

PYRAMID 2753757191 IGX Control System Framework User Manual

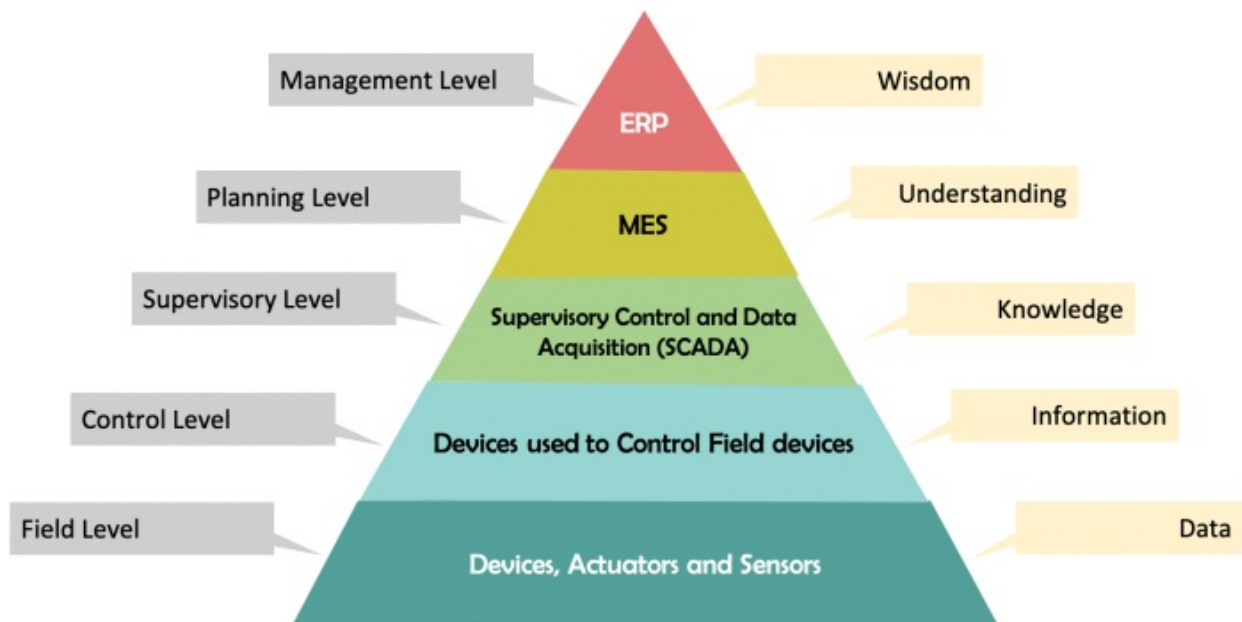
[Home](#) » [PYRAMID](#) » PYRAMID 2753757191 IGX Control System Framework User Manual 

Contents

- 1 PYRAMID 2753757191 IGX Control System Framework
- 2 Introduction
- 3 IGX Functionality and Features
- 4 IGX Interlock and Permit User Guide
- 5 IGX Expression Language Guide
- 6 Introduction
- 7 Uploading the Firmware
- 8 BeagleBone Black Replacement Guide
- 9 Document Control
- 10 Documents / Resources
 - 10.1 References
- 11 Related Posts



PYRAMID 2753757191 IGX Control System Framework



Introduction

Author	Matthew Nichols
Owner	Project Lead
Purpose	Provide clear, concise, and comprehensive information for the safe and effective use, maintenance, and troubleshooting of IGX-based products.
Scope	The use of an IGX product by an end user.
Intended Audience	IGX end-users
Process	Standard Manual Creation Process
Training	NOT APPLICABLE

IGX Control System Framework

IGX is Pyramid's latest software control system framework. It's designed from the ground up for high reliability applications including medical and light industrial settings. The framework is monolithic and modular, meaning that it comes with many features by default and that the features are all encapsulated such that they can be reused and extended easily. Some key features are a Battle-proven real-time QNX operating system with a highly reliable microkernel.

Custom web server with low latency and high bandwidth Harmonized IO data is available over multiple Ethernet protocols. Powerful data acquisition and processing tools. Built-in and user-configurable safety interlocking system allows for automatic safety actions. User-definable expression language for simple scripting.

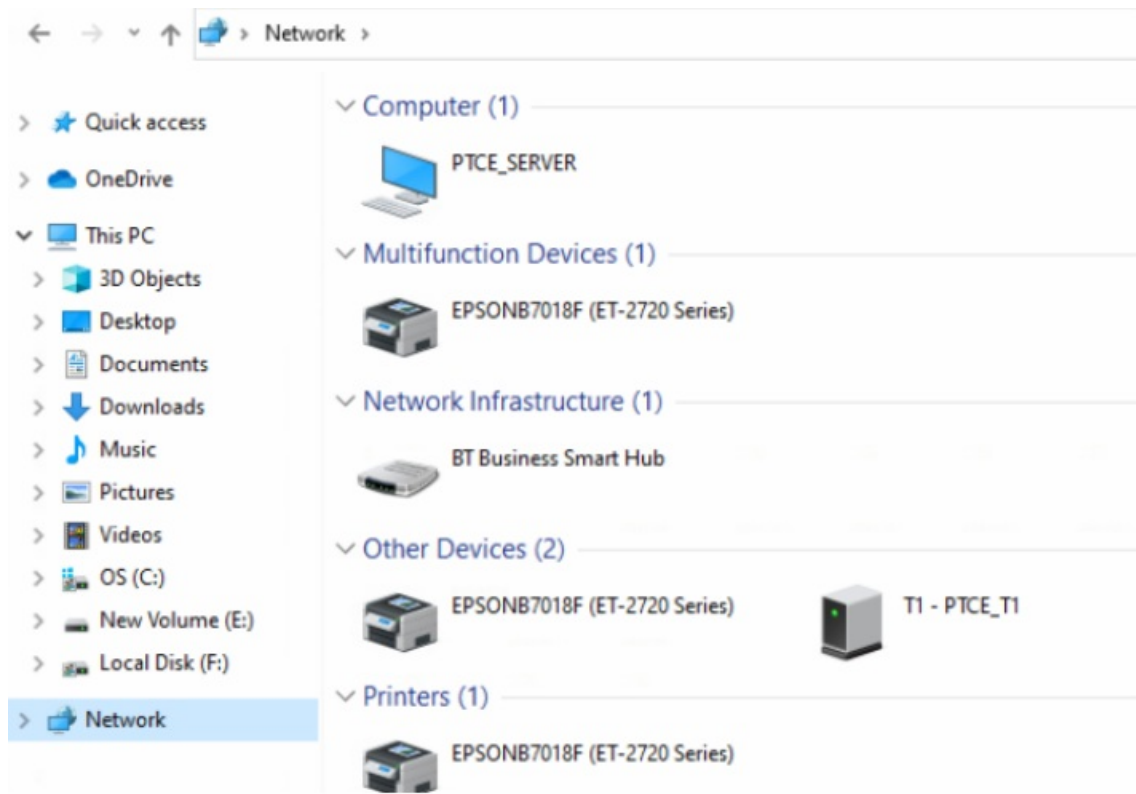
IGX Functionality and Features

Network Introduction

The IGX framework utilizes Ethernet-based communication as the primary medium for data exchange to the outside world. It's used for GUIs, control systems, databasing, scripting, and more. The most common form of communication used by IGX is the HTTP protocol, which is what is used to service the embedded GUI interfaces. To connect with an IGX device, you must have an understanding of basic Ethernet network concepts. This guide will take you through the step-by-step process of connecting to IGX devices and maintaining larger networks with multiple Pyramid and third-party products.

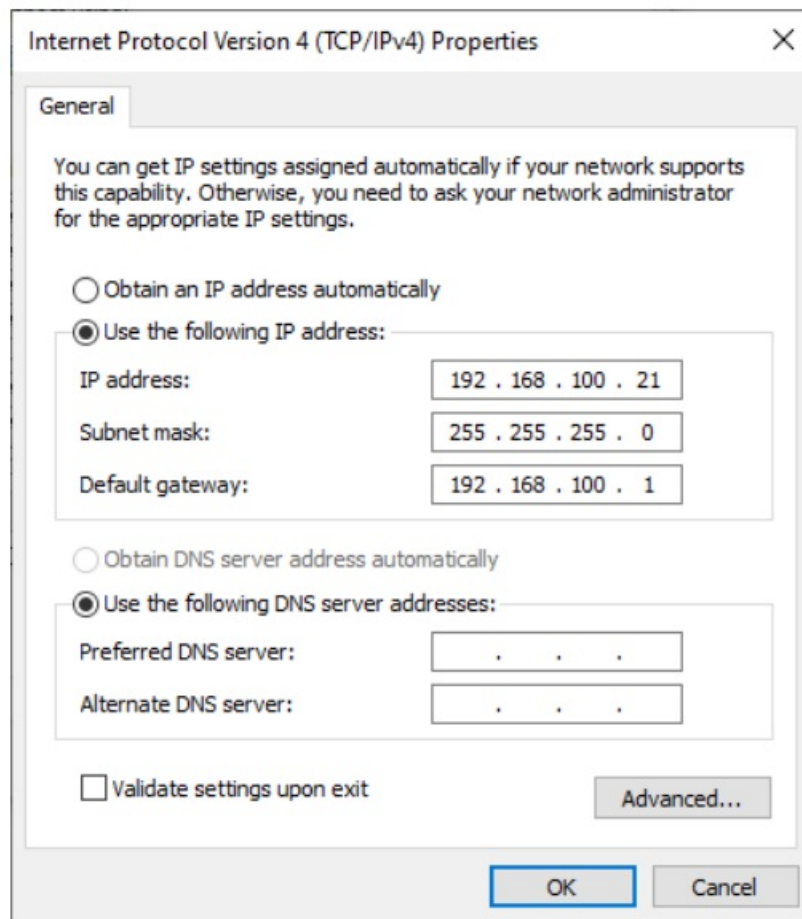
Quick Start

If you are already familiar with network concepts, then you can jump right into using IGX devices. Our products come with DHCP enabled and will attempt to automatically get assigned an IP address if possible. If there is no response to the DHCP request, the device will fall back to the static IP address, which is 192.168.100.20 with a netmask of 255.255.255.0 by default. Once the device has an IP address, it will be discoverable on your network using UPnP, Windows 10 comes with a discovery tool built into the file explorer.

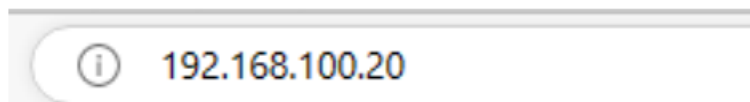


Make sure network discovery is turned on. You may need to press the refresh button if the device has been added very recently. Once you see the device icon on the Network page, double-click it to open a new page on your web browser with the device's URL. You can then log in and start controlling the device's settings and collecting data.

If the previous steps don't work, or you want to explicitly connect directly to the IGX device, you'll need to configure your computer's IP address settings to cooperate with the device's settings.



Refer to your operating system's documentation for specific instructions on how to do this. Once you've set your computer's Ethernet adapter to the right subnet, type the device's IP address directly into the address bar of a new tab on your web browser to open the web interface.



You can then log in and start controlling the device's settings and collecting data. 3 Browsers let you directly connect to the device's web GUI. Note that depending on your device's specific settings and your network configuration, you may need to perform additional configuration steps to ensure that your device is properly connected and configured to work with your computer. Refer to your device's documentation or user guide for more information on how to set up and connect to your device.

IP Networking Basics

Internet Protocol (IP) networks are the foundation of modern digital communication, enabling devices to exchange data over a shared network. IP networks facilitate communication among various devices like computers, smartphones, printers, and of course IGX devices, allowing them to share resources and information.

Here are the basics of IP networks and what a typical user needs to know:

IP Address: Each device on an IP network is assigned a unique identifier called an IP address. It is used to identify and locate devices on the network. There are two versions of IP addresses: IPv4 and IPv6. IPv4 addresses are written as four sets of numbers separated by periods (e.g., 192.168.100.20), while IPv6 addresses use eight sets of four hexadecimal characters separated by colons (e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334).

Subnet Mask: A subnet mask is a number that defines the range of IP addresses within a network. It helps devices determine whether they are on the same network or a different one. Subnet masks are written in the same format as IP addresses. For one computer to talk to another computer without an advanced router scheme, they both must have the same subnet mask.

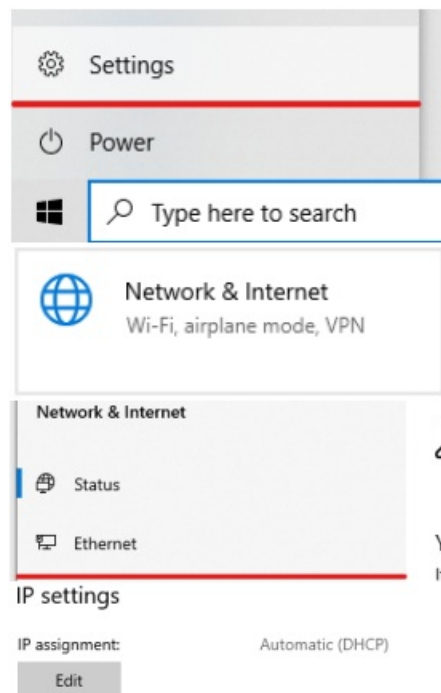
Routers: Routers are devices that connect multiple networks and direct data packets between them. They use IP addresses and subnet masks to determine the best path for forwarding the packets. Home routers also often act as Wi-Fi access points, enabling wireless devices to connect to the network.

DHCP: Dynamic Host Configuration Protocol (DHCP) is a network service that automatically assigns IP addresses and other network settings to devices as they join the network. This simplifies the process of connecting devices and ensures that they have the correct settings.

DNS: Domain Name System (DNS) is a service that translates human-readable domain names (e.g., www.example.com) into IP addresses. This makes it easier for users to access websites and services without having to remember their numerical IP addresses.

Setting your Windows 10 IP Address

This guide will walk you through the steps to set up Ethernet settings on Windows 10 for both Dynamic Host Configuration Protocol (DHCP) and manual static IP configurations.

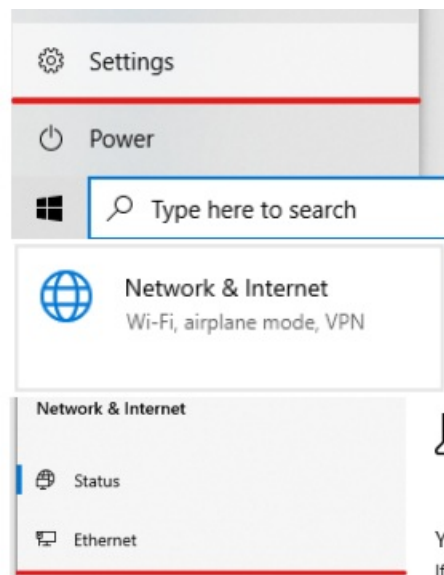


Using DHCP (Automatic IP Configuration)

Click on the “Start” button (Windows icon) in the lower-left corner of your screen and select “Settings” (the gear icon). In the “Settings” window, click on “Network & Internet.” On the left-hand menu, click on “Ethernet,” then select your Ethernet connection from the list on the right. Scroll down to the “IP settings” section and click on “Edit.” In the “Edit IP settings” window, select “Automatic (DHCP)” from the drop-down menu, then click on “Save.” Windows 10 will now automatically obtain an IP address, subnet mask, default gateway, and DNS server addresses from your network’s DHCP server. This is frequently a centralized router managed by an IT department or office administrator.

Using Manual Static IP Configuration

Click on the “Start” button (Windows icon) in the lower-left corner of your screen and select “Settings” (the gear icon). In the “Settings” window, click on “Network & Internet.” On the left-hand menu, click on “Ethernet,” then select your Ethernet connection from the list on the right. Scroll down to the “IP settings” section and click on “Edit.” In the “Edit IP settings” window, select “Manual” from the dropdown menu. You will now see options for “IPv4” and “IPv6.” Toggle the switch to “On” for the IP version you want to configure (usually IPv4).



Enter the following information, which can be obtained from your network administrator or ISP:

IP address: The static IP address you want to assign to your device (192.168.100.21 if trying to make a direct connection to 192.168.100.20). **The subnet mask or prefix length:** The subnet mask for your network (usually 255.255.255.0 for basic networks). Prefix length serves the same purpose as the mask but with a different format. Use “24” for 255.255.255.0.

Edit IP settings

Manual

IPv4

☒ On

IP address

192.168.100.21

Subnet prefix length

24

Gateway

192.168.100.1

Preferred DNS

8.8.8.8

Alternate DNS

Gateway: The IP address of your router or default gateway.

This is optional and only needed if there is a need for internet access and multiple network hops.

Note: Make sure the IP address you assign is unique and not used by another device on your network.

Scroll down to the “Preferred DNS” and “Alternate DNS” fields. Enter the DNS server addresses provided by your ISP or network administrator. If you don’t have specific DNS server addresses, you can use public DNS servers like Google’s (8.8.8.8 for Preferred DNS and 8.8.4.4 for Alternate DNS). This is only required if internet

connections are needed. Click on “Save” to apply the changes.

Windows 10 will now use the static IP address, subnet mask, default gateway, and DNS server addresses you entered for your Ethernet connection. Remember to verify your settings with your network administrator or ISP to ensure proper network connectivity. If you encounter any issues or need to revert to DHCP, follow the steps for “Using DHCP (Automatic IP Configuration)” to switch back.

IGX Interlock and Permit User Guide

Introduction to the Interlock and Permit System

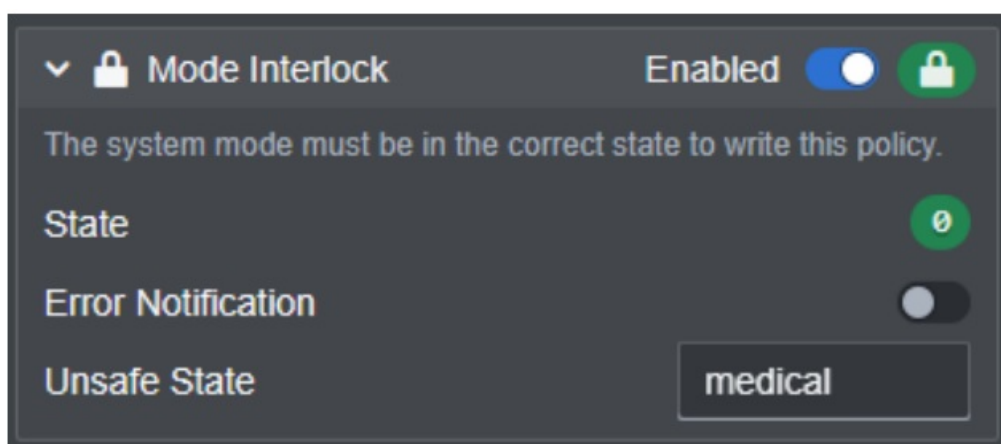
This guide provides an overview of the IGX interlock and permit system, which ensures the correct and safe operation of the IGX system. Software-based interlocks are a type of safety mechanism used in industrial environments to prevent accidents, ensure proper operation, and maintain system integrity. They are designed to coordinate the operation of various equipment, processes, or systems by enforcing predefined conditions or rules. In contrast to hardware-based interlocks, which rely on physical components such as switches or relays, software-based interlocks are implemented using a microprocessor and programmable logic. Frequently software-based interlocks are cheaper and easier to maintain than hardware-based interlocks, but they rarely eliminate the need for hardware-based interlocks.

In real-world scenarios, these interlock logic conditions must be regularly monitored and adjusted according to specific situations. This software interlock safety system offers several features: Configure interlock parameters through a GUI with descriptive text. Enable or disable interlocks, allowing temporary or permanent override of interlocks. Create latching interlocks that remain in error until disabled and re-enabled. Adjust interlock condition parameters or limits.

Combine multiple interlocks into a single permit, allowing the operation of protected outputs (e.g., enabling high voltage or radiation sources).

Interlocks

Interlocks are responsible for checking their conditions and reporting their status. To execute any action in response to an interlock, a permit or additional software is required. Interlocks are IGX IO with an integer type. This interlock detects when the IGX system is in medical mode and prevents otherwise unsafe actions from occurring. From this GUI interface, you can enable or disable the interlock and configure the interlock’s parameters.



Interlock Value

The interlock value is defined by an integer value corresponding to a particular meaning. As of this writing, there are 3 values defined, however, the system may be extended to support more values. Future versions will add these new values to the end of the list, and use new numbers, leaving the old meanings unchanged.

Value	State	Meaning
0	OK	Interlock is OK, all error and warning conditions are unmet.
1	ERROR	Interlock is in error; the error condition has been met.
2	WARNING	Interlock is in warning; the warning condition has been met. Not all interlocks will have a possible warning condition.

Interlock Types

There are several types of interlocks built into the IGX framework. These pre-made types allow for highly uniform and predictable condition checking that is robust and well-tested.

Range Interlocks

These interlocks will check if a given numeric IO is within optional range limits. All of the following parameters are optional. If the parameter is omitted, it will not be used in the conditional checking logic. These limits will be defined using the same units of the IO that it is checking against.

tolerance Interlock

Parameter	Meaning
Upper Limit	The value that if exceeded will cause an error.
Lower Limit	The value that if not exceeded will cause an error.
Upper Warning	The value that if exceeded will cause a warning.
Lower Warning	The value that if not exceeded will cause a warning.

Tolerance

These interlocks will check if a given numeric IO is within the acceptable tolerance limit of a given command IO. The tolerance is defined as a percentage. For example, you may want a readback input to match a given command output within 5%. The minimum is defined in the units of the command IO. The tolerance values are always defined as a percentage.

Parameter	Meaning
Minimum	The minimum absolute value that the command must have to enable the conditional checking. For example, a command of 0 should not count, as it will always be out of tolerance.
Tolerance Limit	The value of percentage error that if exceeded will cause an error.
Tolerance Warning	The value of percentage error that if exceeded will cause a warning.

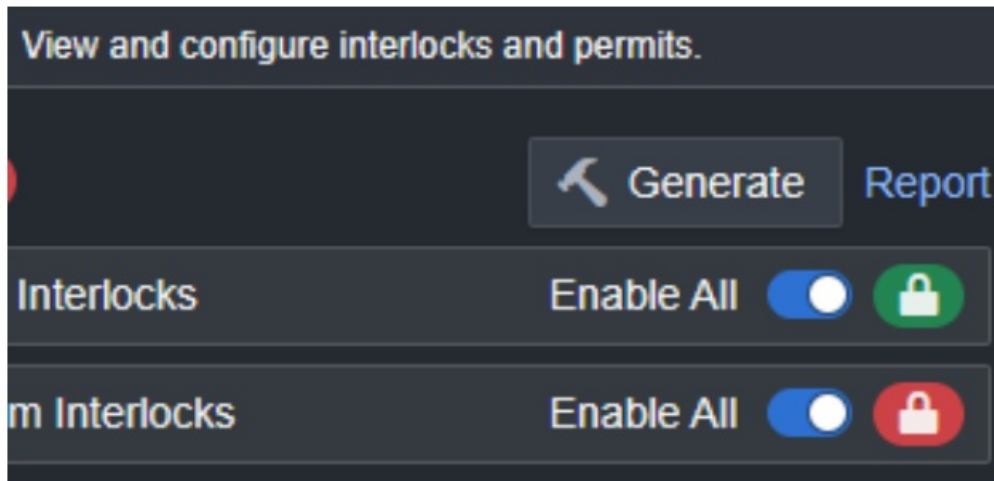
Permits

Permits are responsible for checking the status of their child interlocks and reporting the combined state. They can be in one of two possible states: granted or revoked. A granted permit will have a Boolean value of true, while a revoked permit will have a Boolean value of false.

Value	State	Meaning
true	GRANTED	A permit is granted. All interlocks are OK or in warning.
false	REVOKED	A permit is revoked. At least on interlock is in error.

Reporting and Auditing

It is critical to system safety to regularly conduct interlock parameter audits to ensure that the configuration is still meeting the requirements. IGX includes a system for generating interlock reports in the form of a CSV file containing all the relevant fields. IGX will have a nested system of interlock managers which each have their scope that they can report on. To generate a report of the global interlock scope, navigate to the following URL <https://io/interlocks>. To generate an interlock report, use the following steps.



5 Interlock report interface

First, click the “Generate” button to create a new report. Then click the report link to download the newly generated report. Once the report is on your system you can open it using Excel to see the values.

IGX Expression Language Guide

Introduction to ExprTk Language

ExprTk is a powerful and flexible mathematical expression parsing and evaluation library for C++. It allows users to create, manipulate, and evaluate complex mathematical expressions through a simple and intuitive language. In this guide, we’ll provide an overview of how you can use the ExprTk language to create expressions, incorporate variables, and use built-in functions and operators.

Arithmetic Operators

Expressions can be built using a variety of built-in operators. The following is a list of the mathematical and logical operators. These operators evaluate into a single numerical value.

Operator	Description
+	Addition: Adds two values together.
−	Subtraction: Subtracts the second value from the first value.
*	Multiplication: Multiplies two values together.
/	Division: Divides the first value by the second value.
%	Modulus: Returns the remainder of the division of the first value by the second value.
^	Exponentiation: Raises the first value to the power of the second value.

Arithmetic Operators

Equalities and Inequalities Operators

The following are operators for checking equalities or inequalities. The results of these operators will always be 0 for false and 1 for true.

Operator	Description
== or =	Equality: Checks if two values are equal.
!= or <>	Inequality: Checks if two values are not equal.
<	Less Than Checks if the first value is less than the second value.
<=	Less Than or Equal To Checks if the first value is less than or equal to the second value.
>	Greater Than: Checks if the first value is greater than the second value.
>=	Greater Than or Equal To Checks if the first value is greater than or equal to the second value.

Logical Operators

The following are logical operators. The results of these operators will always be 0 for false and 1 for true.

Operator	Description
true	True state or any value other than zero, typically 1.
false	False state, the value of exactly zero.
and	Logical AND, True only if x and y are both true. Example: x and y
&	Similar to AND but with left-to-right expression short-circuiting optimization.
and	Logical NAND, True only if either x or y is false. Example: x and y
mand(x, y, ...)	Multi-input logical AND, True only if all inputs are true. Left to right short-circuiting of expressions. Example: mand(x > y, z < w, u or v, w and x)
or	Logical OR, True if either x or y is true. Example: x or y
	Similar to OR but with left-to-right expression short-circuiting optimization.
nor	Logical NOR, True only if the result of x or y is false. Example: x nor y
mor(x, y, ...)	Multi-input logical OR, True if at least one of the inputs is true. Left to right short-circuiting of expressions. Example: more(x > y, z < w, u or v, w and x)
xor	Logical XOR, True only if the logical states of x and y differ. Example: x xor y
xnor	Logical XNOR, True if the biconditional of x and y is satisfied. Example: x xnor y
not(x)	Logical NOT, Negate the logical sense of the input.

Variable Assignment Operators

In addition to these operators, there is also a selection of assignment operators to pick from. Unlike the operators above, these will assign the evaluated value to a variable on the lefthand side of the operator.

Operator	Description
<code>:=</code>	Assignment: Assign the value on the right to the variable on the left.
<code>+=</code>	Add and Assign: Adds the value on the right to the variable on the left, and assigns the result to the variable on the left.
<code>-=</code>	Subtract and Assign: Subtracts the value on the right from the variable on the left, and assigns the result to the variable on the left.
<code>*=</code>	Multiply and Assign: Multiplies the value on the right with the variable on the left and assigns the result to the variable on the left.
<code>/=</code>	Divide and Assign: Divide the variable on the left by the value on the right and assign the result to the variable on the left.
<code>%=</code>	Modulus and Assign: This takes the modulus of the variable on the left by the value on the right and assigns the result to the variable on the left.

Built-in Functions

ExprTk comes with a wide range of built-in functions and operators to help you create more complex expressions. These include trigonometric functions (e.g., `sin`, `cos`, `tan`), logarithmic functions (e.g., `log`, `log10`, `exp`), and many others. To use a function, simply write its name followed by its arguments in parentheses. For example: `sin(x) * cos(y)`. Some functions accept a variable number of arguments. These functions will be defined with an argument list ending in `...` to show that more arguments can be added.

Function	Description
abs(x)	Returns the absolute value of x.
sgn(x)	Sign of x, -1 where $x < 0$, +1 where $x > 0$, else zero.
sqrt(x)	Returns the square root of x.
root(x, n)	Nth-Root of x. where n is a positive integer.
pow(x, y)	Raises x to the power of y (x^y).
exp(x)	Returns the exponential function of x (e^x).
log(x)	Returns the natural logarithm (base e) of x.
log10(x)	Returns the logarithm (base 10) of x.
log2(x)	Returns the logarithm (base 2) of x.
login(x, n)	Base N logarithm of x. where n is a positive integer.
ceil(x)	The smallest integer that is greater than or equal to x.
floor(x)	Largest integer that is less than or equal to x.
round(x)	Round x to the nearest integer.
round(x, n)	Round x to n decimal places where $n > 0$ and is an integer. Example: <code>round(1.2345678, 4) == 1.2346</code>
frac(x)	The fractional portion of x.
trunc(x)	The integer portion of x.
max(x, y, ...)	The largest value of all the inputs.
min(x, y, ...)	The smallest value of all the inputs.

Function	Description
sum(x, y, ...)	The sum of all the inputs.
mul(x, y, ...)	Product of all the inputs.
avg(x, y, ...)	Average of all the inputs.
hypot(x, y)	Hypotenuse of x and y. $\text{hypot}(x, y) = \sqrt{x^2 + y^2}$
erf(x)	Error function of x.
erfc(x)	Complimentary error function of x.
df(x)	Normal cumulative distribution function.

Function	Description
$\cos(x)$	Returns the arc cosine of x in radians.
$\text{ASIN}(x)$	Returns the arc sine of x in radians.
$\text{atan}(x)$	Returns the arc tangent of x in radians.
$\text{atan2}(x, y)$	Returns the arc tangent of y/x in radians.
$\cos(x)$	Returns the cosine of x , where x is in radians.
$\cosh(x)$	Returns the hyperbolic cosine of x .
$\sin(x)$	Returns the sine of x , where x is in radians.
$\sinh(x)$	Returns the hyperbolic sine of x .
$\tan(x)$	Returns the tangent of x , where x is in radians.
$\tanh(x)$	Returns the hyperbolic tangent of x .

ExprTk library supports a wide range of built-in functions for various mathematical operations and calculations. Here's a table listing some of the commonly used built-in functions in ExprTk along with a brief description of their behavior:

General Purpose Functions Trigonometry Functions

Conditional Expressions

You can create conditional expressions using the if-then-else construct. The syntax for this construct is as follows: `if (condition) then (value_if_true) else (value_if_false)`. The condition is an expression that evaluates to a Boolean value, while `value_if_true` and `value_if_false` are expressions that will be evaluated depending on the outcome of the condition. For example: `if (x > y) then (x - y) else (y - x)`.

Creating Expressions

To create an expression using ExprTk, simply write a mathematical expression using standard mathematical notation. ExprTk supports basic arithmetic operations such as addition (+), subtraction (-), multiplication (*), and division (/). You can also use parentheses to control the order of operations. For example, an expression could look like this: `3 + 2 * (7 - 5)`.

Incorporating Variables

ExprTk allows you to use variables within expressions. To define a variable, use a single-letter or multi-letter alphanumeric name, beginning with a lowercase letter. For example, you can use `x`, `y`, or `length` as variable names. Once defined, you can use these variables within your expressions. For example, an expression with variables could look like this: `x + y * (length - width)`.

Introduction

IGX devices run the QNX operating system and can be accessed via standard SSH and SFTP protocols. The default login credentials for IGX devices are username: root and password: root. This guide provides step-by-step instructions on how to access IGX devices via SSH and SFTP using the command prompt in Windows 10 and WinSCP. Additionally, it covers basic QNX console commands and how to start and stop the IGX application.

Secure Shell (SSH) Protocol

The SSH protocol is a secure network protocol used for remote access to computers. It is designed to provide secure communication between two untrusted networks by using encryption to protect the contents of the communication. SSH is widely used by network administrators to manage remote systems and by developers to access remote development environments.

The SSH protocol uses the TCP/IP protocol suite for communication, and by default, uses port 22. The protocol supports several authentication methods, including password-based authentication, public key authentication, and host-based authentication. SSH also supports tunneling of TCP/IP connections, allowing applications to securely communicate over the SSH connection. The SSH protocol is defined in several RFCs, including RFC 4250 (The Secure Shell (SSH) Protocol Assigned Numbers), RFC 4251 (The Secure Shell (SSH) Protocol Architecture), and RFC 4253 (The Secure Shell (SSH) Transport Layer Protocol).

Secure File Transfer Protocol (SFTP)

The Secure File Transfer Protocol (SFTP) is a secure alternative to the File Transfer Protocol (FTP) for transferring files between computers. SFTP is based on the SSH protocol and uses the same encryption and authentication mechanisms to protect the contents of the communication.

SFTP uses the TCP/IP protocol suite for communication and typically uses port 22, the same port as SSH. SFTP supports several operations, including file upload, file download, and directory listing.

SFTP also supports the resume of interrupted transfers and can preserve file attributes such as modification time, permissions, and ownership.

The SFTP protocol is defined in several RFCs, including RFC 913 (FTP server extension), RFC 913 (FTP client extension), and RFC 4253 (The Secure Shell (SSH) Transport Layer Protocol).

Further Reading

If you're interested in learning more about the SSH and SFTP protocols, here are some resources to get you started:

- SSH Protocol Specification: IETF RFC24502
- SFTP Protocol Specification: IETF RFC9133
- OpenSSH: <https://www.openssh.com/>

1. 2. 3. 1. 2. 3. 4. PuTTY: <https://www.chiark.greenend.org.uk/~sgtatham/putty/>

WinSCP: <https://winscp.net/eng/docs/start> 2.4.2 Accessing IGX Devices via SSH To access an IGX device via SSH, follow these steps: Open the Command Prompt on Windows 10 by typing cmd into the Start menu. In the Command Prompt, type the following command to connect to the IGX device via SSH: ssh root Replace with the IP address of the IGX device you wish to connect to. When prompted, enter the password for the root user (default password is root). You should now be connected to the IGX device via SSH and can execute QNX commands.

Accessing IGX Devices via SFTP To access an IGX device via SFTP, follow these steps

Download and install WinSCP from the official website: <https://winscp.net/eng/download.php>

Open WinSCP and click on the "New Site" button. In the "New Site" window, enter the following information:

File protocol: SFTP Hostname:

Port number: 22 User name: root Password: root Replace with the IP address of the IGX device you wish to connect to. Click the "Save" button and then "Login" to connect to the IGX device via SFTP. You should now be able to browse and transfer files to/from the IGX device via WinSCP.

Basic QNX Console Commands

To learn more about these commands and their options, check out the official QNX documentation at Official QNX Documentation4

Command	Description	Example
ls	Lists the contents of the current directory.	ls
cd <directory>	Changes the current directory.	cd /root/igx
pwd	Prints the current working directory.	pwd
ifconfig	Displays network interface configuration information.	ifconfig
ping <ip>	Tests network connectivity.	ping 192.168.0.1
netstat	Displays network statistics and active connections.	netstat
kill <pid>	Terminates the process with the specified process ID.	kill 1234
slay <process>	Like kill except you can use process names.	slay igx

Command	Description	Example
ps	Displays information about running processes.	ps
tar	Compresses and decompresses files and directories.	tar -cf archive.tar file
top	Displays the system resource usage (CPU, memory, etc .) in real-time.	top
shutdown	Immediately restarts the operating system. IGX should automatically start if using the default configuration.	shutdown

Starting and Stopping the IGX Application

The IGX application is located at [/root/igx](#) and can be started and stopped using the following commands:

To start the IGX application, use the following command: [/root/igx](#)

This will automatically kill any other instance of IGX that may be running. To stop the IGX application, use the following command: slay igx

This will gracefully stop the IGX application and release all resources.

Note that only one instance of the IGX application can run at a time, so it is important to stop the currently running instance before starting a new one.

Conclusion

Accessing IGX devices via SSH and SFTP is a powerful way to configure and run applications on these devices. By following the steps outlined in this programmer's guide, you can quickly and easily access IGX devices using the command prompt in Windows 10 and WinSCP. Additionally, learning some basic QNX console commands will allow you to configure the device and run applications with ease

IGX User-Managed System Care

This section of the manual is designed to assist users who wish to service their IGX devices independently. While user-managed system care can be a rewarding and cost-effective way to maintain your device, it is essential to follow the guidelines provided in this manual to ensure the safety and longevity of your IGX device. Should you require assistance with any of these advanced service mode activities, Pyramid's expert technicians are always available to help.

Before attempting any user-managed system care tasks, make sure to review the safety precautions outlined in the manual, and ensure that you have the necessary tools, equipment, and knowledge to perform the tasks safely and effectively. Additionally, it is essential to work in a clean, well-lit, and organized environment to avoid potential damage to your device. Here are some general guidelines to follow when performing user-managed system care on your IGX device:

Regular Inspection: Regularly inspect your IGX device for any signs of wear, damage, or loose connections. This can help you identify potential issues before they become more significant problems. Be sure to consult the device's user manual for specific inspection intervals and procedures.

Firmware Updates: Keep your IGX device up-to-date with the latest firmware releases from Pyramid. Regular updates can enhance the performance, stability, and security of your device. Refer to the manual for instructions on how to check and install firmware updates. Some medically controlled systems will be restricted to certain versions of IGX that have more formal testing. Please consult your Pyramid representativity in this case. **Cleaning:** Keep your IGX device clean and free of dust, debris, and other contaminants. Regular cleaning can improve the device's performance, prevent overheating, and reduce the risk of damage. Be sure to follow the recommended cleaning procedures outlined in the user manual.

Component Replacement: Over time, some components in your IGX device may need to be replaced due to wear or damage. Always use genuine Pyramid replacement parts to ensure optimal performance and compatibility. Consult the user manual for guidance on identifying and replacing worn or damaged components.

Troubleshooting: If you encounter any issues with your IGX device, consult the troubleshooting section of the user manual for possible solutions. If the issue persists, don't hesitate to contact Pyramid's support team for assistance. Remember, while user-managed system care can be an effective way to maintain your IGX device, Pyramid's team of skilled technicians is always available to provide support and assistance when needed. If you are unsure about any aspect of servicing your device, it is best to consult with Pyramid's experts to ensure the safety and longevity of your investment.

IGX Firmware Update Guide

Introduction

Firmware updates are typically carried out by trained professionals in a Pyramid-controlled facility. However, there may be circumstances where our clients need to perform firmware updates themselves. It is essential to note that improper updating may result in permanent data loss or damage to the device's functionality. Pyramid does not provide any warranty for data or time lost due to improper firmware installations. With this in mind, this guide will outline the steps necessary to successfully update your device's firmware.

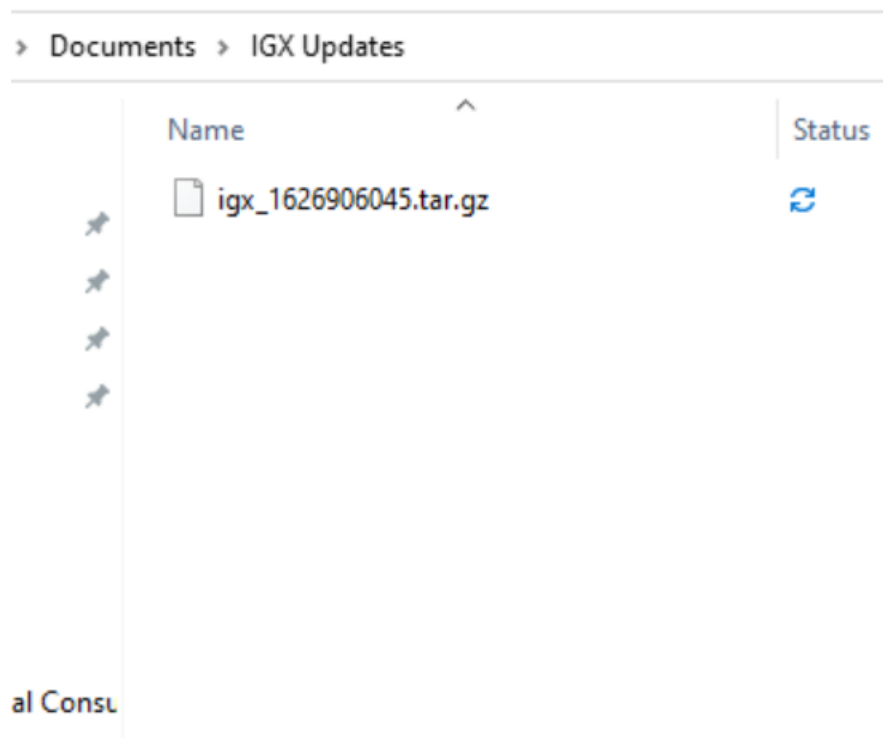
Identify the Firmware File

An IGX firmware file is usually stored in a .tar.gz format, with the name "igx" followed by the version number. This file contains all the binaries and configurations required for updating the device. The operating system (QNX) and supporting libraries are stored separately in a different file system. Do not decompress the file before uploading it to the device, as the device expects a compressed file and will fail to update without one.

Uploading the Firmware

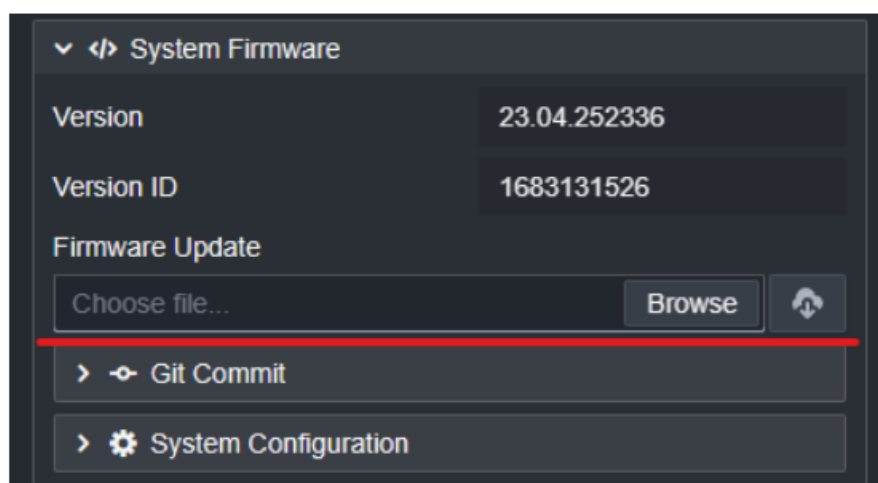
The process for accessing the update screen may vary between devices and software versions. Typically, there will be an "Admin" link that directs you to the relevant page. If unsure, you can enter the direct URL into your

browser to access the admin page, such as <http://192.168.100.20/io/admin>). Most modern browsers do not require so you can omit it. Upon accessing the admin page, you will see an interface similar to the images shown in the original documentation. The procedure for updating the firmware remains the same, regardless of the interface's appearance.



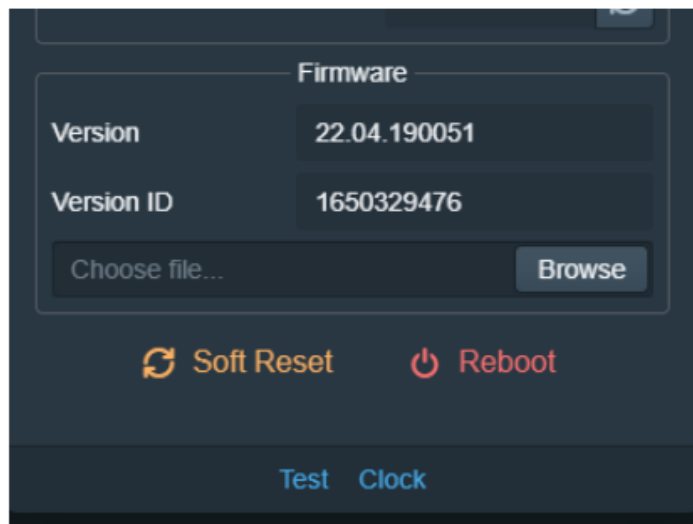
6 A typical IGX firmware file

On the latest versions of IGX, the GUI should look like the following: 7 IGX firmware upload interface. Optionally you can click the download button to download a copy of the last uploaded firmware. This is useful for recovering back to an older version. Click on the “Browse” button to select your update file from your local computer. Then the download button should change into an upload button. Click the upload button to install the selected update. In older versions of IGX, you may find the admin page looking like this. While the interface itself is different, the procedure of updating will be the same.

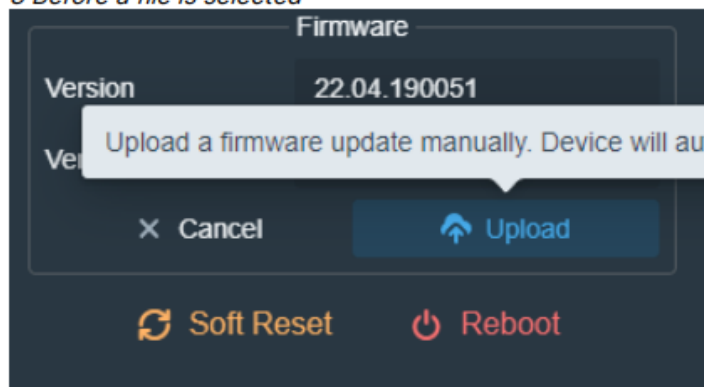


7 IGX firmware upload interface.

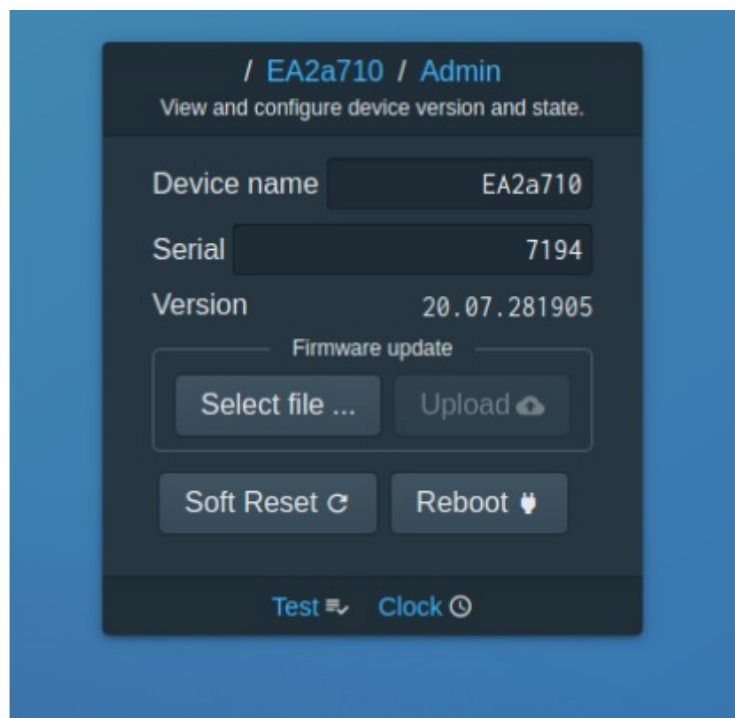
Click on “Browse” and select the update file from your local machine.



8 Before a file is selected



9 After the file is selected



10 Historic file upload interface

After selecting your file, the “Upload” button should appear.
Click the “Upload” button and the update process will begin.

Historic file upload interface

Press the “Select file ...” button and select the compressed update file on your system.
Then press the “Upload” button. The update will automatically start and then reset the device.

Post-Firmware Update

The device will automatically restart after the firmware update. You will temporarily lose your connection to the device, which is normal. Do not manually refresh the page during the update or power down the device until it has restarted. If the device has not restarted after 3 minutes or you cannot reconnect, try power cycling the device to resolve any unexpected issues. If the device still malfunctions, contact Pyramid's support team for assistance.

Update Validation and Troubleshooting

Verify that the device is functioning as expected. If necessary, power cycle the device to resolve any issues that may have arisen after the update. If you are upgrading from a significantly older firmware version, manually power cycle the device might be required to complete the update. Contact Pyramid's support team if you require further assistance.

IGX SD Card Flashing Guide

Introduction

This guide will take you step by step through the standardized procedure of flashing an SD card for either updating or creating for the first time a usable card for Pyramid devices. Our SD cards contain all the firmware and data our devices use. You might be creating an SD card for the first time for manufacturing or trying to fix a card that has been broken beyond simple repair.

Get the Micro SD Card

Any micro SD card that is 32GB should work. We use a microSDHC UHS-I Class 10 U1 card made by Samsung, this may change in the future as SD card technology moves quite quickly and manufacturers stop making and supporting obsolete cards.



Get the Desired Image File

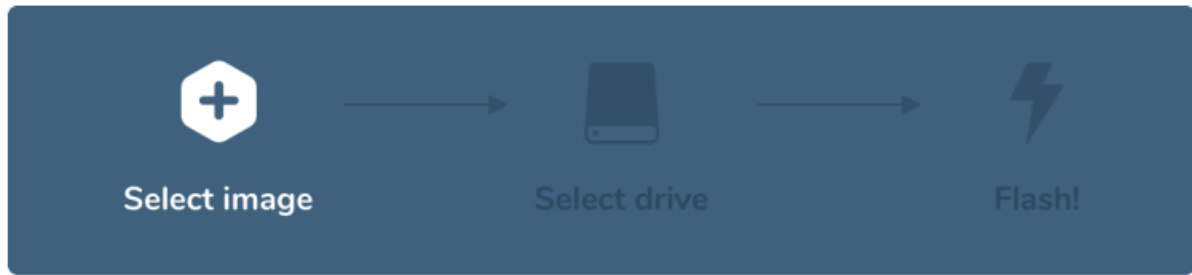
First, you are going to need a file called an 'image', the name comes from the fact that it's a snapshot of the device's state at one time. An image file is simply a copy of another SD card contained in a single file. A common file extension used for images is '.iso' although Pyramid images are usually compressed using gzip so our file extensions are '.iso.gz'. If the image is compressed do not decompress the file before flashing. Do not try and flash an image located on a network drive. In the past, we have found issues with this, just make sure you use a local copy of the file before using it.

Get the Required Tool

There are many programs and tools to flash SD cards, but we can't make any guarantees they will work correctly. For example, we have found Win32 Disk Imager to sometimes fail and not report the failure correctly. For the sake of constancy and standardization, everyone at Pyramid uses the same tool called Etcher. Download Etcher [here](#). Etcher works on Windows, Linux, and Mac, it flashes quickly, handles compressed images without configuration, and validates the card after writing to confirm the flash worked correctly. It also has a nice simple interface to boot!

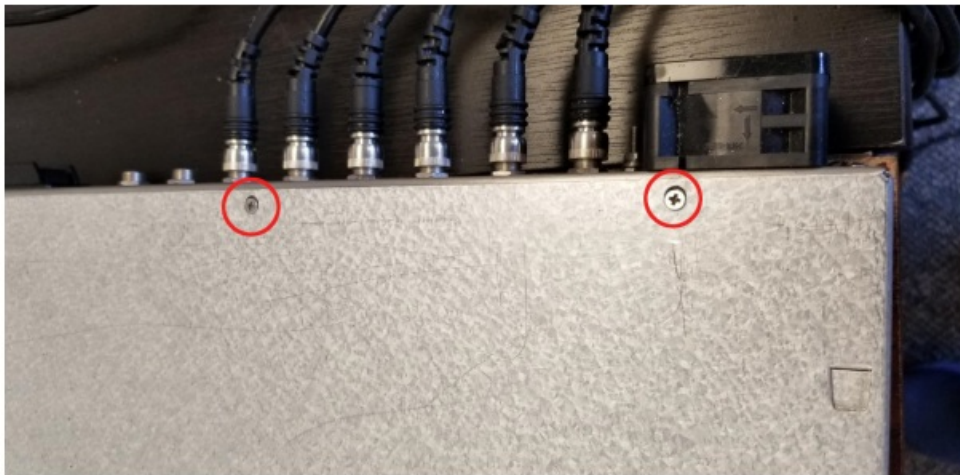
Flash the Card

Start up Etcher and then select the image file on your computer's file system. Insert the card into your computer, you may need a USB or full-size SD card adapter depending on the type of computer you have. Etcher should immediately recognize the card but you may have to select the correct drive if there are multiple connected to your computer. Then just press flash, and wait. This should take at least several minutes so feel free to get a coffee. If all went well you should be ready to install the freshly made card. If Etcher gave you an error, just try again, or try a different card.

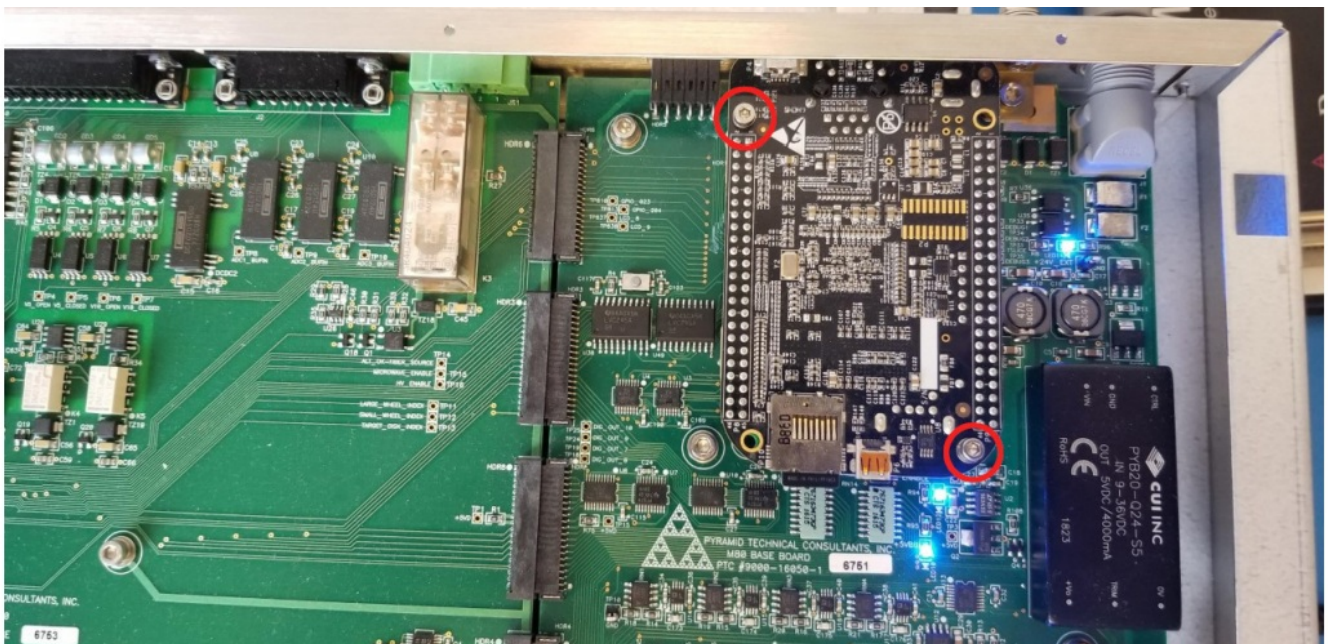


Card Installation or Removal

To install or remove a card from a Pyramid device you must first remove the top panel to gain access to the SD card reader and control board. For our 1U or 2U rack mountable devices, the top panel can be removed by unscrewing the four small Phillips-head screws located on the top of the rear end of the device.



Then slide the cover off by pushing back towards the rear end of the device. Depending on the device the card reader might be located in different locations, but it should look something like in the two example pictures below.



12 Mounting Screw Locations

To remove the card gently press the card in like a button and it should then pop back out and be easily removed. To install a card do the same but in reverse where you gently press in to lock the card back into place.

Reboot the Device

Once the new card has been installed, you may power on the device normally. The LEDs on the Ethernet port (if there is Ethernet on this device) should be on and flickering with an Ethernet cable attached.

IGX SD Card Imaging Guide

Summary

This document provides instructions for creating and compressing SD card disk images on Linux and Windows 10 systems.

For Linux, the procedure uses the `dd` and `gzip` command-line tools to create and compress the disk image. Users must identify the SD card device using the `lsblk` command and then execute the `dd` command to create the disk image with the `.iso.gz` extension. For Windows 10, the process requires the use of Win32 Disk Imager to create the disk image and 7-Zip to compress it. Users must install and run Win32 Disk Imager, select the appropriate SD card device, and create the disk image with the `.iso` extension. Afterward, 7-Zip is used to compress the disk image into a `.iso.gz` file. Both procedures require sufficient free space on the computer to store the disk image and its compressed version.

Linux Procedure

Creating a disk image of an SD card on Linux involves the use of two command-line tools: `dd` and `gzip`. These tools are typically pre-installed on most Linux distributions. The `dd` command is used for copying and converting data, while `gzip` is a data compression tool. In this procedure, you will create an image of the SD card and compress it as you go. Identify the SD card device: To find the correct device file for your SD card, run the following command:

lsblk

This command lists all the block devices on your system. Look for the SD card in the list, typically named `mmcblk0` or `mmcblk0p1`. Note that the actual name may vary depending on your system.

Create and compress the disk image: Run the following command to create a disk image of the SD card and compress it using Standard Command `sudo` Command With Progress `sudo dd status=progress if=/dev/mmcblk0 | gzip>filename.iso.gz` with the appropriate device file for your SD card and replace the filename with your desired name for the disk image. This command may take around 20 minutes to complete, depending on the size of the SD card. Ensure that the destination for the output file has enough free space to accommodate the compressed image. For more information on the `dd` command, refer to the official GNU Coreutils documentation.

For additional details on `gzip`, consult the `gzip` manual page

Windows 10 Procedure To create and compress an SD card disk image that results in a `.iso.gz` file on Windows 10, follow these steps:

Download and install Win32 Disk Imager: Download the latest version of Win32 Disk Imager from the official website and install it on your computer. Insert the SD card: Insert the SD card into your computer's SD card slot or an external card reader.

Launch Win32 Disk Imager: Run Win32 Disk Imager and select the appropriate drive letter for your SD card in the "Device" dropdown menu.

Create the disk image: Click on the folder icon next to the "Image File" field and choose a destination for the output file. Enter a file name with the `.iso` extension, for example, `filename.iso`. Click the "Read" button to start creating the disk image.

This process may take some time, depending on the size of the SD card. Download and install 7-Zip: Download and install the latest version of 7-Zip9 if you don't already have it on your computer. Compress the disk image: After the disk image has been created, you can compress it using 7-Zip. Right-click the `.iso` file, choose "7-Zip" from the context menu, and select "Add to archive...". In the "Archive format" dropdown, choose "gzip", and click "OK" to start compressing the image. Once the compression is complete, the file will have the `.iso.gz` extension, for example, `filename.iso.gz`. Ensure that you have enough free space on your computer to store the disk image and

its compressed version.

For more information about Win32 Disk Imager, visit the official documentation¹⁰. For further details on using 7-Zip, consult the 7-Zip help documentation¹¹

BeagleBone Black Replacement Guide

Purpose

This document provides step-by-step instructions for replacing the beagle bone found in some Pyramid devices. The beagle bone is a processing unit that Pyramid uses in its designs. If it becomes damaged, it is easy to replace as it is a standard component. This guide ensures that the replacement process is completed safely and correctly.

Replacing a beagle bone in a Pyramid device is a straightforward process, but it is essential to handle the components with caution and care. By following these instructions and taking the necessary precautions, you can successfully replace the beagle bone and ensure the continued operation of your device. If you require further assistance or are unsure about any of the steps, it is recommended to contact Pyramid support or refer to their documentation.

Procuring a BeagleBone Black

To procure a BeagleBone Black, users have several options. They can either purchase it directly from Pyramid or BeagleBoard.org¹³. BeagleBone Blacks that are purchased from BeagleBoard.org¹⁴ will not come with any SD card. If you are simply going to swap the SD card from your original unit, then this will not be a problem, but if you want a complete replacement, then we recommend buying from Pyramid.

Purchase from Pyramid

Email sales@ptcusa.com¹⁵, and request a quote for a BeagleBone Black, pre-loaded with an SD card containing the software of your choice. We will reach out to you to verify the software choice and complete the sales order.

Purchase from BeagleBoard.org

Visit the official website: <https://beagleboard.org/black> On the website, you'll find detailed information about the BeagleBone Black, its features, and specifications. Click the "Buy Now" button to see a list of authorized distributors.

Choose your preferred distributor from the list and follow their purchase process.

Replacement Procedure

For our 1U or 2U rack mountable devices, the top panel can be removed by unscrewing the four small Phillips-head screws located on the top of the rear end of the device. Then slide the cover off by pushing back towards the rear end of the device.

Top Panel Screw Locations

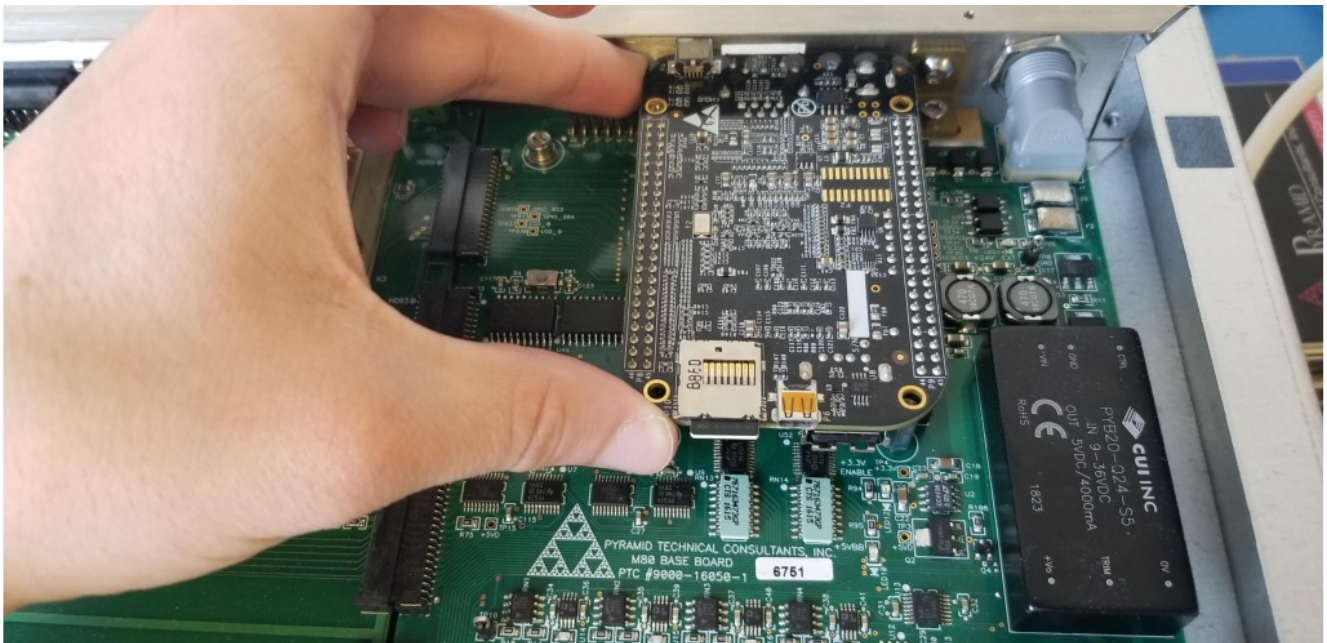
Before the beagle bone can be removed, there will be two or four M3 hex screws that have to be unscrewed. These screws hold the beagle bone in place.

Mounting Screw Locations

After unscrewing the mounting screws, grab the board by the corners, and pull it directly up. The fit may be tight, so you might have to wiggle the board slightly to remove it.

Where to Grab the Device

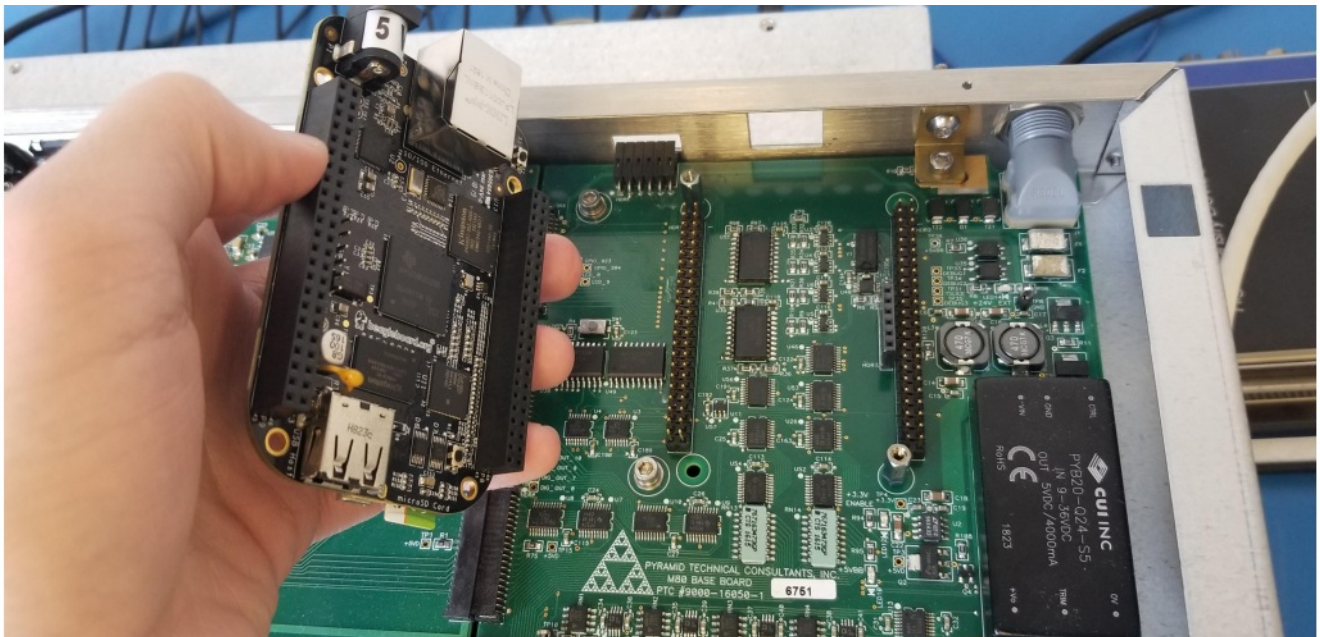
Once the old board has been removed, make sure to inspect the pins on the main board. They should all be straight, and nothing should be missing. If there is any damage to the pins or sockets, they should be fixed before installing the new beagle bone.



13 Where to Grab the Device

Inspecting the Connector Pins

Take the new beagle bone and put it into position. Press down directly on the corners, making sure that each pin is seated completely into the corresponding socket. Screw back in the two or four M3 screws to hold the beagle bone in place. Do not over-tighten these screws as it could cause damage to the board. Slide the top panel back on and re-screw the four small screws into place. Be careful not to overtighten these screws, as they can strip very easily.



14 Inspecting the Connector Pins

Document Control

Approvals

Document Control

Current document version: v.1


No reviewers were assigned.

Signatures

Revisions

Version	Description	Saved by	Saved on	Status
v2	SD card imaging instructions and expression language documentation.	Matthew Nichols	Apr 1, 2024, 3:51 PM	APPROVED
v1	Initial release	Harvey Jules Nett	Dec 1, 2023, 5:51 PM	OUTDATED

Documents / Resources

	PYRAMID 2753757191 IGX Control System Framework [pdf] User Manual 2753757191 IGX Control System Framework, 2753757191, IGX Control System Framework, C control System Framework, System Framework, Framework
---	---

References

- 🌐 [Example Domain](#)
- 🐶 [BeagleBone® Black - BeagleBoard](#)
- 💎 [Win32 Disk Imager / Wiki / Home](#)
- 💎 [Win32 Disk Imager download | SourceForge.net](#)
- 📄 [Table of Contents :: WinSCP](#)
- 📄 [WinSCP :: Official Site :: Download](#)
- 📦 [balenaEtcher - Flash OS images to SD cards & USB drives](#)
- 🌐 [PuTTY: a free SSH and Telnet client](#)
- 🌐 [ietf.org/rfc/rfc4250.txt](#)
- 🌐 [ietf.org/rfc/rfc913.txt](#)
- 🌟 [OpenSSH](#)
- 🌐 [qnx.com/developers/docs/7.1/#com.qnx.doc.neutrino.user_guide/topic/cmdline.html](#)
- [User Manual](#)