

UMD Driver

UMDDRV.exe

Written by Steve Leatherbarrow

Contents

Contents.....	2
1 Introduction	3
1.1 Overview	3
2 Setup.....	3
2.1 Overview.....	3
2.2 Com Port Settings.....	3
2.3 BNCS configuration.....	3
2.4 .INI File Settings.....	3
3 Operation.....	4
3.1 Slot allocation.....	4
3.2 Command characteristics	5
3.3 Limitations	5
3.4 Using Driver in Virtual Mode.....	5
3.5 Resilience and redundancy	5
4 Version history.....	6
4.1 Driver version.....	6
4.2 Document Version	6

1 Introduction

1.1 Overview

The control of UMDs was historically achieved by using Applcore. This driver now gives control via a standard driver. The later versions of the driver have a UMD virtual mode and simple resilience.

2 Setup

2.1 Overview

The driver works by connecting to an External Infodriver. The start-up parameter is the device number of the Infodriver.

2.2 Com Port Settings

Serial connection to each device is RS422/485.
UMDDRV.EXE uses the following com port parameters

38400 baud

8 bits

1 stop bit

Even parity

2.3 BNCS configuration

This driver is compatible with configuration paths:

- V4.5 environmental variables
- Path set in C:\bncs_config.ini
- C:\Windows

2.4 .INI File Settings

The following settings are placed in the relevant DEV_XXX.INI file if they are not present and can be used as defaults

[UMDDRV]

DebugMode=1

LogMode=0

Port=1

Speed=38400
DataBits=8
StopBits=1
Parity=E
RefreshStart=1
RefreshEnd=128
UMDTYPE=TSL
UMDVirtualisation=0
ProtocolVersion = 3.1 or ProtocolVersion = 4.0

Most of the settings are self explanatory. RefreshStart and RefreshEnd are the start and end infodriver slots to which UMDs are connected. These are used for the background updates to be sent via the serial port to the devices (UMDs). If in non-virtual mode, UMD address is same as the slot to which it is connected i.e., there is one-to one mapping. The minimum value of RefreshStart is '1' because the infodriver slots start from '1'. The UMD type describes the manufacturer of UMDs that are connected to the chain. UMD virtualisation sets an expansion mode to accommodate dual display UMDs so that each half of the display is set through its own slot.

The driver is now compatible with both firmware versions, 3.1 and 4.0. The driver can be made to work with any of these protocols with a key/value combination in the ini file - ProtocolVersion=3.1 (e.g. in SCAR) or ProtocolVersion=4.0 (e.g. in TC4). The protocol defaults to firmware version 3.1, if not explicitly specified.

If using an additional instance of the driver for resilience, the changeover in event of UMD driver closure or failure can be assisted by the following host Infodriver settings:

[Device]
DefaultDriverMode=EXTERNAL
GoRxOnlyOnExternalFail=1

3 Operation

3.1 Slot allocation

The following lists the slot allocation

Slot Index	Read / Write	Description
1 – 256	RW	Text
501 – 756	RW	Cue, Brightness

1001 – 1256	RW	Cue 0 = off 1 = left cue on 2 = right cue on 3 = both cues on
1501 - 1756	RW	Brightness 0 = off 1 = 1/7 th brightness 2 = 1/2brightness 3 = full brightness

3.2 Command characteristics

Cue and Brightness slots cross update each other so that there is a full choice on entering cue and brightness or just cue or just brightness.

If virtualisation is not enabled then the slot ranges are limited to 128 as this is the maximum amount of physical UMDs allowed on the chain.

In virtualisation mode, slots 1 and 2 are concatenated to the first UMD, slots 3 and 4 for the second and so on.

3.3 Limitations

Version 1.00.02.00 and later supports only TSL protocol.

3.4 Using Driver in Virtual Mode

If the driver is used with Dual Dynamic Displays, the virtual mode should be enabled.

Everything remains the same except that the information to be sent to UMD consists of the contents of two consecutive slots (starting with info-driver slots 1 and 2). The slot contents are concatenated and the CUE (Tally 1 and 2 for left hand display and Tally 3 and 4 for right hand side) values sent as combined 'one-packet' to the UMD.

Dual 8 character displays are treated as a single display of 16 characters, the first 8 characters for the left-hand side and the second 8 characters for the right-hand side. Tallies 1&2 are for the left display and tallies 3&4 for the right display. They have one physical address.

Though each half has got the same physical address, the UMD driver treats them as separate entities. The text, CUE and brightness should be set as it would be done for a single 16-character UMD.

3.5 Resilience and redundancy

The driver supports Tx/Rx switching. UMD hardware generally has only a single comms port, an additional instance of this driver can be run on a spare workstation and the control cables will need to be manually swapped or connected via an external changeover switch.

The UMD protocol is unidirectional so the driver does not receive any acknowledgment from the hardware to assist with fail detection. The driver, irrespective of its interface mode, will continue to process network commands, update its host infodriver and send updates to the serial connection; the only difference is that network reverts are not issued by an RxOnly driver.

If the Tx/Rx driver closes or disappears from the network CSI will manage the promotion of the RxOnly driver. Setting the Infodriver options for 'DefaultDriverMode=EXTERNAL' and 'GoRxOnlyOnExternalFail=1' will assist the changeover.

4 Version history

4.1 Driver version

Version	Date	Details	Name
1.00.00	Aug 2004	Initial development	Steve Leatherbarrow
1.00.01	15.04.04	First release for W12 project – TSL Protocol only	Steve Leatherbarrow
1.00.02	09.12.08	Adds UMD Virtual mode	Kunal Dutta
1.00.03	23.11.09	Modifications for UMD Virtual mode	Kunal Dutta
1.00.04	04.08.11	Adds V3 configuration paths	Atul Arora
1.00.05	19.03.12	Adds simple resilience functionality	Steve Lowe
2.00.00	21.06.2012	Merges two versions of the driver that had been separately worked on. Includes provision for v4.1 firmware.	Atul Arora

4.2 Document Version

Version	Remarks	Date
0.1	First draft for comment	08.07.04
0.2	Changes for UMD Virtual mode	23.11.09
0.3	Adds use of bnccs_config.ini for V3 compatibility	04.08.11
0.4	Resilience and branding updates	19.03.12
0.5	Updates for the combined version to include both v3.1 and v4.1 versions of firmware	05.07.12

Atos IT Services UK Limited
4 Triton Square
Regents Place
London NW1 3HG