

PLC—PLC 之间的 MPI 通讯----调用系统功能的通讯方式

全局数据包通讯的组态必须是在一个项目下，非常不灵活，有一些用户喜欢调用系统功能（SFC65~69）来实现 PLC 之间的 MPI 的通讯，这种通讯方式适合于 S7-300, S7-400 和 S7-200 之间的通讯，一些非常老的 S7-300/400CPU 不含有 SFC65~69，所以不能用这种方式通讯，只能用全局数据包的方式来通讯，判断一个 CPU 是否含有通讯用的 SFC，可以在联机的情况下，在线查看所用的程序块，看一看是否包含有通讯用的 SFC65~69。通过调用 SFC 来实现的 MPI 通讯又可分为两种方式：双向通讯方式和单向通讯方式。调用系统功能通讯方式不能和全局数据通讯方式混合使用。本例中所需硬件：CPU315-2DP，CPU416-2DP。所需软件：STEP7 V5.2 SP1。

1). 双向通讯方式 在通讯的双方都需要调用通讯块，一方调用发送块，另一方就要调用接收块来接收数据。这种通讯方式适用 S7-300/400 之间通讯，发送块是 SFC65(X_SEND)，接收块是 SFC66(X_RCV)。下面我们将以举例的形式说明怎样调用系统功能来实现通讯：在 STEP7 中创建两个站 STATION1，CPU 为 S7-416，MPI 站地址为 2；STATION2，CPU 为 S7-315-2DP，MPI 站号为 4，2 号站发送 2 包数据给 4 号站，4 号站判断后放在相应的数据区中。

在 2 号站 OB35 中须调用 SFC65，如果扫描时间太短，发送频率太块，对方没有响应，将加重 CPU 的负荷，在 OB35 中调用发送块，发送任务将间隔 100MS 执行一次，编写发送程序如下：

```
CALL "X_SEND"           SFC65
  REQ      :=M1.1
  CONT     :=TRUE
  DEST_ID :=W#16#4
  REQ_ID  :=DW#16#1
  SD      :=P#DB1.DBX0.0 BYTE 76
  RET_VAL:=MW2
  BUSY    :=M1.2
```

```
CALL "X_SEND"           SFC65
  REQ      :=M1.3
  CONT     :=TRUE
  DEST_ID :=W#16#4
  REQ_ID  :=DW#16#2
  SD      :=P#DB2.DBX0.0 BYTE 76
  RET_VAL:=MW4
  BUSY    :=M1.4
```

```
CALL "X_ABORT"          SFC69
  REQ      :=M1.5
  DEST_ID :=W#16#4
  RET_VAL:=MW6
  BUSY    :=M1.6
```

参数中 REQ 为发送请求,该参数为 1 时发送。

CONT 为 1 时表示发送数据是连续的一个整体。

DEST_ID 表示对方的 MPI 地址。

REQ_ID 表示一包数据的标识符 , 标识符自己定义 , 例子中两包数据的标识符分别为 “ 1 ” , “ 2 ” 。

SD 表示发送区 , 以指针的格式表示 , 例子中第一包数据为 DB1 中从 DBX0.0 (DBB0) 以后的 76 个字节数据,发送区最大为 76 个字节。

RET_VAL 表示发送的状态

BUSY 为 1 时表示发送中止。

在这个例子中, M1.1, M1.3 为 1 时, CPU416 将发送标识符为 “ 1 ” 和 “ 2 ” 的两包数据给 4 号站的 CPU315-2DP。

用户可能会问:一个 CPU 究竟可以和能几个 CPU 通讯? 这和 CPU 的通讯资源有关系,这也决定 SFC 的调用的次数,在选项手册中,常常可以看到“动态连接”的术语,这个术语与 SFC 的调用有关,以上例作说明, M1.1, M1.3 为 1

时，与 4 号站的连接就建立起来了，反之 4 号站发送，2 号站接收同样要建立一个连接，也就是说两个站通讯时，若都需要发送和接收数据，则须占用两个动态连接。

具体可参考下图：

General	Diagnostic Buffer	Memory	Scan Cycle Time
Time System	Performance Data	Communication	Stacks
Transmission Rate			
Interface X1:		187.5 Kbps	
Connection Resources			
Maximum Number:		Not Assigned: 10	
	Reserved	Assigned	
PG Communication:	1	1	
OP Communication:	1	0	
S7 Standard Communication:	8	1	
Other Communication:	--	0	

连接建立

M1.1，M1.3 为 0 时，此时建立的连接并没有释放，必须调用 SFC69 来释放连接，在上例中 M1.5 为 1 时，与 4 号站建立的连接断开，如下图：

	Reserved	Assigned
PG Communication:	1	1
OP Communication:	1	0
S7 Standard Communication:	8	0
Other Communication:	--	0

连接释放

有的用户编写多个连接时，由于 CPU 的资源有限，而不能通讯，可以用这种方法试一试。

在 4 号站编写接收程序如下：

```

CALL "X_RCV"                                SFC66
  EN_DT :=M1.1
  RET_VAL:=MW2
  REQ_ID :=MD4
  NDA    :=M1.2
  RD     :=P#DB1.DBX0.0 BYTE 76

L      MD      4
L      DW#16#1
==D
=      M      1.3

L      MD      4
L      DW#16#2
==D
=      M      1.4

A      M      1.3
JCN    M1
CALL "BLKMOV"                                SFC20
  SRCBLK :=P#DB1.DBX0.0 BYTE 76
  RET_VAL:=MW10
  DSTBLK :=P#DB2.DBX0.0 BYTE 76
M1:   A      M      1.4
      JCN    M2

CALL "BLKMOV"                                SFC20
  SRCBLK :=P#DB1.DBX0.0 BYTE 76
  RET_VAL:=MW12
  DSTBLK :=P#DB3.DBX0.0 BYTE 76

M2:   NOP    0

```

在 OB1 中调用 SFC66 (X_RCV)，参数 EN_DT 表示接收使能，RET_VAL 表示接收状态字，REQ_ID 为接收数据包的标识符，NDA 为 1 时表示有新的数据包，为 0 时则表示没有新的数据包，RD 表示接收区，接收区放在 DB1 中从 DBB0 以后 76 个字节中。例子中，接收块只识别数据的标识符，而不管是哪一个 CPU 发送的，接收从 2 号站 CPU416 发送的两包数据，当标识符为“1”且 M1.3 为 1 时，复制接收区的数据到 DB2 的前 76 个字节中（调用 SFC20），当标识符为“2”时，M1.4 为 1，复制接收区的数据到 DB3 的前 76 个字节中。例子程序参见光盘，项目名为 MPI_SFC_SR

2). 单向通讯

与双向通讯时两方都需要编写发送和接收块不同，单向通讯只在一方编写通讯程序，这也是客户机与服务器的关系，编写程序一方的 CPU 作为客户机，

没有编写程序一方的 CPU 作为服务器，客户机调用 SFC 通讯块对服务器的数据进行读写操作，这种通讯方式适合 S7-300/400/200 之间通讯，S7-300/400 的 CPU 可以同时作为客户机和服务器，S7-200 只能作服务器。SFC67 (X_GET) 用来读回服务器指定数据区中的数据并存放本地的数据区中，SFC68 (X_PUT) 用来写本地数据区中的数据到服务器中指定的数据区中，我们以举例的方式说明怎样调用 SFC 通讯块来实现单向通讯，先介绍 S7-300/400 之间的通讯，与上例一样，建立两个 S7 站，STATION1，CPU 为 S7-416-2，MPI 地址为 2 作为客户机；STATION2，CPU 为 S7315-2DP，MPI 地址为 4 作为服务器，调用 SFC68，CPU416 把本地数据区的数据 DB1.DBB0 以后的 76 个字节存储在 CPU315 的 DB1.DBB0 以后的 76 个字节中，调用 SFC67，CPU416 读出 CPU315 的数据 DB1.DBB0 以后的 76 个字节，放到本地 DB2.DBB0 以后的 76 个字节中，例子程序如下：

```

CALL "X_PUT"                                SFC68
  REQ      :=M1.1
  CONT     :=TRUE
  DEST_ID  :=W#16#4
  VAR_ADDR:=P#DB1.DBX0.0 BYTE 76
  SD       :=P#DB1.DBX0.0 BYTE 76
  RET_VAL  :=MW2
  BUSY     :=M1.2

CALL "X_GET"                                SFC67
  REQ      :=M1.3
  CONT     :=TRUE
  DEST_ID  :=W#16#4
  VAR_ADDR:=P#DB1.DBX0.0 BYTE 76
  RET_VAL  :=MW4
  BUSY     :=M1.4
  RD       :=P#DB2.DBX0.0 BYTE 76

CALL "X_ABORT"                              SFC69
  REQ      :=M1.5
  DEST_ID  :=W#16#4
  RET_VAL  :=MW6
  BUSY     :=M1.6

```

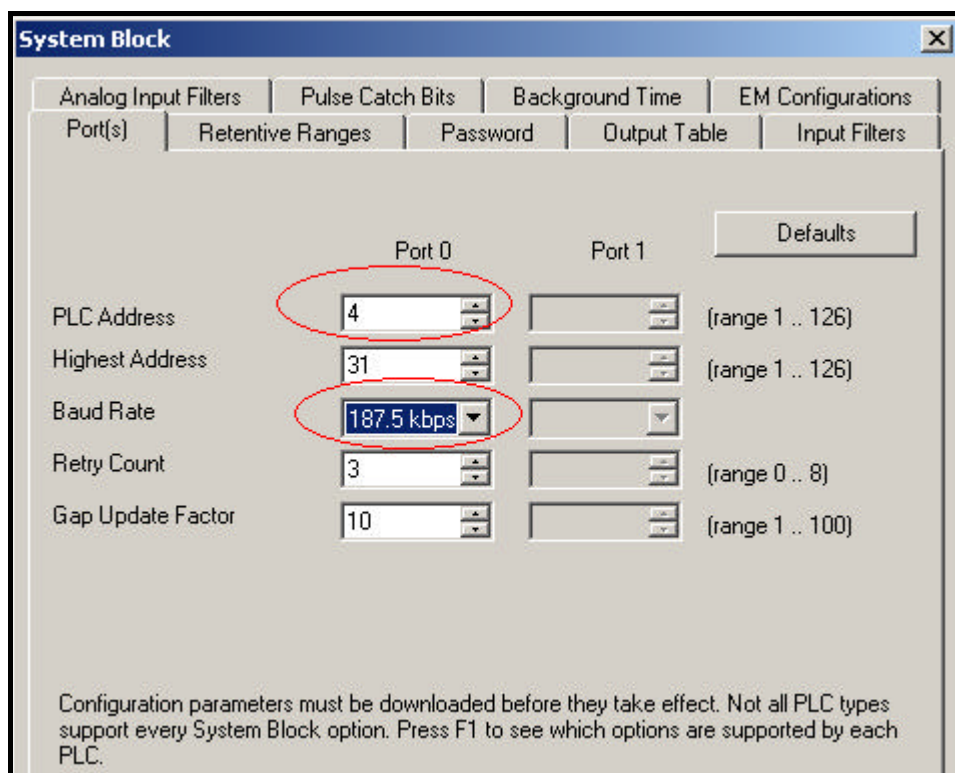
参数 DEST_ID 表示对方的 MPI 地址，VAR_ADDR 表示指定服务器的数据区，SD 表示本地数据区（数据源），当 M1.1 为 1 时，CPU416 将数据区的数据从 DB1.DBB0 以后的 76 个字节的数据存放到 CPU315 的数据区 DB1.DBB0 以后的 76 个字节中。

RD 为本地数接收区，当 M1.3 为 1 时，CPU416 将 CPU315 中从 DB1.DBB0 以后的 76 个字节中的数据存放到本地数据区 DB2.DBB0 以后的 76 个字节中。数

据区最大为 76 个字节。同时在一个 CPU 中调用 SFC67 和 SFC68 将占用一个动态连接，当 M1.5 为 1 时中断通讯并释放连接。例子程序参见光盘，项目名为 MPI_SFC_PG300。同样 S7-300CPU 也可以作为客户机，S7-400CPU 也可以作为服务器。

S7-300/400 与 S7-200 之间的通讯：大家可能知道在 S7-200 中不能调用 SFC 通讯块，只能在 S7-300/400 中调用，所以只有 S7-300/400 可以作为客户机，S7-200 只能作为服务器。下面将以举例的方式介绍实现 S7-300/400 与 S7-200 之间的通讯的过程，所需硬件：S7-300 CPU315-2DP，S7-200 CPU224 和通讯卡 CP5611；所需软件：STEP7 V5.2 SP1，MICROWIN V3.2。

首先要设定各 PLC 的 MPI 的站地址和通讯速率，S7-300 的 MPI 的站地址为 2，S7-200 的 MPI 的站地址为 4，通讯速率为 187.5K/S，在 S7-300 侧：打开 STEP7，插入一个新的 S7-300 站，组态站地址及通讯速率并下载到 CPU 中，在 S7-200 侧：打开 MICROWIN，在“SYSTEM BLOCK”中设定 S7-200 的站地址和通讯速率，具体参照下图：



然后把组态数据下载到 S7-200CPU 中，这样所有的硬件组态就完成了，接下来的工作是在 S7-300CPU 中编写通讯程序，在 OB1 中调用 SFC68 和 SFC67。如下图：

```

CALL "X_PUT"                                SFC68
  REQ      :=M1.1
  CONT     :=M1.2
  DEST_ID  :=W#16#4
  VAR_ADDR:=QB0
  SD       :=IB0
  RET_VAL  :=MW2
  BUSY     :=M1.3

CALL "X_GET"                                SFC67
  REQ      :=M1.4
  CONT     :=M1.5
  DEST_ID  :=W#16#4
  VAR_ADDR:=IB0
  RET_VAL  :=MW4
  BUSY     :=M1.6
  RD       :=QB0

```

当 M1.1 为 1 时把 S7-300 中的输入信号 IB0 发送到 S7-200 的 QB0 中，同时当 M1.4 为 1 时把 S7-200 的输入信号 IB0 发送到 S7-300 的输出信号 QB0 中。例子程序参见光盘，项目名为 MPI_SFC_200_PPI。

如果用户想与 S7-200 的 V 区通讯，在 S7-300CPU 中相对应的是 DB1，例如读出 S7-200 中 VB100 以后的 16 个字节并且放到 S7-300 的 MB20 以后的 16 个字节中，需要调用系统功能 SFC67(X_GET)，在参数 VAR_ADDR：中写入 S7-200 的地址区如：P#DB1.DBX100.0 BYTE 16。在参数 RD：中写入 S7-300 的地址区如：P#MB20.0 BYTE 16，这样就可以完成通讯了。

另外还可以通过 S7-200 的 PROFIBUS-DP 模块 EM277 与 S7-300/400 的 MPI 口通讯，其设置更为简单，在 S7-300/400 侧调用 SFC67/68，与上例参数相同，地址变成 S7-200 EM277 的 MPI 地址就可以了，在 S7-200 侧，用拨码开关设定 EM277 的站地址而不用软件下载设定，连接好以后，从新上电通讯速率就可以自动匹配。