

图2 与步进电机的连接方法

如图 1 所示，脉冲信号和步进电机控制信号通过 J8 与采集卡相连。CLKA,CLKB,CLKC 这三路脉冲内部有 10K 的上拉电阻，并且有施密特反相器，用于给信号整形，CLKD 内部只有上拉电阻。信号接入板卡后，进入 CPLD，在 CPU 的控制下采集脉冲信息。

步进电机控制信号有 DIR 信号和 PLUSE 信号。一般步进电机驱动器包括三个信号，分别是 DIR+,DIR-控制电机的方向，CP+,CP-用于输入脉冲，EN+,EN-用于使能。将控制器的 DIR+,CP+,EN+连接在一起，然后接到 J8 的 8，将 DIR-接到 J8 的 2，将 CP-接到 J8 的 4 即可。如果外部有电源+5V，请不要使用内部 3.3V 电源，将外部+5V 接到电机驱动器的 EN+,DIR+,CP+上，将外部电源地与本板的数字地相连即可。

三、软件使用

1、步进电机控制软件

步进电机的控制通过 DLLIOCTL 函数实现，具体功能如下：

'设定电机的走步方向

```
Sub SetDir(Dir As Integer)
    Dim i As Integer '7
    InBuff(0) = 71 'Function No 功能号
    InBuff(1) = 1 '1 数据个数
    InBuff(2) = &HA6 '
    InBuff(3) = 0
    InBuff(4) = Dir And 1 '0 为正向，1 为反向
    i = DLLIOctl(100, InBuff(0), 100, OutBuff(0))
End Sub
```

'设定电机是否连续走步

```
'Flag_Countinue=1 表示连续
'Flag_Countinue=0 表示运行到设定步数后就停止
'缺省情况下是非连续模式
Sub Motor_RunCountinue(Flag_Countinue As Integer)
    Dim i As Integer '7
    InBuff(0) = 71 'Function No 功能号
```

```
InBuff(1) = 1 '1 数据个数
InBuff(2) = &HA7 'low 8 是否连续走步
InBuff(3) = 0
InBuff(4) = Flag_Countinue And 1
i = DllIOctl(100, InBuff(0), 100, OutBuff(0))
End Sub
```

'启动走步功能,电机开始走步,到走完设定步数

```
Sub MotorStep_Start(Action As Integer)
  Dim i As Integer '7
  InBuff(0) = 71 'Function No 功能号, 输出 CPLD 数据
  InBuff(1) = 1 '数据个数
  InBuff(2) = &HA8 '
  InBuff(3) = 0 'high 8
  InBuff(4) = Action '1 启动, 2: 停止
  i = DllIOctl(100, InBuff(0), 100, OutBuff(0))
End Sub
```

'设定电机的走步数

```
Sub MotorStep_set(num As Integer)
  Dim i As Integer
  If num < 1 Then Exit Sub
  If num > 1000000 Then num = 1000000
  InBuff(0) = 71 'Function No 功能号
  InBuff(1) = 3 '数据个数
  InBuff(2) = &HA3
  InBuff(3) = 0 'high 8
  InBuff(4) = num And &HFF '低 8 位数据
  InBuff(5) = Int((num And &HFF00) / 256) '中 8 位数据
  InBuff(6) = Int(num / 65536) '高 4 位数据
  i = DllIOctl(100, InBuff(0), 100, OutBuff(0))End Sub
```

'设定电机的速度

```
Sub MotorStep_Speed(num As Integer)
  Dim i As Integer
  If num < 1 Then Exit Sub
  If num > 65535 Then num = 65535
  num = num - 1
  InBuff(0) = 71 'Function No 功能号
  InBuff(1) = 2 '数据个数
  InBuff(2) = &HA0
  InBuff(3) = 0 'high 8
  InBuff(4) = num And 255 '低 8 位数据
```

```
InBuff(5) = Int(num / 65536) '高 8 位数据
i = DIIOCtl(100, InBuff(0), 100, OutBuff(0))
End Sub
```

'读取当前电机状态

```
Sub MotorStatus(Dat() As Long)
    Dim i As Integer
    InBuff(0) = 70 '命令号
    InBuff(1) = 16 '数据个数低 8 位
    InBuff(2) = 0 '偏移地址
    InBuff(3) = 0 '偏移地址
    i = DIIOCtl(100, InBuff(0), 100, OutBuff(0))
    Dat(0) = OutBuff(10) '电机已经走步步数的低 8 位
    Dat(1) = OutBuff(11) '电机已经走步步数的中 8 位
    Dat(2) = OutBuff(12) And &HF '电机已经走步步数的高 4 位
    Dat(3) = (OutBuff(12) / 16) And 1 '电机运行模式，是否是连续方式
    Dat(4) = (OutBuff(12) / 32) And 1 '电机方向
    Dat(5) = (OutBuff(12) / 64) And 1 'CLKC 的状态
    Dat(6) = (OutBuff(12) / 128) And 1 'CLKD 的状态
End Sub
```

读取电机控制的状态，可以协助用户进行点击控制，如 CLKC，CLKD 可以用于检测位置开关，当达到指定位置后，就可以启动或停止电机。检测电机已经走步数，便于判断控制的位置，这对一些开环控制提供方便。

2、数据采集软件

采集软件与标准的本公司标准的采集软件相同，但在通道设定和读取上有差别。

通道数 NumChn 必须在 8 至 39 之间变化。

通道的定义如下：

通道 0：16 位开关量输入，位 0 对应第一通道开关量输入，位 15 对应第 16 通道的开关量输入。

通道 1：CLKA 脉冲的瞬态周期，单位是微秒 us

通道 2：CLKA 脉冲的正向脉冲计数

通道 3：CLKA 脉冲的方向脉冲计数

通道 4：CLKB 脉冲计数

通道 5：CLKC 脉冲计数

通道 6：CLKD 脉冲计数

通道 7：模拟量通道 0 结果

通道 8-39：模拟量通道 1-31 的采集结果

从上面可以看出，用户可以设定通道数在 8-39 之间变化，其中脉冲信号不变，模拟量通道数可以改变。

四、带外壳方式的接线表

为了用户方便使用，本公司提供外壳，对应的接线表如下：

USB 数据采集器 RBH8251-9 接线表---脉冲型	
端子 1-16	模拟量输入：单端方式 CH0-CH15，双端方式 CH0-CH15 高端
端子 17-20	模拟量地线
端子 21-25	开关量输出：DO0-DO4
端子 26-28	26：3.3V 输出，27：方向-，28：脉冲- （特别注意：不要接错）！！！！
端子 29-30	开关量地线
端子 31-46	模拟量输入：单端方式 CH16-CH31，双端方式 CH0-CH15 低端
端子 47-50	模拟量地线
端子 51-54	开关量输入：DI0-DI3
端子 55-58	55：CLKA, 56：CLKB, 57：CLKC, 58：CLKD 脉冲输入
端子 59-60	开关量地线

JB1 的 1-16 接端子 1-16；

JB1 的 21-36 接端子的 31-46；

JB1 的 37-40 接 17-20，47-50 为模拟量地线；

JOUT 的 1-5 接 21-25

JOUT 的 19，20 接 29，30

JIN 的 1-4 接 51-54

JIN 的 19，20 接 59，60

J8 的 1，3，5，7 接端子的 55，56，57，58

J8 的 2，4，8 接端子的 28，27，26

五、编程实例

如图 3 所示，本板卡提供完善的编程实例，可以实现电机的方向切换，设定设定，方向设定，电机启动与停止等功能

在数据采集方面，可以实现波形显示、数据存盘，波形滤波显示等。

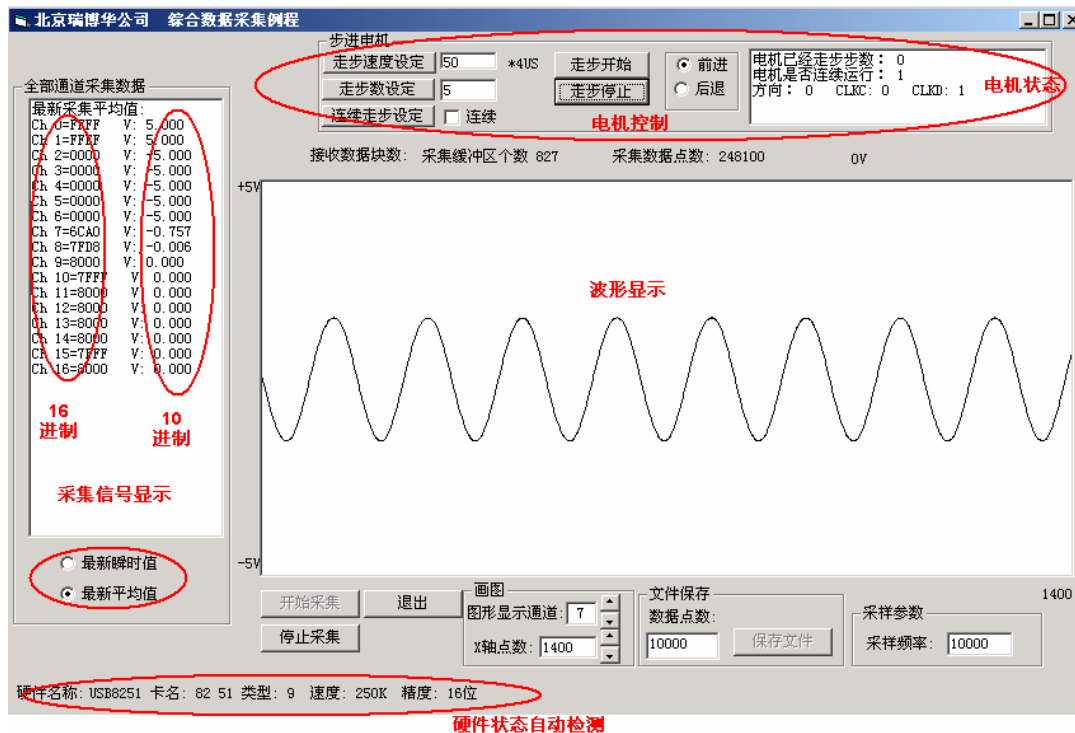


图 3 编程实例

六、注意事项

- 1、硬件接线时，不要将 3.3V 接到地线上，如果能够外部提供电源，请不要使用板上的 3.3V。
- 2、信号没有隔离，如果有条件，请添加保护。
- 3、软件设置通道时，通道数大于 7 个。