ANALOG WAY MIDRA

INTRODUCTION

AMX NETLINX

Date: **October 16, 2017**
Driver version: **V1.11**
Tested with: **Midra Firmware v02.00.15**

INTRODUCTION

This document describes the driver interface provided between an AMX NetLinx system and a MIDRA series switcher (IP protocol only).

The package provided includes:

- *.tko library files
- *.axi file for parameters definition
- A Netlinx Studio project sample
- A TPDesign project sample for iPad and a TPDesign sample project for AMX MST-701 touch panel
- Modules documentation

Analog Way Worldwide
Europe, Middle East & Africa: +33 (0)1 81 89 08 60
The Americas: +1 212 269 1902
Asia Pacific: +65 6292 5800
www.analogway.com

USING SAMPLE PROJECT

1 - Launch the AMX NetLinx Studio program then connect the AMX controller (see menu **Settings** / **Master Communication Settings** menu). For more information about this topic, read the corresponding AMX documentation.

2 - Load the Midra.apw project (located in the provided package **Driver** directory) then edit the file Midra_User_Definitions.axi to change the default Midra IP address with the configured Midra switcher IP address. Compile the sample project then transfer it to the AMX controller (see menu **Tools** / **File Transfer**). If successful, the controller automatically reboots and runs the sample program.

3 - Load the TPDesign sample project. Two projects are available, one for iPad, one for an AMX 16/9 touch panel (both located in the provided package **Panels** directory). Once the selected file loaded, transfer it to the panel (see menu **Transfer** / **Send to Panel**). **Important**: In order to transfer the project to an iPad device, it is mandatory to use the AMX TPTransfer program (and not TPDesign).

IMPLEMENTATION

In order to use this driver in any Midra AMX program, the programmer must perform the following tasks:

- Include both the Midra_User_Definitions.axi and Midra_Definitions.axi files in the application project. These two files must appear before the declaration of the driver modules themselves.
- Open the Midra_User_Definitions.axi file and configure the IP address and the port of the Midra machine as well as the different device numbers (touch panel, Midra machine, AMX controller ...). For each module used in the program, you need to assign the value 1 to the corresponding variable Midra_*Module*_Usage (*Module* is the module name). If an optional module is not used, the value of this variable must remain at 0. In order to avoid overloading the processor, it is strongly advised not to 'load' any optional modules that will never be used by the main program.
- DO NOT CHANGE the file Midra Definitions.axi.
- Include all necessary driver modules in the main program project (see sample program available with this package) :

  - The Midra_Proc_Com module is **required** (communication module)
  - The Midra_General module is **required** (core module)
  - The Midra_Video_Inputs module is optional. Required to change connector input plugs
  - The Midra_Audio module is optional. Required to control audio inputs/outputs
  - The Midra_Frame_Logo module is optional. Required to retrieve frames and logos properties (availability, dimensions ...)
  - The Midra_Screen module is optional. Required to access Midra screen information. This module must be implemented as many times as the number of screens configured and controlled on the Midra swicther.
  - The Midra_Screen_Presets module is optional. Required to recall presets on the Main or on the Preview of a given screen. This module must be implemented as many times as the number of screens configured and controlled on the Midra swicther.

USER VARIABLES

Some driver variables can be used in the main program for getting feedback.

X is screen number (1 to 2)
Y is layer number (1 to 8)
Z is video input (1 to 10)

Screen**X**_Main_Layers_Source[**Y**]	Integer array	Main source number (0 to 10) See table below
Screen**X**_Preview_Layers_Source[**Y**]	Integer array	Preview source number (0 to 10) See table below
Active_Video_Mode	Integer	Switcher Mode 1 : Mixer 2 : Matrix 3 : Quadravision
In_Active_Plug_FB[**Z**]	Integer array	Active plug for input Z 0 : analog (HD15) 1 : DVI 2 : SDI 3 : HDMI 4 : HDBaseT

Sources (depending on layer type)

0	No input
1	Input or Frame 1
2	Input or Frame 2
3	Input or Frame 3
4	Input or Frame 4
5	Input or Frame 5
6	Input or Frame 6
7	Input or Frame 7
8	Input or Frame 8
9	Input or Frame 9
10	Input or Frame 10
11	Color (or Black) fill the PiP