Trio Motion Technology Ltd.
Shannon Way, Tewkesbury,
Gloucestershire. GL20 8ND
United Kingdom
Tel: +44 (0)1684 292 333
Fax: +44 (0)1684 297 929

1000 Gamma Drive
Suite 206
Pittsburgh, PA 15238
United States of America
Tel: +1 412 968 9744
Fax: +1 412 968 9746

B1602 Tomson Centre
188 Zhang Yang Rd.,
Pudong New Area, Shanghai,
Postal code: 200122
CHINA
Tel/Fax: +86 21 587 97659

Doc No.: AN-255
Version: 1.0
Date: 27 April 2010
Subject: How to set up a RS422 Hostlink Network

APPLICATION NOTE

1. Introduction

Hostlink is a serial communication protocol used by many devices. It was developed by Omron for communication between their PLCs, HMIs and other peripherals. Hostlink is a standard built-in protocol for most Trio Motion Coordinators. It allows the Hostlink Master to exchange data with the VR global variables and the TABLE memory of the Slaves.

This document describes how to set up and wire a Hostlink network with one Master Motion Coordinator and one or more Slave Motion Coordinators.

2. RS422 network wiring

RS422 is a serial data communications standard which uses differential signals in a 4 wire configuration. The use of differential line drivers and receivers makes for improved noise immunity and thus reduces data corruption. 4 wires are split into 2 pairs. One pair is for the master to send commands to the slaves. The second pair returns the data from the slave to the master. All Slaves are connected to the return pair of wires, so they go to a high resistance state when not transmitting. In this way the slaves share the same wires to make a network.

RS422 is really a 5 wire connection. It is vital to connect the 0V with a wire included in the cable. Do not use the screen of the cable as the 0V. Usually this means that the cable must have 3 pairs. 2 pairs are for the data and one wire of the third pair for 0V.

2.1. Physical construction

Each Motion Coordinator has one RS422/RS485 port on an 8-pin mini-DIN socket.

The 0V of each Motion Coordinator MUST be connected to a common 0V wire in the cable.

The cable must be shielded, twisted pair. Each pair separately shielded and with a nominal characteristic impedance of 120 Ohms.

The optional terminating resistors must be placed at the master end of the network and the 5V connected ONLY to the master +5V (pin 6).

4-Wire Rs422 with Termination network

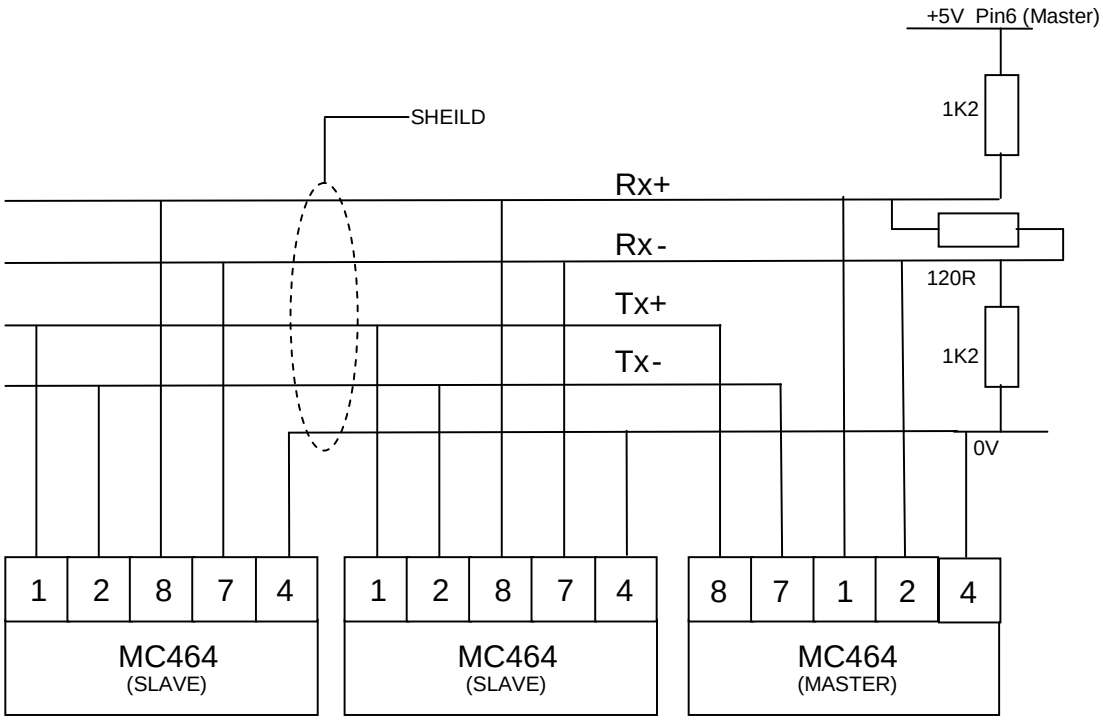

Figure 1 – Master and slave wiring

3. Configuring the Slaves

Each slave Motion Coordinator must be set up with a short BASIC program set to run on power-up.

```
baud_rate = 9600
data_bits = 7
stop_bits = 2
parity = 2 ' Even parity
option =5 ' Host Link Slave
SETCOM(baud_rate,data_bits,stop_bits,parity,2,option)

' Host Link Slave settings
'-----------------------------
HLS_NODE=1
HLS_MODEL=48
```

The first slave should be set as HLS_NODE=0, the second one as HLS_NODE=1, third as HLS_NODE=2 etc.

4. Configuring the Master

The Master is configured with a single SETCOM command like this:

```
' set up host link master for PORT 2
SETCOM(9600,7,2,2,2,6)
```

After configuration there are various Hoslink Master commands that can be used in the BASIC. For example: HLM_READ and HLM_WRITE.

5. Transferring data

The Hostlink commands are described in the Motion Coordinator Technical Reference Manual. See Chapter 8, section "system parameters and commands". There follows some examples of reading and writing data.

5.1. Read 2 VRs from slave at node 0

This reads 2 16 bit words from VR(2) and VR(3) from the slave and puts the data into VR(0) and VR(1) of the master.

```
' HLM_READ(port,node,pc_area,pc_offset,length,mc_area,mc_offset)
' source ADDRESS: VR(2)
' amount of data: 2 words
' destination ADDRESS: VR(0)
HLM_READ(2,0,PLC_IR,2,2,MC_VR,0)
```

5.2. Read 4 VRs from slave at node 1

4 x 16 bit words are read from VR(2) to VR(5) in slave 1 and put into VR(100) to VR(103) in the master.

```
' Node 1
' source ADDRESS: VR(2)
' amount of data: 4 words
' destination ADDRESS: VR(100)
HLM_READ(2,1,PLC_IR,2,4,MC_VR,100)
```

5.3. Read 4 VRs from slave at node 5

30 x 16 bit words are read from TABLE(22) to TABLE(25) in slave 1 and put into VR(1200) to VR(1203) in the master.

```
' source ADDRESS: TABLE(22)
' amount of data: 30 words
' destination ADDRESS: VR(2)
HLM_READ(2,5,PLC_DM,22,40,MC_VR,1200)
```

5.4. Write 10 values from TABLE in master to TABLE in slave node 0

10 x 16 bit words from TABLE(18) to TABLE(27) in the master are written to the slave 0, into TABLE(14) to TABLE(23)

```
' Source address: Table(18)
' Amount of data: 2 words
' Destination address: TABLE(14)
HLM_WRITE(2,0,PLC_DM,14,10,MC_TABLE,18)
```

6. Error checking

The Hostlink protocol has some error checking. HLM_STATUS returns the value of the error number if there is an error, or 0 if there is no error.

This example shows a typical error routine.

```
…..
' Source address: Table(18)
' Amount of data: 2 words
' Destination address: TABLE(14)
HLM_WRITE(2,0,PLC_DM,14,2,MC_TABLE,18)
IF HLM_STATUS<>0 THEN GOSUB report_status
…..


report_status:
  ' decodes the HLM_STATUS word and prints to terminal
  hst = HLM_STATUS
  IF hst AND $200 THEN
    PRINT #term,"HLM: Command not Recognised"
  ENDIF
  IF hst AND $100 THEN
    PRINT #term,"HLM: Timeout error"
  ENDIF
  hst=hst AND $ff
  IF hst=0 THEN
    PRINT #term,"HLM: Normal completion"
  ELSEIF hst=1 THEN
    PRINT #term,"HLM: Not executable in RUN mode"
  ELSEIF hst=13 THEN
    PRINT #term,"HLM: FCS error"
  ELSEIF hst=14 THEN
    PRINT #term,"HLM: Format error"
  ELSEIF hst=15 THEN
    PRINT #term,"HLM: Entry number data error"
  ELSEIF hst=18 THEN
    PRINT #term,"HLM: Frame length error"
  ELSEIF hst=19 THEN
    PRINT #term,"HLM: Not executable"
  ELSEIF hst=21 THEN
    PRINT #term,"HLM: CPU error"
  ELSE
    PRINT #term,"HLM: Unknown error"
  ENDIF
RETURN
```