Installation and Operation Manual

June 2017

Congratulations with your SKAARHOJ controller! Our mission is to ease the use of broadcast hardware for people making live video and doing so by means of awesome tactile industrial strength hardware controllers. We are really proud of how much we have been able to stuff into this device, and we hope you can see our love and passion for cool and helpful technology shine through when you browse this manual.

While we really want to make everything intuitive for you, we still need to document some not so obvious facts and conventions and we have tried to put everything you need to know into this document along with a few extra tips too.

Please notice this manual is aimed at UniSketch powered SKAARHOJ controllers. If you have a device not running UniSketch please see other manual at http://skaarhoj.com/support/manual/. In order to run UniSketch you need a device with the SKAARDUINO Due MCU.

Have fun!

Quick info

- **Connecting everything** - see "Connecting SKAARHOJ Hardware with Devices"
- **Accessing web interface** - see "Configuration Mode/Accessing Web Interface"
- **Updating Firmware** - see "Uploading New Firmware"

Supported Hardware Devices in UniSketch

To get latest updates on supported hardware devices go to: www.skaarhoj.com/support/device-cores/

Contents

Connecting SKAARHOJ Hardware with Devices

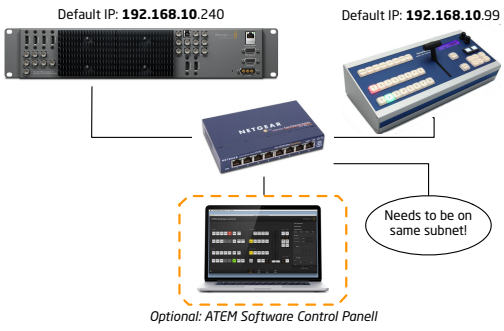
A SKAARHOJ controller needs a network connection to connect to devices you want to control. If you want to control an ATEM Switcher you can connect it directly to the ATEM, or via a network hub or switch that connects the devices.

If you had your controller delivered with a standard configuration, the IP address is set to **192.168.10.99**, with subnet 255.255.255.0.

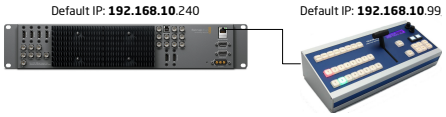
Remember that the IP address for all the devices you wish to communicate with should be in the same subnet (192.168.10.*).

If you want to control for example an ATEM switcher you do *not* need a PC/Mac running the ATEM Software Control Panel for using the SKAARHOJ controller since it connects directly to the ATEM. But you can run both at the same time if you want, and the changes made in either the SKAARHOJ controller or in the ATEM Software Control panel will correlate with the other unit. This is the case for most device cores you can install on your controller, but for more information, consult the support pages for the individual device cores.

Please notice if you have multiple SKAARHOJ units connected they *need* to have different IP addresses.



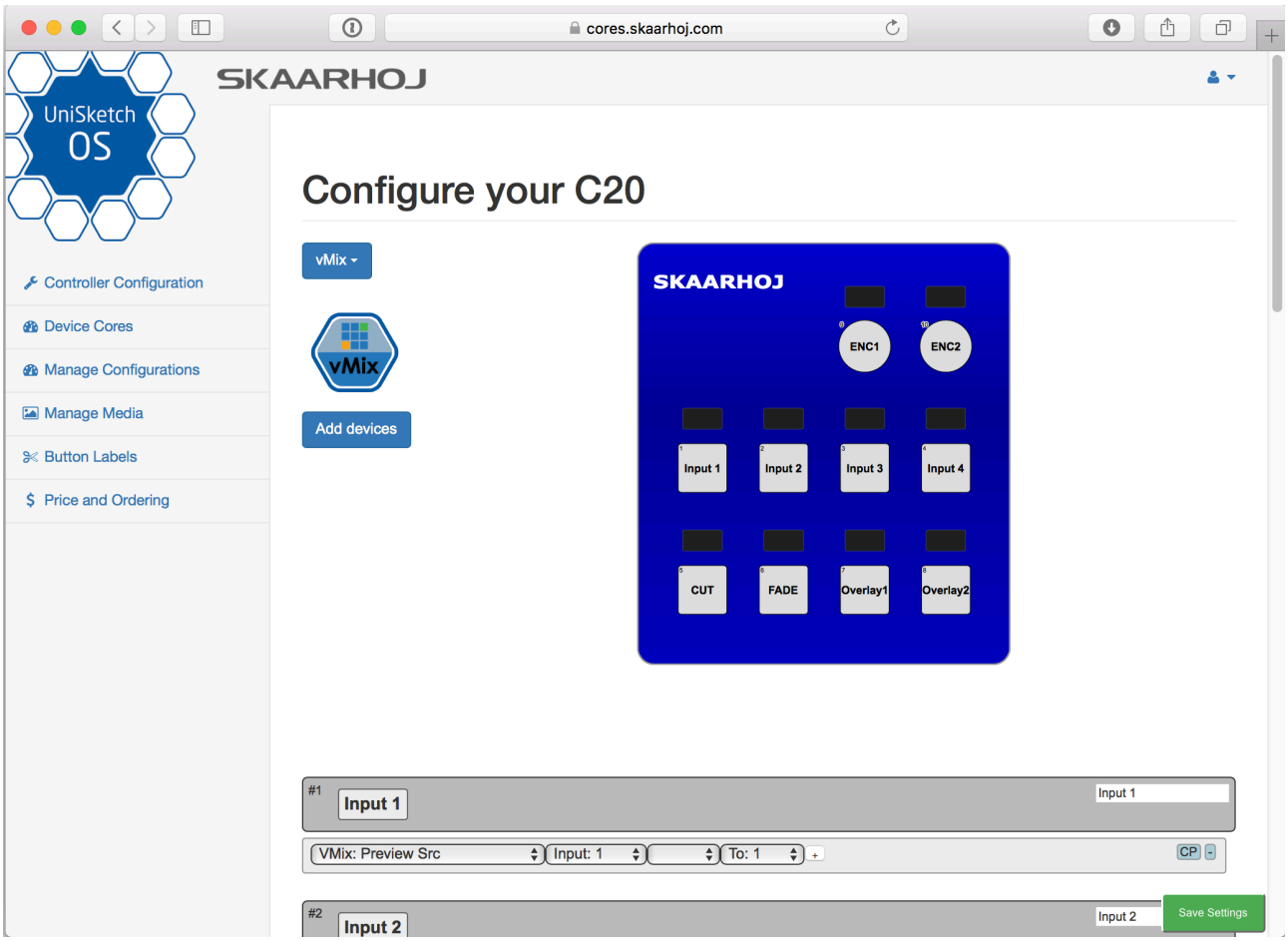
Connection via hub/switch



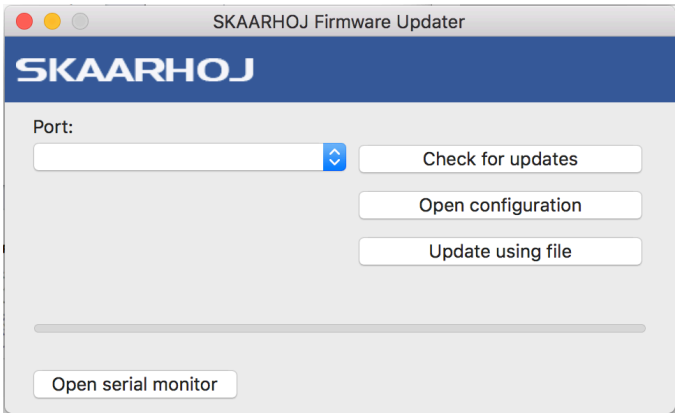
Direct Connection

cores.skaarhoj.com

A SKAARHOJ controller running UniSketch OS has a license profile in our online repo at cores.skaarhoj.com. This is where its software (firmware) came from that was programmed onto the unit when it was delivered. Whenever you need a firmware upgrade for whatever reason, this is where the firmware will be downloaded from by the firmware application. But this is only the most basic case. cores.skaarhoj.com provides you a fantastic way to expand the features of your controller; you can install new device cores (support for more broadcast hardware), change configuration of the controllers behaviour, add media strings and graphics, add modules, print button labels and manage multiple configurations for your controller.



On “Cores” (cores.skaarhoj.com) the configuration of your controller shares the same interface as the one documented in this manual in the UniSketch section. The configuration you set up on “Cores” will become the default configuration of your controller with the next firmware download while the configuration set locally on the controllers own web interface will stay there. We recommend using “Cores” for configuring your controller since you have access to more options and the latests software updates to UniSketch.

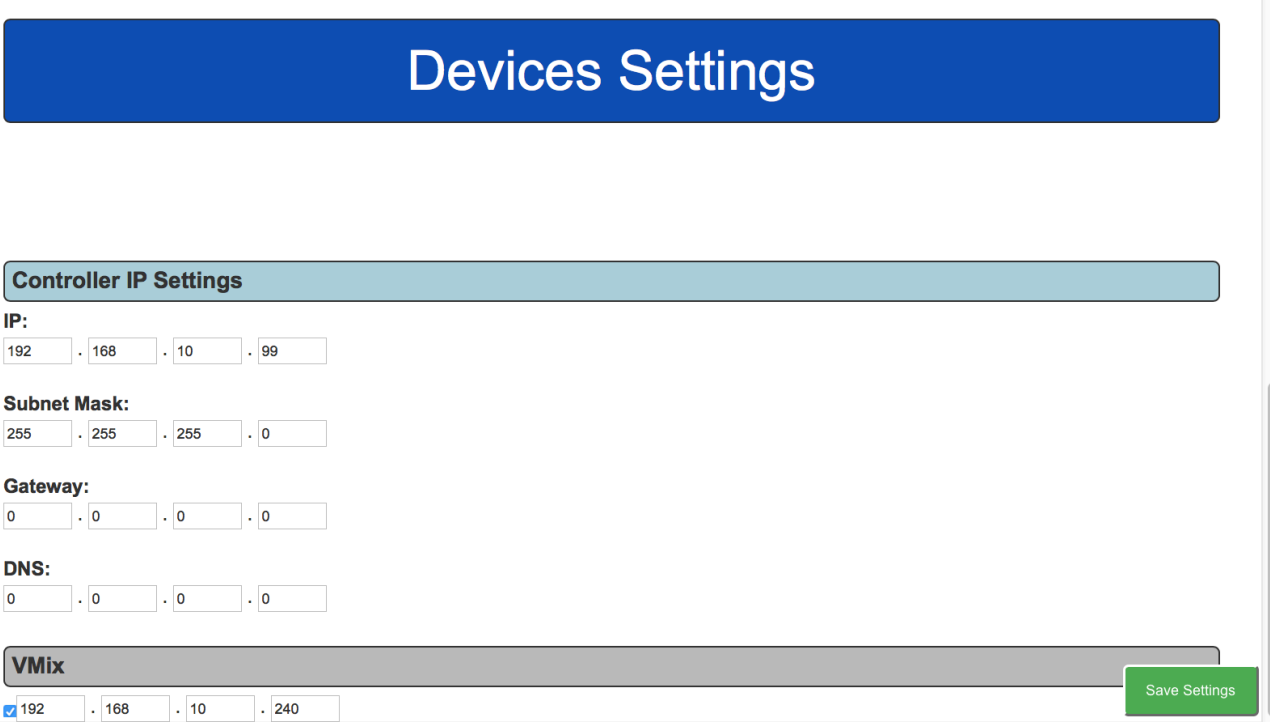


To access the “Cores” configuration for your controller, simply connect it by USB to your computer and start the SKAARHOJ Firmware Application, then press “Open Configuration”. If the controller correctly reboots and returns

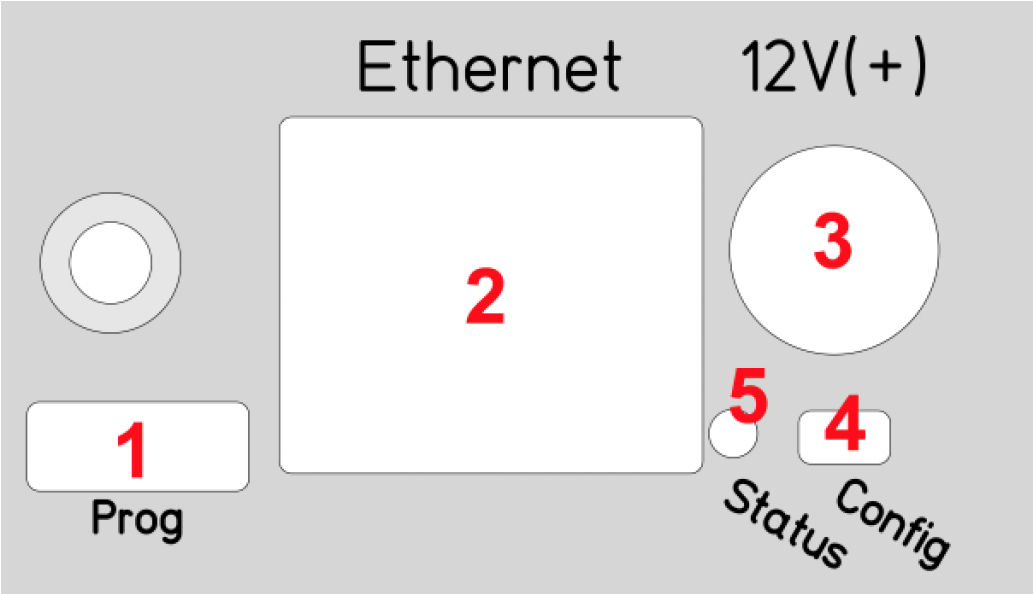
its unique ID, a web browser will open and take you to the configuration page on cores.skaarhoj.com. Whenever you are done changing the configuration online, return to the SKAARHOJ Firmware App and press "Check for updates". A new firmware reflecting the changes will be created and downloaded to your controller.

Checking the default configuration

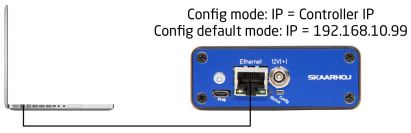
Whenever this manual refers to your controllers default configuration, it's a reference to the active configuration for the controller on "Cores". So, to check if the default IP address really is "192.168.10.99" as is often mentioned, you can simply use the Firmware Application, press "Open Configuration" and browse your controllers set up online, including such as the IP addresses:



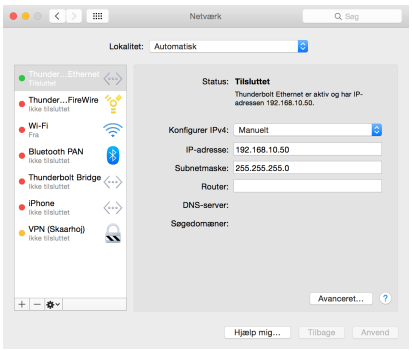
Back Connections and Config Mode



1. **Micro USB plug.** Used for service monitoring and programming new software into the unit ("firmware upgrades"). For some device cores this may also be used for controlling equipment. See "Programming"
2. **Ethernet Jack.** Connect this to your ethernet switch. This jack may also support PoE (Power over Ethernet) if your SKAARHOJ controller was delivered with that option. When connected to a network switch, the yellow LED (lower left) will be on. When data is sent to/from the controller, the green LED (lower right) will blink. If the device in the other end supports TX/RX auto detection you may be able to connect the SKAARHOJ controller directly to your device, otherwise use a crossed cable or a network switch (the supported setup).
3. **DC input.** Use a standard 2.1mm center pin plug (center = "+"). Allowed voltage range is 7-18V. We test controllers will work at 12V. The device uses max 1A at 12V. Units with BlackMagic 3G-SDI Arduino Shields needs 12V.
4. **Configuration/Reset button.** Use a pencil or tooth pick to press the button. When you press the button shortly, the controller will reset (same as pulling the power plug). If you press and hold the button, you can reset the controller into configuration mode:
    1. Press and hold the button until the status LED becomes blue after a few seconds. Release the button and the controller is in *config* mode. You can access the controller web interface with a web browser on "http://[CONTROLLER IP]/" where CONTROLLER IP is the IP address used for the currently loaded preset. The controller will run a diagnostics mode after 2 minutes where displays and buttons will light up.
    2. Press and hold the button longer until the status LED becomes white (which is 2 seconds after becoming blue). Release the button and the controller is in config default mode. You can access the controller web interface with a web browser on "http://192.168.10.99/". The controller will immediately run a diagnostics mode where displays and buttons will light up.



Direct connection to PC or Mac



Manual IP settings for Mac

Manual IP settings for PC

3. Press and hold the button even longer until the status LED becomes red (which is 10 seconds after becoming blue and 8 seconds after becoming white). This will clear all presets in memory thereby resetting all configuration made in the controllers own web interface (this corresponds to the serial monitor command “clearpresets” and should only be necessary in case a firmware update requires it or if there is another tricky error state present).

In rare cases you cannot rely on the reset button but have to turn of the power to the controller instead (“cold start”).

5. **Status LED:** When the controller is just powered up, you will see the status LED blink purple during the boot process. In this process, the hardware is initialized. Eventually the LED should end up blinking slowly (2 sec period) steady green (or blue or white if in config modes). If the LED blinks yellow quickly it indicates that connection to one or more devices is not established. This is perfectly normal for a few seconds between the boot up process (purple blinks) and the operational state (green blinks) when the controller connects to all devices. The status LED should never be permanently on or off, this indicates a potential freeze in the system. In fact, in normal healthy operation the LED should blink with a steady 2 sec period, otherwise it could indicate trouble with connections or hardware. The LED will also light up red for a split second whenever an analog hardware component (such as a T-bar) is operated (this feature is helpful to determine if calibration is needed).

Status LED Overview

Purple blinks, uneven durations	The controller is booting up (and for each blink a given step has been completed).
Yellow blinks, quickly	The controller hasn't established necessary connection to one or more devices. At the end of the boot process this is natural for a few seconds as the controller connects to devices for the first time. If you unplug the network cable or turn off an external device the controller is connected to, you will also see this state. Just turn on the external device again or re-insert the cable and the error state should restore itself to normal operation (green, steady blinking) after some time. If this happens during normal operation and without obvious explanations (like removal of a network cable or shutting down an external device), it's an error state you need to pay attention to and bug-fix further. If the controller boots up and never stops blinking yellow, you may want to check if you have configured devices for the controller which are not currently present in the network set up. Go to config mode, enter the web interface and check which devices are enabled and their IP addresses.
Green blinks, steady, period of 2 seconds	Normal mode, everything is connected and working properly. Just bliss
Blue or white blinks, steady, period of 2 seconds	Config mode (white: “config default”) where you can access the controller web interface.

Red blinks (interrupting green or yellow blinks)	This happens if you move an analog hardware interface component like a T-bar or knob and is totally normal in that case. However if such blinks happen without you touching any analog components it indicates the need for calibration.
Quick red blinks and no response from controller	A problem with the preset memory checksum indicates that the preset memory may be corrupt. However, in most cases a “cold start” by removing the power supply, waiting 10 seconds and connecting the power supply will solve the problem. If after 2-3 attemps with this solution it still remains a problem, you must clear the preset memory. This is done by holding the config button pressed, then apply power to the controller and wait for around 15 seconds until the status LED becoems solid red (before that state, the LED should be first purple, then blue, then white for 8 seconds and finally solid red). When the LED is red, release the button and the status LED should start blinking again and the controller should boot up. Notice that your controller will be reset to factory settings in this case and you may need to reload or recreate your configuration.

Connection Troubleshooting

When you have a "blinking-yellow-quickly" situation, you need to figure out which device is not connected. Try some or all of these things:

- You should bring the controller into config mode and access the web interface in order to check which devices are enabled and what their IP addresses are.
- Make sure the SKAARHOJ controller itself has the expected IP address and subnet mask.
- Make sure the devices you have setup actually are on the network, can be ping'ed and responds to their respective other types of software connecting over IP.
- Unpower all devices and your network switch for 10 seconds and power them up again.
- Connect a computer to the USB port of the SKAARHOJ controller and open the serial monitor using the firmware application to see the output from that. This provides the most direct information about which devices are not answering and at which IP. You can also see the controller IP and Mac address here. See "Serial Monitor" section.

Configuration Mode/Accessing Web Interface

In Configuration Mode all device communication is disabled and instead the controller provides a **web interface** for configuration of the interface component behaviors. The IP address of the controller in configuration mode (or "config" mode) depends on how config mode was entered. There are two options: "config" mode or "config default" mode. You can enter either mode using

A. The Config/Reset button on the controller

B. Using the serial monitor command "config" or "configd".

- **"config" mode:** The controller IP address is the "last used" IP address; the one set up for the currently loaded preset. This is convenient most of the time since you probably know your controller IP and just need to boot in config mode and access the web interface with a browser. After 2 minutes in config mode, the controller will run cyclic test programs on the hardware interface components, typically a lot of blinking.
- **"config default"** mode changes the IP address to 192.168.10.99 and this is useful if for some reason you forgot the controller IP or otherwise want to make absolutely sure you know the right IP address for the controller. The controller will run cyclic test programs on the hardware interface components immediately as it has booted up.

Notice that the IP address of your SKAARHOJ controller can be different for each preset you have! This makes it easy to have presets for completely different network and device configurations.

We recommend connecting your SKAARHOJ controller directly to your PC/Mac with a ethernet cable in order to access the web interface

Turn off your Wi-Fi and set your IP manually: (these numbers are only valid if you are using the default IP address configuration from the factory)
192.168.10.50 PC/Mac IP address
255.255.255.0 Netmask
192.168.10.1 Gateway/Router IP address if necessary

Enter configuration mode

- **Step 1** - Connect the SKAARHOJ controller to PC/Mac with ethernet cable and power up the unit. Let it boot up.
- **Step 2** - Press and hold the config button until the status LED becomes blue. Release the button.
- **Step 3** - Access web interface by entering http://[CONTROLLER IP]/" where CONTROLLER IP is the IP address used for the currently loaded preset in your browser.

Access to web interface if SKAARHOJ controller is connected via network router

We always recommend a direct ethernet connection with manual IP in order to gain access to the web interface. But depending on your network router settings you might be able to connect to the web interface without setting a manual IP address on your PC/Mac.

Web Interface Troubleshooting

If you are having problems connecting to the web interface there are numerous things that can trick you:

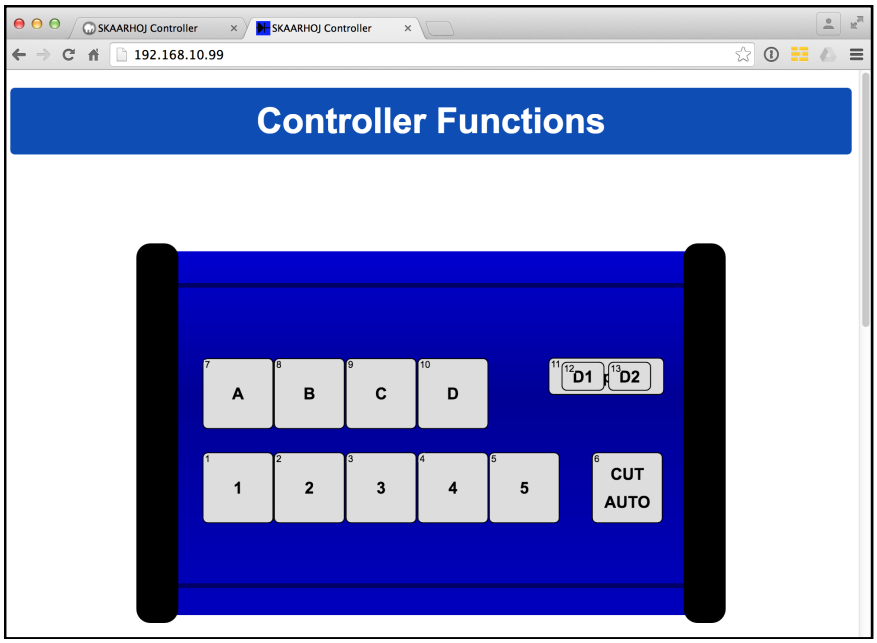
- Reboot your SKAARHOJ controller completely (disconnect for 10 seconds, re-power)
- Reboot your network switch. Sometimes it may hold old information about the controller IP and MAC addresses and to clear this out, reboot it (disconnect for 10 seconds, re-power)
- Check connection to the assumed SKAARHOJ controller IP from your computer with the "ping" command. You computer IP settings must be correct too.

UniSketch OS Explained

Web Interface

When your controller is in **configuration mode**, you will be able to access it's **web interface** with a web browser. We recommend using the Chrome browser because it's faster, but we have successfully tested it with other modern browsers such as Firefox, Safari and Internet Explorer in the latests version.

Notice that the web interface heavily depends on modern JavaScript and may malfunction with older browsers!



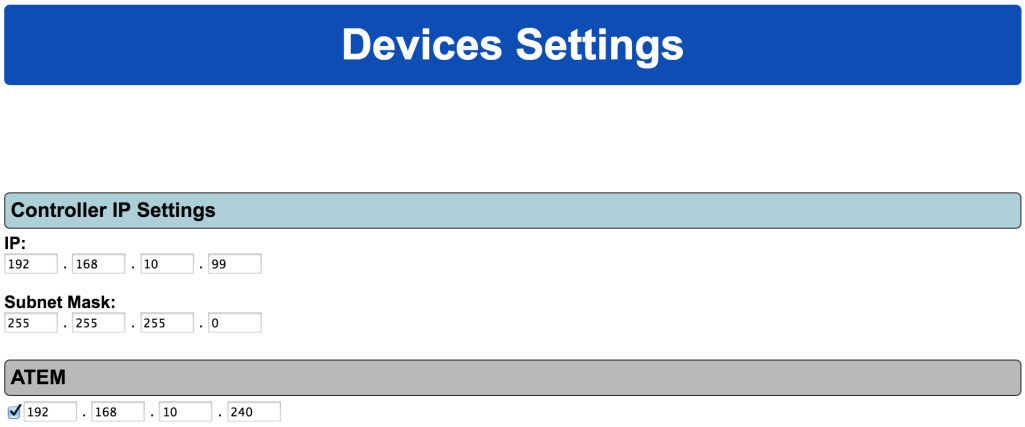
Example of web interface for SKAARHOJ controller.

Device Settings

You set up the IP address, Subnet mask, Gateway and DNS of your SKAARHOJ controller in the web interface. Likewise any external IP device your controller is configured to work with is listed here. Any device you want to be active must be enabled here and have a valid IP address set up.

Notice that all these IP settings will be saved with each preset in the controller. This is quite awesome because different presets allows you different IP configurations so a controller can easily move between different hardware contexts.

The number and type of external devices listed is compiled into your controller when it is delivered. The list may vary from controller to controller and can potentially be expanded or change to include other hardware by time. The "default" configuration for your controller is determined by the settings for it made on cores.skaarhoj.com.



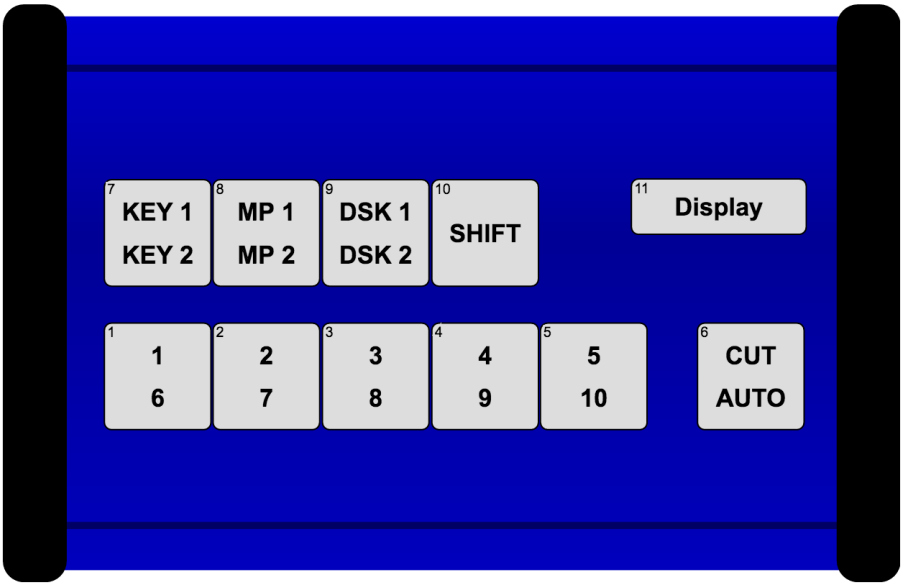
Hardware Interface Components (HWC)

Your SKAARHOJ controller consists of a number of hardware interface components such as buttons, knobs, dials, levers, displays, LEDs, joysticks, plugs, etc. They generally fall into broad categories such as inputs (eg. buttons and knobs, GPI) or outputs (eg. displays, LEDs, relay). Sometimes they can be both (most buttons have an illumination color, some even have a display on them). Input elements can be sub categorized as:

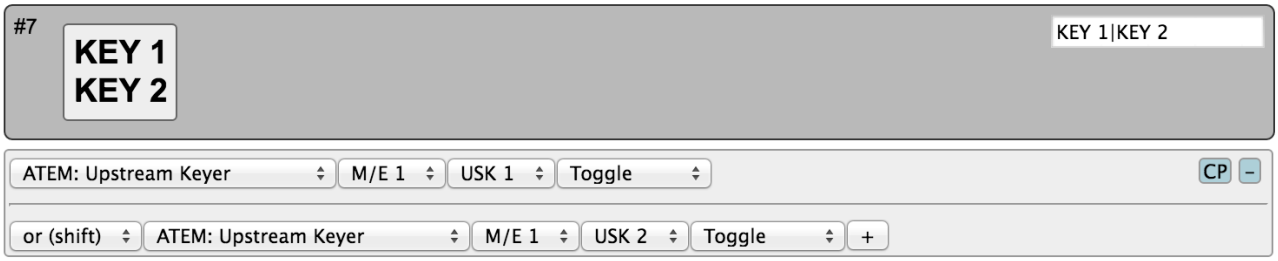
- "binary" such as a button or GPI trigger which is either pushed/triggered or not. Sometimes holding down a binary input has a special function.
- pulse-generators such as encoders which are knobs that can rotate indefinitely in both directions and sends a corresponding number of pulses. Encoders typically has a fine/coarse adjustment mode which is toggled by pressing it. Pressing and holding an encoder down for 1 second will typically send a binary "button down" signal to the interface component. Often this can function as a "reset" feature.
- analog signals such as T-bars or joysticks which provides a free value within some range.

Outputs range from a simple binary output like a relay to red/green LEDs, an array of LEDs (like a VU meter) or a graphical or text based display.

Hardware Interface Components are configured with actions which is what give them their function. This is described in the next sections.



An example of the schematic drawing of a controller as found in the controllers web interface which is available when booted in configuration mode.

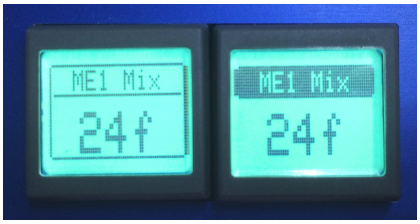


This is how the button "KEY1 / KEY2" is configured in the web interface. As it appears, this button will toggle upstream keyer 1 or 2 on an ATEM switcher depending on whether the controller is in shift-state or not.

Graphical Displays

Displays are found in many configurations on SKAARHOJ controllers. One of them is Smart Switches which are buttons with a display on. Otherwise displays are typically stand alone but can be configured to reflect a certain button on the controller (through the "Tie to HWC#" system action). An important convention with displays is whether it works as a label or displays a current status. Take the picture below as an example. Here there are two SmartSwitch buttons apparently showing the same thing:

- **A label:** The button to the left is configured to set the frame rate of Mix transitions to a fixed "24f". This can be seen from the fact that the button has a non-solid header bar. This is a *label* that simply tells us what will happen if you push the button: You will set a 24 frame mix transition rate.
- **A status:** The button to the right is configured to also set the frame rate of transitions but is configured to act in "Cycle" mode so when you press the button you will cycle through transition types and values by some scheme. The important thing is that the button shows the current value for Mix transitions – 24 frames. And if we change the value to 30 frames, the button will show 30 frames. This is a *status* that informs you about the current value of this ATEM feature and this can be seen from the fact that the button has a *solid* header bar.



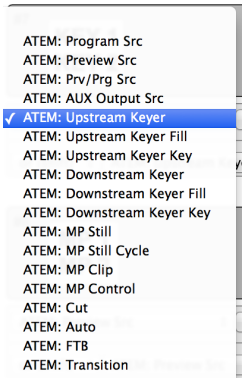
This convention works throughout all displays on your SKAARHOJ controller. Generally, a display will show the value status unless it has been tied to a button-type interface component in which case a label is typically shown – unless the particular button operates in some sort of cyclic mode where a status makes more sense to display.

Devices

You configure your controller by assigning actions to interface components. An action is most typically a command sent to an external device, such as a video switcher, router, recording deck, monitor etc. External hardware support is made available on your controller as what we call *device cores*. A device core is the driver - or akin to an "app" - installed on the your controller that enables communication support with a given piece of broadcast gear.

Some actions may also relate to internal registers or "system functions". For instance you can have a button set or clear a "shift" value which the rest of the interface components will adapt itself to.

Since interface components can be inputs and/or outputs and of various types, the way they affect any given device via an action is a fixed interpretation coded into the system.



An excerpt of the list of ATEM switcher related actions.

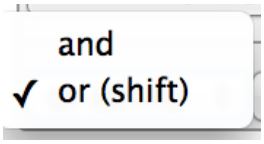
Multiple Actions

You can assign multiple actions – even on different devices – to any interface component. This is done by simply pressing the "+" button and setting up the new action:

In this case, the media player 1 (MP1) is brought on Preview on an ATEM switcher and right after the still number 5 is selected for the media player 1.

The return values, including those driving a display, will always come from the first action in the list (of the current shift level).

Notice that multiple actions are separated by an "operator" which is either "and" or "or (shift)":



This is explained in the following.

Shift

You can assign a button to set a shift state on your controller. Even though a shift state sounds like an either/or option, we have implemented the possibility to have multiple *shift-levels*. However, in the simple case, a shift button would be configured as shown below:

This will set the shift-level "1" in the system as long as the button is held down, otherwise it will be "0" (normal)

This means another button on the same controller could be configured like this now:

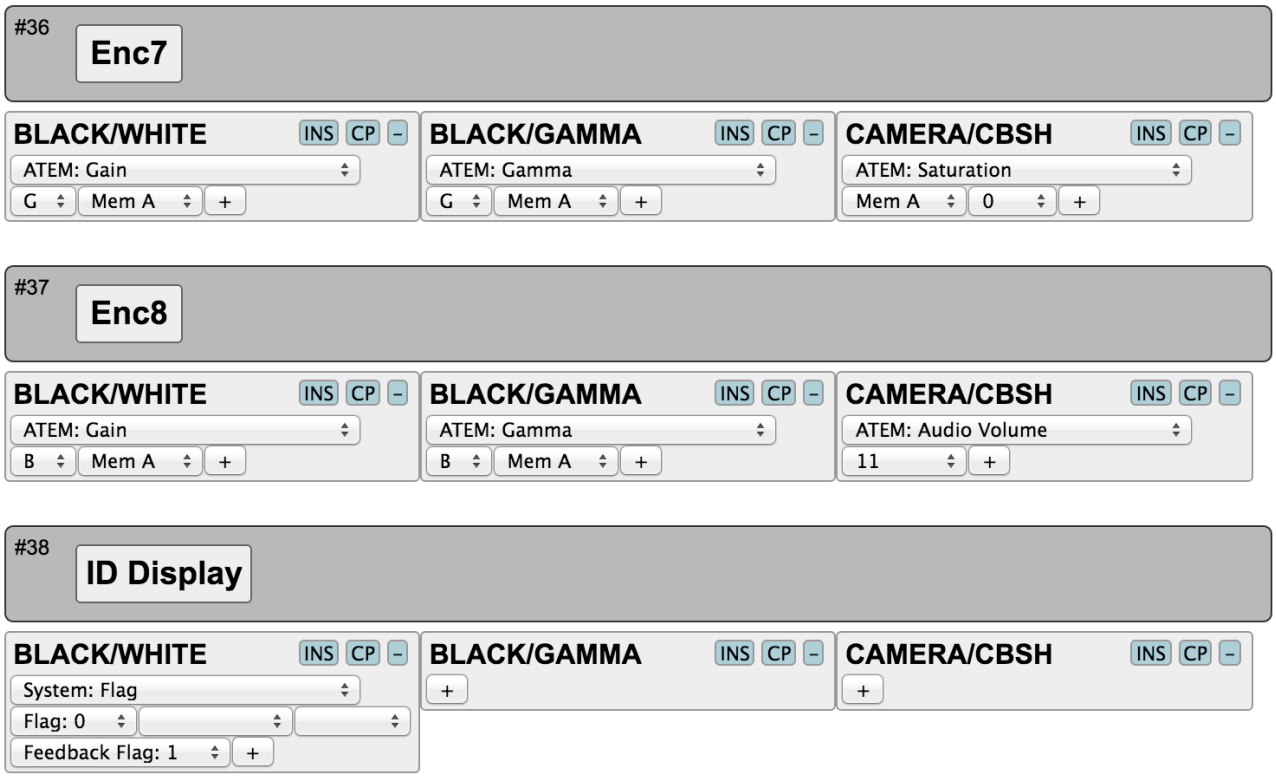
And because the divider between them is "or (shift)", the second action is only active when the shift button is held down.

If no specific action is defined for a shift level, the interface component will use the default list of actions.

If you insert additional "or (shift)" dividers, it will define how shift levels 2 and beyond will act. Within each shift level you can have multiple actions (see previous section).

States

Similar to shift levels you can put your controller in various states. States are mainly different from shift levels by the way the interface lets you set them up. With three states you see three columns of actions for each interface component:



Like with shift levels you can assign other interface components to change the state of the controller. If actions are not defined in any given state, they will fall back to the action list in the first column (Normal state).

States and shift levels can be combined of course; you can have unique shift levels inside each state.

States can also be named. In the above example, they are named "BLACK/WHITE", "BLACK/GAMMA", "CAMERA/CBSH". This is done in the controller web interface as well:

This is where you select how many states the controller should support.

Copy / Paste

Often you will find yourself needing to set up almost the same function on multiple interface components (such as a row of buttons, all sending inputs to an AUX channel). To make this easy, make sure to use the Insert / Copy / Delete functions:

As soon as you make any change to a given interface components action list, this will be copied to memory so you just need to go to the next interface component and press "INS" for insert.

Presets

At the bottom of the web interface you can load, save and reset your presets. Your controller can theoretically hold any number of presets only subject to the memory usage related to storing them. Pressing the save button in the web interface will save the configuration to the currently selected preset, but you can also select a new or different preset to save to using the selector box.

Load the "(Default)" preset to get back to the factory configuration.

If your presets seems to be messed up for some reason (could be memory overflow which there is currently no protection against) you may need to clear the entire memory by using the serial monitor command "clearpresets" (see later).

Presets are a very powerful way to make use of your controller in multiple places since it can change the entire behavior of the controller including which devices to connect to and which IPs they are on.

There is a nifty way to load presets on most controllers if you have created more than 1 preset: When you boot your controller, you may see that a number of buttons (corresponding to the number of available presets) light up for a few seconds and one of them being highlighted. The highlighted button indicate the currently loaded preset and the other buttons represent other presets. If at this moment you press and hold any of the other buttons down until that button light up, you will then change the preset of the device (corresponding to selecting it in the web interface and press the "Load" button).

Presets are a great way to manage multiple configurations in off-line situations. However, using cores.skaarhoj.com and the firmware application to manage multiple configurations for your controller is more powerful in the sense that you can include changes to installed device cores, string and image media and also have the latest software updates installed.

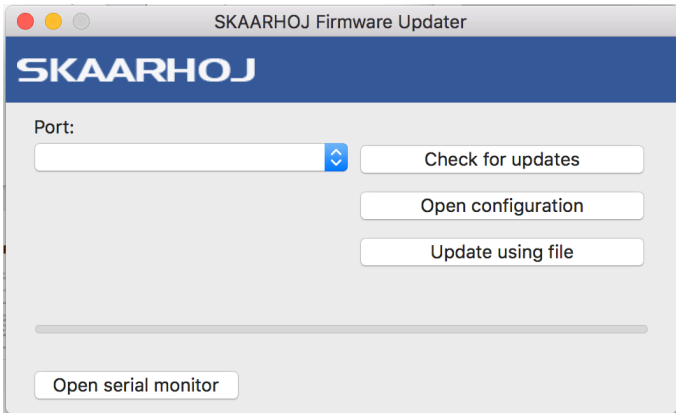
Uploading New Firmware

For uploading a new firmware please use our Firmware Updater Application. Go to:

http://skaarhoj.com/support/firmware-updater/

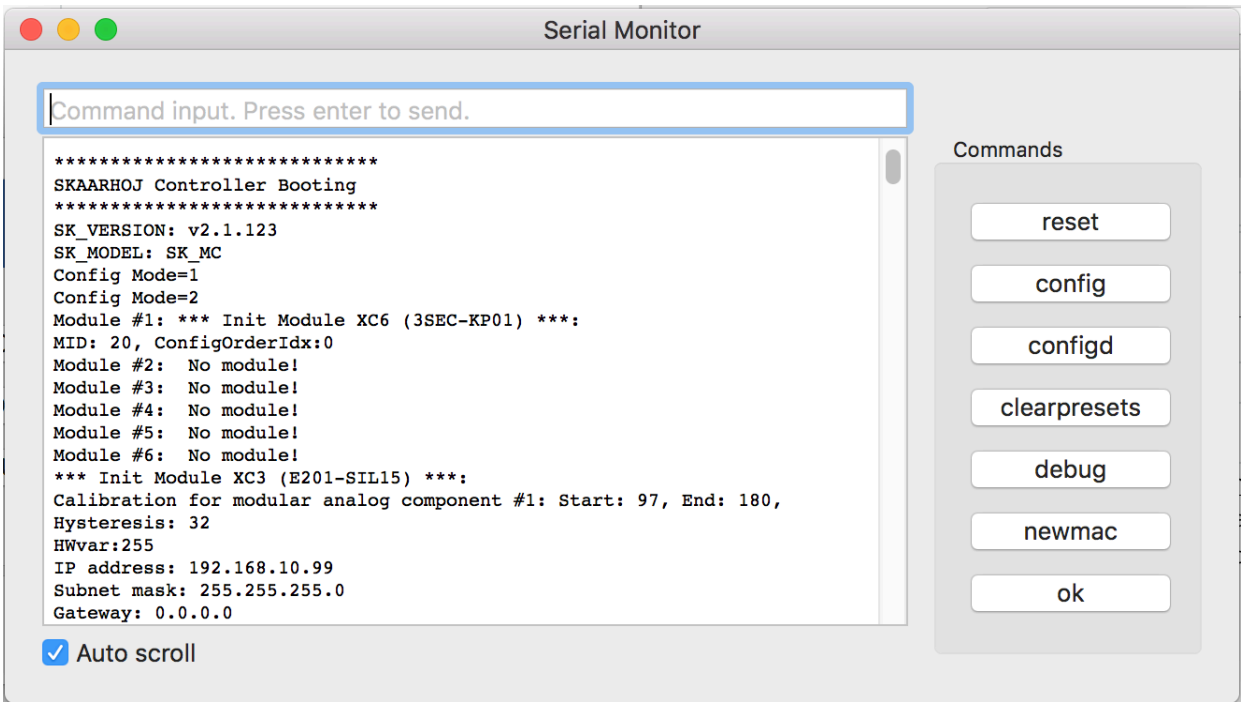
When the application is installed and open, connect your controller with USB to the computer. It should now be displayed in the "Port" dropdown box. Then press "Check for updates". After a while a new firmware should be downloaded and installed.

In case you have a firmware hex-file to upload, use the "Update using file" button that will let you use a file based workflow.



Serial Monitor

The serial monitor in the Firmware Application is an indispensable tool for bug fixing any problems with your SKAARHOJ controller. The serial monitor runs at 115200 baud and a typical output from the boot process looks like this:



This tells us the model name of the controller, how a number of hardware components have been initialized, that preset 1 is loaded, which IP address, subnet mask and MAC address the controller has, which date the software was compiled.

It also shows us which hardware devices it will try to connect to.

During this process until the "setup() Done" message is output, the status LED will blink purple.

After the setup, the controller enters normal operational state. You see that it tries to connect to the devices and that it succeeds in this. During this process, the status LED blinks yellow and eventually it will blink green.

The serial monitor will continuously output a small dot and a number every second. If this is not the case permanently, it indicates a crash of the controller. The number indicates the number of times a second the controller manages to check all device connections and hardware components. It should be higher than 25. The higher the better. This number may/will drop if there are problems, if something slows down the controller, if devices are not connected properly or in the process of being connected, if a lot of displays needs to be updated etc. The lower this value, the less responsive the interface will feel. This value will also be impacted by the number and type of actions configured for interface components in the web interface. Network problems may also impact this value. If this value is too low, the controller may further loose connections to devices and may seem unresponsive to interface operations.

Commands in the Serial Monitor

You can enter commands in the serial monitor to do certain things with the controller. This is particularly useful for developers and also for bug-fixing and calibration. If you are using the Arduino IDE Enable CR/LF on the serial monitor dropdown menu in order to send the commands.

List of Commands

"config"	Reboots the device into config mode with its current IP. Similar to holding the config button until the LED becomes blue.
"configd"	Reboots the device into config default mode (IP always 192.168.10.99). Similar to holding the config button until the LED becomes white.
"debug"	Reboots and enables debug output to serial monitor
"newmac"	Generates a new random MAC address to EEPROM. Power cycle both your controller and network switch after this operation. Useful if you have network problems.
"clearpresets"	Clears the preset memory completely (flushes all!). Useful/necessary after a firmware upgrade. Similar to holding the config button until the LED becomes red.
"reset"	Reboots the controller
"HWvar=XXX"	Set Hardware Variant (byte). This value shouldn't be changed by users. It's significance is to inform the UniSketch software about which hardware revision it's running on in order to take certain specifics into account. Bit 0: Determines model of status LED on SKAARDUINO-AVR models.
"list analog"	Lists analog hardware components on the controller with number, description and three calibration values (start/end/tolerance)
"show analog X"	Shows readings from analog component X where X is the number given by "list analog". The readings indicate the value and noise level for the read out. This is useful for debugging. If you move the analog component you should see values change. Write "hide analog" to stop the display.
"hide analog"	Stops the display of "show analog"
"calibrate analog X"	Starts calibration of analog component X. Instructions will be posted in the serial monitor. The steps involve moving the analog component to various positions.
"clear analog X"	Resets calibration data for analog component X to default. If X is not given it resets calibration data for all components.
"set analog X=start,end,tolerance"	Forces calibration data "start", "end", and "tolerance" for component X
"exportPresets"	Will dump a large amount of data representing the entire configuration with presets in the controller. Last two bytes is a checksum
"importPresets"	Will import configuration into the internal EEPROM memory. After sending the command, the controller will instruct to paste configuration into the serial monitor. It must be formated like the output from "exportPresets"

"preset X"	Will select the given preset number X if it exists and reboot the controller
"ip=A.B.C.D"	Sets the controller IP address (for the current preset)
"ipDeviceX=A.B.C.D"	Sets the IP address for device index X (see boot up output) for the current preset.
"enableDeviceX=[0/1]"	Enable or disable device X for the current preset.
"clearusermemory"	Clears user memory space (used for various types of device setting presets).
"getVersion"	Shows the current version of UniSketch
"getCID"	Returns the controllers unique ID, which is used to access the configuration pages on cores.skaarhoj.com. Since this ID gives access to the online configuration, do not share it publicly.

Technical Notice: Any operation from the serial monitor that reboots the controller, does so with a "soft" reset and the Ethernet chip in the controller may still hold old settings. In most cases this is no problem, but at other times it may lead to strange behaviors and connection problems. In that case; power cycle the unit, press the reset button shortly or close down and reopen the serial monitor which will also act as a hardware reset.

Contact Support

You are always welcome to contact us for support questions - write an email to support@skaarhoj.com and we will do our best to accommodate your request.

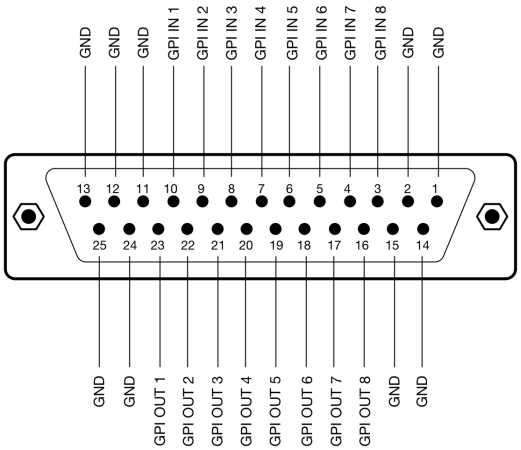
In order for us to provide the best support please state:

- Which SKAARHOJ unit it is about
- The serial number of your device if there is one (small silver label with 6 digits)
- The nature of the problem
- Which hardware device(s) you are controlling and their firmware version
- If you have successfully installed the Firmware Updater Application and made contact with your device though the Serial Monitor (you need the USB programming cable)
- If you have made changes to the default web interface, please include a print of the settings (print to pdf or similar)
- Your operating system

Hardware Notes:

DB-25 Connector Configuration for GPI

This is the pinout for the DB25 Connector we use as a GPI module:



This is the pinout for the DB25 Connector we use on the SDI-GPI Link: