >> 6) & 0x01) == 0x01) DI6 = true; else DI6 = false;
    if(((RxByteData[0] >> 7) & 0x01) == 0x01) DI7 = true; else DI7 = false;

    if((RxByteData[1] & 0x01) == 0x01) DI8 = true; else DI8 = false;
    if(((RxByteData[1] >> 1) & 0x01) == 0x01) DI9 = true; else DI9 = false;
    if(((RxByteData[1] >> 2) & 0x01) == 0x01) DI10 = true; else DI10 = false;
    if(((RxByteData[1] >> 3) & 0x01) == 0x01) DI11 = true; else DI11 = false;
    if(((RxByteData[1] >> 4) & 0x01) == 0x01) DI12 = true; else DI12 = false;
    if(((RxByteData[1] >> 5) & 0x01) == 0x01) DI13 = true; else DI13 = false;
    if(((RxByteData[1] >> 6) & 0x01) == 0x01) DI14 = true; else DI14 = false;
    if(((RxByteData[1] >> 7) & 0x01) == 0x01) DI15 = true; else DI15 = false;
}
```

```
@Override
public void UartData(int i, char[] chars) {
    int j, crc_data, crc_check;

    //client.sendUartByte(1, chars, chars.length);
    for(j=0;j< chars.length;j++)
        RxByteData[j] = chars[j];

    crc_data = uartTools.GenerateCRC(RxByteData, 9);
    crc_check = uartTools.Get2BytesData(4, RxByteData, false);
    if( crc_data == crc_check )
    {
        switch(RxByteData[0])
        {
            case 100 ://Input Status
                GetInputStatus();
                break;
```

DO 範例程式

```
void SendDOCommand(int output)
{
    int cmd, length, crc_value;
    char[] TxByteData=new char[11];
    cmd = 16;

    TxByteData[0] = (char)(output & 0x000000FF);
    TxByteData[1] = (char)((output >> 8) & 0x000000FF);
    TxByteData[2] = (char)((output >> 16) & 0x000000FF);
    TxByteData[3] = (char)((output >> 24) & 0x000000FF);

    TxByteData[4] = 0;
    TxByteData[5] = 0;
    TxByteData[6] = 0;
    TxByteData[7] = 0;
    length = 8;

    RegisterSendingCommand(cmd, TxByteData, length);
}
```

Servo Motor 範例程式

```
void SendGoCommand(int id, int position)
{
    int cmd, length, crc_value;
    char[] TxByteData=new char[11];
    cmd = 11;
    switch(id)
    {
        case 0 ://X Axis
            TxByteData[0] = 0;
            break;
        case 1 ://Y Axis
            TxByteData[0] = 1;
            break;
    }
    TxByteData[1] = 0;
    TxByteData[2] = 0;
    TxByteData[3] = 0;

    TxByteData[4] = (char)(position & 0x000000FF);
    TxByteData[5] = (char)((position >> 8) & 0x000000FF);
    TxByteData[6] = (char)((position >> 16) & 0x000000FF);
    TxByteData[7] = (char)((position >> 24) & 0x000000FF);
    length = 8;

    RegisterSendingCommand(cmd, TxByteData, length);
}
```