Synapse

# TSX30-20

Triple/Dual-channel-enhanced-TS/ASI-monitor

## Installation and Operation manual

Powered by LINUX

DESIGNED FOR ASI/DVB

AXON THE HEART OF BROADCAST

Synapse

# TECHNICAL MANUAL

TSX30-20

**Hercules 28**

**NL-5126 RK Gilze**

**The Netherlands**

**Phone: +31 161 850 450**

**Fax: +31 161 850 499**

**E-mail: Info@axon.tv**

**Web: www.axon.tv**

**WARNING:** TO REDUCE THE RISK OF FIRE OR ELECTRICAL SHOCK, DO NOT EXPOSE THIS APPLIANCE TO RAIN OR MOISTURE

● ALWAYS disconnect your entire system from the AC mains before cleaning any component. The product frame (SFR18, SFR08 or SFR04) must be terminated with three-conductor AC mains power cord that includes an earth ground connection. To prevent shock hazard, all three connections must always be used.
● NEVER use flammable or combustible chemicals for cleaning components.
● NEVER operate this product if any cover is removed.
● NEVER wet the inside of this product with any liquid.
● NEVER pour or spill liquids directly onto this unit.
● NEVER block airflow through ventilation slots.
● NEVER bypass any fuse.
● NEVER replace any fuse with a value or type other than those specified.
● NEVER attempt to repair this product. If a problem occurs, contact your local Axon distributor.
● NEVER expose this product to extremely high or low temperatures.
● NEVER operate this product in an explosive atmosphere.

**Warranty:** Axon warrants their products according to the warranty policy as described in the general terms. That means that Axon Digital Design BV can only warrant the products as long as the serial numbers are not removed.



Date created: 08-06-2016
Date last revised: 16-04-2018



This product complies with the requirements of the product family standards for audio, video, audio-visual entertainment lighting control apparatus for professional use as mentioned below.

| | |
|---|---|
| EN60950 | Safety |
| EN55103-1: 1996 | Emission |
| EN55103-2: 1996 | Immunity |

Axon Digital Design TSX20 TSX30

Tested To Comply With FCC Standards

FOR HOME OR OFFICE USE

This device complies with part 15 of the FCC Rules Operation is subject to the following two conditions: (1) This device may cause harmful interference, and (2) This device must accept any interference received, including interference that may cause undesired operation.

# Table of Contents

# 1 Introduction to Synapse

### An Introduction to Synapse

Synapse is a modular system designed for the broadcast industry. High density, intuitive operation and high quality processing are key features of this system. Synapse offers a full range of converters and processing modules. Please visit the AXON Digital Design Website at www.axon.tv to obtain the latest information on our new products and updates.

### Local Control Panel

The local control panel gives access to all adjustable parameters and provides status information for any of the cards in the Synapse frame, including the Synapse rack controller. The local control panel is also used to back-up and restore card settings. Please refer to the rack controller manuals for a detailed description of the local control panel, the way to set-up remote control over IP and for frame related settings and status information.

### Remote Control Capabilities

The remote control options are explained in the rack controller (RRC, RRS, ERC or ERS) manual. The method of connecting to a computer using Ethernet is also described in these manuals.

!

**CHECK-OUT:** "AXON CORTEX" SOFTWARE WILL INCREASE SYSTEM FLEXIBILITY OF ONE OR MORE SYNAPSE FRAMES

Although not required to use Cortex with a Synapse frame, you are strongly advised to use a remote personal computer or laptop PC with Axon Cortex installed, as this increases the ease of use and understanding of the modules.

# 2 Unpacking and Placement

## Unpacking

The Axon Synapse card must be unpacked in an anti-static environment. Care must be taken NOT to touch components on the card – always handle the card carefully by the edges. The card must be stored and shipped in anti-static packaging. Ensuring that these precautions are followed will prevent premature failure of components mounted on the board.

## Placing the card

The Synapse card can be placed vertically in an SFR18 frame or horizontally in an SFR04, SFR08 and SFR Mobile frame. Locate the two guide slots to be used, slide in the mounted circuit board, and push it firmly to locate the connectors.

Correct insertion of card is essential as a card that is not located properly may show valid indicators, but not function correctly.

**Note**

On power up all LED’s will light for a few seconds, this is the time it takes to initialize the card

# 3 A Quick Start

## When powering-up

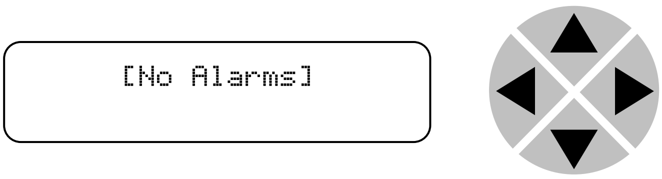
On powering up the Synapse frame, the card set will use basic data and default initialisation settings. All LED's will light during this process. After initialisation, several LED's will remain lit – the exact number and configuration is dependent upon the number of inputs connected and the status of the inputs.

## Changing settings and parameters

The front panel controls or the Synapse Cortex can be used to change settings and view status. A graphical view of the settings and status elements is described in chapters 5 and 6 of this manual. A detailed listing can be found in chapters 7, 8 and 9.

## Front Panel Control

Front Panel Display and Cursor




Settings are displayed and changed as follows;

Use the cursor 'arrows' on the front panel to select the menu and parameter to be displayed and/or changed.

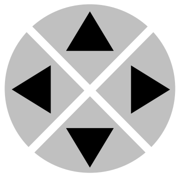
| | |
|---|---|
| Press ► | To go forward through the menu structure. |
| Press ◄ | To go back through the menu structure. |
| Press ▲ | To move up within a menu or increase the value of a parameter. |
| Press ▼ | To move down through a menu or decrease the value of a parameter. |

**Note** Whilst editing a setting, pressing ► twice will reset the value to its default

## Example of changing parameters using front panel control

With the display as shown below

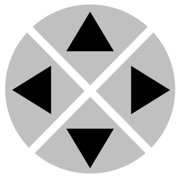
```
RRC18 [Select Card]
>S01=SFS10
```



Pressing the ► selects the SFS10 in frame slot 01.

The display changes to indicate that the SFS10 has been selected. In this example, the `Settings` menu item is indicated.

```
SFS10 [Select Menu]
>Settings
```

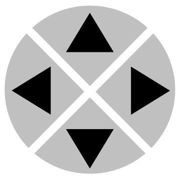


Pressing the ► selects the menu item shown, in this example `Settings`.

(Pressing ▲ or ▼ will change to a different menu e.g. `Status`, `Events`).

The display changes to indicate that the SFS10 `Settings` menu item `SDI-Format` has been selected and shows that it current setting is `Auto`.

```
SFS10 [Settings]
>SDI-Format=Auto
```

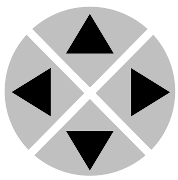


Pressing the ► selects the settings item shown, in this example `SDI-Format`.

(Pressing ▲ or ▼ will change to a different setting, e.g. `Mode`, `H-Delay`).

The display changes to indicate that the SFS10 `Edit Setting` menu item `SDI-Format` has been selected.

```
SFS10 Edit Setting]
SDI-Format>Auto
```



To edit the setting of the menu item press ▲ or ▼.

All menu items can be monitored and/or changed in this way. Changing a setting has an immediate effect.

## Axon Cortex

Synapse Cortex can be used to change the settings of Synapse

## Software

modules from a PC, either locally or remotely. The software enables communication based on TCP/IP between the Setup PC and Synapse frames/modules.

Each Synapse frame is addressed through its rack controller’s unique IP address, giving access to each module, its menus and adjustment items. Axon Cortex has access to data contained within the Synapse module and displays it on a GUI. The software has an intuitive structure following that of the module that it is controlling.

For details of the TSX30-20 GUI, see the User Guide & GUI Reference chapters of this manual.

For general operation of Axon Cortex, please refer to the Cortex help files.

## Menu Structure Example

| Slot | Module | | Item | | Parameter | | Setting |
|---|---|---|---|---|---|---|---|
| ▲ | | | | | | | |
| ▲ | | | | | | | |
| S02 | | | Identity | | | | |
| ▲ | | | ▲ | | | | |
| S01 | SFS10 | ► | Set-tings | ► | Standard_dig | ► | Auto |
| ▼ | | | ▼ | | ▼ | | ▼ |
| S00 | RRC18 | | Status | | Mode | | 625 |
| | | | ▼ | | ▼ | | ▼ |
| | | | Events | | Ref-Input | | 525 |
| | | | | | ▼ | | |
| | | | | | H-Delay | | |
| | | | | | ▼ | | |
| | | | | | ▼ | | |

**Note** Further information about Front Panel Control and Synapse Cortex can be obtained from the RRC, RRS, ERC and ERS operational manuals and the Cortex help files

# 4 The TSX30 & TSX20 Modules

## Introduction

The TSX30 & TSX20 modules are designed for real-time, in-line monitoring and switching of MPEG-2 Transport Streams on Asynchronous Serial Interface inputs. Intended for deployment at headends, they pass live broadcast signals and provide monitoring and continuity in the case of input failure by means of redundancy switching. Headends may be at content provider, telco, satellite uplink, cable or terrestrial broadcast sites.

The TSX30 is a triple channel TS/ASI integrity checker. The TSX20 is a dual input version. Both modules share a common hardware platform. A TSX20 is firmware upgradeable to a TSX30. The TSX20 has the same output options as the TSX30 (only the number of inputs is limited).

The hardware can be fitted with 2 SFP modules that can be used for fibre I/O or additional coaxial outputs. Up to 3 fibre inputs can be enabled. A combination of 2 fibre inputs and 2 fibre outputs is also possible. See the Block Schematic below for details.

## Features

**Standard Signal I/O**

- ASI Transport Stream (coaxial)
- 3 logical inputs with 3 analysis cores (TSX30)
- 2 logical inputs with 2 analysis cores (TSX20)
- 3 outputs (2 x main, 1 x monitoring)

**Optional Signal I/O**

- ASI Transport Stream (fibre)
- Extra outputs (coaxial / fibre)
- Logical inputs are independently configurable for coax or fibre

**Switching Features**

- Two independent switches - MAIN and MONITORING
- MAIN switch uses auto-switching, and/or external control
- MON switch can follow MAIN or use independent control
- Switching of MAIN is near-seamless, preserving TS sync

**Auto-Switching Features**

- Innovative auto-switching algorithm distinguishes CRITICAL FAILURE from PARTIAL FAILURE of each input
- PARTIAL FAILURE definition is configurable from palette of switchable alarms; additional monitoring-only alarms are available
- Auto-switching can be set to prefer lower number inputs, or to minimise switching
- Auto-switching may be configured to avoid input 3 unless others critical (TSX30)
- Auto-switching is near-seamless, blocking TS_sync_loss if a good input is available

- Optional null TS output on loss of all ins (10 Mbps, 188/204 mode)

**Monitoring Features**

*Critical sub-alarms are marked * below. Switchable sub-alarms are highlighted in **bold**. Items not in bold are monitoring-only.*

- Non-critical sub-alarms that are switchable *(i.e.* ***bold****, without *)* have the option of driving PARTIAL FAILURE input status, or being switched off. Alternatively, they may be set to PROBE, which allows them to report their state without affecting the input status or the switch.
- The initial condition of multi-second test results after TS_sync_loss may be specified (GUPI/IUPG).

ASI Datalink Monitoring (per input)

| | |
|---|---|
| **ASI Link*** | > nine ASI errors /s |
| ASI Error | 8b10b violation / running disparity error |
| ASI Mode | Empty, Byte, Burst, Packet |
| ASI Min Stuffing Error | < two k28.5 chars between TS packets |
| Packet Periodicity w/history | Min, Max Sync Byte Spacing |
| Byte Periodicity w/history | Min Byte Spacing, Max Burst Len |

Transport Stream Monitoring (per input)

| | |
|---|---|
| **TS Stopped*** | |
| **TS Sync Loss*** | |
| **PAT Upper Distance*** | configurable |
| **TS Mode** | 188, 204 |
| **Transport Stream ID** | |
| **TS Rate** | customisable hi and lo threshold alarms |
| **Data Rate** | customisable hi and lo threshold alarms |
| **Table Fail** | 64 fully configurable section detection tests, shared between inputs |
| **PID Fail** | 64 configurable PID detection tests in 4 upper distance groups, shared between i/ps |
| **Total PID/s** | customisable hi and lo threshold alarms - legacy feature |
| Network ID | |
| Null / Data Ratio | |
| Sync_byte_error | |
| Transport_error | |

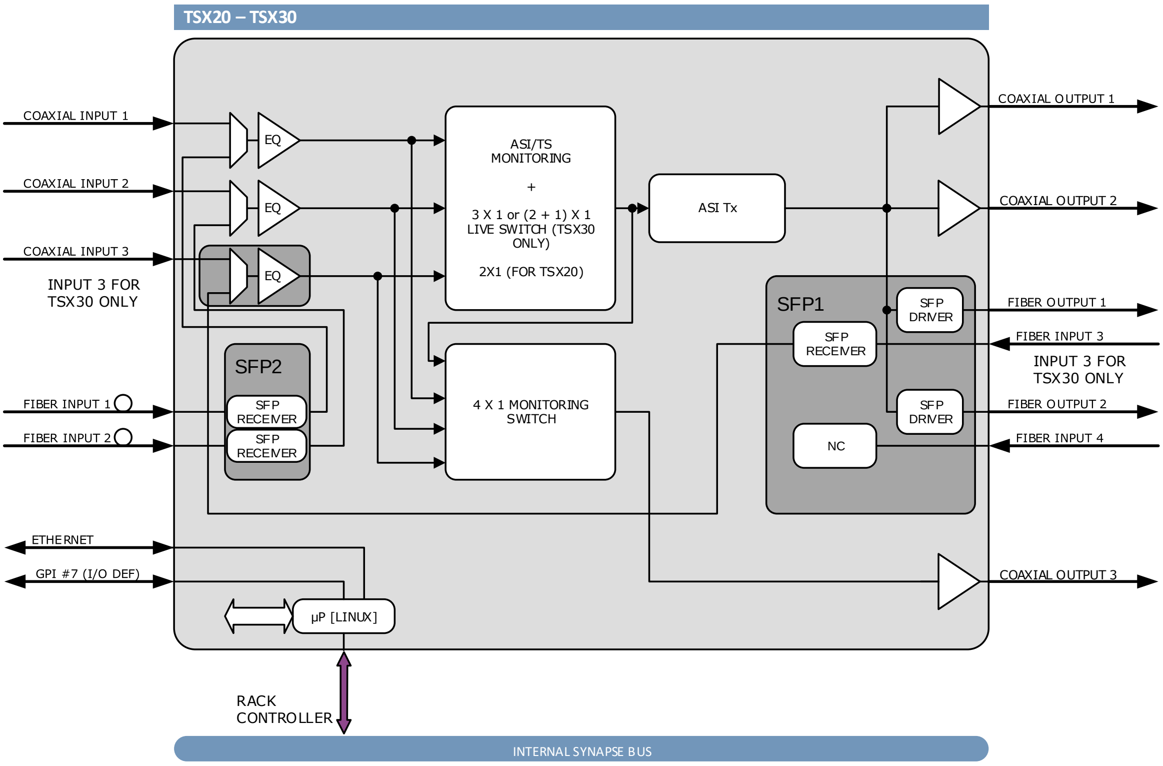
**Flexible Interfacing**

- Full control and status monitoring through the front panel of the SFR04/SFR08/SFR18 frame and the Ethernet port (ACP)
- Supported by Axon Cortex & Cerebrum remote control software
- SNMP interfacing supported with the correct frame controller
- Basic Control & Monitoring over GPIO
- SFP2 provides fibre in for inputs 1 and 2
- SFP1 provides fibre input 3, or extra MAIN outputs (fibre/coax)

## Applications

- DVB TS/ASI integrity monitoring and backup switching
- Ensuring continuity of service via autonomous automatic switching
- Redundancy switching at playout centres, headends and encoding / multiplexing systems
- Input monitoring and switching at DVB-T and DVB-T2 transmitter sites
- Input checking after IP/ASI boundaries
- Mixing input distribution system types (dedicated fibre, satellite, TS extracted from IP etc.)
- Prioritising input distribution systems according to reliability
- Use of public internet as tertiary backup, if converted to ASI
- TS_sync_loss blocking for systems with long recovery times
- Support for non-DVB Transport Streams through custom PID & table tests
- PSI/SI/PSIP verification after Service Information Processing stages
- Detection of custom / unreferenced elementary streams by PID
- Detailed detection of tables down to individual section level
- Detection of custom payloads with private table syntax
- Providing audit trail through comprehensive error logging

## Block schematic




Note: Fibre Input 4 may exist if a dual input fibre SFP module is inserted. However, it is not connected to the module and will be ignored.

# 5 User Guide

## Introduction

This chapter introduces the concepts needed to successfully deploy and utilise the TSX modules covered by this manual. It is task-orientated and progressive, like a tutorial. (For a getting started guide on the modules and the rack, see the Quick Start chapter.)

The operation of the modules is best understood by studying the GUI, so we begin with a brief guide to setting up and using the remote control software, before diving into the signal and device concepts. The same GUI is offered for both Cortex and Cerebrum editions of the Axon control software. Reference material on the GUI is included in the next chapter, and later chapters of the manual act as a reference for interfacing with the modules by other means, such as over SNMP.

## Installing Cortex / Cerebrum

Cortex and Cerebrum are available from the Axon website. Please refer to the detailed installation instructions that come with the download. These instructions are available as .htm files, once you unpack the software. In addition, note the following:

- The SQL part of the installation may fail if the path to the installation files is too long on your system. Depending on the length of your user name, this may mean that you cannot simply unpack it to the desktop and run it. For best results, move and/or rename the unpacked folder to shorten this path before running setup.exe
- Remember to run the setup.exe as Administrator
- Remember to allow communication through the firewall of the host

## Updating the device GUI

After installation, to ensure you have the latest GUI for these products, please update your software installation with the latest clf files. To obtain the clf files, search “TSX” on the Axon website, and download the latest firmware for the device. The clf files are included in the firmware zip.

- Remove any old files or subfolders with names of the form TSX30* or TSX20* from the .\Forms\Device folder of your software installation
- Copy in the latest versions

**Note** If you have version 1.12 or earlier of Cortex, Cerebrum or the Cortex Forms Library, you will need to update your .clf files.

The GUI comprises the following files: TSX20.clf, TSX30.clf and the TSX30 sub-folder, which contains the remaining forms. Contact Axon support if you are unable to locate the latest files.

## Ethernet connectivity

There are two options for providing a connection from your control PC or server to the card:

- via the rack controller using its rear panel Ethernet connector

  and /or

- directly, using the rear panel Ethernet connector of the card

Use the rack controller connection to manage multiple cards in the frame. The IP address of the rack controller should be set appropriately using the front panel controls of the frame to be reachable from the machine hosting the control software. Having done this, the frame should automatically appear under Network Devices in the System View of your control software. Various types of rack controller are available.

A direct connection is normally reserved for performing firmware updates and requires that the card itself be manually added as an ACP device in the control software. This requires knowledge of the card's IP address, which can be statically set, but you do this using the rack controller. Therefore, set up a physical connection to the rack controller first. Unless you plan to update the firmware, such a connection will normally suffice.

The remainder of this guide assumes the control software is up and running and you have successful communications with the card.

## Software overview

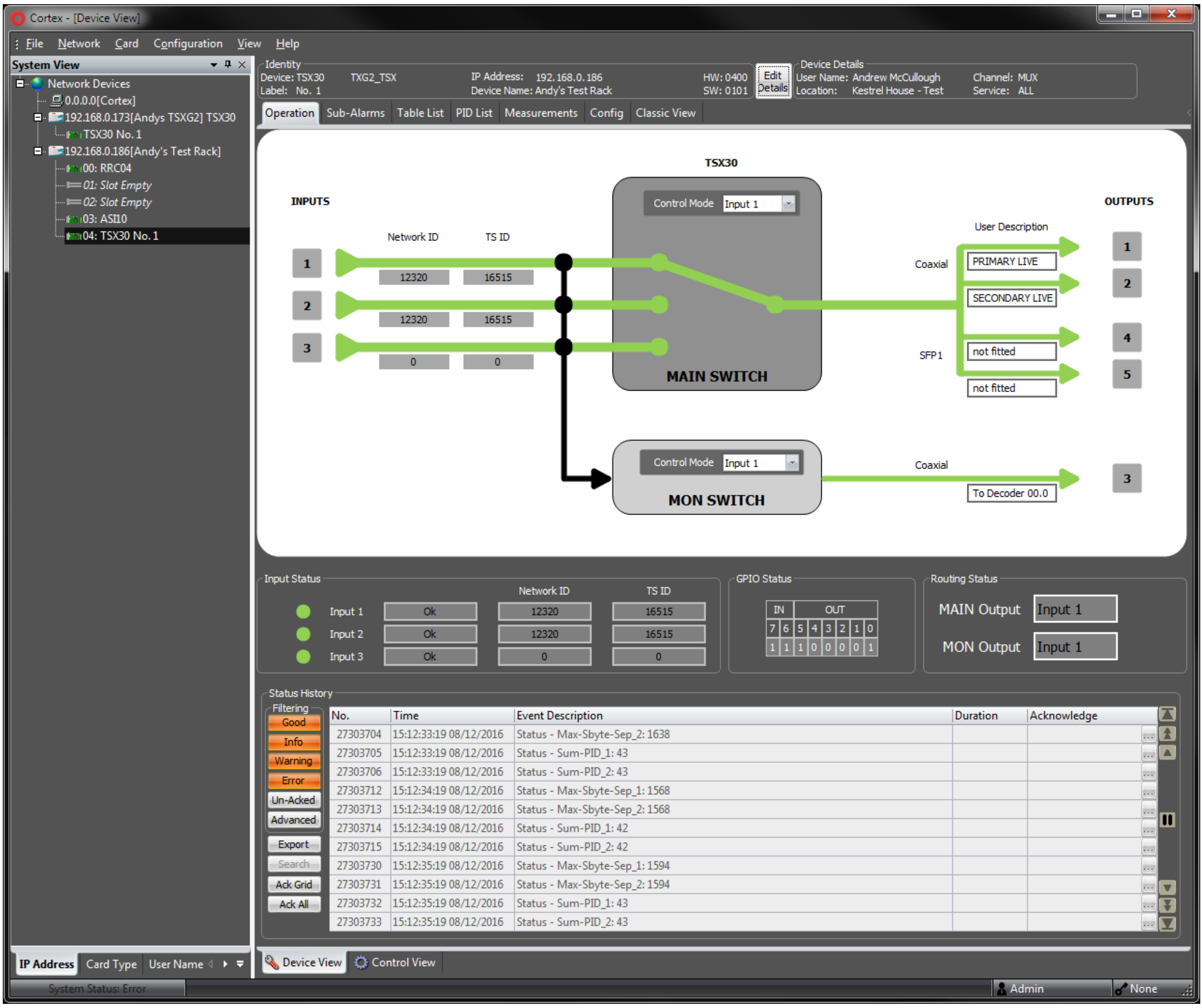
Both Cortex and Cerebrum have several display panels: System View, Device View, Control View, Navigation View and the Event Log.

When a TSX device is selected in System View, the TSX GUI appears as a series of tabs within Device View. The rightmost tab supplements the GUI with a Classic View - a non-graphical interface, which has access to all of the device's monitoring and control objects by name. Since the object names form the basis of all non-GUI interaction with the device (e.g. via Front Panel, SNMP etc.) they are listed and explained in Chapters 7-9 of this manual.

The Event Log is a separate dockable window (available from the View menu if hidden). Use the Event Log to keep track of events from all devices in the system. The Status History pane on the other hand, refers to the current device only and is available on both Operation and Measurement tabs.

Further information on Cortex / Cerebrum is available in the help.

For the purposes of this User Guide, we will examine the GUI of the TSX30. The TSX20 GUI has the monitoring and controls pertaining to the third input greyed out or removed, but is otherwise identical.

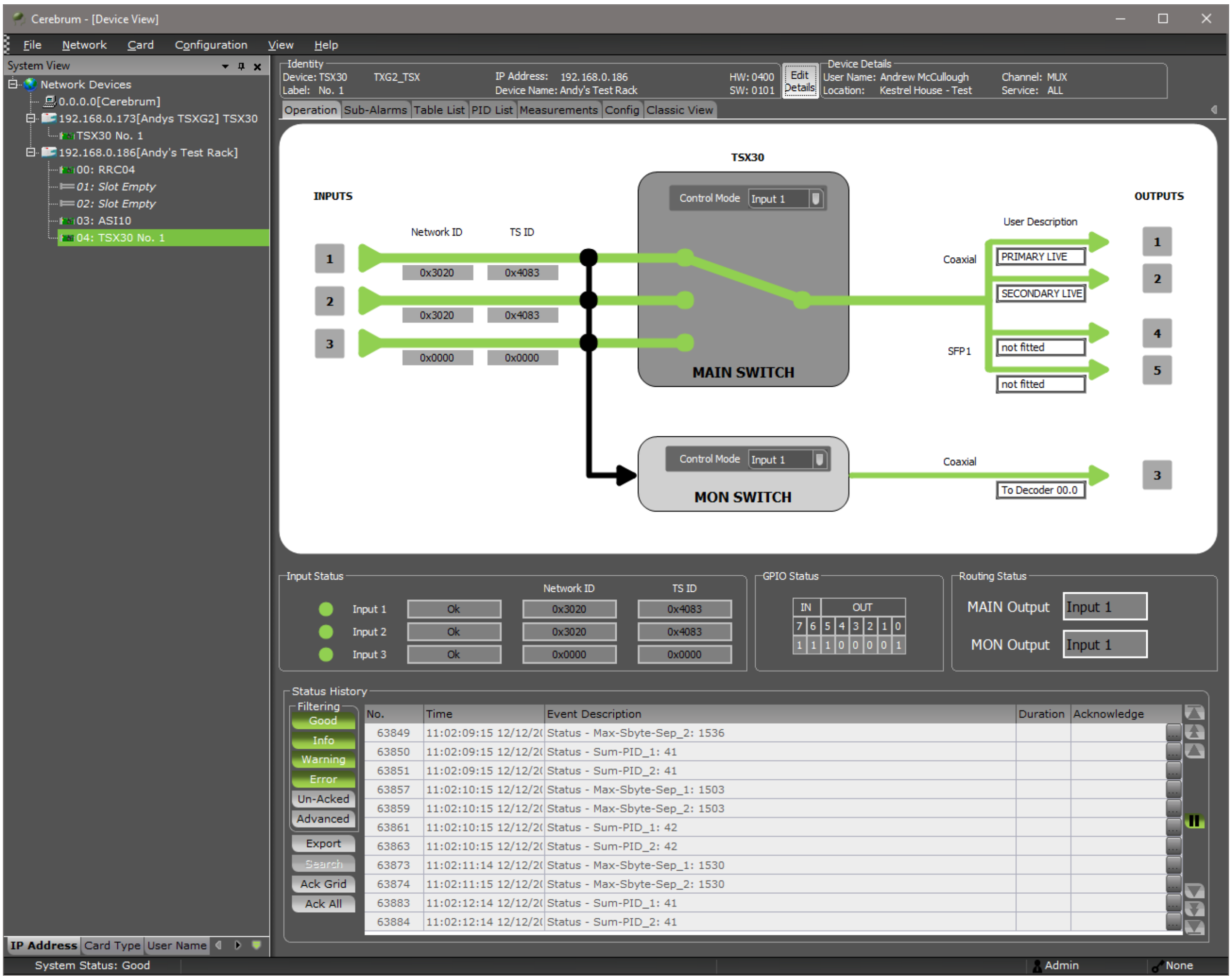



Device View appears on the right hand side by default. If there are multiple tabs at the bottom of the main panel, ensure Device View is selected in order to see the GUI.

The System View appears on the left. Here we have made an optional direct connection to the card (`192.168.0.173`) as well as a standard connection via the RRC04 rack controller (`192.168.0.186`). We have expanded both trees and selected the GUI for slot 4 of the rack by clicking on it. This rack also happens to have an ASI10 card in slot 3.

In this example, the Navigation View and the Event Log are hidden.

The Identity and Device Details boxes at the top of the display provide a method of labelling the frame (or module) with its location and usage. Usage may be labelled in terms of "Channel" and/or "Service". Since TSX cards carry multiplexes with many services, you may choose to ignore these fields, or add something appropriate. Notes that are more detailed can be viewed and/or edited by clicking the Edit Details button.



Here we have the same setup in Cerebrum, except that we have additionally hidden Control View so there are no tabs at the bottom of the main panel.

Some of the GUI elements in Cerebrum have a slightly different look when compared with the GUI in Cortex, but their operation is the same.

The GUI reference in the next chapter zooms in to each tab in turn and examines the elements of the GUI in detail. It uses Cortex for its illustrations.

Before that, we will look at some key concepts for the devices, starting with the nature of the input signals.

## TS/ASI concepts

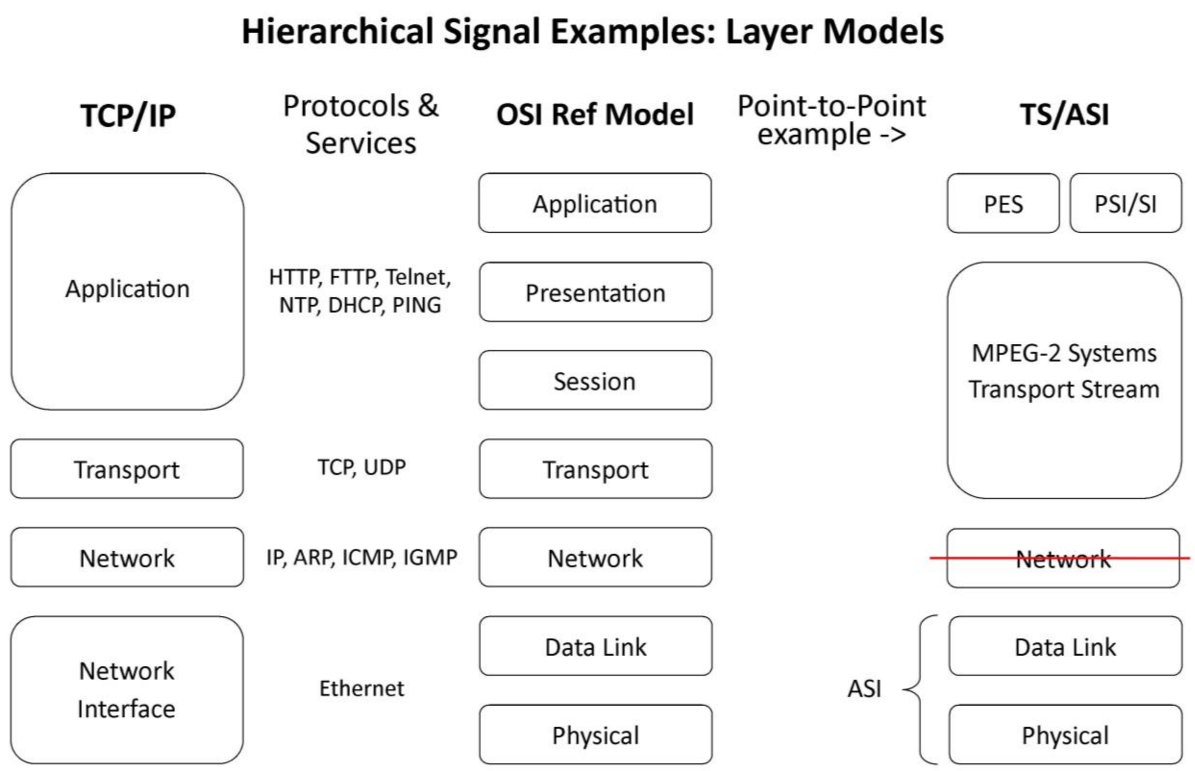
### Layers

TSX modules are so named because their focus is on monitoring and switching Transport Streams [1]. The TSX20 & TSX30 do this for Transport Streams each carried on an Asynchronous Serial Interface [10]. TS/ASI signals have more in common with telecoms and computer networking signals than they do traditional baseband broadcasting signals such as SDI because they are inherently hierarchical. That is, they must be thought of in layers.

ASI is designed to carry only Transport Streams, and for that reason, it is accurate to describe the entire signal as ASI. However, the ASI standard [10] does not define the Transport Stream. It simply includes it by reference for its upper layers. Therefore, in a world where Transport Streams may be distributed by various means, including over Ethernet using Internet Protocol, it is important to remember that ASI is actually just a specification for the lower layers of the signal in a particular application. The application is the point-to-point connection of broadcast system hardware.

ASI provides a simple, reliable, capable and unidirectional method of connecting a Transport Stream source (e.g. multiplexer or service information processor) to its destination, typically a channel adaption device for cable, satellite, terrestrial or IP broadcast.

The performance of the ASI datalink layer of each input is monitored on the Measurements tab, and by the ASI Link sub-alarm.



**Multiplex**

Transport Streams are multiplexes. That is, the TS layer exists to enable multiple programmes (i.e. services) to be combined into a single byte stream ready for transmission. The multiplex makes efficient use of the available bandwidth of the transmission channel. The primary application is broadcast, i.e. one-to-many transmission over a terrestrial, satellite or cable channel. Such channels have a reliable and fixed bandwidth, which depends on the channel and the modulation techniques employed, rather than the number of receivers. The imperative is to make the best use of the bandwidth afforded by the channel (e.g. a satellite transponder) and to fill the available space with interesting content - hence the multiplex.

By contrast, consider interactive streaming services where the user has a TCP/IP connection with the content provider over which they receive their content. In this case the quality, responsiveness and even availability of the service depends on the number of users as the return channels must be processed and the available bandwidth shared out between the connections. A multiplex of services is of little use in this situation as the imperative is to serve only the data that is required and each connection requires only one service.

The situation is different with UDP/IP multicast systems however. These systems are much more akin to traditional broadcast systems as they are one-to-many. Transport Streams can be used in this context. The ASI datalink layer is removed and the Transport Stream is instead inserted into UDP packets, which are in turn inserted into IP and Ethernet layers for transmission. Since UDP is connectionless, some packets may be lost by the network, or arrive out of sequence.

IP may also be conveniently used as part of a distribution system that transfers Transport Streams to their final transmission point, even if the final stage is to be a traditional satellite, cable or terrestrial channel.

The Transport Stream multiplex is therefore ubiquitous. It is at home within both traditional and IP-based broadcast systems and is in use everywhere a signal is required to be broadcast. Despite the growth of interactive and on-demand viewing, the broadcast paradigm retains a role in support of contexts where the end-user requires curated content because interactivity is undesirable, and for live events. Every major digital television system in the world uses Transport Streams of one form or another [11][21][31].

The Transport Stream also provides a stable multiplexing scheme for program elements that are yet to be specified, thus future proofing the layer. As the systems standard [1] is separated out from coding and compression standards, TSX20 & TSX30 modules are able to monitor video, audio and data streams in new formats as they emerge, without requiring an upgrade.

The TSX20 & TSX30 modules provide monitoring and switching of Transport Streams on ASI links based on the integrity and contents of each input. ASI is a way of connecting professional equipment that creates, processes or transmits Transport Streams.

**PES v PSI/SI/PSIP**

A Transport Stream contains a multiplex of programs (services). A program may consist of video, audio and data elements. Elementary Streams from all services are multiplexed together at the TS layer, together with information describing how the elements relate to one another and should be organised into programs for presentation.

There two types of data then that feed into the TS layer: **P**acketized **E**lementary **S**treams (content), and **P**rogram **S**pecific **I**nformation (tables). Elementary Streams can be detected in the TS layer by Packet Identifier (PID), and tables can be detected by PID and table_id. The TSX20 & TSX30 GUI has a separate tab and sub-alarm for the detection of each:

- The PID List is intended for detection of content
- The Table List is intended for detection of tables

Use the PID List tab on the GUI to set up a list of PIDs that contain the content you wish to monitor. Each PID in the list will define a test for a specific Elementary Stream (e.g. BBC1 audio). You will need an understanding of how your stream is organised to achieve this. If in future the stream is re-organised for some reason, you will need to manually update the PID List.

Use the Table List tab on the GUI to set up a list of tests for the tables you wish to monitor. The PSI tables concept created by the MPEG-2 Systems standard [1] that defines the Transport Stream is extended as follows:

- by DVB as Service Information [11]
- by ATSC as Program and System Information Protocol [21]
- by ARIB (ISDB) as Service Information [31]

Although this gives rise to 3 different systems globally for TS layer tables, each is based on the PSI private data section syntax, which allows the TSX20 & TSX30 to detect tables from all three systems, when correctly configured.

TSX20 & TSX30 modules can be configured to detect specific video, audio and data Elementary Streams that are considered important, as well as any specific table sections that may be required. The devices can be configured to switch on the basis of the detection results.

## Severity concepts

Cortex and Cerebrum offer the ability to customise the severity of monitoring events for the purposes of logging. This is done under **Status Severities** on the Configuration menu. By classifying potential events as Good, Ignore, Info, Warning or Error, the concept of Status Severities allows the user to filter announcements by severity in both the Event Log window and the Status History pane. This system is extremely flexible. A default setting is also available for each item, should customisation no longer be required. It is important to note that Status Severities are a Cortex / Cerebrum setting that do not affect the modules themselves, or indeed the rest of the GUI. They are merely a convenience to help organise reporting.

In parallel to Status Severity, a separate and more important concept of criticality exists within the module itself. This is necessary in order to support real-time auto-switching. This concept is known as **Auto-Switch Severity**. The Auto-Switch Severity of an event determines whether the auto-switch will see a particular monitoring result as an error or not. This allows the module to simultaneously provide flexible comprehensive monitoring and a programmable switch.

With flexibility comes complexity. Whilst this may be an asset when it comes to reporting, for switching we wish to tie down the available options somewhat so the user can be confident they have configured auto-switching as intended. To achieve this, certain critical events have a fixed Auto-Switch Severity, as explained below.

In addition, a strict hierarchy of **Alarm Types** exists to provide a vocabulary with which to understand the Auto-Switch Severity settings, as well as harmonisation of Status Severities if required.

## Alarm types

Monitoring results consist of **numerical measurements**, **data field reporting** and **status alarms**. Status Alarms are organised into a hierarchy of types to support both detailed analysis and the extraction of summary information.

At the top of the hierarchy are the **board LEDs**. These summarise the status of each input using a single bit, as good or failed (green or off). These alarms are not available anywhere else and are merely a convenience for checking hardware integrity.

Next, are the **Input Status Alarms**. These are the most important alarms. They summarise input status using 2 bits per input. This allows us to expand the failed state into ‘critically failed’ and ‘partially failed’. Understanding the Input Status Alarms is vital to understanding the operation of the auto-switch. These alarms are prominent on the GUI.

Next, are the **Sub-alarms**. Sub-alarms are single bit quantities that feed into the Input Status Alarms. There are two classes of sub-alarm: critical and optional. Critical sub-alarms drive critical input status.

At the bottom, are the **Sub-sub-alarms**. Sub-sub-alarms are generally two-bit quantities, which optionally feed into the Sub-alarms.

The sections below describe each alarm type in detail.

**Summary alarms (Board LEDs)**

The Input Status is implemented within the module using the following two bits per channel, the first of which is used to drive the board LEDs:

- INPUT_GOOD (single-bit summary for board LEDs)
- INPUT_CRITICAL (describes how bad each failure is)

The truth table for Input Status bits is shown below along with the colour coding for the hardware LEDs on the edge of the board. Since these LEDs are single colour, the failure states are combined.

| COLOUR | INPUT_GOOD | INPUT_CRITICAL | Description |
|---|---|---|---|
| Off | 0 | 0 | Partial Failure |
| Off | 0 | 1 | Critical Failure |
| Green | 1 | 0 | Good |
| N/A | 1 | 1 | Illegal State |

**Summary alarms (Input Status)**

As demonstrated above, the board LEDs provide a single-bit view of the Input Status. However, two bits are required to see it in full. Inputs can be good or failed, but the failed state can be partial or critical. This leads to a traffic light style system for the presentation of Input Status on the GUI. Remember this is independent of Cortex Status Severities.

| COLOUR | INPUT STATUS |
|---|---|
| GREEN | GOOD |
| ORANGE | PARTIAL FAILURE |
| RED | CRITICAL FAILURE |

Each Input Status is a summary alarm that determines the state of an input for the purposes of Auto-Switching. Accordingly, these alarms are called **Input-Sum_x** in the Status Objects list in chapter 8.

**Sub-alarms**

Sub-alarms are single-bit status results reported by input monitoring on both the ASI datalink layer and the Transport Stream layer of each input signal. The status of all sub-alarms may be seen at glance on the Sub-Alarms tab in the GUI. Separate tests are performed for each input. The list for each channel is split into ‘Critical’ and ‘Optional’ sub-alarms. In each case, the alarm state is shown as red, with the non-alarm state shown as green. A third, grey state is used to show when a sub-alarm test is not running. Critical sub-alarms have Critical Auto-Switch Severity. That is, whenever a critical sub-alarm is in the alarm state, the corresponding Input Status will always be in the CRITICAL FAILURE state. Optional sub-alarms have selectable Auto-Switch Severity. They may be specified as OFF, PROBE or 1+2+3. This setting affects the corresponding sub-alarm test on all channels. When a test is set to OFF, the LEDs will grey out. When set to PROBE, the tests run but are disconnected from the Input Status. When set to 1+2+3, the results of the tests are included in the Input Status calculations. Any optional sub-alarm set to 1+2+3 produces a PARTIAL FAILURE in the Input Status when in the alarm state, provided no critical sub-alarms are active. Note that if any of the first 3 critical sub-alarms is active, the remaining sub-alarms on that channel are greyed out as their tests are automatically disabled, their results being meaningless under these conditions.

**Sub-sub-alarms**

Some sub-alarms present the result of a single test. Others summarise the results of a number of tests. Where the component tests are available for the user to individually monitor and configure, their results are referred to as sub-sub-alarms:

- The Table List sub-alarm is an OR sum of up to 64 individual table detection tests. The individual test results are sub-sub-alarms. The tests may be monitored and configured on the Table List tab of the GUI.

- The PID List sub-alarm is an OR sum of up to 64 individual PID detection tests. The individual test results are sub-sub-alarms. The tests may be monitored and configured on the PID List tab of the GUI.

An echo of the corresponding sub-alarm status for each channel is available at the top right of each tab, being a summary of the sub-sub-alarms. This allows the user to easily see the effect of modifying a test.

On each tab, the test list is broken down into 4 groups of 16 for manageability. The tests are then configured using 4 configuration windows, one for each group of tests.

The list is shared between all channels. However, each test may be individually enabled or disabled for each channel.

The GUI features a clickable led matrix showing the status of the sub-sub-alarms on both the Table List and PID List tabs. See the GUI Reference chapter for details.

**Monitoring-only alarms**

Monitoring-only alarms cannot be configured to affect auto-switching. They appear on the Measurements tab with the numerical measurements. They include standard TS layer alarms [51] that are inappropriate for driving a switch in their standard form, as well as datalink layer alarms representing individual errors, which the module is designed to ignore if they occur in isolation.

Standard alarms [51] :

- Sync Byte Error (Sync_byte_error)
- Transport Error (Transport_error)

ASI datalink alarms :

- ASI Error
- ASI Min Stuffing Error

Persistent or recurring instances of these alarms indicate a problem with the source signal that should be investigated.

## Input Status equations

The Input Status bits are a summary of what is happening on a given input. More fundamentally, they define the Auto-Switch view of the health of each input.

Each summary bit is calculated using a logic equation, based on the OR of a number of sub-alarms. The INPUT_CRITICAL equation is defined internally, so its logic is fixed. However, INPUT_GOOD is defined as (the inverse of) an OR list of sub-alarms supplied by the user. The user builds the equation by selecting sub-alarms from the full list of potentially switchable sub-alarms. This is done using the Auto-switch Severity setting for each sub-alarm.

A few sub-alarms are mandatory and cannot be removed from the INPUT_GOOD equation, these are ASI Link, TS Stopped and TS Sync Loss. INPUT_CRITICAL has all of these sub-alarms, and adds the PAT Upper Distance.  PAT UD can be disabled for exceptional uses. By default PAT UD is enabled in the menu.

The logic equations are as follows. Sub-alarms have alarm sense, being ‘1’ in the error state, regardless of name (i.e. ASI Link detects link failure).

***INPUT_CRITICAL = ASI Link OR***
***TS Stopped OR***
***TS Sync Loss OR***
***PAT Upper Distance***

***!(INPUT_GOOD) = INPUT_CRITICAL OR***
***(sub-alarm1 OR sub-alarm2... OR sub-alarmN)***

The INPUT_GOOD equation is defined once and shared between all the inputs. Note, this does not mean you cannot test for different things on each input, just that you cannot specify that a specific sub-alarm is switchable for input 1 but not input 2 etc. Each input will still have its own Table List, PID List, rate thresholds and so on, but the list of switchable sub-alarms will be shared.

## Control arbitration

The GUI is just one of a number of control surfaces capable of controlling the MAIN switch. A general list is as follows:

- Networked Control Panels }
- Cortex / Cerebrum GUI } Configuration
- Synapse Front Panel } Controllers
- SNMP }
- GPIO
- Auto-Switching

This demands a clear set of rules for how the various control surfaces should be arbitrated and how control of the MAIN switch may pass from one to another.

In general, with the exception of GPIO and Auto-Switching, controllers can also modify configuration settings and control the MON switch. It is clear therefore that a number of classes of controller exist.

The easiest way to approach this is to classify controllers according to their technology domain. Using this method, things become a lot simpler: the Synapse Front Panel, also known as the Local control surface, is a mirror of the Cortex Classic View and therefore synchronises with the GUI. The same is true of the SNMP interface and all currently envisaged networked controller devices. In short, for the purposes of arbitration, any control surface handled by the on-board micro-controller software is treated equally, as belonging to a single class of Software Controllers.

GPI and Auto controllers are distinct. These are low-level and must be handled separately. This results in the following classification:

| **CONTROLLER** | **CLASS** |
|---|---|
| Networked Control Panels | **Software Controllers** |
| Cortex / Cerebrum GUI | |
| Synapse Front Panel | |
| SNMP | |
| GPIO | **Hardware** |
| Auto-Switching | **Auto Controllers** |

Software Controllers are treated with equal and top priority. They override commands from the other classes. Within the software class, routing commands and settings are applied in the order they arrive at the device.

The low-level details of GPIO and Auto-Switching are covered later. In this section, we are concerned purely with arbitration.

### Control Modes

Control Modes are used to specify which classes of controller are currently authorised to control a switch. Software Controllers are the only class allowed to modify the Control Mode. This effectively means that such controllers are always authorised, but not always active.

Control Modes are necessary because we may wish to authorise more than one low-level class (i.e. GPI and Auto).

Control Modes are a switch property. The Control Mode of each switch may be different.

The Control Modes for TSX devices are constructed by combining the control surfaces as follows.

Control Modes for MAIN switch:

- Forced (Cortex/Local/SNMP/Networked Panels)
- GPI-only
- Auto-GPI
- Auto

Control Modes for MON switch:

- Forced (Cortex/Local/SNMP/Networked Panels)
- Following

**Control priority**

Forced Mode has the highest priority. A switch will enter Forced Mode whenever a routing request is received from a Software Controller. When multiple such devices exist, the most recent request wins. For example, the Synapse frame front panel and the Cortex GUI both operate at this level. If the GUI calls a switch to input 2 and then a front panel user subsequently calls a switch to input 1, both requests will be executed in sequence.

Forced Mode will persist until a Software Controller selects an alternative Control Mode for the switch (e.g. setting it back to Auto).

**Control Mode selection**

Control Modes can be selected on the Cortex GUI, Synapse Front Panel, over SNMP and via Networked control panels.

Since the Control Mode list is different for MAIN and MON, separate selection controls are necessary.

The Forced Control Mode appears expanded (wherever Control Mode selection is exposed) as a list of options to force the switch, rather than a single entry in the list labelled "Forced Mode". These options are combined into a single list with the other Control Modes. This allows the GUI to offer a single drop-down box combining Control Mode selection and Route selection in the Forced Control Mode. This feature will be familiar from other Axon devices such as the ASI10.

MAIN switch control selections:

- Auto
- Auto-GPI
- GPI-only
- Input 1
- Input 2
- Input 3

MON switch control selections:

- Following
- Input 1
- Input 2
- Input 3

"Following" forces the MON switch to an internal feedback of the current MAIN switch output. This setting allows the MON switch to automatically follow the MAIN switch. Since the real output TS is tapped, the MON switch will inherit the switch type specified for MAIN, for MAIN switch changeovers which occur when MON is set to follow. If the MAIN switch changes its route near-seamlessly, the same near-seamless transition will also appear on the MON output. The MON switch is slaved off the MAIN switch on this setting, regardless of which Control Mode the MAIN switch is in. The two outputs will always be the same. As a result, the device caters for both systems that require output monitoring and those that require increased fan-out for the MAIN switch, but do not require monitoring.

## Auto-switch concepts

This section describes the operation of the AUTO Control Mode, and the AUTO component of the AUTO-GPI Control Mode. For details on how control modes interact, please refer to the section on Control Arbitration above.

When enabled, AUTO determines the behaviour of the MAIN switch routing in response to the Input Status Alarms and the Switching Configuration. Here we describe the switch configuration settings and their effects.

Note that Control Modes and Switch Modes are not the same thing. Control Modes are about which controller controls the switch. Switch Modes are about how the switch logic operates when the Forced Control Mode is NOT in play (i.e. in AUTO, AUTO-GPI or GPI-Only)

The MON switch has no auto-control. (It can be made to shadow the live output as described in the Control Mode Handling Algorithm, so that the user will be able to manually select input or output monitoring.) This chapter concerns the MAIN switch only.

### Switch Modes

At the top level, we have the following three mutually exclusive Switch Modes. The user selects one of these modes to specify the role the TSX30 will play in the system

| AS MODE | USAGE | DESCRIPTION |
|---|---|---|
| I) 2x1 | main, reserve | DUAL |
| II) 3x1 | main, reserve, spare | SYMMETRIC THIRD |
| III) (2+1)x1 | main, reserve, emergency | SPECIAL THIRD |

Selection of Switch Modes applies only to the TSX30. The TSX20 supports only Switch Mode I).

**Note** Selecting the 2x1 Switch Mode on the TSX30 does **not** prevent you from remotely forcing MAIN to Input 3. However, it does mean that Auto-Switching will always route to Input 1 or Input 2 when it is enabled and Input 3 selection from GPI is blocked.

**Auto-switch behaviour**

SWITCH MODE

Here is the logic of the Auto-Switch. We start with the assumption that the MIN_SW and OP_NULL_TS_ALL_CRITICAL modifiers are unselected (i.e. OFF). The modifiers are described in the next section.

Modes I & II
These are "normal" modes. There are three general principles that drive the logic. They are applied in order:

RULE1: A good input takes priority over a failed input.
then
RULE2: Inputs that are not critical are preferred over inputs that are.
then
{RULE3: Only applies in mode III.}
then
RULE4: A low-numbered input takes priority over a high-numbered input.

Mode III
This is a "special" mode. It is for use on systems where input 3 should only be selected in "emergencies".

The same rules apply, except for the addition of rule 3.

RULE3: Input 3 will only be selected iff both inputs 1 and 2 are critical, and input 3 is not critical.

The concept of partial failure is still valid for input 3 in mode III. However, because of rules 1-4 it will not affect the routing. It will only affect the input 3 status colour, which is appropriate.

Input 3 will still have its own PID list etc. which will mean there will be something to probe if the PID list sub-alarm is not globally set to OFF, but such input 3 sub-alarms will have no effect on auto-switching in mode III.

Similarly, in mode I on the TSX30, full input 3 analysis and status is still available even though AUTO and GPI may not select this route.

For the TSX20, no input 3 analysis or routing is offered.

MINIMISE SWITCHING PREFERENCE

The user can choose to minimise switching by setting a further preference bit. This modifier (MIN_SW) may be switched on using the Minimise Auto-Switching drop down box on the GUI.

MIN_SW = ON instructs the device to minimise switching operations. This setting modifies Rule 4 slightly so that switches are not called purely to enforce number priority when the device is already routed to an input with the best available status. Rule 4 will still be used to resolve the routing when switching away from a non-optimal input if there are multiple better alternatives, as there may be in mode II.

In mode III, the ordering of the rules and the if in Rule 3 dictate that MIN_SW will be ignored for switches in and out of input 3, because you will leave input 3 as soon as input 1 or 2 becomes non-critical, or input 3 becomes critical. It is still a useful setting for mode III, as it allows you to prevent unnecessary switching between inputs 1 and 2 on systems where these inputs are to be treated equally.

OUTPUT NULL TS PREFERENCES

The concept of INPUT_CRITICAL allows us to support the following feature that requires some settings. This feature modifies all modes in the same way:

The user has the option of outputting a null TS when all inputs are critical. For mode I, ‘all’ means inputs 1 and 2. For modes II and III, this means all 3 inputs.

The single-bit enable setting has positive sense and known internally as OP_NULL_TS_ALL_CRITICAL. Check the “Output Null TS when all Inputs Critical” box on the GUI to switch it on.

When the feature is set to ON, the output will switch near-seamlessly to an internally generated null TS, preconfigured with a defined TS rate (10 Mbps) and ASI mode (PACKET). The TS mode may be specified using a drop down box on the GUI (188 or 204). The TS rate is displayed as a read-only quantity on the GUI.

This setting may be used to protect downstream equipment from capturing an excess of corrupted TS data during a failure. However, it should be used with caution.

For systems with the same TS on all inputs, it is probably wise to avoid the combination of a strict PAT Upper Distance setting with this option. Otherwise, you might end up nulling the output TS for 1s in response to a single delayed PAT section, which downstream equipment may actually be able to handle. The purpose of our device is not to strictly report minor non-compliance, but to improve system resilience.

## GPIO control & monitoring

This section describes the operation of the GPIO, including how GPI Control should operate when authorised by Control Mode handling.

Synapse modules may be deployed with hardware control panels in the following ways:

- Cerebrum + Axon Control Panel
- Custom panel using GPIO

For Cerebrum systems, hardware control panel support is provided by connecting an Axon Control Panel to the Cerebrum server. These panels are configurable in much the same way as the soft control panels that can be constructed in the Cerebrum client software. As a result, the switch requests generated by such panels interface with TSX devices at the Forced Mode level, along with other Cerebrum clients. Cortex does not support such panels.

In addition, direct connection of a custom hardware panel directly to the module via its GPIO interface is also supported. This second use case is covered in this section.

The GPIO interface has 8 GPIO lines whose directions are firmwired. It is assumed that devices connected to GP outputs will have weak pull-ups so that under module power-fail they will float high.

The TSX30 & TSX20 will use 6 lines as outputs and 2 as inputs. They will be defined as follows:

| DBH15/F pin (BHX45 / BPH45) | GPIO line | Direction | Signal Name |
|---|---|---|---|
| 3 | GPI/O_0 | Output | n_PRDY |
| 5 | GPI/O_1 | Output | n_INPUT1_GOOD |
| 4 | GPI/O_2 | Output | n_INPUT2_GOOD |
| 9 | GPI/O_3 | Output | n_INPUT3_GOOD |
| 13 | GPI/O_4 | Output | n_LS0 |
| 15 | GPI/O_5 | Output | n_LS1 |
| 10 | GPI/O_6 | Input | LIVE_SW_GPI_CTRL_0 |
| 14 | GPI/O_7 | Input | LIVE_SW_GPI_CTRL_1 |
| 6,1,11,2,12,8 | RESERVED | | |
| 7 | GND | | |

- The input status signals are the inverse of the INPUT_GOOD signals, so that inputs are reported failed on power-failure.

- The n_LS1 .. n_LS0 outputs are the inverse of the MAIN switch status. This is so that the power-fail state can be coded when OP_NULL_TS_ALL_CRITICAL is not in use. These outputs will continue to work regardless of whether GPIO control is enabled or not.

- The LIVE_SW_GPI_CTRL_0 and LIVE_SW_GPI_CTRL_1 inputs will have no effect unless the MAIN switch is in Auto-GPI or GPI-only Control Mode. LIVE switch is just another name for the MAIN switch.

- It is important for a GP Control Panel to know if it will have any effect. Therefore, there is an additional output (n_PRDY) to show the panel that it is authorised. PRDY is short for “panel ready”.

n_PRDY = !( Auto_GPI OR GPI_only)

The ready output is negative true to be suitable for driving an active low LED circuit, as we want to indicate the PRDY condition.

**GPI Control Modes**

The two GPI control modes are:

- GPI-only
- Auto-GPI

For context, we start by looking at how these modes operate in the related ASI10 device (2 input ASI switch):

ASI10 GPI-only mode: On entering GPI-only mode, the switch would keep its current routing until the assertion of the GP (1) input caused a switch to input 2. Only this one action was permitted. After the input was de-asserted, the new route was maintained as auto is disabled. The only way to change the routing thereafter was to call a new mode, e.g. forced, auto, or auto-gpi.

ASI10 Auto-GPI mode: On entering Auto-GPI mode, the switch would act as if in auto mode until the assertion of the GP (1) input caused a switch to input 2. Only this one action was permitted. This action effectively lifted the device out of auto whilst the assertion was in place. After the GP input was de-asserted, auto was re-enabled.
There was no facility in the ASI10 to switch to 1 using GPI control. Since it was the GP input level that was important in ASI10, there was no need for a trigger input.

The TSX30 extends this design using a 2-bit code to specify the desired routing on the GPIs. Unasserted inputs pull high. The external GPI switch requests should persist for the minimum GPI time to guarantee acceptance. With the exception of this requirement and increased number of selectable routes, the behaviour of GPI modes is the same as for ASI10

**GPI coding**

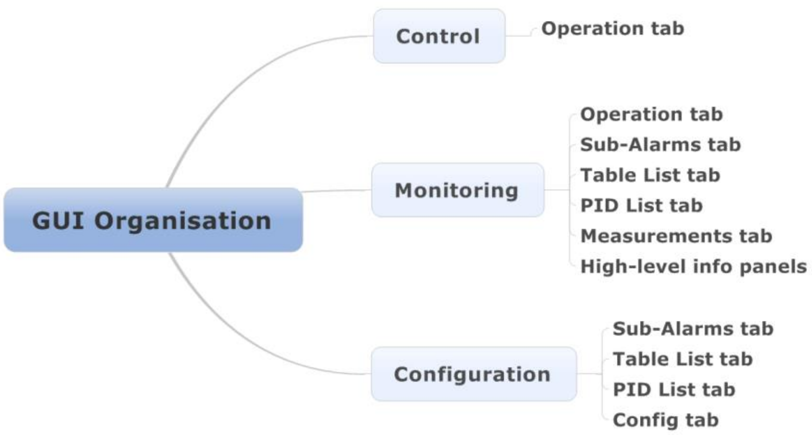
| GPI_CTRL_1 | GPI_CTRL_0 | Requested Route |
|---|---|---|
| 1 | 1 | Not Asserted |
| 1 | 0 | Input A |
| 0 | 1 | Input B |
| 0 | 0 | Input C |

Let us combine these bits into the set GPI_CTRL(1..0) which we can use for shorthand.

# 6 GUI Reference

## Organisation

The GUI comprises a series of tabbed displays and is organised by task as follows:

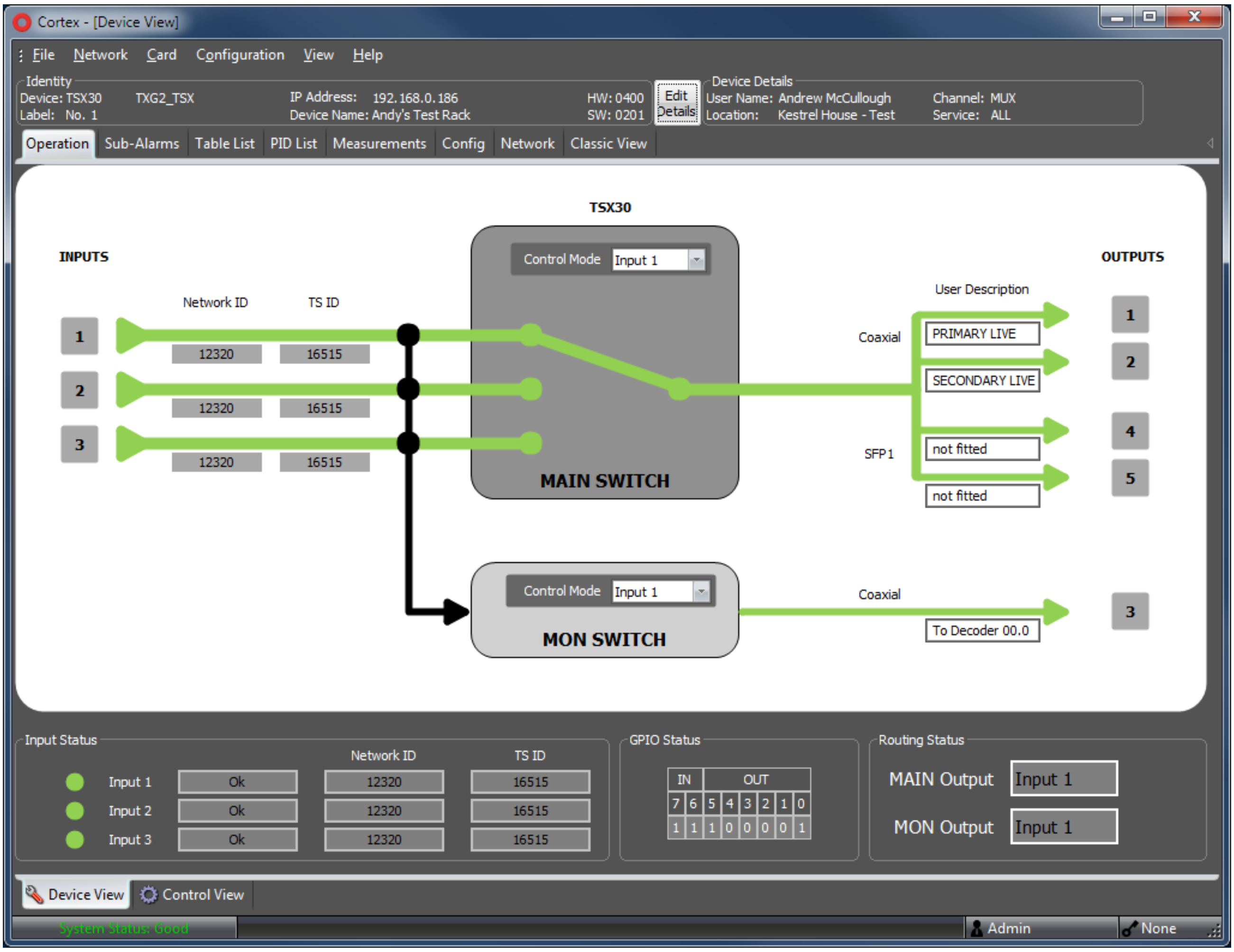



In addition, a Classic View tab is available on the right of the GUI, which lists all of the Settings, Status Objects and Alarm Events by their internal names. Classic View is synchronised with the GUI, and current values are shown. These names are the basis of all non-GUI interaction with the device (e.g. Rack Front Panel, SNMP) and are the subject of chapters 7-9.

Finally, a Network Tab is inserted just before the Classic View tab, for easy access to the Network parameters of the card. These parameters also appear under Classic View. They are relevant only for direct connections to the card and do not affect operation via a rack controller.

In this chapter, we will look at the graphical elements of the GUI and focus on usage and workflow. For each element, we will also provide a cross reference between the GUI and the Classic View name. We will also refer to concepts and logic described in the previous chapter.

## Operation tab




### Operator interface

The Operation Tab lays out the basic concept of the device in graphical form. It also doubles as an operator interface, allowing an operator to quickly assess the state of play in a fault situation and make manual overrides to the routing if necessary.

TSX devices are deployed inline, i.e. in the TS signal path of a live broadcast system. Such systems are unidirectional. Signal flow is from left to right in the graphic. Each input signal is a multiplex containing multiple services.

The MAIN switch routes one input to all of the main outputs, thereby feeding the rest of the broadcast chain.

A secondary switch is provided to allow the operator to tap off one of the inputs to external device such as a decoder, or capture device. The MON switch is designed to be used freely as required. Its status will never affect the routing of the main outputs.

**Input identifiers**

On the left hand side of the graphic, the logical input numbers are displayed.

Next to these are the input signal identifiers: network_id and transport_stream_id. These fields are obtained directly from each input Transport Stream so the operator can confidently identify the muxes. A valid Network Information Table (actual TS) must be present in the stream to obtain the network_id. A valid Program Association Table must be available to obtain the transport_stream_id.

**Note** Physical inputs are mapped to logical inputs by the Input Mux control on the Config Tab.

**Output identifiers**

On the output side of the graphic, User Description fields are available to allow the installing engineer to add a reminder for each output as to the location or name of the next equipment in the system. This can provide the operator with a sanity check - particularly useful when performing a manual override to support system maintenance.

Note, that these labels must be updated if your system wiring changes.

On the right hand side the logical output numbers are given. Logical output numbers refer to fixed physical locations.

**Control Mode of MAIN**

Manual control of the MAIN switch may be performed by simply selecting a new route using the drop-down box labelled "Control Mode" on the MAIN switch graphic. If a route is shown in this field (e.g. Input 1) then Forced Mode is in play, otherwise the name of the active Control Mode is displayed (e.g. Auto). Control Modes are covered later in this guide.

**Control Mode of MON**

Manual control of the MON switch may be performed by simply selecting a new route using the drop-down box labelled "Control Mode" on the MON switch graphic. All options for the MON switch are considered to be Forced Mode selections, including the special "Following" setting. Further detail is given in the next section.

**Note** Control Modes are set on the Operation Tab. These determine who may control each switch. Auto-Switch details are specified on the Config Tab.

**Routing graphic & Input Status**

The MAIN switch routing graphic shows the currently selected route. Its inputs are coloured to reflect their status. The colours Green (OK), Orange (Partial Failure) and Red (Critical Failure) are used. The colour of the selected input will flow through to the output graphic to show the status of the output.

To simplify the diagram, MON switch routing is not shown: its inputs are shown as a bus (black). However, the status (colour) of the input selected for monitoring will flow through to its output graphic.

**Note**

*** If the routing graphic displays grey lines at the output, then the device is emitting a NULL TS, because all inputs are critical and a check box on the config tab has instructed it to do so.

**High-level panels**

At the bottom of each GUI tab, three panels are dedicated to high-level status information to support configuration editing on a live system. This information is provided to help a user spot unintended consequences of any changes that are being made.

**Input Status panel**

At the bottom left is the Input Status Panel. The colour of the Input Status LEDs will track the colour of each input in the routing graphic. For those with limited colour vision, a text version of the status is added to the right of each LED. This panel also has a copy of the Input Identifiers.

**GPIO status panel**

At the bottom in the centre is the GPIO Status Panel. This panel is designed primarily to allow the user to spot unexpected control inputs coming from the GPIs. Input Status and Routing Status outputs are also displayed. Both inputs and outputs display active low values. So, a value of "11" for the IN field implies no GPI control is asserted.

**Routing status panel**

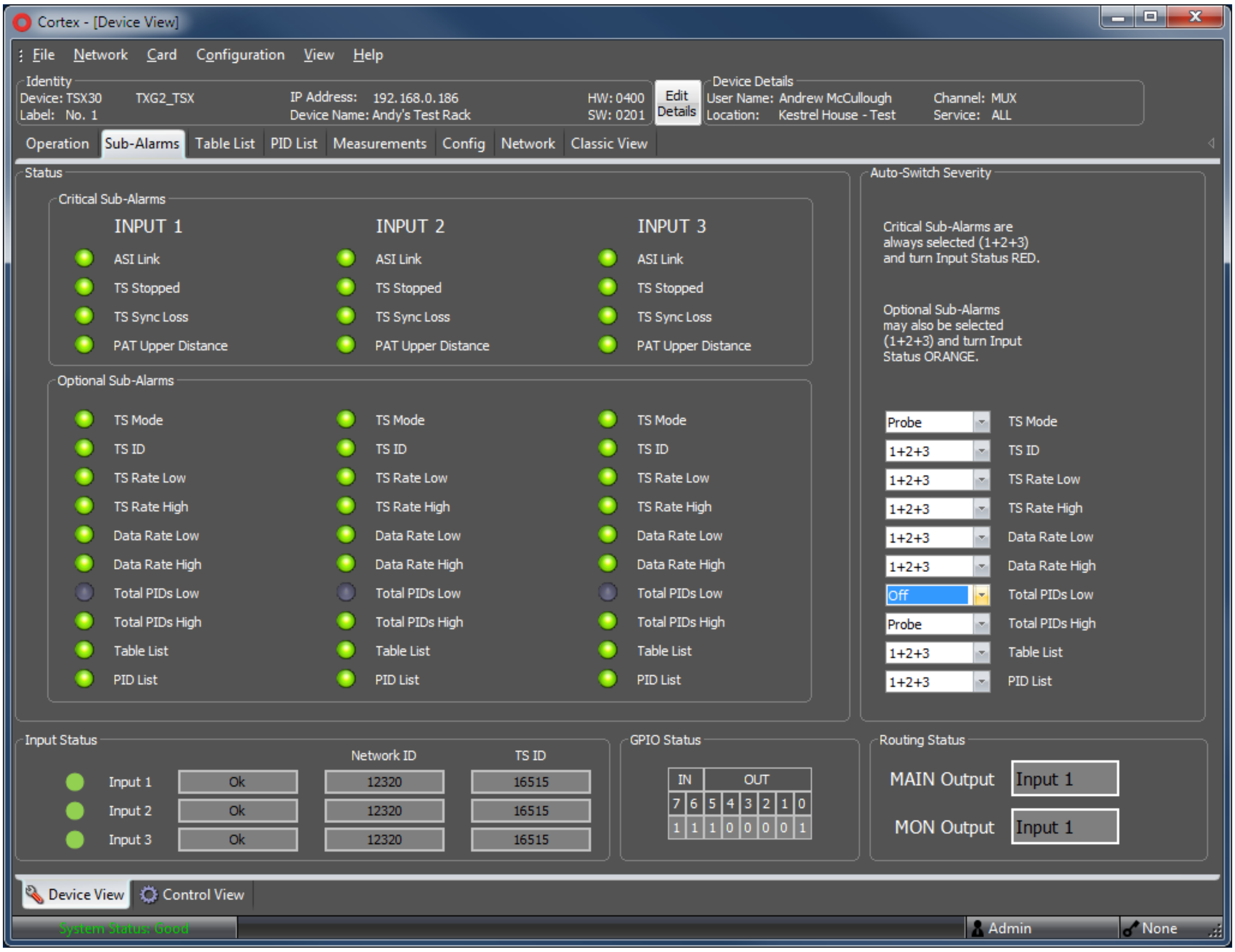
At the bottom right, is the Routing Status Panel. This panel enables the user to keep track of the routing status whilst viewing / editing other tabs. Note that when the MON switch Control Mode is set to Following, this will also be indicated here.

**Note**

In the special case where a NULL TS is deliberately emitted (due to all inputs being critical and the config being set up to do so), the routing status panel indicates a preview of the route that will be made should the check box be deselected.

AXON

## Sub-Alarms tab




**Overview**

This tab shows the status of the sub-alarms for each input. For an explanation of the concept of sub-alarms and how they relate to Input Status, see the “Alarm Types” and “Input Status Equations” sections in the previous chapter. For an explanation of each sub-alarm in detail, refer to Chapter 8 - Status Menu.

Sub-alarms are red in the alarm state, green in the good state and grey when disabled or muted.

The sub-alarms tab shows the complete list of alarms that can affect Input Status and therefore potentially drive an automatic changeover. Each column contains the results for a single input channel and is divided into critical sub-alarms and optional sub-alarms.

Critical sub-alarms always drive the Input Status. Optional sub-alarms may or may not drive the Input Status, depending on their Auto-Switch Severity setting.

**Auto-switch severity**

Critical sub-alarms always drive the Input Status. That is, when a critical sub-alarm is in the alarm state (red), the corresponding Input Status will also be in the CRITICAL_FAILURE state (red).

Remember that the Input Status for each channel is visible in the Input Status panel at the bottom of this tab as both a colour and in text.

Optional sub-alarms may or may not drive the Input Status, depending on their Auto-Switch Severity setting. This setting is selected for each sub-alarm by a control on the right hand side of the tab, at the end of each row. Settings are applied for the given sub-alarm across all channels. A sub-alarm may be set to Off, Probe or On. The On setting is called “1+2+3” on TSX30 and “1+2” on TSX20 as a reminder that all channels are affected. It is possible to configure each sub-alarm with different parameters for each channel, but how important each sub-alarm is deemed to be from an auto-switching perspective is set here and will be the same for all channels.

For example, the TS ID sub-alarm can be configured (via the config tab) to look for a *different* transport_stream_id on each input. However, if you want the result of the analysis to affect Input Status then you must specify this for all channels here. The Auto-Switch Severity of a sub-alarm is the *same* for all channels.

Optional sub-alarms will be red in the alarm state, but they will drive the Input Status to the PARTIAL FAILURE state (orange). However, if a critical sub-alarm is alarming on the same channel, this will override the optional alarm and drive the Input Status to CRITICAL FAILURE (red).

The Auto-Switch Severity of the critical sub-alarms is hardwired to the On setting.

For further information, see the “Severity Concepts” section in the previous chapter.

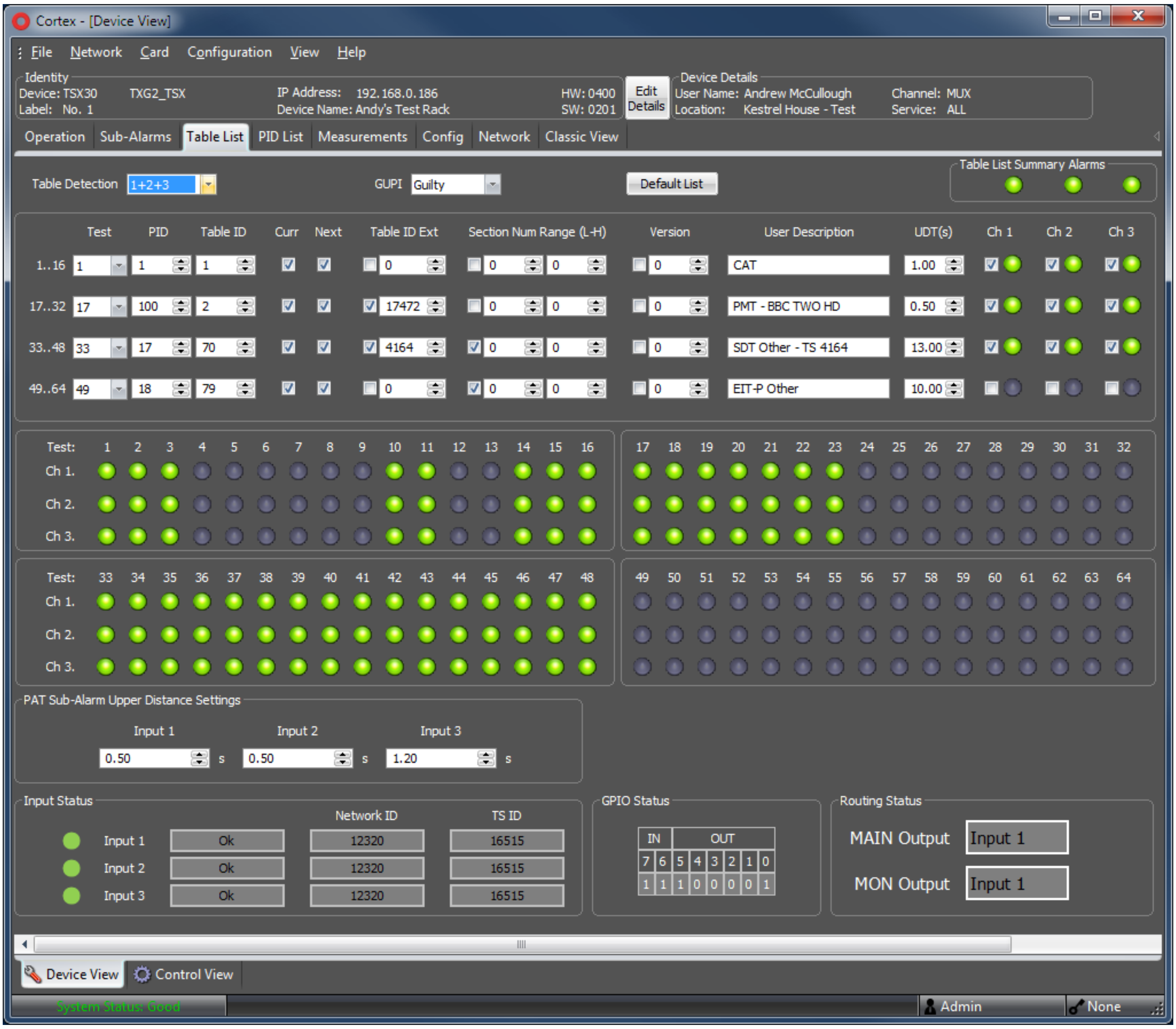
**Analysis & presentation period**

Sub-alarm results are collected during analysis periods of 1s. These are then reported for the following 1s.

**Sub-alarm muting**

The ASI Link, TS Stopped and TS Sync Loss critical sub-alarms are also known as muting sub-alarms. If one of these reports as being in the alarm state, the reporting of all other sub-alarms is disabled for the duration of that presentation period. The disabled sub-alarms will turn grey. (The Auto-Switch Severity setting is unaffected.) It is standard practice to derive no further information from a stream that exhibits a TS_sync_loss. We apply the same principle to ASI Link failure and TS Stopped. PAT Upper Distance however does not mute any sub-alarms.

## Table List tab



### Configuration overview

The Table List tab is used for both monitoring and configuration.

The table list itself is split into 4 groups. These groups allow the user to conceive of the 64 tests in batches of 16. It is not possible to see the settings for all 64 tests at once, due to the amount of configuration data involved. The grouping allows for a compact view of the configuration. A configuration row is provided for each batch.

Each configuration row starts with a drop-down box for selecting a test to configure, followed by the configuration settings for that test and finally a check box for each input. The check boxes are for specifying to which inputs the test should be applied. These checkboxes allow different tests to be applied to each channel, up to a maximum of 64 for the device. Selecting a test using the drop down allows you to inspect its settings and modify them if required.

The Table List is shared between all inputs.

**Monitoring matrix**

Underneath the configuration rows is the monitoring matrix. This pincushion style display of LEDs shows the status of all the tests in real time. Greyed out LEDs show that a test has been disabled for a particular input. The matrix is split into four boxes, mirroring the organisation of the configuration rows. When a test is selected in a configuration row, the LEDs on the right of the configuration row show the status of the selected test. These horizontal LEDs are mirrored vertically in the pincushion. The pincushion shows the status of all tests, including those that are hidden to the configuration view.

To quickly see the settings that give rise to a particular result, simply click on the result in the LED matrix, and the relevant configuration row will update to display the settings. This is a quicker way of browsing the complete configuration, than repeatedly using the test number drop down box.

**Note** The LED matrix is clickable. The test indices will automatically adjust to display the setup of the clicked test.

**GUPI setting**

Table list tests are upper distance tests. That is, for a test to be good, the gap between occurrences of the specified table sections in the input stream must be less than the specified upper distance. Upper distance is measured in units of time (s).

Because the entire gap time must elapse before a test can be definitively declared a failure, the test must support an unproven state, which is displayed in yellow. Note: this has a different meaning to the orange state used for Input Status. Yellow status is only used for the sub-sub-alarms themselves. To understand how this yellow state affects auto-switching, you must understand the GUPI setting.

The GUPI setting tells the device to interpret unproven states as “Guilty” or “Innocent”. When set to Guilty, all unproven states are treated as failures when calculating the single-bit Table List Summary Alarms. When set to Innocent, unproven states are treated as good.

The GUPI control also appears on the PID List tab. It is a device-wide, universal setting for all multi-second tests. Multi-second tests are tests that may require more than one analysis second to establish a result.

Note that all upper distances are measured accurately and independently of the 1s analysis / presentation schedule.

**Table List Summary Alarms**

The Table List Summary Alarms at the top right of the tab are the same thing as the Table List sub-alarms. They are “summaries” from the point of view of the Table List tab, but “sub-alarms” from the point of view of Input Status. They are included here so that a user can see the effect of any changes to the configuration on the Table List sub-alarms.

**Configuring tests**

Tables are identified by PID and table_id. To set up a table test, the minimum information you need to specify is the PID, table_id and state of the Current and Next checkboxes (see Current / Next below).

**Sub-tables**

Tables may be organised as one or more sub-tables. Sub-tables are identified by table_id_extension and may be distinguished if required. The table_id_extension exists at the same location in the syntax for all tables, though it may be called different things in each context. For example, in the DVB Service Description Table the table_id_extension is a transport_stream_id. In the DVB Event Information Table, the table_id_extension is a service_id. By using this symmetry, and allowing the user to specify which sub-tables are important, the TSX20 & TSX30 modules are able to provide a very powerful, flexible and efficient table detection algorithm. To deploy these advanced features successfully, system designers are encouraged to refer to the latest version of [11], [21] or [31]. The table_id_extension is always a 16-bit field, and can be identified by its location within the table section syntax: the 3rd and 4th bytes after the table_id. Remember to check the box on the GUI to enable this field and enter a value if you wish to detect a specific sub-table.

**Current / Next**

Sub-tables are labelled in the Transport Stream as either Current or Next. Current sub-tables are currently applicable, next sub-tables are not yet applicable but will be the next sub-table to be applicable. Decide for each test in the table list if you wish to detect only Current sections, only Next sections or both. Select the check boxes on the GUI accordingly.

**Short-syntax tables**

Certain tables do not consist of sub-tables and therefore have a short syntax e.g. DVB TDT and TOT. A section_syntax_indicator (= ’0’) flags these tables. To configure a test to detect such tables, deselect both Current and Next checkboxes on the GUI. The unnecessary fields will then grey out. Remember to select either Current or Next or both for normal syntax tables even if you do not care about table_id_extension.

**Table sections**

Within each sub-table, there are also sections. Sections are the atomic units of tables. Enable this field and select a specific section or range of sections for detection if required.

**Table versions**

Multiple versions of a sub-table may also exist in a stream. Again, enable the version field if you require a specific version, otherwise any version will satisfy the test.

**User description**

A user-text field is provided so that a label may be associated with each table test. This is independent of the test configuration data.

**Upper Distance Time**

Each test looks for its target to occur at regular intervals. The upper distance is the maximum interval allowed before the target is declared to be in error. Enter standard [51] or custom values as required.

**PAT Upper Distance**

The PAT Upper Distance test is a table test like the others, except that it drives its own critical sub-alarm. This is because it detects Program Association Table sections, which are fundamental to the stream.

The upper distance specified here for the PAT also affects the Transport Stream ID sub-alarm as the transport_stream_id field is carried in PAT table sections.

Since PAT Upper Distance is a sub-alarm, it already has the GUPI setting factored in. There is no way to see a yellow state here unless you set up a duplicate test for it in the Table List. There is little point doing this however, as standard PAT sections should occur every 0.5s. The Upper Distance is configurable to allow a slight relaxation of this test to avoid aggressive switching on systems that are known to operate close to the standard upper distance. There should not normally be a need to make this test a multi-second test.

**Maintenance**

After the initial installation, it may become necessary to modify or add some tests over the course of time if there are changes to the nature of the input streams. Circumstances may require you to do so whilst the system is live-to-air. The recommended method for doing this is to first take the MAIN switch out of Auto (or Auto-GPI) by forcing it remotely to a particular route before applying the changes. This is to protect the system from unnecessary Auto-Switching that may occur if the user makes a mistake entering the changes. This way, the changes can be previewed for their effect before they are applied.

For further peace of mind, in the top left hand corner of the tab, next to the words Table Detection, is a drop down box containing the text "1+2+3", "Probe" or "Off". This is a copy of the Auto-Switch Severity control for the Table List Sub-alarms, which also appears on the Sub-Alarms tab. When set to "Probe" or "Off" you can be confident that the test results on the Table List tab are disconnected from the Input Status and the Auto-switch.

(There is a Table List sub-alarm for each input but their Auto-Switch Severity setting is shared, being set by a single control. This is why the ON state is called "1+2+3".)

This control is included at the top of the Table List tab so that the user can see if changes on this tab might result in a switch, were AUTO to be enabled by mistake by another operator on another client, whilst the changes are in progress. If set to “Probe” the entire tab is monitoring-only. Therefore, in addition to forcing the MAIN switch to a particular route before commencing maintenance, you also have the option of modifying this setting to provide a second method of isolation.

Be aware that this method will only provide additional protection if all the inputs are good. If an input is already partial failure, and this is due exclusively to Table List failures, then changing this setting will give the input a clean bill of health, such that if AUTO were enabled, the input would be available to switch to, which may not be the intention.

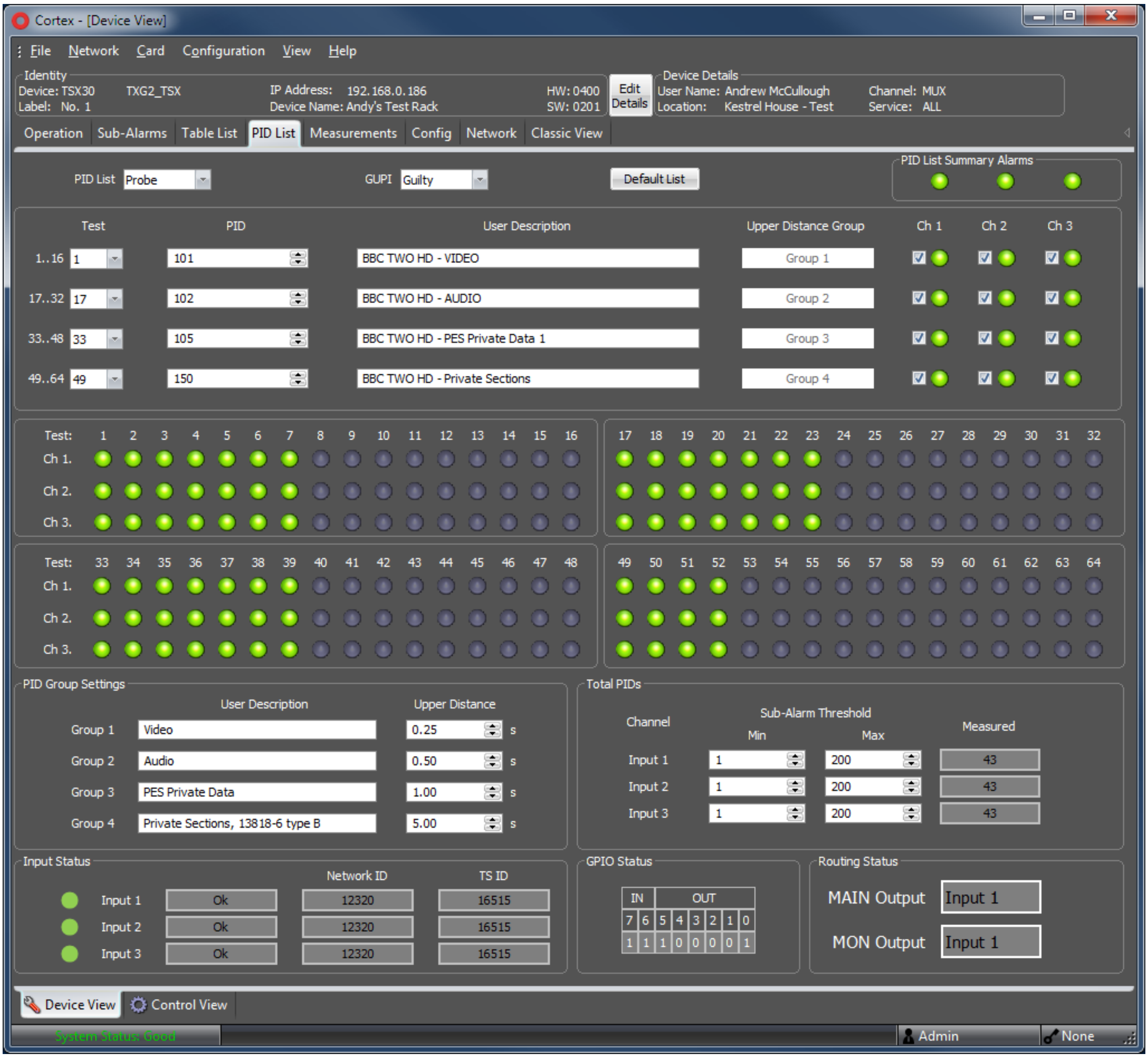
When the changes are complete and you are confident they are correct, then you can re-select “1+2+3” if you wish the table tests to affect Input Status and Auto-Switching. Finally, you can re-enable Auto or Auto-GPI control mode by removing the switch forcing.

**Note**

The Table List tab is intended for setting up tests to detect Programme Specific Information / Service Information tables.

Tests can be quite specific down to an individual table section or range of sections. Each test looks for its target to occur at regular intervals. The upper distance is the maximum interval allowed before the target is declared to be in error.

## PID List tab



**Configuration overview**

The PID List tab is used for both monitoring and configuration.

The PID list itself is split into 4 groups. These groups allow the user to conceive of the 64 tests in batches of 16. It is not possible to see the settings for all 64 tests at once. The grouping allows for a compact view of the configuration. A configuration row is provided for each batch.

Each configuration row starts with a drop-down box for selecting a test to configure, followed by the configuration settings for that test and finally a check box for each input. The check boxes are for specifying to which inputs the test should be applied. These checkboxes allow different tests to be applied to each channel, up to a maximum of 64 for the device. Selecting a test using the drop down allows you to inspect its settings and modify them if required.

The PID List is shared between all inputs.

**Monitoring matrix**

Underneath the configuration rows is the monitoring matrix. This pincushion style display of LEDs shows the status of all the tests in real time. Greyed out LEDs show that a test has been disabled for a particular input. The matrix is split into four boxes, mirroring the organisation of the configuration rows. When a test is selected in a configuration row, the LEDs on the right of the configuration row show the status of the selected test. These horizontal LEDs are mirrored vertically in the pincushion. The pincushion shows the status of all tests, including those that are hidden to the configuration view.

To quickly see the settings that give rise to a particular result, simply click on the result in the LED matrix, and the relevant configuration row will update to display the settings. This is a quicker way of browsing the complete configuration, than repeatedly using the test number drop down box.

**Note**

The LED matrix is clickable. The test indices will automatically adjust to display the setup of the clicked test.

**GUPI setting**

PID list tests are upper distance tests. That is, for a test to be good, the gap between occurrences of TS packets with the specified PID in the input stream must be less than the specified upper distance. Upper distance is measured in units of time (s).

Because the entire gap time must elapse before a test can be definitively declared a failure, the test must support an unproven state, which is displayed in yellow. Note: this has a different meaning to the orange state used for Input Status. Yellow status is only used for the sub-sub-alarms themselves. To understand how this yellow state affects auto-switching, you must understand the GUPI setting.

The GUPI setting tells the device to interpret unproven states as "Guilty" or "Innocent". When set to Guilty, all unproven states are treated as failures when calculating the single-bit PID List Summary Alarms. When set to Innocent, unproven states are treated as good.

The GUPI control also appears on the Table List tab. It is a device-wide, universal setting for all multi-second tests. Multi-second tests are tests that may require more than one analysis second to establish a result.

Note that all upper distances are measured accurately and independently of the 1s analysis / presentation schedule.

**PID List Summary Alarms**

The PID List Summary Alarms at the top right of the tab are the same thing as the PID List sub-alarms. They are "summaries" from the point of view of the PID List tab, but "sub-alarms" from the point of view of Input Status. They are included here so that a user can see the effect of any changes to the configuration on the PID List sub-alarms.

**Upper Distance Groups**

The PID list is split into 4 groups. These groups may be assigned a meaning by the user. Groups exist to strike a balance between flexibility in assigning Upper Distance values, and ease of use. For example, all video elementary streams can be grouped and assigned a common upper distance value. This value could be small and different to the value used for elementary streams containing custom data that are known to be low rate, or bursty at the TS layer.

**Maintenance**

After the initial installation, it may become necessary to modify or add some tests over the course of time if there are changes to the nature of the input streams. Circumstances may require you to do so whilst the system is live-to-air. The recommended method for doing this is to first take the MAIN switch out of Auto (or Auto-GPI) by forcing it remotely to a particular route before applying the changes. This is to protect the system from unnecessary Auto-Switching that may occur if the user makes a mistake entering the changes. This way, the changes can be previewed for their effect before they are applied.

For further peace of mind, in the top left hand corner of the tab, next to the words PID Detection, is a drop down box containing the text "1+2+3", "Probe" or "Off". This is a copy of the Auto-Switch Severity control for the PID List Sub-alarms, which also appears on the Sub-Alarms tab. When set to "Probe" or "Off" you can be confident that the test results on the PID List tab are disconnected from the Input Status and the Auto-switch.

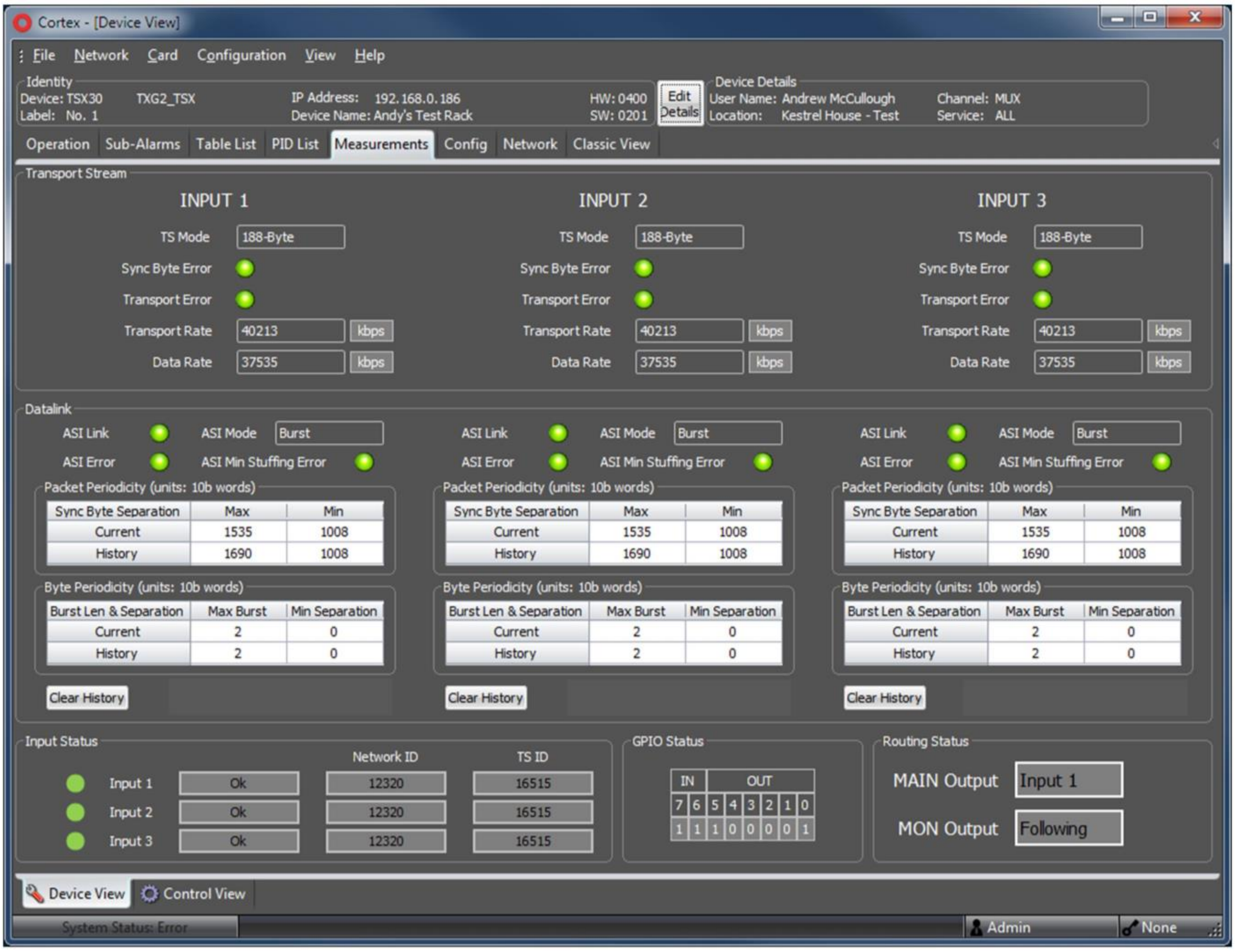
(There is a PID List sub-alarm for each input but their Auto-Switch Severity setting is shared, being set by a single control. This is why the ON state is called "1+2+3".) This control is included at the top of the PID List tab so that the user can see if changes on this tab might result in a switch, were AUTO to be enabled by mistake by another operator on another client, whilst the changes are in progress. If set to "Probe" the entire tab is monitoring-only. Therefore, in addition to forcing the MAIN switch to a particular route before commencing maintenance, you also have the option of modifying this setting to provide a second method of isolation.

Be aware that this method will only provide additional protection if all the inputs are good. If an input is already partial failure, and this is due exclusively to PID List failures, then changing this setting will give the input a clean bill of health, such that if AUTO were enabled, the input would be available to switch to, which may not be the intention. When the changes are complete and you are confident they are correct, then you can re-select "1+2+3" if you wish the PID tests to affect Input Status and Auto-Switching. Finally, you can re-enable Auto or Auto-GPI control mode by removing the switch forcing.

**Note**

The PID List tab is intended for detecting content rather than multiplex information. Content (e.g. video, audio, data) is carried in elementary streams, each with a different Packet IDentifier. The PID list is maintained manually to allow elementary streams not referenced in the Service Information to be added to the list.

## Measurements tab




The Measurements Tab provides further monitoring beyond that provided by the switchable sub-alarms.

It is organised into two layers, reflecting the hierarchical nature of the ASI signals. Measurements made on the TS layer are shown in the top pane. Measurements made on the ASI datalink layer are shown in the bottom pane.

### Transport Stream

The measurement standard for DVB Transport Streams [51] describes a number of standard errors that can be evaluated (e.g. TS_sync_loss). Where we have used standard names in this manual, the standard method has been used. A number of the standard measurements have been omitted because they have been replaced by a more flexible analysis. For example, the standard PID_error covers only PIDs referred to in the Service Information, whereas the PID_List sub-alarm allows the inclusion of unreferenced PIDs if required. Similarly, the Table List sub-alarm covers the Upper Distance component of PMT_error_2 and many of the standard 3rd priority alarms as well as facilitating monitoring of tables not covered by the standard, private sections and specific table sections.

A few standard alarms are not suitable for directly driving a switch as they may represent transient errors in the stream. These have been implemented, but not as switchable sub-alarms. They are therefore included on this tab. Currently we provide Sync_byte_error and Transport_error.

Additionally, the TS packet size is reported. DVB systems may optionally use a 204-byte mode, rather than the usual 188. A sub-alarm has been implemented to allow the expected mode to be specified. The detected mode is reported here.

Transport Stream Rate and Data Rate are reported in kbps or pkt/s according to the units setting on the Config tab. Both are averaged over the analysis second. The TS Rate includes null TS packets, whereas the Data Rate does not.

**Datalink errors**

The ASI datalink measurements give further details about the characteristics and integrity of the upstream connection to the previous equipment in the broadcast chain. In particular, we are concerned with datalink errors, and the way the TS data is distributed on the link. The ASI datalink uses a 8B10B coding scheme to guard against errors, and datalink stuffing characters to maintain the line rate of 270 Mbps. Note that this datalink stuffing mechanism is independent of any TS layer stuffing mechanisms such as null packets or stuffing tables.

An echo of the ASI Link sub-alarm is provided first, for context. If 10 or more ASI Errors are detected in one analysis period, the ASI Link alarm will be triggered.

The ASI Error alarm reports every error detected in the 10b code or in the Running Disparity. This allows transient errors to be logged.

Additionally, the ASI standard [10] states that a minimum of two stuffing characters be present on the ASI link between TS packets. If this rule is violated the ASI Min Stuffing Error alarm will be triggered. This alarm is a warning only and will not trigger the ASI Link sub-alarm in isolation.

**Datalink distribution**

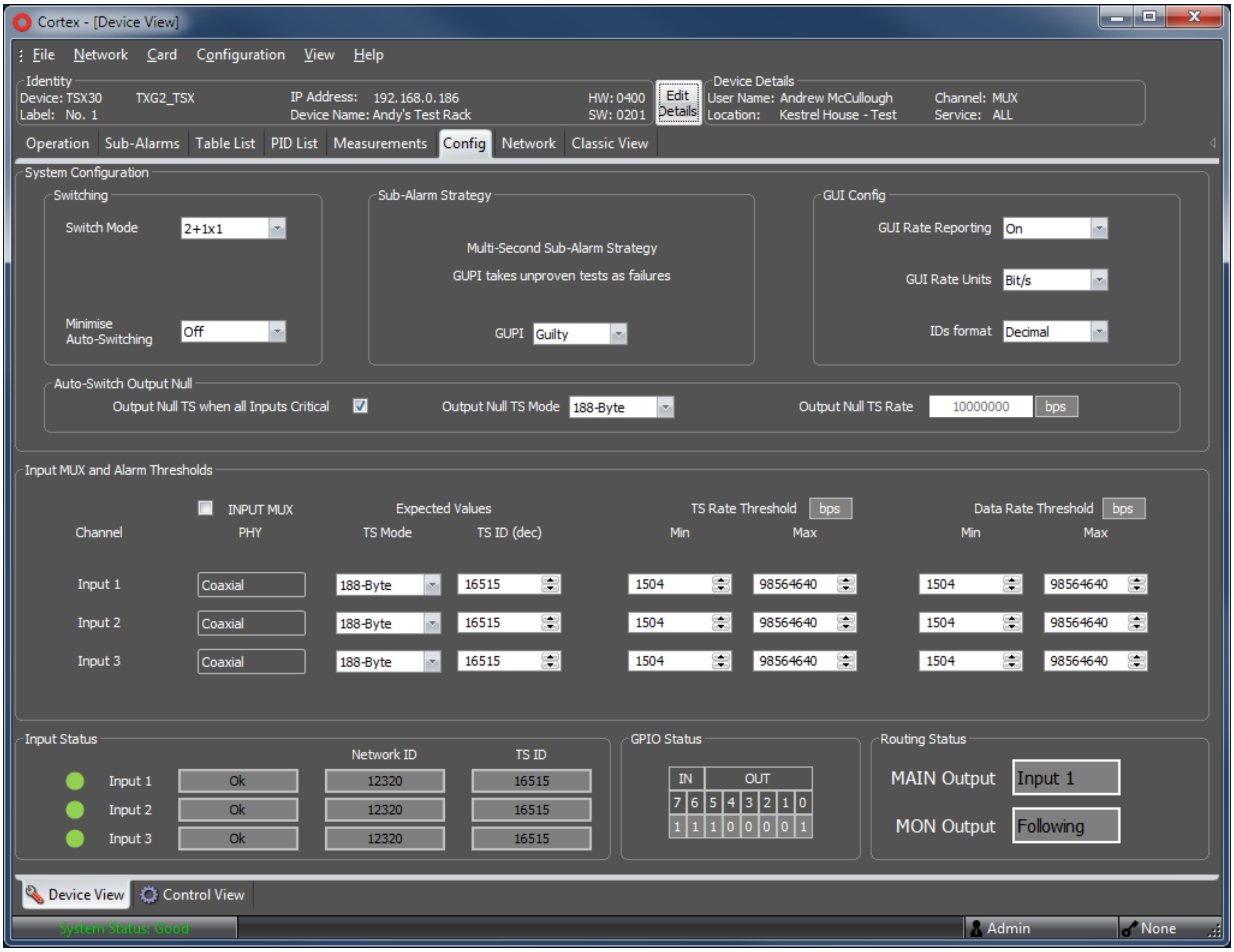
The Asynchronous nature of ASI means that an ASI transmitter is not required to distribute the TS bytes evenly on the link. This can result in a bunching of the TS data, or periodicity. Manufacturers are encouraged to report the maximum periodicity that is likely to occur on an output to avoid the situation where an input stage with shallow buffers may be overwhelmed.

The packet and byte periodicity measurements are intended to provide detail that may assist in debugging ASI outputs in this regard. The TSX20 & TSX30 themselves are reasonably transparent, meaning that the ASI outputs will have similar characteristics to the input streams.

**Loopback function**

The TSX30 offers a useful mode for users with only 2 inputs, who wish to have a rigorous approach to understanding output datalink distribution as well as using the modules inline: simply set the Switch Mode to 2x1 and loop back the secondary main ASI output to the third input. The third input analysis will then show the datalink distribution of the main output as the primary and secondary main outputs are exact copies. The Switch Mode will prevent the third input from being selected by Auto.

## Config tab

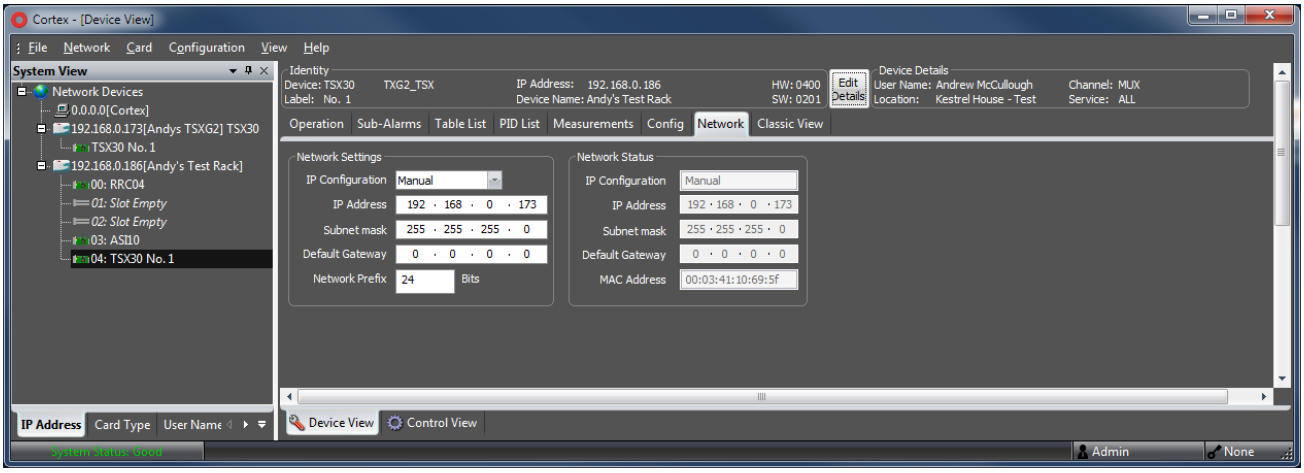



This is where the various settings of the module are configured. At the top left, in the box marked “Switching”, we configure the behaviour of the Auto-Switch.

The Input Mux is normally unchecked to prevent accidental modification. It is used to specify the coaxial or optical route for each logical input. Should it be necessary to modify this setting, check the box to enable the control. It is advisable not to do this on a live system. Such switching is non-seamless and is intended as an installation setting, rather than an operational one.

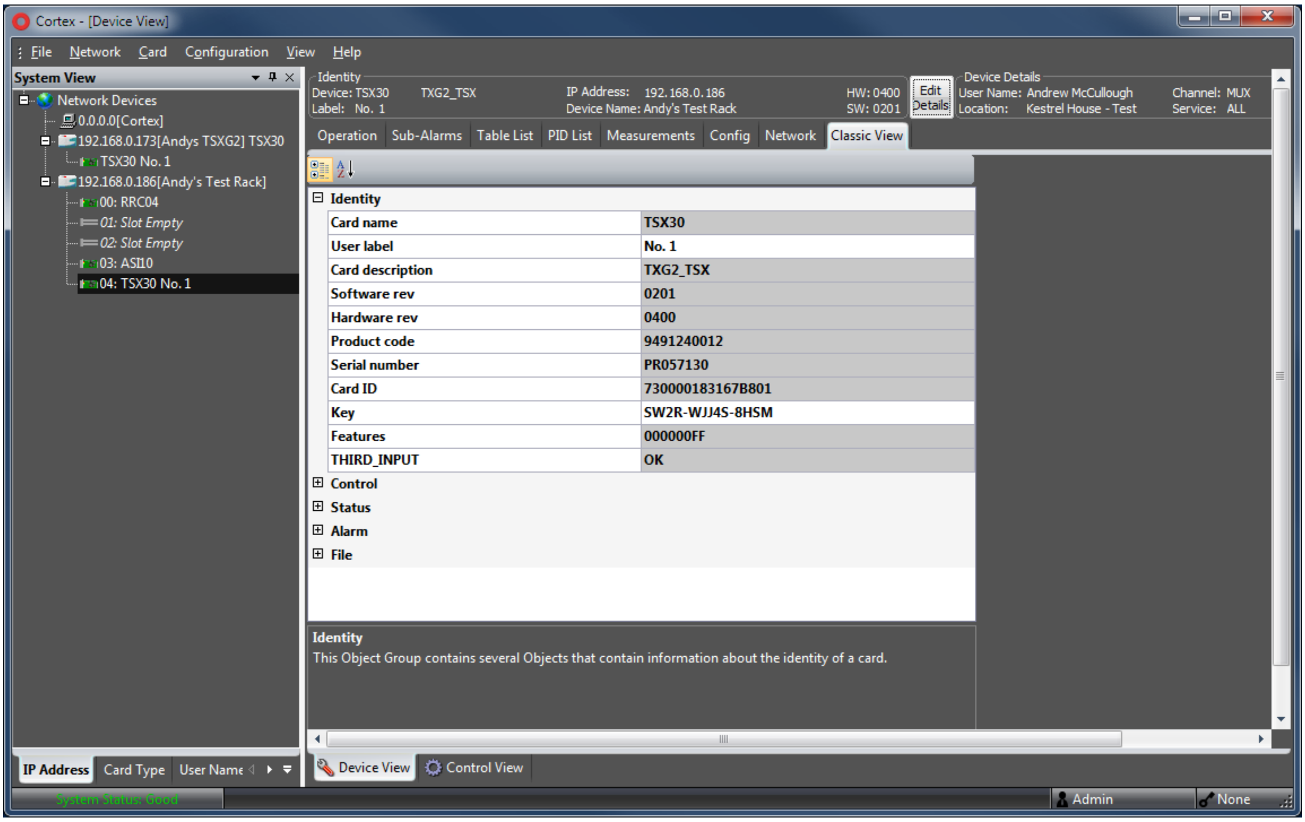
Expected values and thresholds for the sub-alarms are also specified on this tab.

## Network tab




The Network tab allows easy access to the network status and settings of the module (for direct Ethernet connections to the module itself). These objects are also available under Classic View, but are gathered together here for convenience. The System View is also shown above on the left hand side, to illustrate that the settings apply to the direct connection.

## Classic View tab




The final tab of the TSX Device View is the Classic View. This is non-graphical user interface has access to all of the device's monitoring and control objects by name, just like the front panel.

Each section can be expanded by clicking on the plus next to the section name.

The Identity section has useful identifiers for the module currently being probed, such as firmware versions and serial number.

The Control section contains the Settings objects.

The Status section contains the Status objects.

The Alarm section contains details of Announcements and Events.

Announcements can also be managed under Network/ Edit Common Parameters of the main Cortex window.

Settings, Status and Event objects are covered by name in Chapters 7-9 of this manual.

**Note** Announcements must be ON if you want the GUI to update automatically. Otherwise, you must manually refresh the card status by right clicking on the module in the System View and selecting Refresh Card Status. Announcements also allow you to send updates to a specific network client only.

# 7 Settings Menu

**Introduction**

The settings menu displays the current state of each TSX30-20 setting and allows you to change or adjust it. Settings can be changed using the front panel of the Synapse frame (SFR18, SFR08 or SFR04) or with Cortex. Please refer to chapter 3 for information on the Synapse front panel control and Cortex.
***Note:*** All items preceded with a #-sign are part of the presets.

**IP_Conf0**

With this setting, you can let the card obtain an IP address automatically via DHCP, or set a Manual IP address. The default setting is DHCP

**mIP0**

When IP_Conf0 is set to Manual, you can enter the preferred IP address here. The default setting is 0.0.0.0

**mNM0**

When IP_Conf0is set to Manual, you can enter the required Netmask 0.0.0.0..255.255.255.255. Changing NetwPrefix0 will update this value as well. The default setting is 0.0.0.0

**mGW0**

With IP_Conf0 set to Manual, this setting let you set a Standard Gateway. The default setting is 0.0.0.0

**NetwPrefix0**

With IP_Conf0 set to Manual, this item lets you set a network prefix varying from 0 to 30 bit. This is an alternate entry for the actual netmask. Changing mNM0 will update this value as well. The default setting is 0bit

**Output-Config**

With Ouput-Config you can put the card in the following auto-switch modes :

- 2x1, Only input 1 and 2 are used. The card will only switch between these two inputs.
- 3x1, all 3 the inputs are used.
- 2+1x1, Only input 1 and 2 are used. The card will only switch between these two inputs. Input 3 will only be used as an Emergency input when Input 1 and 2 have a critical failure.

Default is 2x1. TSX20 only provides the 2x1 mode. For more information on switching behaviour refer to Chapter 5 Auto-switch concepts and Auto-switch behaviour.

**Live_Switch**

With the `Live-Switch` you select the output of the Live-Switch (see block schematic). You have the following options:

- Auto
- Auto-GPI
- GPI_Only
- Input_1
- Input_2
- Input_3 (Only available when TSX30)

**Monitor-Switch**

With the `Monitor-Switch` you select the output of the Monitoring Switch (see block schematic). You have the following options:

- Following (the output of the Live-Switch)
- Input_1
- Input_2
- Input_3 (Only available when TSX30)

**Min-Switch**

With this setting you can enable the Minimise Auto-Switching. Default is On. Auto-switch behaviour in Chapter 5 explains the effect of this setting on the switch behaviour in greater detail.

**Inp-Selx**

With this setting you can select the coaxial input or SFP module for the input X. TSX30 has 3 inputs, TSX2 only has 2 inputs.
Default is coaxial input

**Rate-Status**

With this setting you switch `on` or `off` the automatic status updates of the Data and TS rates. By default this is switched `off` to reduce control network traffic

**Rate-Units**

When set to `Pkt/s` (packets per second), all reported bitrates and the bitrate thresholds are in packets per second. Set to `Bit/s` (bits per second) the unit changes to bits per second.

The `bit/s` value thresholds for an input channel are calculated using the `TS-Mode_x` status. Set these to the expected packet sizes for these inputs when using `Bit/s`.
Default is `Pkt/s`

**ASI-Link-Det**

This setting is set fixed to All Inputs. The ASI-link detection is always enabled.

**TS-Stopped-Det**

This setting is set fixed to All Inputs. The TS Stopped detection is always enabled.

**TS-SyncLoss-Det**

This setting is set fixed to All Inputs. The TS Sync Loss detection is always enabled.

**ASI-Mode-Det_x**

Alarms can be generated when the contents of the ASI stream on input X does not match this setting. Possible ASI-modes are:

- `Empty`
- `Byte`
- `Packet`
- `Burst`

Default is Empty.

**DataRate-H-Det**

With this setting you can decide whether or not you want to check on too high Data rates and how the card should act to too high data rates (threshold is set in the following menu items). The following settings are possible:

- `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)
- `1+2 / 1+2+3`: will monitor and switch when datarate heights errors are detected on either ASI input.
- `Off`: no data rate height check will be performed.

**Max-DataRate_x**

This item sets the threshold at which data rate maximum the card should signal an error on input X. Can be set between 1 and 109050240 units per second for 208 byte mode (either packets or bits per second, dependant on the `Rate_units` setting). For 188 byte mode the limit is 98564640. For 204 byte mode the limit is 106953120. Default is 65535 unit/s

**DataRate-L-Det**

With this setting you can decide whether or not you want to check on too low Data rates and how the card should act to too low data rates (threshold is set in the following menu items). The following settings are possible:

- `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)
- `1+2 / 1+2+3`: will monitor and switch when too low datarates are detected on either ASI input.
- `Off`: no data rate low check will be performed.

**Min-DataRate_x**

This item sets the threshold at which data rate minimum the card should signal an error on input X. Can be set between 1 and 109050240 units per second (either packets or bits per second, dependant on the `Rate_units` setting). Default is 1 unit/s

**TS-Rate-H-Det**

With this setting you can decide whether or not you want to check on too high transport stream rate and how the card should act to too high rates (threshold is set in the following menu items). The following settings are possible:

- `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)
- `1+2 / 1+2+3`: will monitor and switch when transport rate height errors are detected on either ASI input.
- `Off`: no transport stream rate height check will be performed.

**Max-TS-Rate_x**

This item sets the threshold at which transport stream rate maximum the card should signal an error on input X. Can be set between 1 and 109050240 units per second for 208 byte mode (either packets or bits per second, dependant on the `Rate_units` setting). For 188 byte mode the limit is 98564640. For 204 byte mode the limit is 106953120. Default is 65535 unit/s

**TS-Rate-L-Det**

With this setting you can decide whether or not you want to check on too low transport stream rates and how the card should act to too low rates (threshold is set in the following menu items). The following settings are possible:

- `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)
- `1+2 / 1+2+3`: will monitor and switch when too low transport streams are detected on either ASI input.
- `Off`: no transport stream rate low check will be performed.

**Min-TS-Rate_x**

This item sets the threshold at which transport stream rate minimum the card should signal an error on input X. Can be set between 1 and 109050240 units per second (either packets or bits per second, dependant on the `Rate_units` setting). Default is 1 unit/s

**TS-Mode-Det**

With this setting you can decide whether or not you want to check on TS mode size and how the card should react. The following settings are possible:

- `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)
- `1+2 / 1+2+3`: will monitor and switch when the selected mode isn't detected on either ASI input.
- `Off`: no transport stream TS mode check will be performed.

**TS-Mode-Size_x**

This selects the expected packet size for Input-X. This can either be 188 or 204 byte.

**PAT-UD-Det**

With this setting you can decide whether or not you want to check the Program Association Table (PAT) repetition. The following settings are possible:

- `1+2 / 1+2+3`: will monitor and switch when the PAT repetition is too infrequent on either ASI input. (default)
- `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs. This setting is not recommended for most use cases!
- `Off`: no transport stream PAT Upper Distance check will be performed. This setting is not recommended for most use cases! .

**PAT-UpperDist_x**

This item sets the threshold at which upper distance in time the card should signal an error on input X. Can be set between 0 and 81.91 seconds. Default is 0.5 seconds

**TS-ID-Det**

With this setting you can decide whether or not you want to check on the expected TS ID and how the card should react. The following settings are possible:

- `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)
- `1+2 / 1+2+3`: will monitor and switch when the selected mode isn't detected on either ASI input.
- `Off`: no TS-ID check will be performed.

**Exp-TS-ID_x**

This selects the expected TS-ID for Input-X. Can be set between 0 and 65535. Default is 0.

**Op_Null_TS_En**

With this setting you can enable or disable the Output Null TS when all inputs are critical. Can be set to `On` or `Off`(default)

**Op_Null_TS_Mode**

With this setting you can set the transport stream mode of the Output Null TS option. Can be `188-Byte` (default) or `204-Byte`.

**Op_Null_TS_Rate**

This setting sets the transport stream rate for the Output Null TS option. This option is not implemented. This is a static value of 10.000.000 bps.

**GUPI**

With this setting you can set or the Table and PID List Test are "Guilty Until Proven Innocent". You can set it to `Guilty` or `Innocent`. For more details on GUPI setting and its effect see Chapter 5.

| | |
|---|---|
| **Table-Det** | With this setting you can decide whether or not you want to enable the Table Test and how the card should react. The following settings are possible:<br>▪ `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)<br>▪ `1+2 / 1+2+3`: will monitor and switch when the selected mode isn't detected on either ASI input.<br>▪ `Off`: no Table check will be performed. |
| | **Table-List Test**<br>The following objects together form the Table List Test. This List contains 64 rows. With the x_Table-Test you can select the row. The objects with a # are preset values that follows this index number. The rows are grouped in four parts, marked with a prefix 1_, 2_, 3_ and 4_. |
| **x_Table_Test** | With this index number you can select the row number of the table.<br>▪ 1_Table_Test contains the rows 1 to 16<br>▪ 2_ Table_Test contains the rows 17 to 32<br>▪ 3_ Table_Test contains the rows 33 to 48<br>▪ 4_ Table_Test contains the rows 49 to 64 |
| **#x_PID** | With this setting you can set the packet identifier PID for the selected Table-List row selected with the x_Table-Test object. |
| **#x_TID** | With this setting you can set the table ID TID for the selected Table-List row selected with the x_Table-Test object. |
| **#x_Current** | With this setting you can enable or disable to check the Current. Explanation for this setting van be found in Chapter 6, Sub-tables |
| **#x_Next** | With this setting you can enable or disable to check the Next. Explanation for this setting van be found in Chapter 6, Sub-tables |
| **#x_TID-Ext-Ena** | With this setting you can enable or disable the Table ID extension detection. Explanation for this setting van be found in Chapter 6, Sub-tables |
| **#x_TID-Extension** | With this setting you can set the expected Table ID extension. . Sub-tables are identified by this Table ID extension |
| **#x_Section-Ena** | With this setting you can enable or disable the section detection. Within each sub-table, there are also sections. Sections are the atomic units of tables. Enable this field and select a specific section or range of sections for detection if required. |

| | |
|---|---|
| **#x_Section-Low** | With this setting you set the lower end section number for detection. |
| **#x_Section-High** | With this setting you set the higher end section number for detection. |
| **#x_Version-Ena** | With this setting you can enable or disable the version detection. Enable the version field if you require a specific version, otherwise any version will satisfy the test |
| **#x_Version** | With this setting you set the version number for detection. Note multiple versions of a sub-table may also exist in a stream |
| **#x_Table-descr** | With this label the user can add a description to the row. |
| **#x_UDT** | This sets the Upper Distance in seconds. |
| **#x_En1** | With this setting you can enable or disable the Table test for input 1. |
| **#x_En2** | With this setting you can enable or disable the Table test for input 2. |
| **#x_En3** | With this setting you can enable or disable the Table test for input 3 (only when TSX30). |
| **PID-Det** | With this setting you can decide whether or not you want to check on the PID List Test and how the card should react. The following settings are possible:<br>▪ `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)<br>▪ `1+2 / 1+2+3`: will monitor and switch when the selected mode isn’t detected on either ASI input.<br>▪ `Off`: no PID table check will be performed. |
| | **PID-List Test**<br>The following objects together form the PID List Test. This List contains 64 rows. With the x_PID-List you can select the row. The objects with a # are preset values that follows this index number. All this objects are there 4 times (1_, 2_, 3_ and 4_). |
| **x_PID-List** | With this index number you can select the row number of the table.<br>▪ 1_PID-List contains the rows 1 to 16<br>▪ 2_PID-List contains the rows 17 to 32<br>▪ 3_PID-List contains the rows 33 to 48<br>▪ 4_PID-List contains the rows 49 to 64 |

| | |
|---|---|
| **#x_PID-Field** | With this setting you can set the PID id for the selected PID-List row selected with the x_PID-List object. |
| **#x_PID-Label** | With this setting the user can give a description to the selected row of the PID-List. |
| **#x_Ch1_Enable** | With this setting you can enable or disable the PID test for input 1. |
| **#x_Ch2_Enable** | With this setting you can enable or disable the PID test for input 2. |
| **#x_Ch3_Enable** | With this setting you can enable or disable the PID test for input 3 (only when TSX30). |
| **Grp0-UDT** | Sets the Upper Distance for the PID-List Test in seconds. Grp0 is PID-List 1 to 16. |
| **Grp0-Label** | With this label the user can give the group setting a description |
| **Grp1-UDT** | Sets the Upper Distance for the PID-List Test in seconds. Grp1 is PID-List 17 to 32. |
| **Grp1-Label** | With this label the user can give the group setting a description |
| **Grp2-UDT** | Sets the Upper Distance for the PID-List Test in seconds. Grp2 is PID-List 33 to 48. |
| **Grp2-Label** | With this label the user can give the group setting a description |
| **Grp3-UDT** | Sets the Upper Distance for the PID-List Test in seconds. Grp3 is PID-List 49 to 64. |
| **Grp3-Label** | With this label the user can give the group setting a description |
| **High-PID-Cnt-Det** | With this setting you can decide whether or not you want to check on too many PIDs (packet identifiers) and how the card should act to too many PIDs (threshold is set in the following menu items). The following settings are possible:<br>▪ `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)<br>▪ `1+2 / 1+2+3`: will monitor and switch when PID total height errors are detected on either ASI input.<br>▪ `Off`: no PID maximum check will be performed. |

**Low-PID-Cnt-Det**

With this setting you can decide whether or not you want to check on too few PIDs (packet identifiers) and how the card should act to too few PIDs (threshold is set in the following menu items). The following settings are possible:

- `Probe`: will only do the status detection, which can then be monitored in the status menu, but the card will not do any switches when an error occurs (default)
- `1+2 / 1+2+3`: will monitor and switch when low amounts of PIDs are detected on either ASI input.
- `Off`: no PID minimum check will be performed.

**Total-PID-High_x**

This item sets the threshold at which PID amount maximum the card should signal an error on input X. Can be set between 0 and 8191.

**Total-PID-Low_x**

This item sets the threshold at which PID amount minimum the card should signal an error on input X. Can be set between 0 and 8191.

**Hist-Clear_x**

With this setting you can clear the following min and max History values for each input X:

- `Hist-Min-Byte_x`
- `Hist-Max-Brst_x`
- `Hist-Min-Sbyte_x`
- `Hist-Max-Sbyte_x`

**Label_x**

With this label the user can give the output a description

# 8 Status Menu

**Introduction**

The status menu indicates the current status of each item listed below.

**IP_Addr0**

This item displays the status of the IP address. It can be `Manual, DHCP Asking, DHCP Leased, DHCP Infin.`

**MAC0**

This item displays the MAC address of the card.

**IP0**

This item displays the current IP address of the card.

**NM0**

This item displays the current Netmask of the card.

**GW0**

This item displays the current Standard Gateway of the card.

**Contact-Status**

Displays the Contact-Status of the GPI inputs.

**Main_Active_Out**

This status item displays which input is currently active on the output of the Live Switch (see block schematic). Can be `Input_1`, `Input_2 or Input_3(TSX30 only).`

**Mon_Active_Out**

This status item displays which input is currently active on the output of the Monitoring Switch (see block schematic). Can be `Input_1`, `Input_2, Input_3(TSX30 only) or Following`. In `Following` it is following the Live Switch position.

**Input-Sum_x**

Displays the status of the summary of all the detection options. Can be `OK` (all the detections are good), `Partial Failure` or `Critical Failure` (one of the detections `ASI-Link-Det, TS-Stopped-Det or TS-SyncLoss-Det or PAT-UD-Det` failed).

**ASI-Link_x**

Displays the ASI link status of Input X. Can be `OK` or `Error`.

**ASI-Error_x**

Shows ASI Link 8B10B violations and running disparity error for input X. Can be OK or Error.

**ASI-Mode_x**

Displays the type of ASI stream detected on input X. Can be `Empty`, `Byte`, `Packet` or `Burst`.

**ASI-Stuff-Err_x**

This item indicates the ASI Stuffing Error status of the transport stream on input X. Can be `OK` or `Error`.

| | |
|---|---|
| **Min-Byte-Sep_x** | Displays the minimum byte separation for input X in ASI words. |
| **Hist-Min-Byte_x** | Records the minimum detected byte separation value for input X. Will be reset at the start of an ASI stream or with the clear setting Hist-Clear_x. |
| **Max-Burst-Len_x** | Displays the maximum burst length for input X in ASI words. |
| **Hist-Max-Brst_x** | Records the maximum detected burst length value for input X. Will be reset at the start of an ASI stream or with the clear setting Hist-Clear_x. |
| **Min-Sbyte-Sep_x** | Displays the minimum separation of packets for input X in ASI words. |
| **Hist-Min-Sbyte_x** | Records the minimum detected separation of packet value for input X. Will be reset at the start of an ASI stream or with the clear setting Hist-Clear_x. |
| **Max-Sbyte-Sep_x** | Displays the max separation of packets for input X in ASI words. |
| **Hist-Max-Sbyte_x** | Records the maximum detected separation of packet value for input X. Will be reset at the start of an ASI stream or with the clear setting Hist-Clear_x. |
| **Orig-Netw-ID_x** | Displays the Original Network ID for input X. |
| **Network-ID_x** | Displays the Network ID for input X. |
| **TS-ID_x** | Displays the Transport Stream ID for input X. |
| **No-TS_x** | Indicates the status of the transport stream in input X. Can be `Error` or `OK`. In `Error`, an ASI link can be present, filled with ASI null words, without any TS packets. |
| **TS-Sync-Loss_x** | Displays the status of the transport stream sync in input X. Can be `OK` (TS-sync is OK) or `Error` (TS-sync is lost). |
| **Sync-Byte-Err_x** | This item indicates the Sync Byte Error status bit of the transport stream on input X. Can be `OK`, `Error` or `NA`. |

| | |
|---|---|
| **PAT-Error-2_x** | Indicates the status of the PAT repetition detection of input X. Can be `OK` (PAT distance in time is belowthe threshold set with `PAT-UpperDist_X`), `Error` (PAT distance in time is above the threshold, or absent) or `NA` (not probed ). |
| **Transp-Error_x** | This item indicates the Transport Error Indicator status bit of the transport stream on input X. Can be `OK`, `Error` or `NA`. |
| **TS-Mode-Error_x** | Indicates the status of the transport stream mode detection of input X. Can be OK (TS mode detected as set with TS-Mode-Size_X), Error (wrong TS mode detected) or `NA` (not available). Will be NA when TS-Mode_Det is set to off. |
| **TS-ID-Error_x** | Indicates the status of the transport stream ID detection of input X. Can be OK (TS ID detected as set with Exp-TS-ID_X), Error (wrong TS ID detected) or `NA` (not available). Will be NA when TS-ID-Det is set to off. |
| **TS-Rate-High_x** | Indicates the status of the transport stream rate maximum detection of input X. Can be `OK` (TS rate is under the threshold set with `Max-TS-Rate_X`), `Error` (data rate is above threshold) or `NA` (Not Available). Will be `NA` when `TS-Rate-H-Det` is set to `off`. |
| **TS-Rate-Low_x** | Indicates the status of the transport stream rate minimum detection of input X. Can be `OK` (TS rate is above the threshold set with `Min-TS-Rate_X`), `Error` (TS rate is under threshold) or `NA` (Not Available). Will be `NA` when `TS-Rate-L-Det` is set to `off`. |
| **Data-Rate-High_x** | Indicates the status of the data rate maximum detection of input X. Can be `OK` (data rate is under the threshold set with `Max-DataRate_X`), `Error` (data rate is above threshold) or `NA` (not available). Will be `NA` when `DataRate-H-Det` is set to `off`. |
| **Data-Rate-Low_x** | Indicates the status of the data rate minimum detection of input X. Can be `OK` (data rate is above the threshold set with `Min-DataRate_X`), `Error` (data rate is under threshold) or `NA` (Not Available). Will be `NA` when `DataRate-L-Det` is set to `off`. |
| **PID-List_Fail_x** | Indicates the status of PID List Test detection of input X. Can be `OK` (All enabled PID test are good), `Error` (one or more PID tests failed) or `NA` (Not Available). Will be `NA` when *PID-Det* is set to `off`. |
| **Tab1-List-Fail_x** | Indicates the status of Table List Test detection of input X. Can be `OK` (All enabled Table test are good), `Error` (one or more Table tests failed) or `NA` (Not Available). Will be `NA` when *Table-Det* is set to `off`. |

| | |
|---|---|
| **Table-Tst_x** | Shows the Table List Test status per row(X). The most left character is channel 1, the middle one is channel 2 and the right channel 3(TSX30). The character can be:<br>- 'O' = Off<br>- 'E' = Not proven<br>- 'G' = Good<br>- 'F' = Failed |
| **PIDList-Tst_x** | Shows the PID List Test status per row(X). The most left character is channel 1, the middle one is channel 2 and the right channel 3(TSX30). The characters can be:<br>- 'O' = Off<br>- 'E' = Not proven<br>- 'G' = Good<br>- 'F' = Failed |
| **TS-Mode_x** | Displays the mode of the transport stream detected on input X. Can be `188-bytes`, `204-bytes` or `SyncLoss`. |
| **TS-Rate_x** | Indicates the transport stream rate on input X in packets per second. This is the sum of data and null packet rate. |
| **Data-Rate_x** | Indicates the data rate on input X in packets per second (excluding null packets). |
| **Null-Rate_x** | The null packet rate for input X. |
| **Sum-PID_x** | Shows the amount of PIDs detected on input X. |
| **Low-PID-Count_x** | Indicates the status of the minimum PID amount detection of input X. Can be `OK` (PID amount is above the threshold set with `Min-Total-PID-Low_X`), `Error` (PID amount is under threshold) or `NA` (Not Available). Will be `NA` when `low-PID-Cnt-Det` is set to `off`. |
| **High-PID-Count_x** | Indicates the status of the maximum PID amount detection of input X. Can be `OK` (PID amount is under the threshold set with `Total-PID-High_X`), `Error` (PID amount is above threshold) or `NA` (Not Available). Will be `NA` when `High-PID-Cnt-Det` is set to `off`. |
| **SFP1-Vendor** | Displays the vendor of the SFP1 module when detected. |
| **SFP2-Vendor** | Displays the vendor of the SFP2 module when detected. |
| **SFP1-Type** | Displays the type of the SFP1 module when detected. |

| | |
|---|---|
| **SFP2-Type** | Displays the type of the SFP2 module when detected. |
| **Port1-1-Enabled** | Displays the port status of port 1 of SFP module 1 |
| **Port1-2-Enabled** | Displays the port status of port 2 of SFP module 1 |
| **Port2-1-Enabled** | Displays the port status of port 1 of SFP module 2 |
| **Port2-2-Enabled** | Displays the port status of port 2 of SFP module 2 |

# 9 Events Menu

**Introduction**

An event is a special message that is generated on the card asynchronously. This means that it is not the response to a request to the card, but a spontaneous message.

**What is the Goal of an event?**

The goal of events is to inform the environment about a changing condition on the card. A message may be broadcast to mark the change in status. The message is volatile and cannot be retrieved from the system after it has been broadcast. There are several means by which the message can be filtered.

**Events**

Each event item can be set between 0..255. 0 = no event, 1..255 are the priority setting. If set to 0 no events will be generated. This information is only needed when the GPI16 card is used or when (Cortex) software is implemented. The events reported by the card are as follows:

**Announcements**

`Announcements` is not an event. This item is only used for switching the announcement of status changes on/off. 0=off, other =on

**Input1 ~ Input3**

If for input X status `ASI-Link_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_1~GENERIC_3**

If for input X status `No-TS_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_4~GENERIC_6**

If for input X status `TS-Sync-Loss_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_7~GENERIC_9**

If for input X status `PAT-Error-2_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_10~GENERIC_12**

If for input X status `TS-Rate-High_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_13~GENERIC_15**

If for input X status `TS-Rate-Low_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_16~GENERIC_18**

If for input X status `Data-Rate-High_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_19~GENERIC_21**

If for input X status `Data-Rate-Low_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_22~GENERIC_24**

If for input X status `PID-List_Fail_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_25~GENERIC_27**

If for input X status `Tab1-List-Fail_X` = `Error`, an Event will be generated at the set priority.

**GENERIC_28~GENERIC_30**

If for input X status `ASI-Mode_X` not is `ASI-Mode-Det_X`, an Event will be generated at the set priority.

**What information is available in an event?**

The message consists of the following items;

1) A message string to show what has happened in text, for example: "INP_LOSS", "REF_LOSS", "INP_RETURN".
2) A tag that also shows what happens, but with a predefined number: e.g. 1 (= loss of input), 2 (= loss of reference), 129(= 1+128 = return of input). For a list of these predefined tags see the table on the next page.
3) A priority that marks the importance of an event. This value is defined by the user and can have any value between 1 and 255, or 0 when disabled.
4) A slot number of the source of this event.

**The Message String**

The message string is defined in the card and is therefore fixed. It may be used in controlling software like Synapse Set-up to show the event.

**The Tag**

The tag is also defined in the card. The tag has a fixed meaning. When controlling or monitoring software should make decisions based on events, it is easier to use the tag instead of interpreting a string. The first implementation is the tag controlled switch in the GPI16.

In cases where the event marks a change to fault status (e.g. 1 for Loss of Input) the complement is marked by the tag increased by 128 ($80_{hex}$) (e.g. 129 ($81_{hex}$) for Return of Input).

## Defining Tags

The tags defined for the card are:

| Event Menu Item | Tag | | Description |
|---|---|---|---|
| Announcements | $01_{hex}$ = NA | $81_{hex}$ = 1 | Announcement of report and control values |
| Input_1 | $01_{hex}$ = INP1_LOSS | $81_{hex}$ = INP1_RETURN | ASI-Link_X is status OK/Error |
| Input_2 | $12_{hex}$ = INP2_LOSS | $92_{hex}$ = INP2_RETURN | |
| Input_3 | $13_{hex}$ = INP3_LOSS | $93_{hex}$ = INP3_RETURN | |
| GENERIC_1 | $3a_{hex}$ = GENERIC_1_ON | $ba_{hex}$ = GENERIC_1_OFF | No-TS_X is status OK/Error |
| GENERIC_2 | $3b_{hex}$ = GENERIC_2_ON | $bb_{hex}$ = GENERIC_2_OFF | |
| GENERIC_3 | $3c_{hex}$ = GENERIC_3_ON | $bc_{hex}$ = GENERIC_3_OFF | |
| GENERIC_4 | $3d_{hex}$ = GENERIC_4_ON | $bd_{hex}$ = GENERIC_4_OFF | TS-Sync-Loss_X is status OK/Error |
| GENERIC_5 | $3e_{hex}$ = GENERIC_5_ON | $be_{hex}$ = GENERIC_5_OFF | |
| GENERIC_6 | $3f_{hex}$ = GENERIC_6_ON | $bf_{hex}$ = GENERIC_6_OFF | |
| GENERIC_7 | $40_{hex}$ = GENERIC_7_ON | $c0_{hex}$ = GENERIC_7_OFF | PAT-Error-2_X is status OK/Error |
| GENERIC_8 | $65_{hex}$ = GENERIC_8_ON | $e5_{hex}$ = GENERIC_8_OFF | |
| GENERIC_9 | $66_{hex}$ = GENERIC_9_ON | $e6_{hex}$ = GENERIC_9_OFF | |
| GENERIC_10 | $67_{hex}$ = GENERIC_10_ON | $e7_{hex}$ = GENERIC_10_OFF | TS-Rate-High_X is status OK/Error |
| GENERIC_11 | $67_{hex}$ = GENERIC_11_ON | $e7_{hex}$ = GENERIC_11_OFF | |
| GENERIC_12 | $69_{hex}$ = GENERIC_12_ON | $e9_{hex}$ = GENERIC_12_OFF | |
| GENERIC_13 | $6a_{hex}$ = GENERIC_13_ON | $ea_{hex}$ = GENERIC_13_OFF | TS-Rate-Low_X is status OK/Error |
| GENERIC_14 | $6b_{hex}$ = GENERIC_14_ON | $eb_{hex}$ = GENERIC_14_OFF | |
| GENERIC_15 | $6c_{hex}$ = GENERIC_15_ON | $ec_{hex}$ = GENERIC_15_OFF | |
| GENERIC_16 | $6d_{hex}$ = GENERIC_16_ON | $ed_{hex}$ = GENERIC_16_OFF | Data-Rate-High_X is status OK/Error |
| GENERIC_17 | $6e_{hex}$ = GENERIC_17_ON | $ee_{hex}$ = GENERIC_17_OFF | |
| GENERIC_18 | $6f_{hex}$ = GENERIC_18_ON | $ef_{hex}$ = GENERIC_18_OFF | |
| GENERIC_19 | $70_{hex}$ = GENERIC_19_ON | $f0_{hex}$ = GENERIC_19_OFF | Data-Rate-Low_X is status OK/Error |
| GENERIC_20 | $71_{hex}$ = GENERIC_20_ON | $f1_{hex}$ = GENERIC_20_OFF | |
| GENERIC_21 | $72_{hex}$ = GENERIC_21_ON | $f2_{hex}$ = GENERIC_21_OFF | |
| GENERIC_22 | $73_{hex}$ = GENERIC_22_ON | $f3_{hex}$ = GENERIC_22_OFF | PID-List_Fail_X is status OK/Error |
| GENERIC_23 | $74_{hex}$ = GENERIC_23_ON | $f4_{hex}$ = GENERIC_23_OFF | |
| GENERIC_24 | $75_{hex}$ = GENERIC_24_ON | $f5_{hex}$ = GENERIC_24_OFF | |
| GENERIC_25 | $76_{hex}$ = GENERIC_25_ON | $f6_{hex}$ = GENERIC_25_OFF | Tab1-List-Fail_X is status OK/Error |
| GENERIC_26 | $77_{hex}$ = GENERIC_26_ON | $f7_{hex}$ = GENERIC_26_OFF | |
| GENERIC_27 | $78_{hex}$ = GENERIC_27_ON | $f8_{hex}$ = GENERIC_27_OFF | |
| GENERIC_28 | $79_{hex}$ = GENERIC_28_ON | $f9_{hex}$ = GENERIC_28_OFF | ASI-Mode_X is status OK/Error |
| GENERIC_29 | $7a_{hex}$ = GENERIC_29_ON | $fa_{hex}$ = GENERIC_29_OFF | |
| GENERIC_30 | $7b_{hex}$ = GENERIC_30_ON | $fb_{hex}$ = GENERIC_30_OFF | |

## The Priority

The priority is a user-defined value. The higher the priority of the alarm, the higher this value. Setting the priority to Zero disables the announcement of this alarm. Alarms with priorities equal or higher than the Error Threshold setting of the RRC will cause the error LED on the Synapse rack front panel to light.

## The Address

Together with the message string or the tag, the slot number or address of the card is relevant to be able to assign the event to a certain card.

# 10 LED Indication

**Reserved LED**

Not used in TSX30-20.

**eUSB Err LED**

Not used in TSX30-20.

**Error LED**

Not used in TSX30-20.

**Input_1 ~ 4 LED**

This LED indicated the presence of a valid ASI stream signal on input 1. Inputs 2, and 3(Only used for TSX30). 4 is not available in TSX30, thus this LED stay unused and is off.

**Reference_1 ~ 4 LED**

Indicated the status of the *Mon_Active_Out* status.

- `When Input_1 Only Led 1 is lit`
- `When Input_2 Only Led 2 is lit`
- `When Input_3 Only Led 3 is lit`
- `When Following Only Led 4 is lit`

**ANC Data_1 ~ 4 LED**

Indicated the status of the *Main_Active_Out* status.

- `When Input_1 Only Led 1 is lit`
- `When Input_2 Only Led 2 is lit`
- `When Input_3 Only Led 3 is lit`

**Data Error LED**

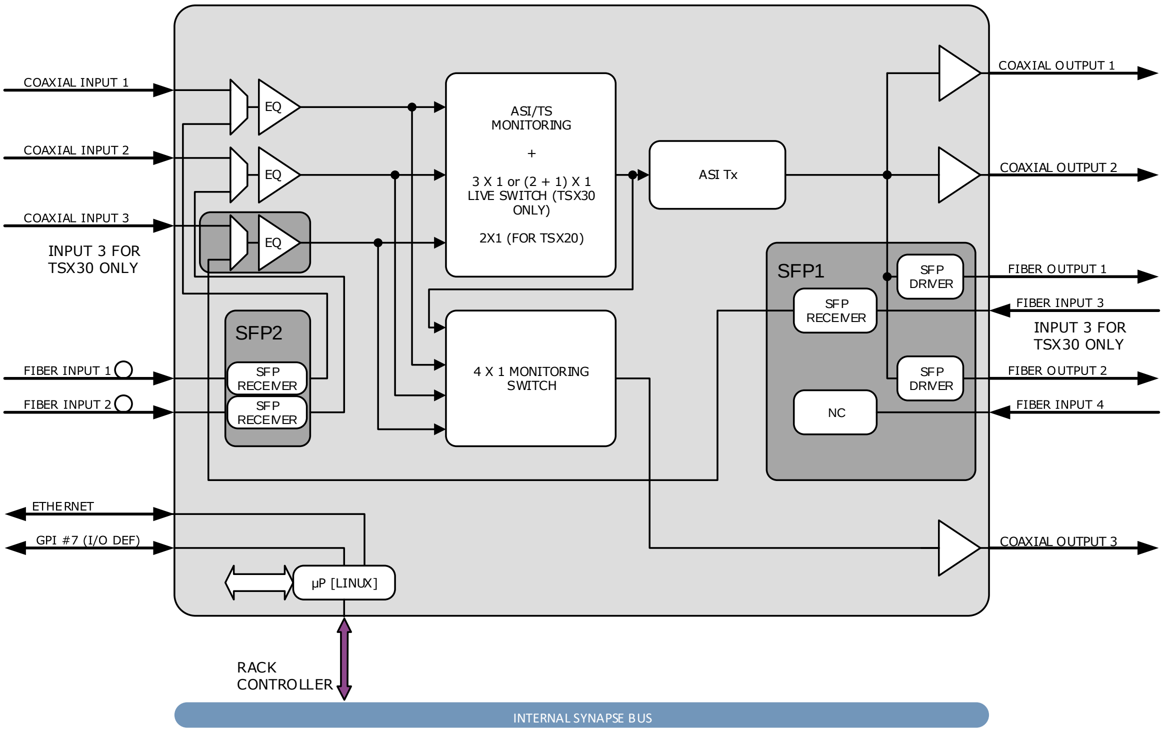
Not used in TSX30-20.

**Connection LED**

This LED illuminates after the card has initialised. The LED lights for 0.5 seconds every time a connection is made to the card.

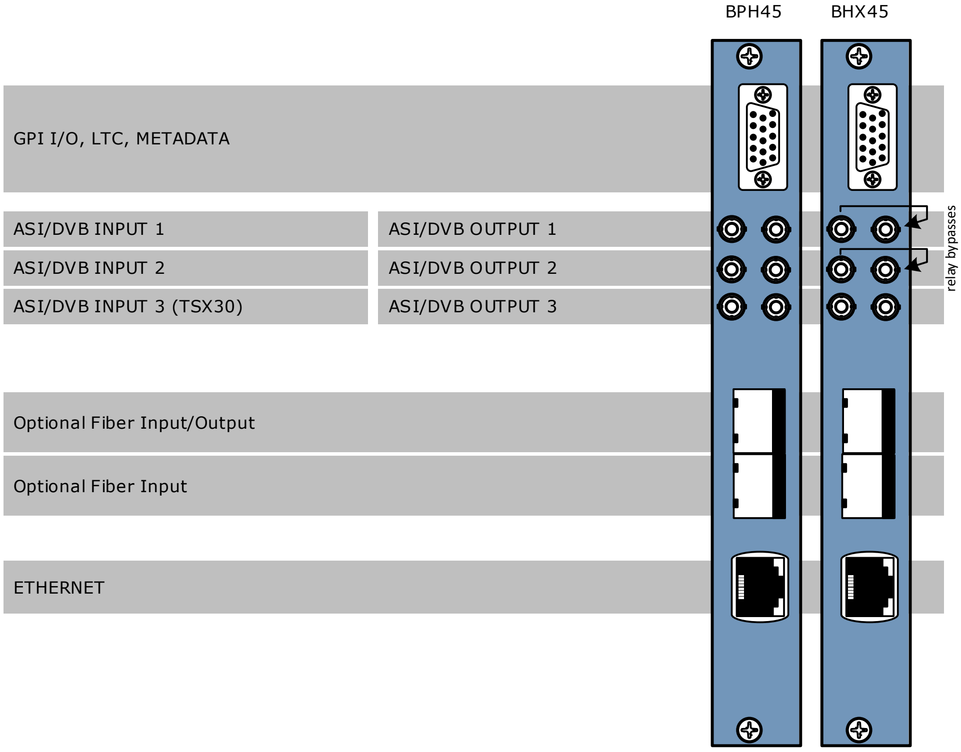
# 11 Block Schematic

TSX20 – TSX30



# 12 Connector Panels

The TSX30-20 can be used with the BPH45 or the bypass relay equivalent BHX45. The following diagram displays the pinout of these backpanels in combination with the card.

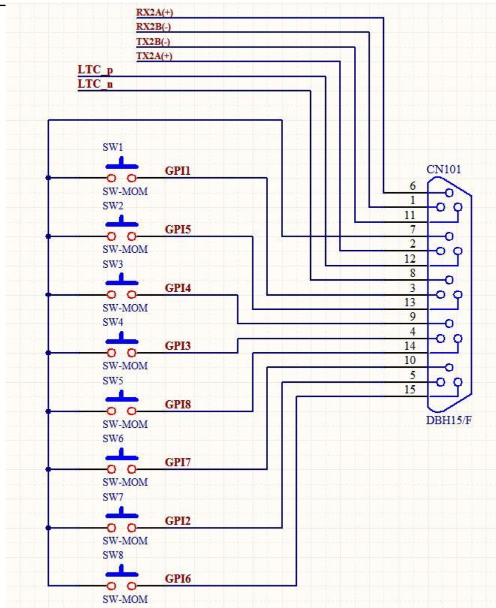



**Note** Unused inputs and outputs must be terminated with the correct impedance

## D-sub pinning

15-pin connector:

pin 01 = reserved
pin 02 = reserved
pin 03 = N_PRDY
pin 04 = N_INPUT2_GOOD
pin 05 = N_INPUT1_GOOD
pin 06 = reserved
pin 07 = GND
pin 08 = reserved
pin 09 = N_INPUT3_GOOD/N_AR
pin 10 = LIVE_SW_GPI_CTRL_0
pin 11 = reserved
pin 12 = reserved
pin 13 = N_LS0
pin 14 = LIVE_SW_GPI_CTRL_1
pin 15 = N_LS1




I ie
card.

N_LS0 and N_LS1 provide the LIVE switch routing status

LIVE_SW_GPI_CTRL0 and LIVE_SW_GPI_CTRL1 control the LIVE switch.

| LIVE_SWITCH_GPI_CTRL_1 | LIVE_SWITCH_GPI_CTRL_0 | Requested route |
|---|---|---|
| 1 | 1 | Not asserted |
| 1 | 0 | Input 1 |
| 0 | 1 | Input 2 |
| 0 | 0 | Input 3 |

N_AR is an alternative GPO mode for the 2x1 Switch Mode. AR is short for Auto-Routed.

| N_AR | N_PRDY | Interpretation |
|---|---|---|
| 1 | 1 | Forced mode |
| 1 | 0 | GPI-only OR Auto-GPI with an active request |
| 0 | 1 | Auto Mode |
| 0 | 0 | Auto-GPI with no active request |

# Appendix 1 Reprogramming the TSX30-20 module

## Before you start

**Functionality explanation**

A Synapse card's functionality is decided by 2 parts: the hardware platform and the software (a.k.a. firmware) that resides on the hardware platform. Changing the firmware of the cards means changing the way the card functions. To keep improving quality and to answer our customer's demands, Axon sometimes releases new software revisions of Synapse cards. These software revisions are formatted in 1 file per revision, with a .spf extension. Customers can download these .spf files from our website, or receive them via e-mail from our support so they can upgrade or reprogram their own cards.

**Choosing .spf files**

Not all .spf files are compatible with all hardware platforms. To know for certain that you are choosing a compatible .spf file you have to know the hardware revision of your card. This revision number can be found in the menu of the card via the control panel on the frames (select card, select 'about', check HW number) or via Cortex (Axon's control software) (select frame, select card, select 'Identity', check 'hardware rev').

Knowing the hardware revision number, you can go to our website (www.axon.tv) and go to our download firmware section. Here you select the card you wish to upgrade. You will see a list of available firmware upgrades of this particular card. The firmware files that are compatible with your card should display your card's hardware revision number in table next to "Hardware versions". If this is not the case you will not be able to upgrade your card with that file.

**Requirements**

For reprogramming or upgrading cards, you need the Cortex program installed on a PC or laptop which is connected to the same network to which the card is connected also. You can download the program free of charge from our website. For this this card you need to use Cortex version v1.091 or later. Updating the card must be done locally (direct connection) through the Ethernet of the backplane. The bottom Ethernet connection must be used.

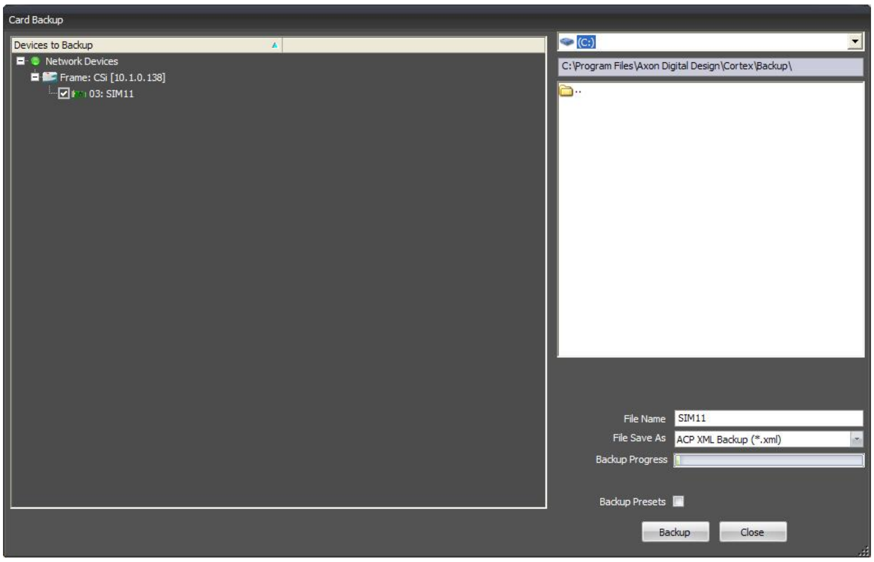
**Using Cortex help files**

This manual describes how to upgrade cards using Cortex. When you are using Cortex and require card further instructions, please refer to the Cortex help files (select 'Card' in the menu > select 'Upload Firmware' (the firmware uploading window will open) > press F1).

## Precautions

**Backup your settings**

It is advised to back up the settings before upgrading the card. To do this, select the frame and card you want to upgrade. Then choose “Card” in the menu and select “Backup card”. An exact copy of the card’s menu can be stored as .xml file in the following window. The next image displays the window where this is done.




**At your own risk**

During the upgrade process, the card will stop functioning for a period of time. Make sure the card you are going to upgrade is currently ***not*** being used by anyone in your company.

**Note**

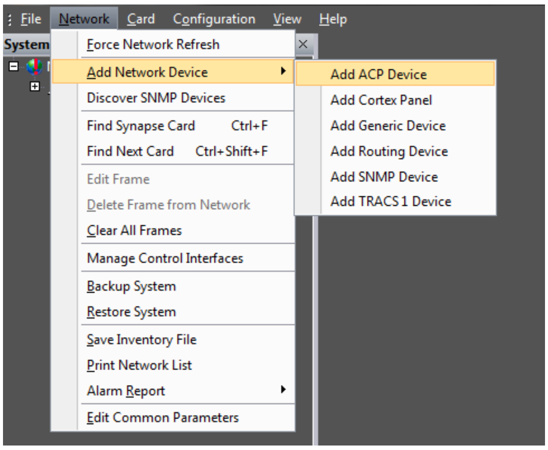
Use cortex version 1.09.01 or later. This software can be downloaded from our website. www.axon.tv

**Setting up card**

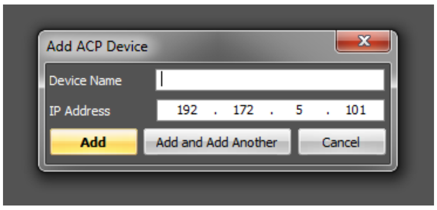
To be able to program the card direct we need to perform two steps. One is setting up of the IP address of the card and second will be making the board recognised as stand alone entity.

To set-up the IP address of the card goto the system view within the Cortex program. Select the GLI100 and goto the device view tab. Within the device tab you will be able to setup the IP address, netmask and gateway.

The next step is to make the card available as a stand alone card within the system. To add this card you need to go to the network tab at the top of the cortex program. Then go to add network device and choose add ACP device.

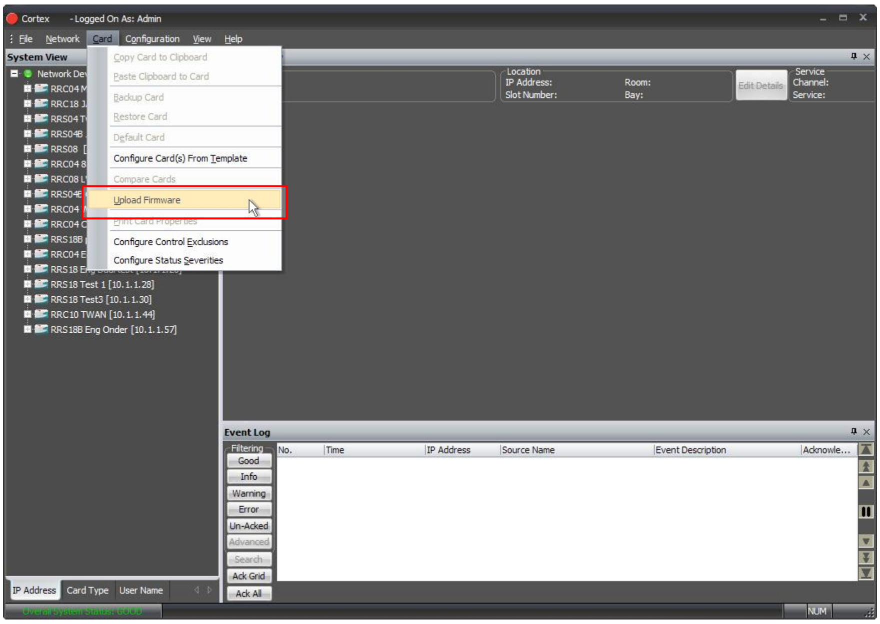



Fill out the name of the card and also the ip address.

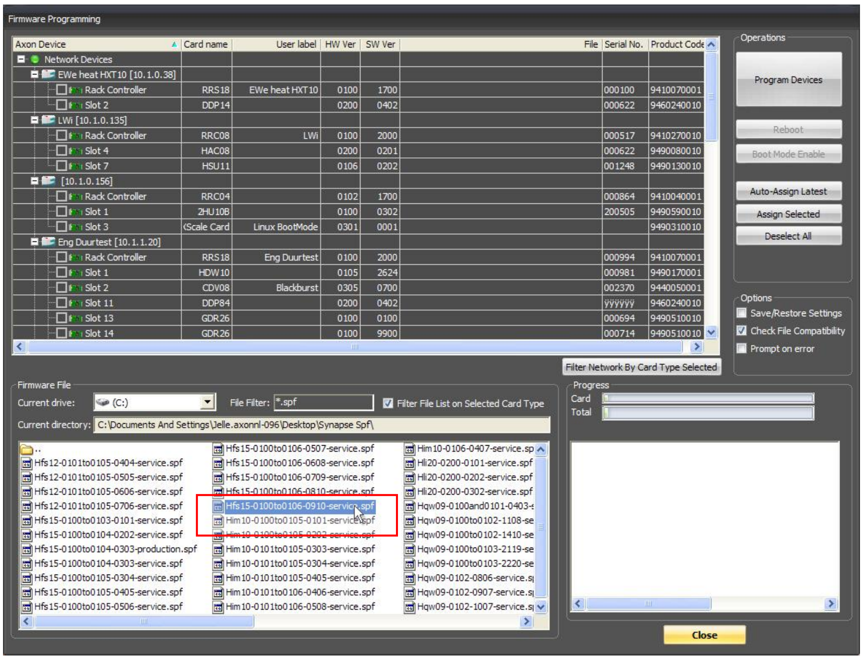


# Upload firmware

You can start upgrading the card. To do this, click ‘Card’ in the top menu and select ‘Upload Firmware’ from the dropdown box as displayed below.

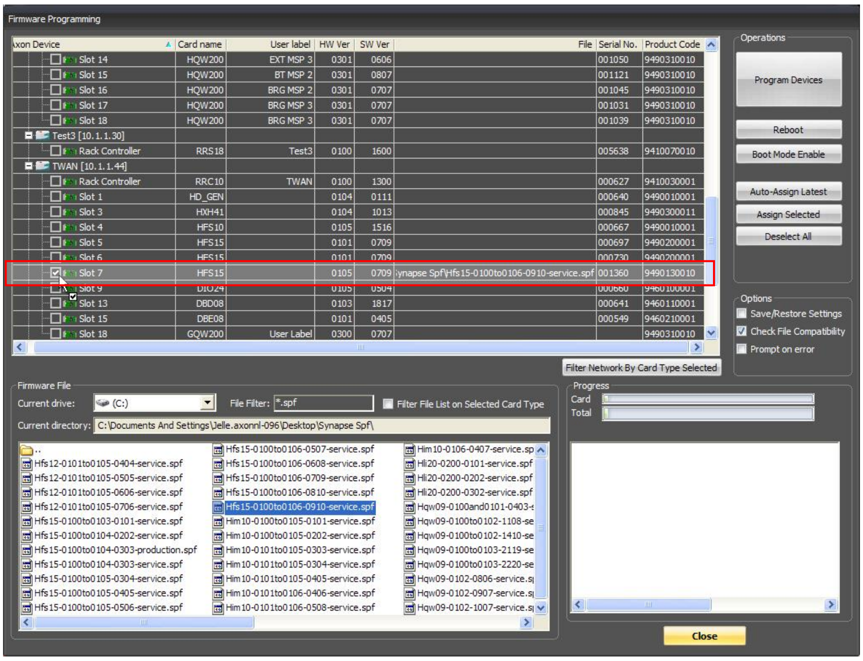



A new window will open, showing you the firmware upload functions. ***At first you must select which .spf file you want to load***. You do this in the buttom dialog as shown below.




To select which .spf you would like to upload into the card, you click the ‘Current drive’ button and select the folder which holds your .spf files.

When you selected the .spf file, check the card(s) in which you want to load this .spf file. You can load multiple cards with the same .spf file at the same time. When the selected .spf file can not be loaded in the card you try to check an error message will appear in the bottom right box. Selecting a card is done as displayed on the next page.




**Testing**

When all previous instructions have been completed the card should be functioning properly. We advise however to test the card’s functionallity before you are going to put it into real on-air use.

# Appendix 2 References

## Global Standards

**[1] ISO/IEC 13818-1** Information technology - Generic coding of moving pictures and associated audio information - Part 1: Systems

## DVB/European Standards

**[10] EN 50083-9** Cable networks for television signals, sound signals and interactive services – Part 9: Interfaces for CATV/SMATV headends and similar professional equipment for DVB/MPEG-2 transport streams

**[11] EN 300 468** Specification for Service Information (SI) in DVB systems

## ATSC Standards

**[20] ATSC A/53** ATSC Digital Television Standard

**[21] ATSC A/65** Program and System Information Protocol for Terrestrial Broadcast and Cable

## ARIB Standards

**[30] ARIB STD-B32** Video Coding, Audio Coding and Multiplexing Specifications for Digital Broadcasting

**[31] ARIB STD-B10** Service Information for Digital Broadcasting System

**[32] ARIB STD-B20** Transmission System for Digital Satellite Broadcasting

## DVB Technical Specifications (EBU/CENELEC/ETSI-JTC)

**[40] ETSI TS 101 211** Guidelines on implementation and usage of Service Information (SI)

**[41] ETSI TS 101 191** DVB mega-frame for Single Frequency Network (SFN) synchronization

## DVB Technical Reports (EBU/CENELEC/ETSI-JTC)

**[50] ETSI TR 101 891** Guidelines for the implementation and usage of the DVB Asynchronous Serial Interface (ASI)

**[51] ETSI TR 101 290** Measurement guidelines for DVB systems

# This product contains open-source software

This product contains open-source software licensed under the GNU Public License (GPL). A copy of the GNU Public License is included below. Under this license you are eligible to receive a copy of the source code of this software including any changes.

Axon Digital Design shall provide the source code on request either through physical distribution or electronic communication. For physical distribution you may be charged a fee that covers distribution costs. This offer is valid up to three years after date of purchase. Please direct your request to the support department of Axon Digital Design.

Axon Digital Design supports open-source software by participating in the development of open-source projects or submitting improvements to these projects. For more information see http://opensource.axon.tv/

# GNU Public License version 2

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifi cations and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modifi cation".) Each licensee is addressed as "you".

Activities other than copying, distribution and modifi cation are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the fi les and the date of any change.
b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in objects code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifi cations to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface defi nition fi les, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.