

CorsairHMI Developer

3/20/2019

Manuals in this series

[CHMI_Overview.pdf](#)

[CHMI_Operator.pdf](#)

CHMI_Developer.pdf – This Manual

[CHMI_Designer.pdf](#)

[CHMI_Experts.pdf](#)

[CHMI_Corrections.pdf](#)

[CHMI_BatteryMon.pdf](#)

[CHMI_Glossary.pdf](#)

Contents

Development.....	6
Installation	6
Development Basics.....	10
Databases.....	14
Sessions.....	14
Drivers	18
Data Sources	20
Register Blocks	21
Tags	22
Tag Types	22
Tag Configuration.....	23
Tag Data Formats	27
Strings	29
Alarms	31
Calls	34
Icons.....	35
Drawings	36
Screens.....	37
Sounds.....	38
Sheets.....	40
Trends	40
Scripts.....	41
Data Logs.....	42
Ethernet Objects	49
SQL Paths	49
FTP Paths.....	50
Driver Paths.....	51
Local and Global Database Views	51
Creating Graphics.....	52

Placement Types	54
Address	55
Arc	55
Bar Graph	56
Bin	57
Checkbox.....	58
Corner	58
Drawing.....	59
Ellipse	59
Frame	60
Icon.....	60
Key.....	60
Line.....	61
Picture.....	62
Pipe	63
Point Name	63
Polygon.....	63
Rectangle.....	64
Rounded Rectangle	65
Spline.....	66
Tank.....	67
Text	68
Trend.....	68
Value	69
Placement Actions	70
Alarm Action	70
Call Action	70
Display Action	70
Entry Action.....	70
I/O Jump Action	70
Log Jump Action.....	70
Source Comms Action	70

Quick Trend Action.....	71
Screen Jump Action.....	71
Sheet Jump Action	71
Task Action	71
Trend Jump Action	71
Door Action	71
Door Command Action.....	72
Developing User Sheets	72
Creating I/O Modules.....	75
Developing with Corsair Blocks.....	75
The Corsair Select and Replace System	77
Database Record Import and Export	81
CSV Data Import.....	83
DXF Import	86
Corsair Micro-CAPs	89
Computer Configuration	91
Model List Group	92
Sessions Group.....	94
Desktop Group	94
Comms Ports Group.....	95
Sounds Group.....	96
Startup Options Group.....	96
Startup Levels Group.....	96
Startup Delays Group.....	97
Reports Printing Group	97
Reports Fonts Group.....	98
Reports SQL Group.....	98
Web Group.....	99
Menus Group	100
Email Group	100
My IP Group	100
Passwords Group	101

Users Group	101
Scramble Group	101
Style Group	102
MBHR Group	103
Serial Group	103
Help Files Group.....	104
EDS Group	106
Shutdown Group.....	107
Developer Preferences.....	107
Printing the Manual	109
The My IP System.....	112
New Versions via Email.....	115
Using CorsairHMI with Linux.....	115

Development

Development is the process of creating a model file for the Corsair program to use. Most Corsair licenses support development. The developer must enter model database information, draw graphic screens, enter scripts, and do many other activities to create a model that is complete enough for use.

Installation

Corsair is a portable application that does not need to be installed into the computer's operating system. It can be run directly from a USB drive or it can be copied to the computer's hard drive. It is uninstalled by deleting the files. It is not copy protected in any way. Users can copy it to another computer for backup purposes. They are expected to honor the limitations of the license that they have purchased.

Typically the first step for a Windows machine is to create a folder named 'corsair' on the root of the C: drive. CorsairHMI will supply a minimum of two files. The first is the Corsair program. Its file name is 'corsair.exe'. It should be copied to the 'c:\corsair' folder. The second file is the license file. Its name has the form 'CHMI*.cky'. The name of the license file begins with 'CHMI' followed by some additional letters. It ends with a '.cky' extension. It must be copied into the same folder as the corsair.exe file. The license file tells the Corsair program what features have been purchased by the customer for his system.

An integrator may have several license files from different customers on his computer. If multiple licenses are in the same folder it is not predictable which one will be used by the program. Common practice is to rename unused license files by inserting a capital 'X' at the beginning of the file name.

The next step is typically to create a desktop shortcut to run the program. Right-click on the corsair.exe file and pick the 'Send to Desktop (Create Shortcut)' option.

The program needs a computer properties file. It is typically named 'corsair.cfg' and it is kept in the same c:\corsair folder. If the system was not supplied with a computer properties file the developer must make one.

The program also needs a developer preferences file. It is named 'corsair.pre' and it is kept in the same folder. If the system was not supplied with a developer preferences file the developer must make one.

Each system needs at least one model file. The model file is typically named 'corsair.cap'. It can be kept in any folder but it usually is in the same folder as the other files.

Some developers may want to place a splash icon in the blank background of the Corsair desktop. This file must be a Windows bitmap. Its name must be 'corsair.bmp'.

The final list of files is typically:

Corsair.exe – the Corsair application

CHMI*.cky – the license

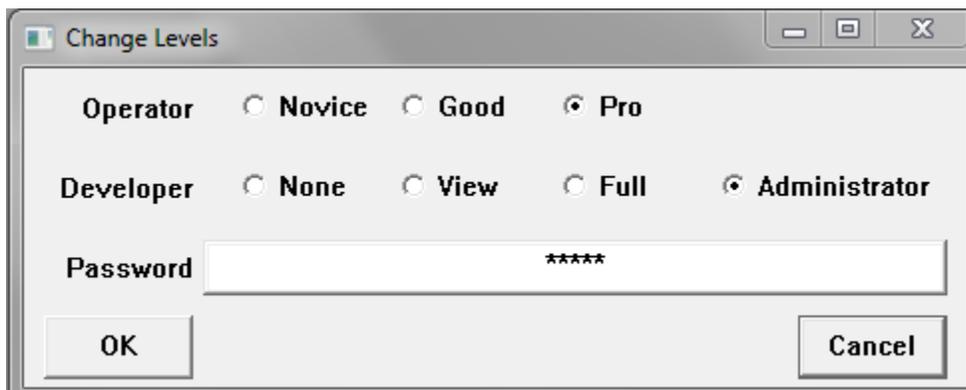
corsair.cfg – computer properties

corsair.pre – developer preferences

corsair.cap – the model

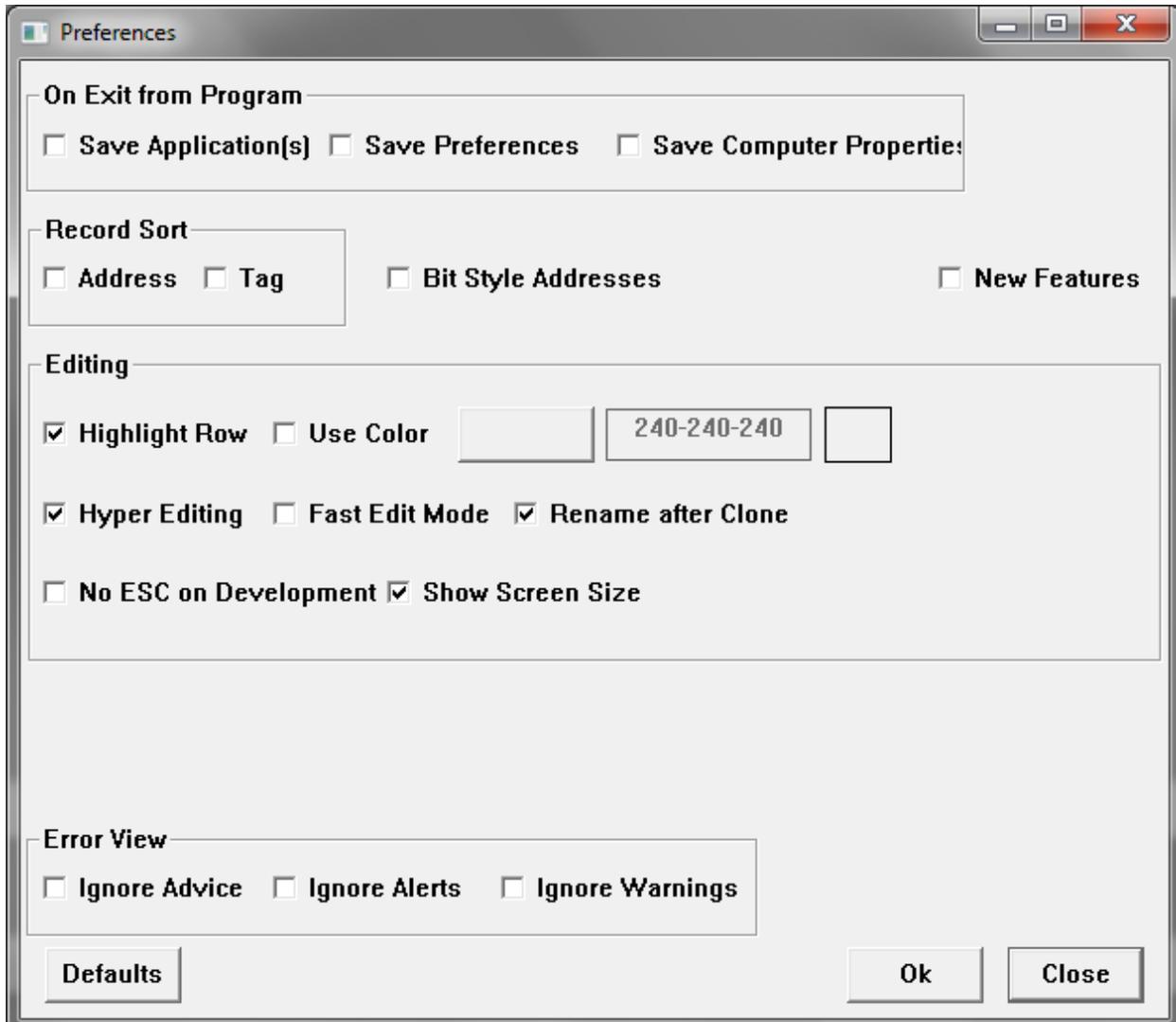
corsair.bmp – the optional splash icon

The suggested procedure for creating the rest of the files is for the developer to start the corsair program from the desktop shortcut. He then picks 'Users', and 'Change Levels' from the main menu.



The developer then clicks on 'Administrator' and enters the default password which is 'admin'. This enables the OK button so he can change the user level to Administrator. After clicking on OK it should say 'Dev: Admin' on the status bar on the lower right side of the Corsair desktop window.

The next step is to create the 'corsair.pre' developer preferences file if one was not provided with your system. From the main menu select 'Setup' 'Default Fields' and answer the prompt. Then do 'Setup' 'Graphic Font' and select something like Courier New, Bold, 12 point. Then do 'Setup' 'Preferences' to open the preferences window.



Select the 'Defaults' option and click on OK. The next step is 'Setup' 'Save Preferences'. This completes creating a preference file.

The next step is to create the 'corsair.cfg' computer properties file if one was not provided with your system. Click on the menu's 'Setup' 'Computer Properties'. The following properties are a suggestion.

From the Interface Tab:

Check 'Allow Click on F9 Corner'

From the Startup Tab:

Select Developer 'Administrator'

Select Operator 'Pro'

From the Report Tab:

Check 'Operator Printing'

Check 'Landscape'

For the Large font put Courier New, Bold, 14 point

For the Medium font put Courier New, Bold, 10 point

For the Small font put Courier New, Normal, 10 point

From the Security Tab:

Check 'Allow Exit'

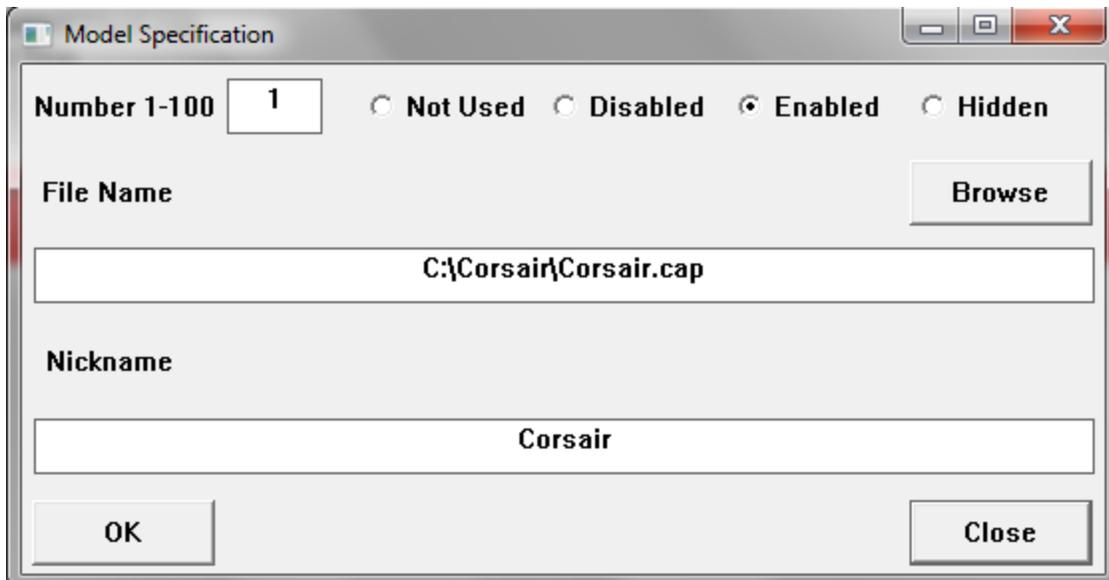
DO NOT check 'Lock Access to Desktop'

Check 'Allow Minimize'

Enter 'corsair' for a Base Session Name

Clicking on 'OK' enters the computer properties you have selected. The 'Setup' 'Save Properties' menu option creates the file. Now when you start the program from the shortcut it initializes with Administrator privileges.

The next step is to create an empty Model file if one was not provided with your system. Select 'Setup' 'Model List' from the main menu. Press F2 to edit the first model specification. Fill it out like this and click on 'OK'.



The 'Setup' 'Save Properties' menu option must be used now to save the Model List part of the computer properties. The model file has been specified. Now it must be created. From the 'Setup' 'Model List' screen menu use first the 'Empty' and then the 'Save' option.

This completes creation of the files that are required for a CorsairHMI system. Now the development work can begin.

Development Basics

Most Corsair development activities can be performed by clicking on a menu item. There are some keystrokes that may be used to do the same things. Some of these keystrokes have alternative keys that do the same thing.

The 'F1' key is used for going to a record, selecting a record for a link, and opening entry assistants.

The 'F2' 'Edit' key is used to start editing a data item.

The 'F3' key is used in some places as an alternative edit key.

The 'F4' 'Create' key is used to create a data item. The alternative keystroke is the 'Ins' key.

The 'F5' 'Clone' key is used to create a duplicate of a data item.

The 'F6' key is used to change database entry modes between a development sheet and single-record entry.

The Control+F6 key combination can be used to jump into a development mode for some windows.

The 'F7' 'Zoom' key is used to zoom on a record. The alternative keystroke is the 'Z' key.

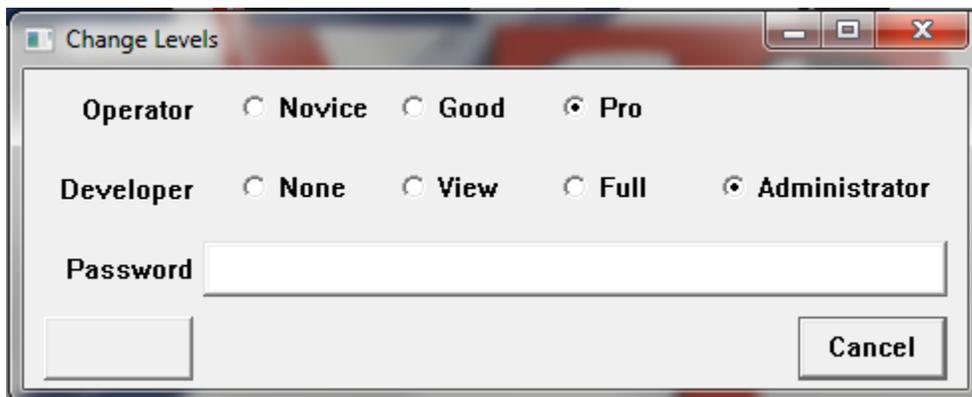
The 'Del' key is used to delete a data item.

'Ctrl' plus 'P' can frequently be used to initiate printing.

Development capabilities can be set to different levels. Those levels are 'None', 'View', 'Full', and 'Administrator'. 'None' means that development cannot be seen. 'View' means that it can be seen but not changed. 'Full' means that most changes can be made. 'Administrator' means that anything can be done. If the current level is greater than 'None' it is shown on the bottom right side of the Corsair desktop status bar. The label 'Dev: Admin' means that the level is set to Administrator.

Development changes are made with the 'File' 'Save File' menu option.

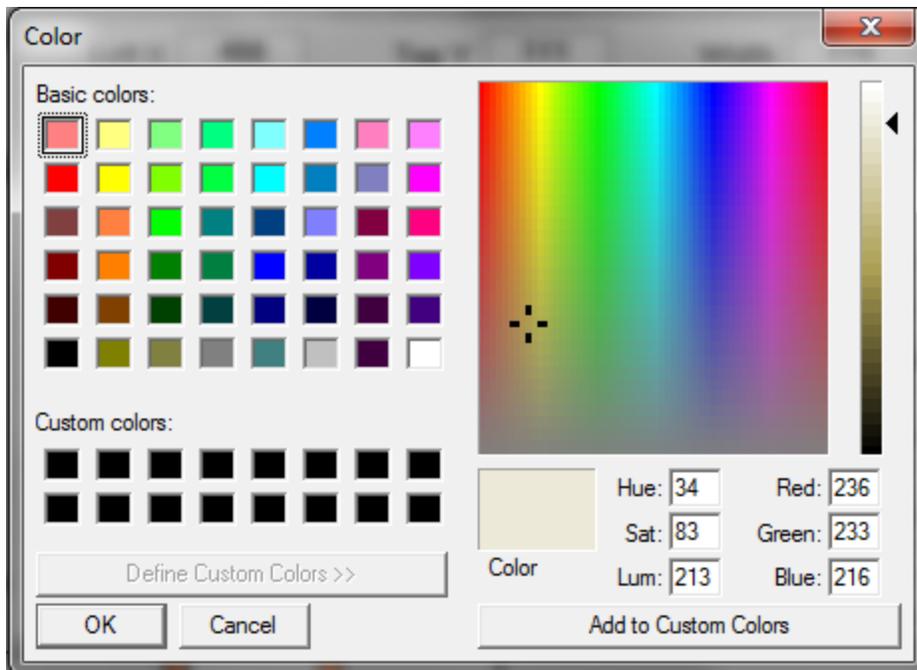
Development level is changed with the 'Users' 'Change Levels' menu option.



This window opens up with the radio buttons set for the current operator and developer levels. Either can be reduced by picking that option and clicking on 'OK'. To increase a level a password has to be entered. The default administrator password is 'ADMIN'. If it hasn't been changed typing it in, clicking on the 'Administrator' radio button, and clicking on 'OK' will enable development. After development is finished the 'Users' 'Change Levels' menu option is used to reopen the window. Clicking on 'None' and OK will shut off development.

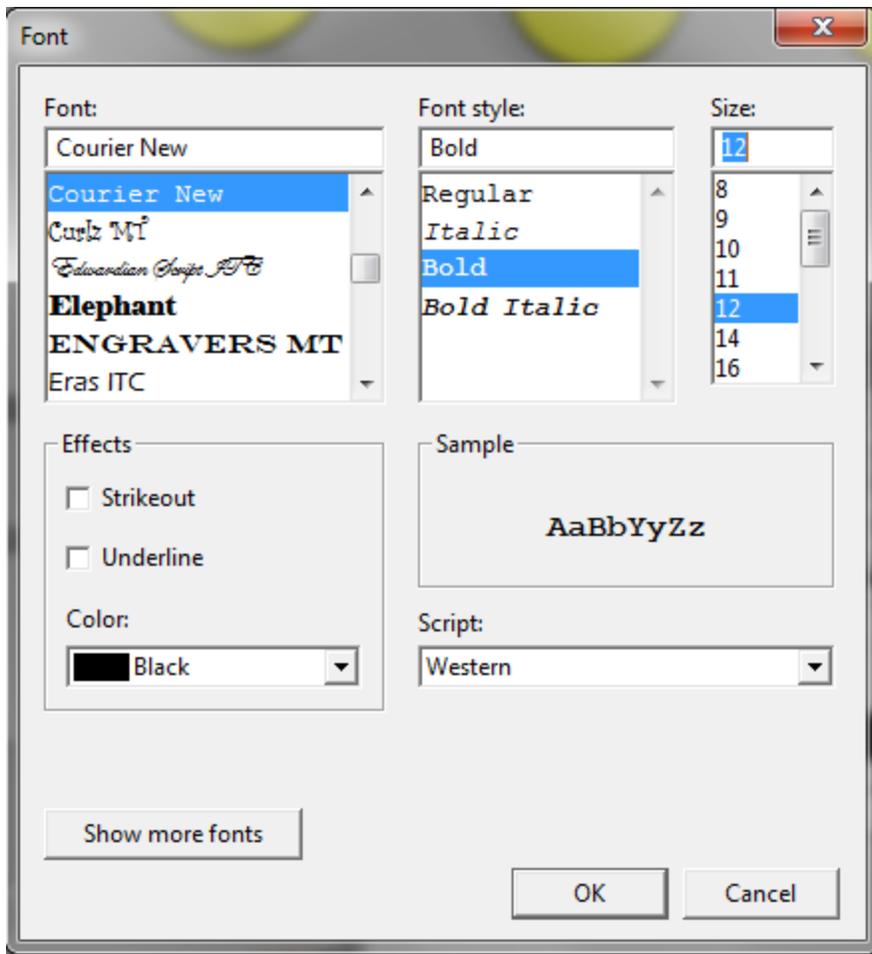
Many development activities involve selecting colors. Colors are specified by three numbers separated by hyphens. Each number can range from 0 to 255. The three numbers correspond to levels of Red, Green, and Blue. 0-0-0 is black. 255-255-255 is white. 0-255-0 is green. Any color can be specified by a three-number combination.

Corsair uses a color selector window to aid the developer when entering color values. One version of this window looks like this:



When the developer finds a color that he wants to reuse it can be added to the 16 custom colors that the window allows. The next time that the color selector is opened that selection will be available.

Many development activities involve selecting a Font for the computer to use to show some text on the screen or for a printout. The operating system has a font selection window. One version of this window looks like this:



A font is specified by a name, by a point size, and by options for Bold, Italic, Strikeout, and Underline.

Development databases are initially viewed as sheet-style development.

Tag	Type	Start	End	Form
Selected Formula Indicators	Switch			
Formula Weld Show Bits	Switch			
Machine Names	String	40001	400384	
Machine Enable Switches	Switch	400385/0	400386/07	
Formula Selection	Integer	400387		U
Formula Names	String	400401	400496	
Formula 1 Weld Count	Integer	400501		U
Formula 2 Weld Count	Integer	400502		U
Formula 3 Weld Count	Integer	400503		U
Formula 4 Weld Count	Integer	400504		U
Formula 5 Weld Count	Integer	400505		U
Formula Weld Times	Integer	400511	400630	U

Each row corresponds to a record. Each column corresponds to a field. The intersection of a row and a column is a single position called a cell. The current cell is shown in a different color. It can be changed with the arrow keys.

The developer uses the F4 Create key to insert a record. He uses the F5 key to clone a record. If he wants to edit a cell he presses the F2 key. The F6 key is used to switch back and forth to single-record editing mode.

Databases

Sessions

A Corsair model (.cap) file can include multiple data records that are known as session records. Some systems may involve several physical computers located in different parts of a facility each running copies of the Corsair software. They would be linked together on a network along with several data sources. Typical data sources would be PLCs - Programmable Logic Controllers. Each computer would have its own copy of the same Corsair application file. Session records are to create differences in how the Corsair program acts on different computers that all use identical application files.

For example, let's assume that a project utilizes 5 computers in areas designated as 'Batching', 'Production', 'QC', 'Warehouse', and 'Office'. There would be 5 records having these labels in the computer database. Each physical computer would have a unique Computer Properties file on its hard drive. That file includes a label that corresponds to one of the session records. This is known as the 'base' computer record for that location. This is a label for the Corsair program and it is not the same name as the operating system's network ID for that computer.

The base computer record for a location can be used to determine which data sources will communicate with that location. There may be a PLC on a molding machine. The data source record includes a list of what computers have been selected to communicate to that data source. 'Batching', 'Production', and 'QC' may need to talk to the molding machine but not 'Warehouse' or 'Office'. The list associated with the molding machine data source is what permits this to happen. The data source is considered to be 'Live' for the Production computer and 'Memory' or 'Not Real' for the warehouse computer. Designers of large systems may need to review how many simultaneous network connections ('sockets') are allowed by their data sources. This may limit how many computers can treat the data source as being 'Live'.

The 'Base' computer record is used to configure data acquisition. The 'View' computer record is used to determine what the operator sees. For simple systems they are the same record.

Initially when the Corsair program is started the view computer record is the same as the base computer record. Corsair offers an operator log-on system. This is not the same as the login that is used by the computer's operating system. The administrator of the Corsair log-in system can create multiple users each with a unique password. He may enter a computer name for each user. When a user logs in Corsair looks to see if a computer record matches what was entered for that user. If one is found that record becomes the new view computer record as long as that user is logged on.

Corsair can act as a web host using its dynamic HTML generation capabilities. Multiple clients can be logged into a Corsair host at the same time with different view computer records. This is regulated by the log-in type setting for the web host. One setting is 'Log in to operate, bypass to view'. The client can choose to log in with a user name if he desires to be able to operate the interface. He can bypass without a login if all he needs to do is view Corsair data.

If the client chooses the bypass option the Corsair program looks for a computer record that has the 'Bypass Web Client' type. That computer record becomes the view computer for that client. If there is not a bypass client record the view computer defaults to the base computer record.

If the web client elects to login he must enter a user name and password. The view computer for that client session is the computer name that was entered for the user name.

Each session record has an associated index number value. These are positive numbers that begin with zero. These values are used with Corsair session-indexed addressing. Changing the view session would then change what address the interface sees. For example, an integer tag on a PLC may begin with Modbus address 40001. If the tag is session-indexed a view session with an index value of 0 would display the value in register 400001. A view session with an index value of 1 would display the value in register 400002.

Session-indexed tags are frequently used for authority tags. The authority system can be used to determine which view computers can operate a tag and which ones can operate and see alarms. The corsair computer can generate different alarm summaries for different users.

ID Field

Hook 1-4 fields

Screen (First Screen)

Index (Addressing Index)

TCP ID

IP (PLC Network IP Address)

Host (MBHR Host?)

MIS IP

Remote (MBHR Remote?)

Host (MBHR Primary Host)

Host (MBHR Backup Host)

Write (Remote can write data?)

Extended Protocol? Y/N

TCP Port (MBHR TCP Port (0 defaults to 502))

Memory Tag MBHR ID 1

Note 1, 2, and 3

Other IP Address

Port Objects Zoom

Models Zoom (Model Specifications)

Icon

Location #

Sequence #

Logging – Do Logging Y/N

X size

Y size

There is a single-record editing window for session records.

The screenshot shows a 'Computer Record' dialog box with the following fields and controls:

- Name:** Corsair
- Mon 1:** Overview
- Mon 2:** ??
- Mon 3:** ??
- Mon 4:** ??
- Hook 1:** ??
- Hook 2:** ??
- Hook 3:** ??
- Hook 4:** ??
- Index:** 0
- Count:** 1
- IP Address:** 0.0.0.0
- MBHR TCP Port:** 0
- Passwords:** Password
- TCP ID:** (empty)
- Is MBHR Host?** MIS IP: 192.168.1.24
- Is MBHR Remote?**
- Can Write?**
- Extended?**
- Primary Host:** ??
- Secondary Host:** ??
- F6 Mode:**
- Alarm Summary:**
- Buttons:** Print, OK, Accept, Cancel

XXXX

Drivers

A driver is a portion of the CorsairHMI program that is most commonly used to communicate with a device. A PLC driver is used to communicate with a Programmable Logic Controller. Each driver has been written to work with a protocol. A protocol is a set of rules as to how communications between the two devices is going to occur. Modbus is one example of a group of protocols. Corsair includes several types of Modbus drivers to talk to different types of Modbus devices. The CorsairHMI Designer manual includes a complete list of available driver types.

Every Corsair program contains all the available drivers. The first step to use a driver is for the developer to create a record in the driver database. This record is given an identifying name and a type that shows which driver it is hooked to. Multiple driver records can use the same or different drivers.

Each driver record should contain at least one data source. A PLC is an example of a data source. One driver record of a Modbus driver may contain 20 data sources. Each of these data sources would be a Modbus-compatible PLC.

ID

Type

PLCs Zoom

Port

Connect Time Out (Seconds)

Retry - Connect Retry (Seconds)

Note 1, 2, 3

PLC IP

MIS IP

Other IP

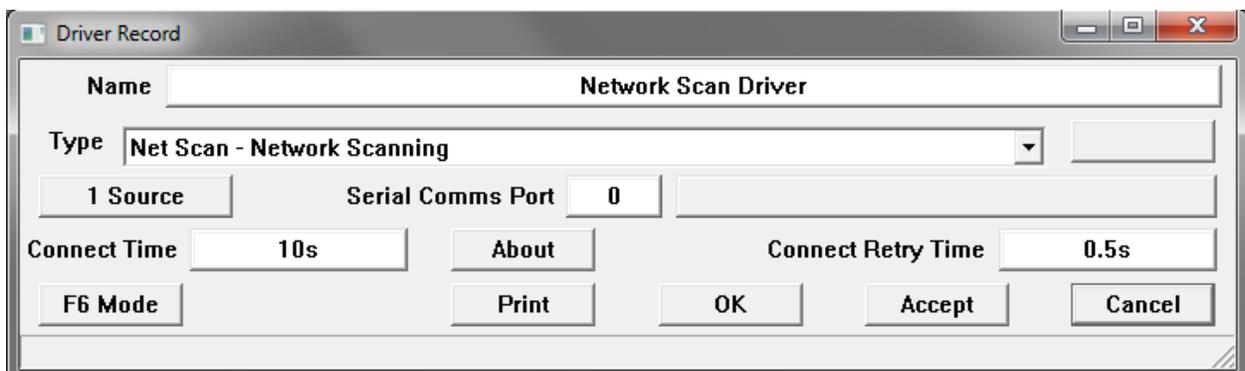
Port Objects Zoom

Icon

Location #

Sequence #

There is a single-record editing window for driver records.



xxx

Data Sources

A data source is a place that the Corsair program goes to get tag data. Data source driver records are created under driver records. A data source under a Modbus driver may be a Modbus-compatible PLC. A data source under a General Electric driver may be a General Electric PLC.

ID

Driver Parent

Register Block Zoom

Tags Zoom

Real Y/N

IP Address

Node Number

Time-Out (seconds)

I/O Zoom

Authority Link

Sessions Zoom

MBHR ID #

Idle Pause (Seconds)

ID (TCP ID)

Doors Zoom

Stream Y/N

Note 1, 2, 3

MIS IP

Other IP

Port Objects Zoom

Icon

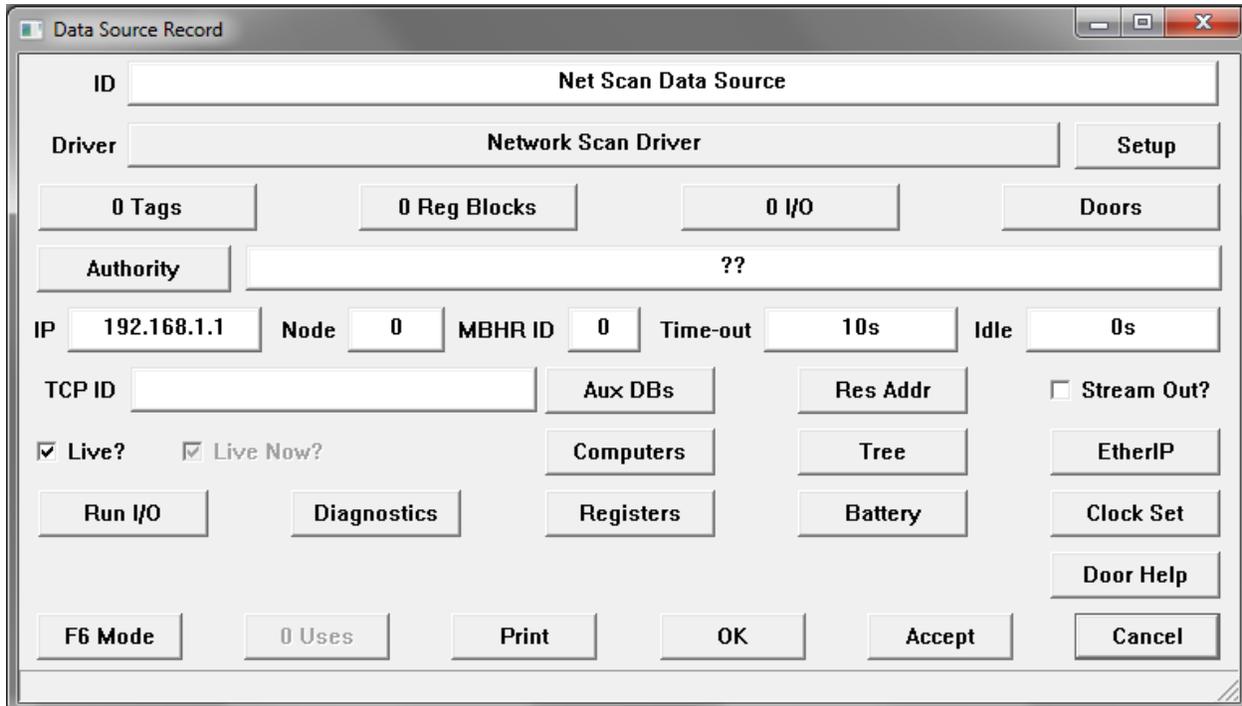
Event Area

Clock Tag Link

Battery Tag Link

Some drivers require a Driver Path specification to determine how to route a communications request to the data source. A Driver Path record of the proper type must be created. The Driver Path Link field is used to link this driver path to the data source record. One Driver Path may be used for multiple Data Sources.

There is a single-record editing window for data source records.



The screenshot shows a software window titled "Data Source Record" with a subtitle "Net Scan Data Source". The window contains several fields and buttons:

- ID:** Net Scan Data Source
- Driver:** Network Scan Driver (with a "Setup" button)
- 0 Tags**, **0 Reg Blocks**, **0 I/O**, and **Doors** buttons.
- Authority:** ??
- IP:** 192.168.1.1, **Node:** 0, **MBHR ID:** 0, **Time-out:** 10s, **Idle:** 0s
- TCP ID:** (empty), **Aux DBs**, **Res Addr**, and **Stream Out?**
- Live?** and **Live Now?**
- Computers**, **Tree**, and **EtherIP** buttons.
- Run I/O**, **Diagnostics**, **Registers**, **Battery**, and **Clock Set** buttons.
- Door Help** button.
- F6 Mode**, **0 Uses**, **Print**, **OK**, **Accept**, and **Cancel** buttons.

XXXXXX

Register Blocks

Name

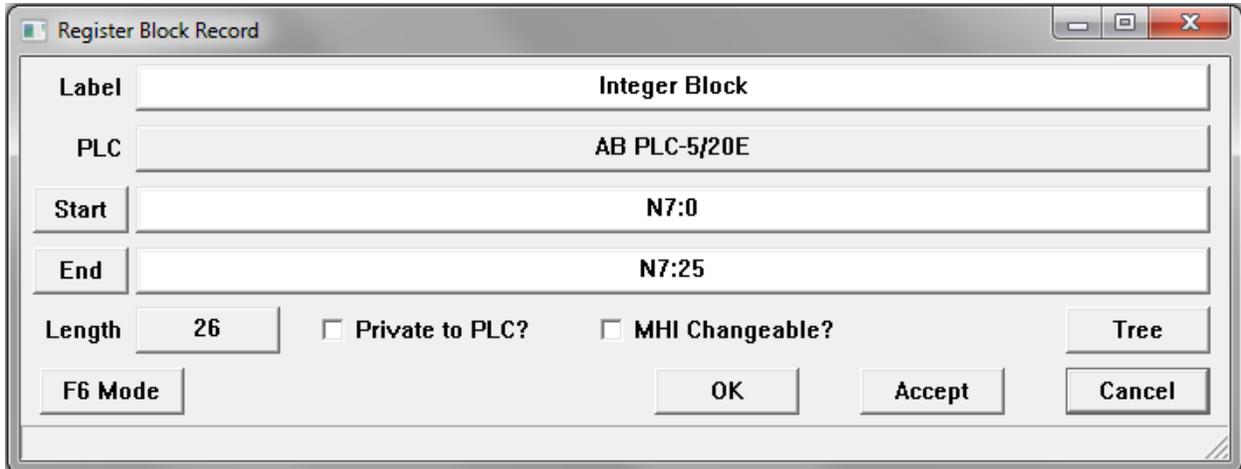
Data Source Parent

The developer must enter both the Start and End address of the block. The format used to enter these addresses varies with the type of the driver that contains the source. The end address should be a location that is after the start address. If the End address is left blank it is assumed that the block starts and ends at a single address.

When the 'Private' flag on a block is set to Yes the developer is marking that the block is used by the PLC and is not to be read or written by the Corsair interface. There should not be any tags or doors located in a private register block. This feature is for documentation purposes.

OIT Changeable Y/N

There is a single-record editing window for register block records.



xxxxxx

Tags

A tag is a storage location in computer memory that holds data while the Corsair program is running.

Tag Types

Corsair tag types describe the way that data is encoded. There are several types for many different purposes. The exact way that each type is addressed depends on the type of driver that is used.

When a tag record is created it initially has the ?? undefined type. The developer can press 'F2' on the type field and enter a type. He can also then press 'F1' to bring up a tag type selector listing all available types.

The 'indicator' tag type is a single-bit logical flag. Indicators can be changed by the PLC and read by the Corsair interface. They indicate on/off conditions. The corsair program cannot change the status of an indicator. The default text description for the program cannot change the status of an indicator. The default text description for the zero state of an indicator is 'Off'. The description for the one state of an indicator is 'On'. Tag enumerations can be used to change these descriptions if desired.

The 'switch' tag type is also a single-bit logical flag. Switches can be changed by either the PLC or the Corsair interface. When a human operator operates a switch tag he picks options from a switch operating window.

The 'button' tag type is a single of data The Corsair program turns the bit on to signal to the PLC to do something. The PLC then shuts the bit off. The Corsair operator does not have an option to shut off a button.

The 'HOA' tag type is a specialized 3-bit data type. One of the bits is controlled by the PLC. The other two are controlled from the PLC or from the Corsair interface. This type is commonly used in control applications where it is desired to force a motor on or off independently of normal automatic logic.

The 'integer' tag type is for a 16-bit integer value. It may be changed by the PLC or by the Corsair program. It may be signed value ranging from -32768 to +32767. It may be an unsigned value ranging from 0 to 65535.

The 'Double Int' tag type is for a 32-bit integer value. It works over a much greater range than the 16-bit integer type.

The 'Float' tag type is for a floating point value encoded in IEE standard 32-bit format.

The '4-bit alarm' type is for alarm status data arranged in Corsair standard bit pattern. The '1-bit alarm' type is for another type of alarm status data. The same bit patterns are available for calls.

String, Loose String, and Reverse String data types are for encoding of ACSII string data in PLC memory. Each type uses a different scheme for arranging the characters in PLC memory. The choice usually depends on the type of PLC and programming system that it uses.

The 'Cascade' type is used for larger-value integer data in PLCs that do not support 32-bit integers. It is useful in cases where integer precision needs to be maintained in totalizing. The Cascade uses two 16-bit registers. The net value is the second register multiplied by 10,000 and then added to the first. A total of 999,999 is possible with a Cascade.

The bit-field types vary from 1 bit to 16. Data encoded in these types is always unsigned and cannot be negative. Bit fields are frequently used to encode sequence options in recipe systems.

Tag Configuration

All tags are either 'Memory' or 'Live'. A memory tag gets its value from what is stored in the Corsair computer's memory. This value can be changed by the operator, by a script, from a block, or written from another computer. A 'Live' tag gets its value from a data source using a driver. The driver performs a scanning function where the value from the data source is continuously read into tag memory.

A company may have a tank that holds water. A pressure transmitter tied to the bottom of the tank is wired to a PLC. The PLC translates the electrical signal from the transmitter into a tank elevation in feet. This elevation is found at a memory address in the PLC. The developer sets up a driver and inserts a data source for the PLC under it. Under the data source he inserts a tag named 'Tank Level'. This tag is linked to a value placement on a screen. Now the operator can see the number of feet of water that are in the tank.

Every Corsair tag is an array. An array is a collection of values of the same type. Each of these values is an element of the array. An integer array with a size of 4 could contain the values 12, 97, -258, and 473. Elements of an array are distinguished by index number. Index numbers start at 0. They can be as large as the size of the array minus one. This array would be:

Element	Value
-----	-----
Index 0	12
Index 1	97
Index 2	-258
Index 3	473

If a developer creates a tag without entering a size for it Corsair defaults to a size of 1. When the developer wants to show the value of a tag on the screen he creates a value placement. He must link it to a tag and specify which index of the tag is to be shown.

Tag

Type

The 'Start' field is where the starting address of the tag is entered. It is left blank for a memory tag. If the tag is on a data source that is on a driver the format of the address depends upon the type of the driver. The developer can press the F2 key to begin editing the address and then F1 to open the address builder. The address builder will help him to enter an address that is correct for the driver type.

The 'End' field is a display-only field that shows the end address of the tag. The end address depends on the size of the tag, the length if it is a string, and the type of indexing that is selected for it.

The 'Format or Length' field is used to enter the data format for numeric tags or the length of string tags. The F1 key can be used while entering a format to open the format builder window. String lengths can vary from 1 to 78 characters long.

The 'Changeable?' field must be set to 'Yes' if the tag data is to be changed from Corsair. Indicator tags cannot be changeable. Button and Switch tags must be changeable.

The 'Size' field is the array element size of the tag. Even if a zero is entered Corsair will correct it to a value of 1.

Corsair tag arrays in data source memory can be indexed in two ways. The default indexing is 'Device size indexing'. The other option is 'Register indexing'. The 'IR Indexing by Register' flag is set to Yes to use Register indexing.

Session indexed tags use a special type of array indexing that is used in systems that have multiple sessions entered into the Corsair database. The 'SI Session Indexing' field is used to turn on this type of indexing.

Data Source Parent

Enumerations Zoom

Trend Sample Time

Trend samples are copies of the tag data taken at regular time intervals. These samples are used to show trend graphs of the tag data. They are also used for Turn-Back-Time history screens and for History review. The sample time parameter represents the time between trend samples. Trend sample times are entered with 1-second resolution.

Trend samples are used as the source of pretrigger data for logs. When multiple tags are on the same log and the pretrigger function is selected it is essential that all of those tags have the same trend sample time.

Trend Period Time

Trend samples are collected at a rate determined by the trend sample time. Data is collected over the total Trend Period. The number of samples is determined by how many trend sample intervals fit within the trend period. Corsair shows how many samples it is using. It typically adds at least one extra sample.

Long trend periods with a large sample count do not load down the Corsair computer because a circular buffer is used. Corsair does not have to move data through the buffer.

Authority Tag link

Alarms and Calls zoom

Block ID

Block Parameters Zoom

Block Variables Monitor Zoom

Min and Max values

When the history of a tag is displayed on a log or a history window its value is plotted versus time. The entered minimum and maximum values are the default scale values for the this tag on the plot.

Address 2, 3, 4

MBHR Address

Event Area

Units

The Units field is for a label describing the measurement units for the tag value. If it is a flow in gallons per minute 'Gpm' would be typical for this field.

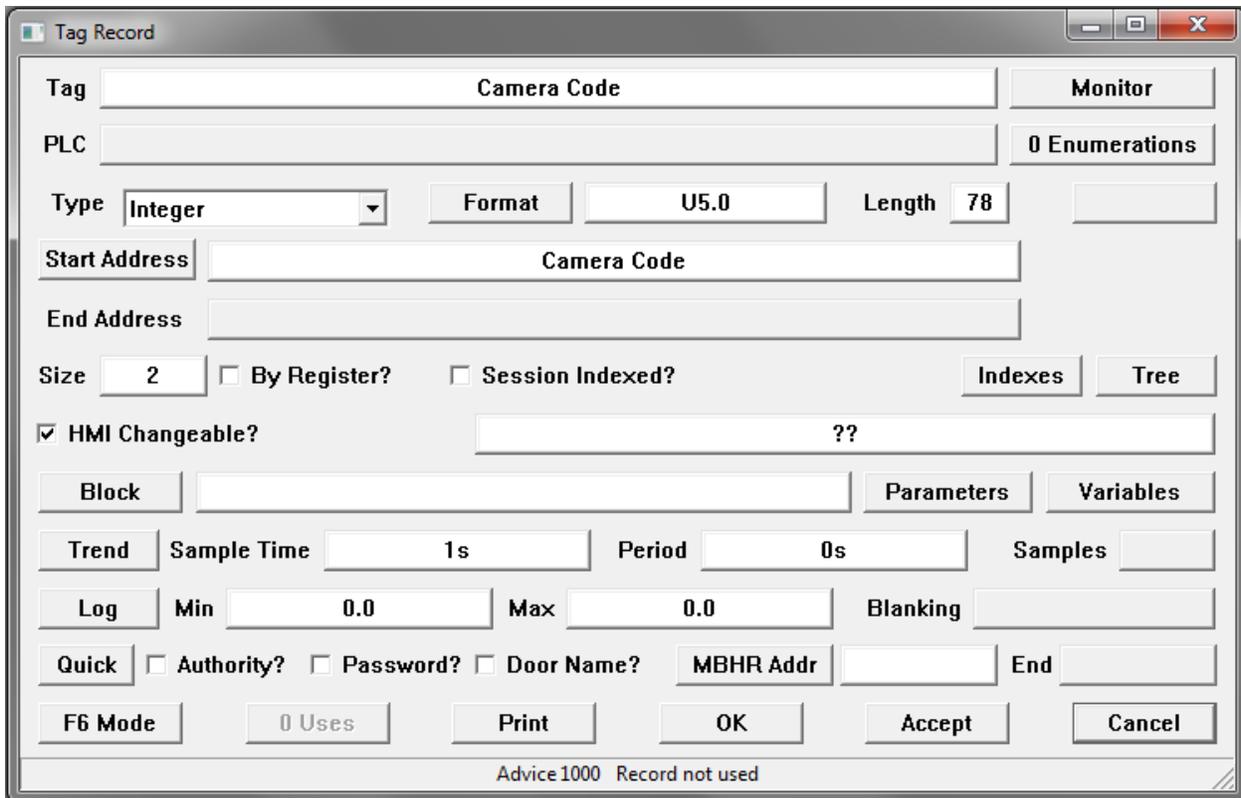
Integration (Msec)

Some tag values represent things that can be integrated over time. A common example is a tag with a flow in gallons per minute. A time-based integration of this flow would result in totalized gallons. A nonzero value in this time enables the integration. For a gpm flow this time should be set to '1m'. The setpoint can be resolved to one millisecond.

Totals Units

When a tag has a nonzero integration period Corsair can calculate totalized values. This field is for a label for the totalized units. If the tag is Gallons per Minute this label is typically Gallons.

There is a single-record editing window for tag records.



Tag Data Formats

Tag numeric formats are entered into the computer with a specification string. '-2.2', 'U5', and 'U,04.1' are all examples of format specification strings. Each specification string has an associated sample string that the computer can display. Examples of sample strings include '-##.##' and '##,###'

The following options may be chosen through the specification string:

- Signed versus unsigned data
- Comma separation of 3 digit groups
- Leading zero padding
- Display digit width
- Display decimal places
- Implied decimal point position for integer types

Specification: -5 or 5

Sample: -#####

A number in the specification shows the number of digits that the computer is to display for the value. The number can be as large as 31. Data is signed by default. This sample occupies 6 character spaces on the screen to allow for a minus sign followed by 5 digits. A 16-bit integer could range in value from -

32768 to 32767. A 32-bit double integer would have a much larger range of values. If the value is positive the minus sign in the sample is replaced by a space on the display.

Specification: U5

Sample: #####

The letter 'U' in a specification stands for unsigned. A 16-bit unsigned integer can range in value from 0 to 65535. No extra space is needed for display of a minus sign.

Specification: U,5

Sample: ##,###

A comma in a specification tells the computer to break digits to the left of the decimal point into groups of three with commas.

Specification: U04

Sample: #####

A zero before the digit count in a specification string tells the computer to display the value with Leading Zero Padding. A value of 5 will be displayed as '0005'. Zeros will be shown as needed to maintain a 4-digit display.

Specification: -2.2

Sample: -##.##

Implied decimal points are used with integer data inside PLCs. Care must be taken to not confuse this with floating point data. If a format shows two places to the right of an implied decimal point, an integer value of 123 in the PLC would be displayed as '1.23'. A value of 5 would be displayed as '0.05'. Corsair supports from 0 to 7 digits to the right of decimal points.

There are also several special purpose formats. These include:

AM/PM: The time will be displayed as '##:##AM'. It is encoded into the PLC register as an hour ranging from 0 to 23 multiplied by 100 and added to a minute value ranging from 0 to 59.

Date: The date will be displayed as 'Jan ##'. It is encoded into the PLC register as a month ranging from 1 to 12 multiplied by 100 and added to a day value ranging from 1 to 31.

MO/DA: The date will be displayed as '##/##'. It is encoded into the PLC register as a month ranging from 1 to 12 multiplied by 100 and added to a day value ranging from 1 to 31.

Weekday: The weekday will be displayed as 'Sun'. It is encoded into the PLC register as 0 to 6 with 0 displayed as 'Sun'

Duration: This format is used to display a time that is encoded in the PLC register as a number of seconds. It is displayed as '#d #h #m #s.'

Msec: This format is used to display a time that is encoded in the PLC register as a number of milliseconds. It is displayed as '#d #h #m #.####s'.

TOD: This is a time of day format that is encoded in the PLC as a number of seconds past midnight. It is displayed as '##:##:##AM'.

Pressing 'F1' while editing a tag format opens a window that helps to enter formats.

The screenshot shows a 'Device Format Help' dialog box. It has a title bar with the text 'Device Format Help' and a close button. The dialog contains three input fields: 'Type' with the value '??', 'Specification' with the value '-5.0', and 'Sample' with the value '-#####'. Below these fields are two radio buttons: 'Standard' (selected) and 'Special'. Under the 'Standard' section, there are two spinners: 'Left Digits 1-31' set to '5' and 'Right Digits 0-7' set to '0'. To the right of these spinners are three checkboxes: 'Unsigned', 'Show Lead Zeros', and 'Use Commas'. Under the 'Special' section, there are ten radio buttons: 'AM/PM (100H+M)', 'Date (100M+D)', 'Duration (Seconds)', 'TOD (Seconds)', 'Latitude', 'Error', 'Month/Day (100M+D)', 'Weekday (0-6 Sun=0)', 'Msec', 'DStime', and 'Longitude'. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

Strings

The Corsair program can work with character string data. When a tag record is given a string data type the developer has to enter in maximum length. This can vary from 1 to 78 characters. A size greater than 1 can be entered to create an array of strings.

There are 3 different string tag types. The first is called simply 'string.' The second is 'reverse string.' The third is 'loose string.' If the tag record is Corsair memory resident and not on a PLC all three types act the same. The difference occurs when the types are assigned to a PLC address. The 8-bit characters are arranged differently in PLC memory for each string type. An example would be the string 'ABCD.' The hexadecimal ACSII code for an uppercase 'A' is 0x41, B is 0x42, C is 0x43 and D is 0x44. Each of these

ASCII codes occupies an 8-bit byte of memory. PLC memory is usually divided into 16-bit registers. In this example the registers are numbered 40001, 40002 and so on.

If our 'ABCD' string is stored as a normal string type the values in the registers would be:

Register	40001	0x4241
Register	40002	0x4443

Each 16-bit register holds two 8-bit ASCII characters. The least significant byte of each register has the first character and the most significant byte has the second character of the pair. This is an 'LSB first' byte ordering. It is the natural byte ordering for the processors in the computers that run the Corsair program. Normal strings are the most efficient strings for the Corsair program to use.

If our 'ABCD' string is stored as a 'reverse' string type the values in the registers would be:

Register	40001	0x4142
Register	40002	0x4344

The most significant byte of each register has the first character and the least significant byte has the second character of the pair. This is a 'MSB first' byte ordering. Some processors use it as a natural byte ordering.

If our 'ABCD' string is stored as a 'loose' string, the values in the registers would be:

Register	40001	0x41
Register	40002	0x42
Register	40003	0x43
Register	40004	0x44

Each character occupies the least significant byte of a register. The most significant byte is ignored by the Corsair computer when it reads data from the PLC. It may be set to zero when Corsair writes to the PLC. Loose strings use twice as much PLC memory as the other string types do.

The Corsair program has experts to help the PLC programmer write code for handling ASCII strings. The first aid is an ASCII code reference chart that may be printed out. The second aid is a window that accepts ASCII strings and converts them to PLC register data for normal, reverse, and loose strings. The third aid is a register value to ASCII string converter.

Alarms

Alarm database records describe items that can appear on the operator's alarm summary display. Alarms do not show until they are tripped. After they are tripped the operator acknowledges them. The next step is for the alarm to be reset. Resets can come from PLC logic or from the operator.

The ID name of the Alarm is used to identify the alarm and enable it to be linked to graphic placements.

The index field is a number that locates the alarm on its parent tag. The parent tag must have a 1-bit or a 4-bit alarm type. If that tag is located on a data source Corsair can calculate and display a Start and End address. These addresses are not changeable by the operator. He can only adjust the index value.

Common Corsair development practice is to initially develop Alarms with zero in the index field. After the alarm list is substantially complete nonzero indexes are entered to place the alarms in their final locations.

Tag Parent

Resettable?

Description

An alarm that is not on a parent tag is considered to be a memory alarm. Memory alarms can be initiated from a Corsair block. The name of the block is entered on the alarm record. Block names entered on alarms that are not memory are not used.

Block Parameter Zoom

Block Variables Monitor Zoom

Latching Y/N

Email Zoom

Sounds Zoom

Play

Repeat – Sound Repeat Time

The default behavior for a Corsair alarm is to be shown on the Alarm Summary window when the alarm is active. This may not be a good idea for some alarms. 'Amount' alarms that are used to log production totals are like that. A field named 'Not on Summary' can be set to 'Yes' to keep it from showing.

Event Area

Sessions Zoom

Ack Authority Link

Reset Authority Link

Priority #

There is a single-record editing window for alarm records.

The screenshot shows a software window titled "Alarm/Call Record". The window contains several input fields and buttons. At the top, there is a field for "ID" with the label "Alarm Identitiy" above it. Below this, there is a row with "Index" (value 1), two checkboxes for "Op Reset" and "Latch", a "Form" field (value Maintenance), and a "Maintenance" button. The next row contains "PLC" and "Tag" fields. The following row contains "Start" and "End" fields. Below these is a "Descrip" field and two buttons: "Events" and "Email". The next row contains a "Block" button, a field, a "Parameters" button, and a "Variables" button. The bottom row contains several buttons: "F6 Mode", "0 Uses", "Tree", "OK", "Accept", and "Cancel".

The 'Events' button leads to a window that characterizes the event logging for the alarm.

Alarm Identitiy - Alarm Event Logging

Show on Summar Area

Class **Standard** Priority **0**

Custom Event Labels

Trip	<input type="text"/>	TripCmd	<input type="text"/>	L
Ack	<input type="text"/>	AckCmd	<input type="text"/>	C
Reset	<input type="text"/>	ResetCmd	<input type="text"/>	

Tag Values

Tag 1	<input type="text"/>	??	Index	<input type="text"/>	0
Tag 2	<input type="text"/>	??	Index	<input type="text"/>	0
Tag 3	<input type="text"/>	??	Index	<input type="text"/>	0
Tag 4	<input type="text"/>	??	Index	<input type="text"/>	0

Trip Event Complete

Good	<input type="text"/>	??	Index	<input type="text"/>	0	Value	<input type="text"/>	0
Bad	<input type="text"/>	??	Index	<input type="text"/>	0	Value	<input type="text"/>	0

OK Accept Cancel

The 'Email' button opens a window that characterizes what the Corsair email system does with the alarm.

	Key	Send Group	Delay	Repeat
Trip Message	0	Grp	0	0s
		Grp	0	0s
Review		Grp	0	0s
Ack Message	0	Grp	0	0s
		Grp	0	0s
Review		Grp	0	0s
Reset Message	0	Grp	0	0s
			0	0s
Review		Grp	0	0s

OK Accept Cancel

Corsair can send emails when an alarm is tripped, when it is acknowledged by the operator, and when it is reset. As many as 3 email send groups can be entered for each of these actions. The 'Review' buttons can be used by the developer to see what each type of email will look like.

Calls

Corsair Calls are very similar to Corsair Alarms. The field options are nearly identical.

ID

Index

Start

End

Tag

Resettable?

Description

Block ID

Block Parameters Zoom

Block Variables Monitor Zoom

Email Zoom

Sounds Zoom

Play

Repeat time

Not on Summary Y/N

Event Area

Sessions Zoom

Ack Authority Link

Reset Authority Link

Priority #

There is a single-record editing window for call records.

Alarm/Call Record

ID

Index Op Reset Latch Form

PLC Tag

Start End

Descrip Events Email

Block Parameters Variables

F6 Mode Tree OK Accept Cancel

xxxxx

Icons

Name

X size

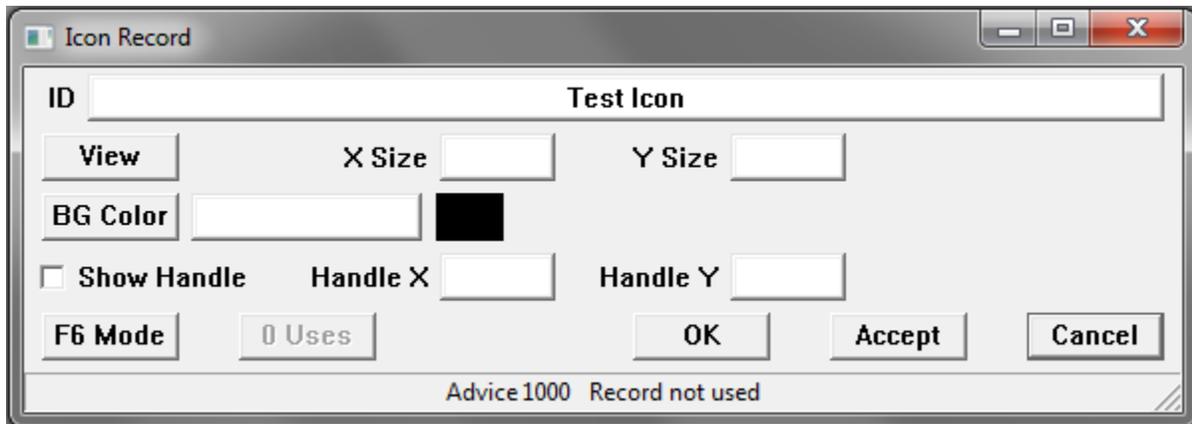
Y size

The color field is used to specify the background color of the window that is used to view the icon.

An icon has a handle point. This is a single point on the icon that is used to determine where the icon is located on a screen.

Options are available to turn the display of the handle point on and off and to enter its X and Y coordinate positions.

There is a single-record editing window for icon records.



The 'View' button allows the developer to see the icon.

Drawings

A drawing is a collection of graphic placements that are used as a set. A developer may make a drawing of a motor. It may be a rectangle with some ellipses on the ends. The motor drawing can then be placed many times on Corsair screens for the operator to see. Changing the drawing will change the appearance of all the motor placements.

The ID Name of the drawing is used by the developer to hook the drawing into placements on screens.

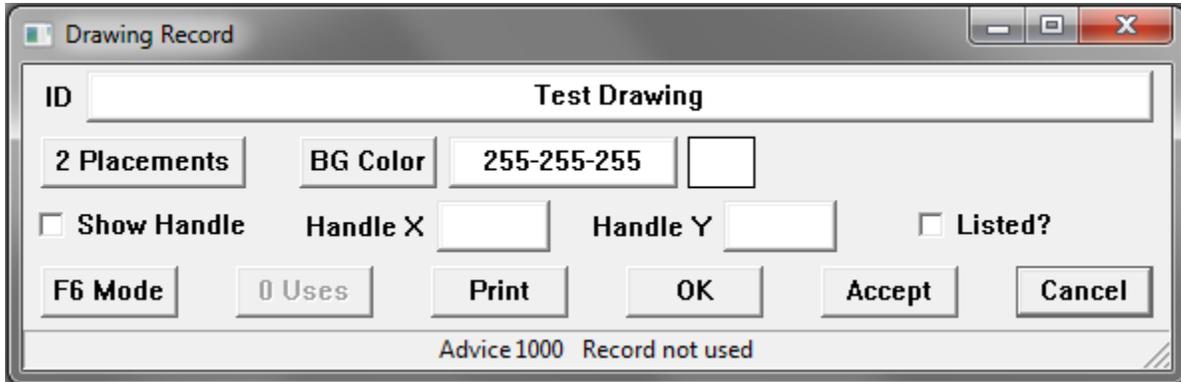
The color field is used to specify the background color of the window that is used to develop the drawing.

A 'listed' drawing is a drawing that is used for part of a project's documentation. Listed drawings can be viewed by the operator or printed.

A drawing has a handle point. This is a single point on the drawing that is used to determine where the drawing is located on a screen. It is not used on a listed drawing.

Options are available to turn the display of the handle point on and off and to enter its X and Y coordinate positions.

There is a single-record editing window for drawing records.



The Placements button on this window opens up editing of graphic placements. This is how drawing images are created.

Screens

A screen is a collection of graphic placements that are viewed by the operator.

The ID name of a screen is used as a label to identify it.

The Escape Screen link is used to specify what screen is displayed when the operator escapes from the screen.

Blank bottom status bar? Y/N

The color field specifies the background color for the screen.

Blank top title? Y/N

Pattern – is it a pattern?

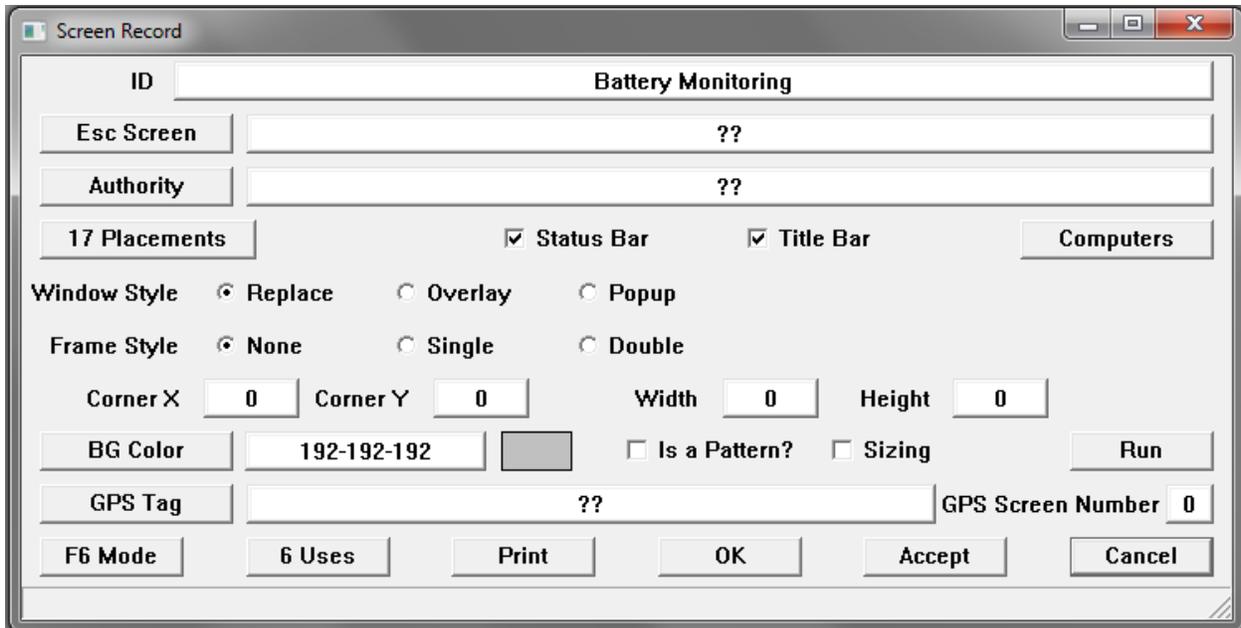
Authority Link

Sessions List

GPS area Tag

GPS Screen Number

There is a single-record editing window for screen records.



The Placements button on this window opens up editing of graphic placements. This is how screen images are created.

Sounds

Corsair can play wave sound files when alarms or calls are tripped. This results in sound from the computer's speakers.

The first step is to create a folder on the computer's hard drive to contain the sounds. A recommended possibility is:

C:\cwaves\

This folder will contain all the wave files that the Corsair program is to use. Each of these files has a .wav extension. Corsair has no provision for generating wave files. The sound recorder that comes with Windows is a common way to generate them.

After the wave files are generated and placed in the proper folder the developer can start the Corsair program. There is an option under the 'Tools' menu called 'Play Sound' that can be used to test if the computers speakers are working properly.

Each Corsair computer must be configured to use sounds. This is done as a part of Setup/Computer Properties. The interface tab includes an option to 'Use Sound' that must be checked. Below that option is an edit box for entering the path to the folder where the wave files are. C:\cwaves\ would be a typical entry for this option.

The next step is to create sound records in the Corsair database. This is done under Edit/Graphic/Sounds. Each sound record gets a name. This is the identity that will be used to hook the sounds to alarms – it is not the name of the wave file.

Zooming under each sound goes to a database of the wave files that are used for that sound. A Sound can consist of several files that are played in consecutive order. It is possible to put a short time delay before each file is played. Normally this timing is left at zero. It can be as short as one millisecond (entered as 0.001s)

Wave file names can be typed into the database or found using a 'browse' function. The browse function will return a full path specification to the file. The path specification in the computer properties is put on the front of the file name that is in the database. Many times if a file name is found with the browse the path specification will have to be removed from the file name. The computer cannot play C:\cwaves\C:\cwaves\ding.wav.

Corsair has 'Play' database fields that can be used to test sounds. They are available for individual wave file records and for sound records that include multiple wave files. The 'Use Sound' checkbox under Computer Properties must be checked for these 'Play' files to work. The play field is activated by arrowing to it and pressing 'F2' edit key. Another option is to click on the field and then click on it a second time.

The Tools/Monitor Sounds menu option is a way to see what is happening with the sound system. It has an option to cancel playing of the current sound.

The next step is to associate sounds with alarm and call records. Each alarm and call record has a zoom field that allows entry of one or more sounds. A 'Play' field is available for testing sound.

Sound Record Fields

ID

Files Zoom

Play

RTP Tag – Request Talk Path

RTP Index #

RTP Value #

TPR Tag – Talk Path Ready Tag

TPR Index

CTP Tag – Clear Talk Path Tag

CTP Index

CTP Value

Timeout (seconds)

Sheets

A user sheet is a collection of data in a row and column format for use by the operator. The developer must specify a name for the sheet and how many rows and columns it has.

There is a single-record editing window for sheet records.

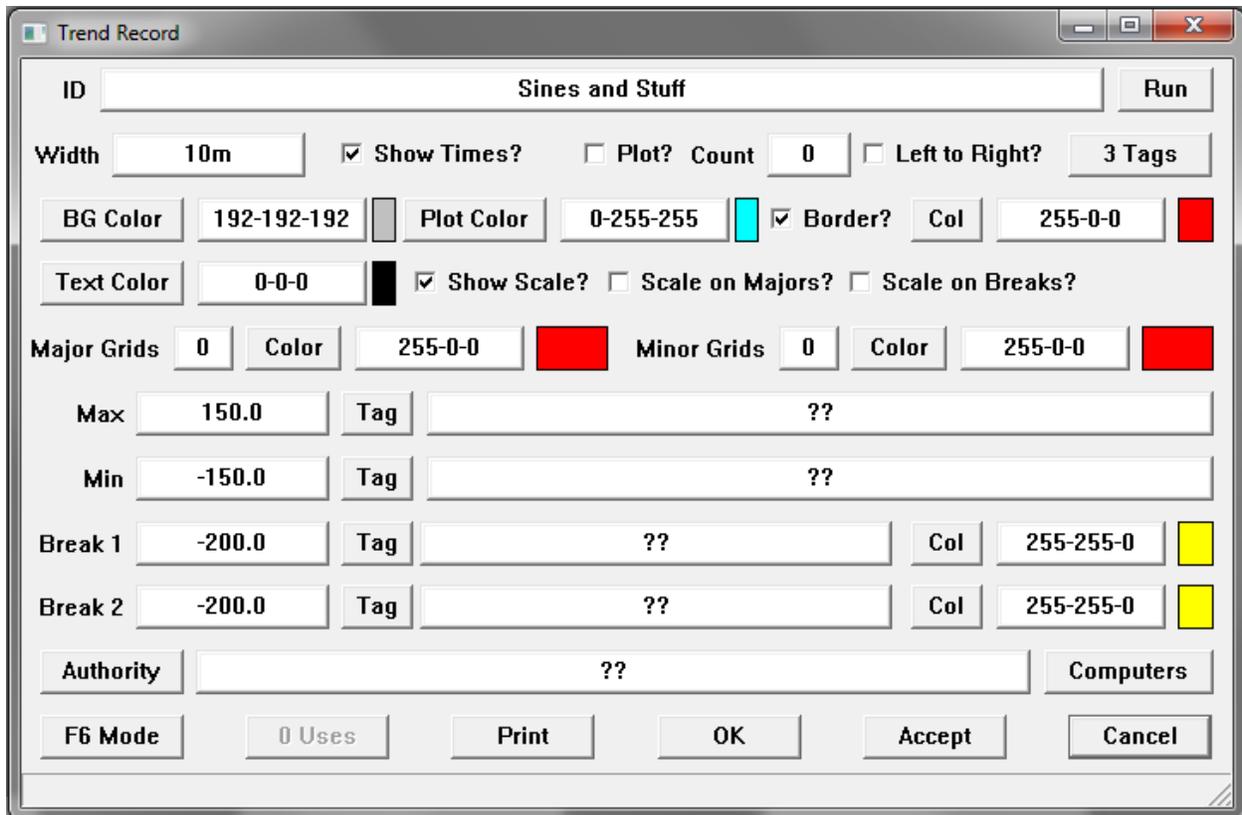
The screenshot shows a 'Sheet Record' dialog box with the following fields and controls:

- Name:** Discrete Input Conditioning
- Rows:** 80
- Columns:** 15
- Computers:** (button)
- Run:** (button)
- Page Move Row Count:** 0
- File Name:** (empty text field)
- Allow Export?
- Allow Import?
- Is a Pattern?
- Authority:** ??
- F6 Mode:** 1 Use
- OK** (button)
- Accept** (button)
- Cancel** (button)

XXXX

Trends

There is a single-record editing window for trend records.



The 'Tags' button opens a window that is used to place tags on the trend. These tags should be set up to have trend sample data. Typically the Trend Period on the tag is set to be the same as or longer than the width of the trend.

Scripts

Scripts are programs that are created by the Corsair developer using script steps. The type of the step determines what the step does. Different step types require different parameters. The CorsairHMI Designer manual describes the available script step types.

There is a single-record editing window for script records.



XXXXX

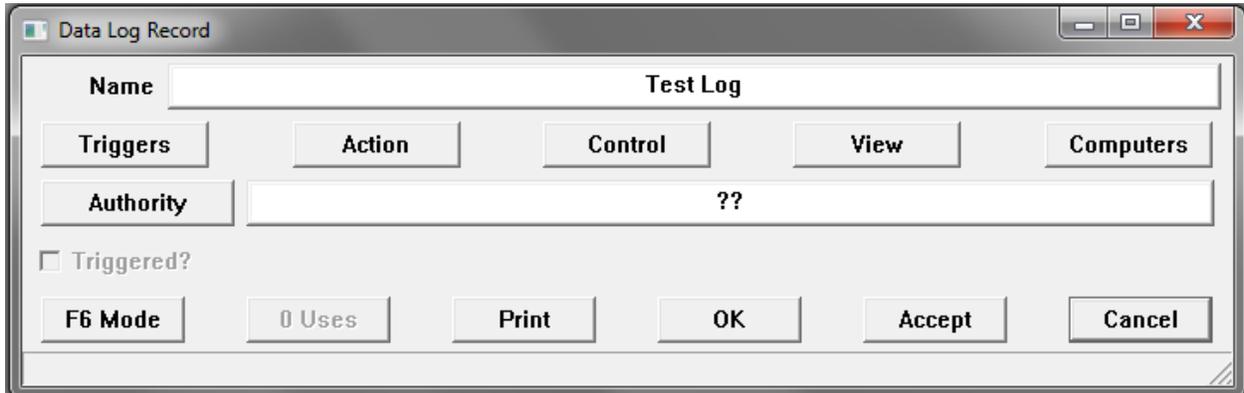
Data Logs

Corsair Logs are tasks that the Corsair computer can perform based upon trigger conditions. Logs can be used to record historical data in disk files, to print, and do other tasks.

Creating a Log

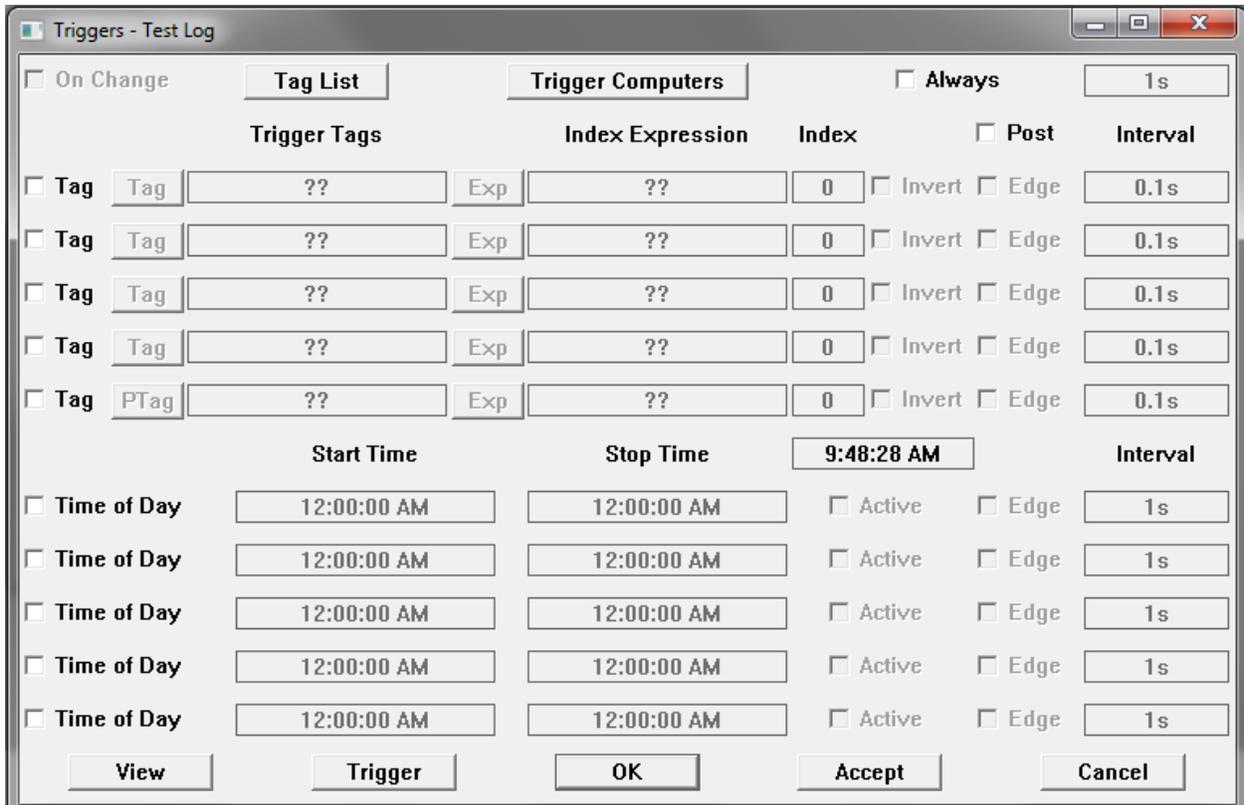
Logs are created from the main menu's Edit/ More/ Logs selection. The F4 Create key can be used to create a log record. There are 5 zoom fields.

There is a single-record editing window for data log records.



Each of the zoom windows has a 'trigger' button on the bottom. These buttons are for the Corsair developer in case he wants to manually trigger execution of the log.

The Triggers Zoom



The Triggers zoom opens a window that is used to configure the triggering of the log. Logs can be triggered continually, upon changes in tag data, from trigger tags, or based upon the time of day. Combinations of any or all of these triggers are possible.

The Trigger Computers button on the top of the window determines which computers in a multiple computer system are allowed to trigger the log. A group of 4 computers may all need to look at the data in a log file that is on a server. Each computer can view the file but only one of them is to write to it. That computer should be the only one that is highlighted in the Trigger Computers list.

The simplest form of triggering is Always Triggering. The Always checkbox must be checked and a nonzero time interval entered to the right of it. If it is set for '30s' the log will trigger on interface startup and every 30 seconds after that.

Data Change triggering occurs when the data in a tag changes by more than an entered value. The Tag List button is used to enter a list of tags that are used by the log. As each tag is entered the developer can signify if it is to be used for data change triggering. The 'On Change' checkbox on the Triggers zoom window shows if any tags have been configured with data change triggering enabled.

A log can also be triggered based upon the status of up to 5 tags. Each of these tags can be entered as a 'Corsair Triplet' consisting of a tag, and index expression tag, and an index constant. The 'Tag' checkbox on the left side of the window must be checked for this entry to happen. These tags are normally indicators or switches as the log looks for an on-off value to determine triggering. Triggering occurs when the tag is 'On' unless the 'Invert' checkbox is checked. Invert will cause the log to trigger when the tag is 'Off'.

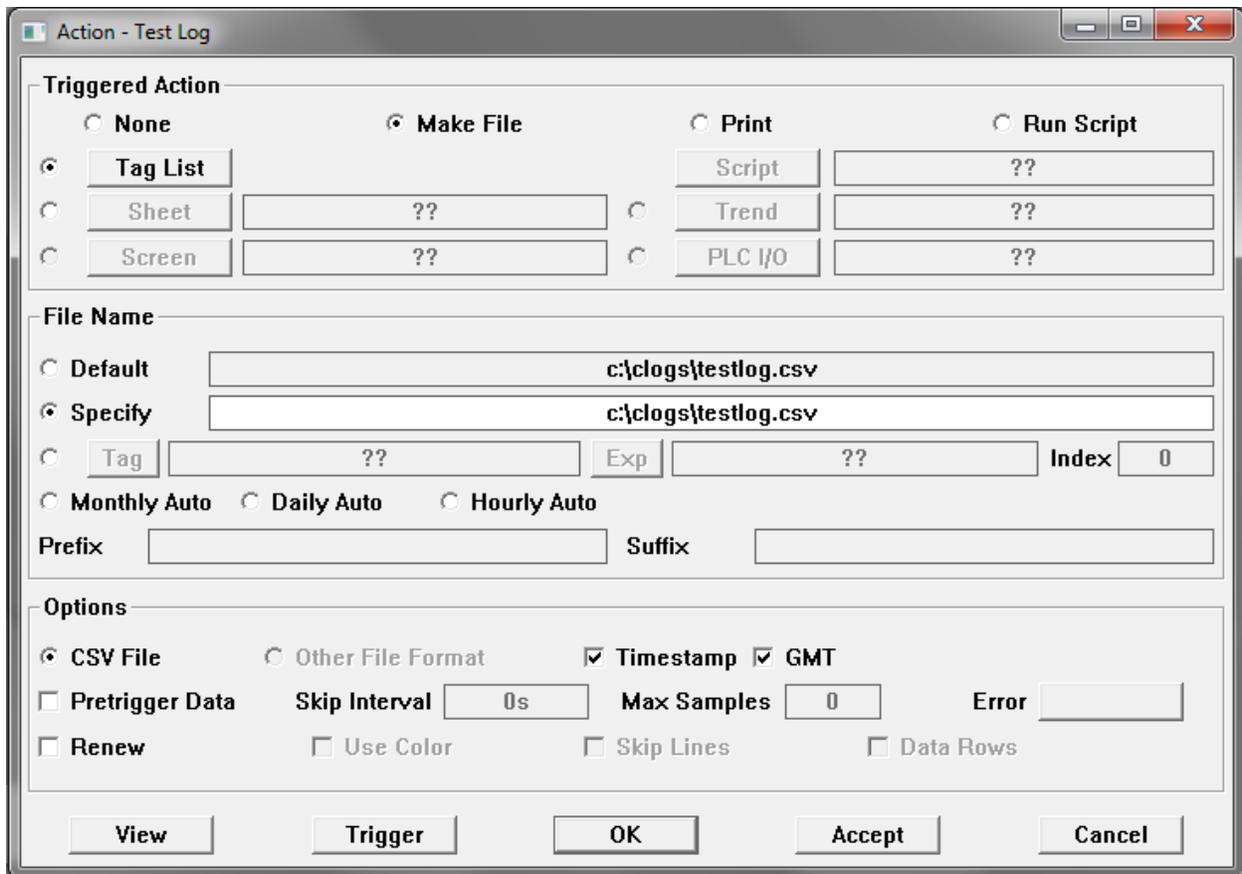
When the tag is On (or Off if inverted) Corsair can do two possible types of triggering. Edge triggering means that that log only triggers on the Off to On transition of the tag data. For it to trigger again the tag must go Off and then back On. Repeat triggering happens with the Edge checkbox is not checked. It uses a time interval. If a tag has 30 second repeat triggering the log triggers when the tag comes on and at 30 second intervals until the tag goes off.

Post triggering is an option that can be used with tag triggering. If 'Post' is checked the log will trigger one more time after all of the trigger tags go to Off.

Some logs may be set up to output pretrigger data to a log file. This is set up under the log Action zoom. Pretrigger data is only used when the fifth of the five possible trigger tags is turned on. The find button for this tag says 'PTag' instead of 'Tag'.

There are 5 possible time of day triggers for each log. If they are checked the developer enters a start and stop time for each one. The computers current clock time is displayed. Each time of day trigger has an 'Active' checkbox that shows if it is currently active based upon the computers clock. Edge and Repeat modes work just like with the tag triggers.

The Action Zoom



The Action zoom window for a log determines what the log does. It needs a combination of an action and an object. The possibilities are:

Action – None

Action – Make File Object – Tag list

Action – Make File Object – Sheet

Action – Print Object – Sheet

Action – Print Object – Screen

Action – Print Object – Trend

Action – Print Object – PLC I/O

Action – Run Script Object – Script

Make File – Tag List – is one of the most common log tasks. It is used to log historical data to a CSV file. The Tag List button is used to enter the list of tags. This is the same tag list that is used for data change triggering. A tag list can be used for change triggering even if the Object is not Tag List.

The Run Script action can be used for time of day scheduling of a script.

A Make File action requires the specifying of a file name. The file name can be the default as shown. It can be set to 'Specify' and entered. It can come from a tag triplet. It can be varied automatically based upon month, day, or hour. The Prefix entry is typically used for a path specification. The Suffix entry is typically used to specify an extension like '.csv'.

The Options group box shows other options for the log action. The 'Timestamp' checkbox is used with Tag List files to place date and time fields at the start of each line of the file. This can be in the local computer time or in GMT (UCT) time. GMT is nearly always a better option because it eliminates confusion in the data records when changing to or from Daylight Savings Time. There are 6 date and time fields. They are all numbers. The first is the year, the second is the month as 1 through 12, the third is the day of the month, the fourth is the hour as 0 through 23, the fifth is the minutes from 0 through 59, and the sixth is the seconds. The seconds value may have up to three decimal places. '23.456' means 456 milliseconds after 23 seconds after the minute.

The normal mode for a disk file of tag data is for Corsair to create the file and then append records to it. The 'Renew' option tells Corsair to create the file each time that it is triggered. Renew log files will only have 1 record of data in them. They are designed for real-time file updates that are read by other software.

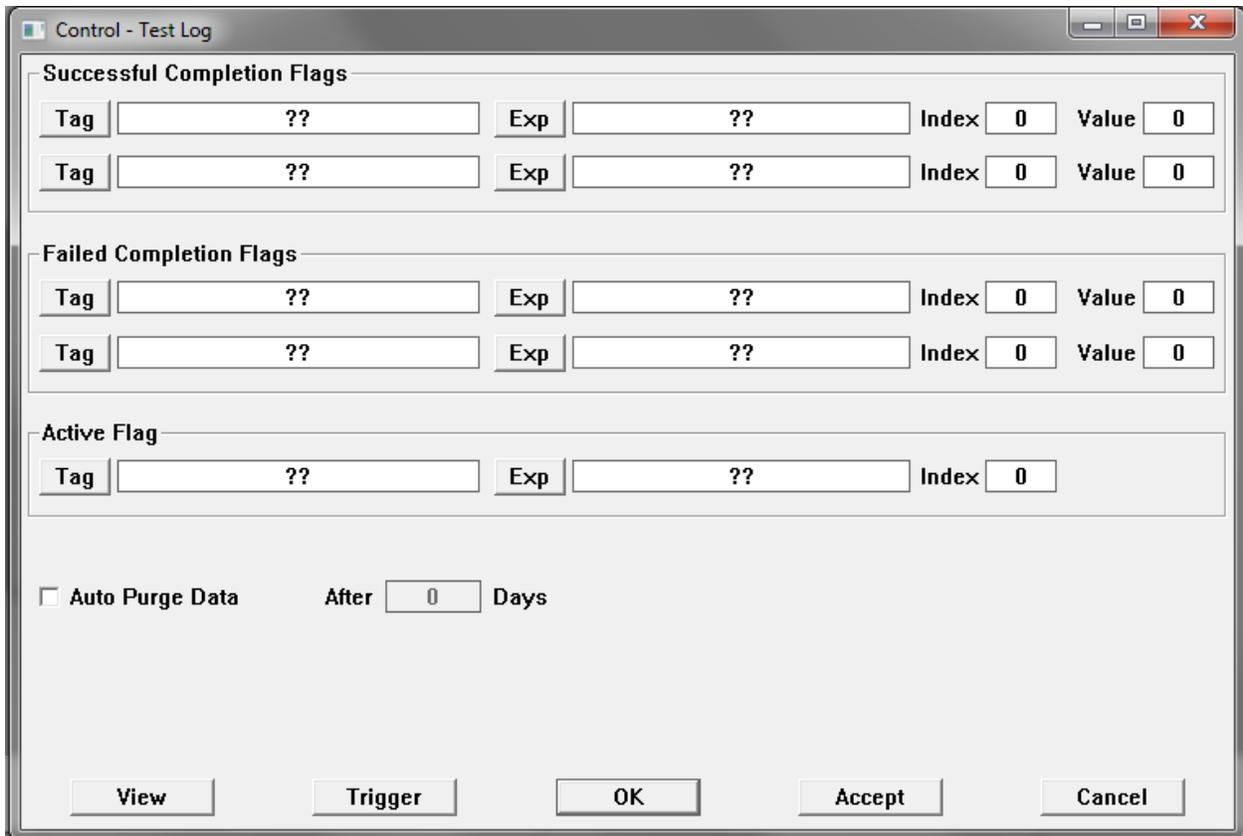
The 'Use Color' option is used with print logs to tell Corsair to use color printing whenever possible instead of Black and White.

The Data Rows option is used with Sheet File and Sheet Print actions. If Data Rows is selected Corsair will look up from the bottom of the sheet to see how many of the bottom rows are blank. They will not be printed.

The Other File Format radio button is not available. CSV files are the only currently supported format for logs.

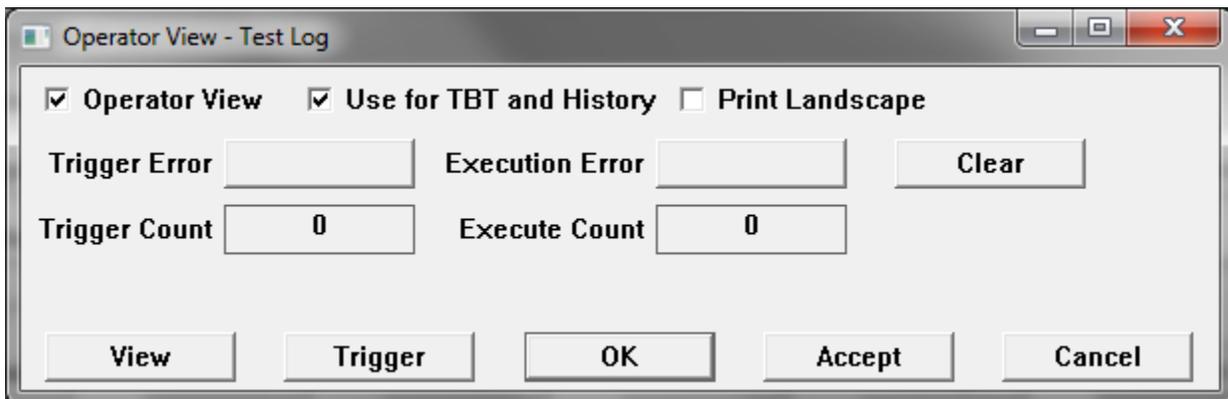
Pretrigger Data is an option with Tag List files. If it is checked and the log is triggered from the fifth of the five possible tag triggers Corsair will attempt to insert pretrigger data into the file. This is trend data that Corsair remembered in the form of trend samples from the time before the trigger occurred. Each tag that is on the log must be set up with the same trend sampling period and trend interval. When the trigger occurs Corsair 'steps back' into the trend data by the entered 'Skip Interval' time. It puts up to the 'Max Samples' count of samples of pretrigger data into the log file. If Max Samples is set to zero Corsair will insert as many pretrigger samples as it has available.

The Control Zoom



The Auto Purge Data option is currently not available for Disk File logs.

The View Zoom



The View zoom window is used to control viewing options for the log. The Operator View checkbox is used to allow operator viewing of the log data if some form of view is available for the logs Action and Object.

The Print Landscape checkbox is used to tell Corsair to print landscape instead of portrait if landscape printing is available for that type of log.

The Trigger button on the bottom is used to manually trigger the log. If it is successfully triggered the Trigger Count will increment. It then goes into a list for execution. If it executes successfully the Execute Count will increment. The last Trigger or Execution error is shown. The Clear button clears the error code and the counts.

The Computer Zoom

The Computer Zoom field is used to access a list of computers that can use the log. It is used in multi-computer Corsair systems. Computers that are not selected in this list will not be able to view the logs files.

Variable Rate Logging

CSV data files are not the most efficient way to store data. Data stored at a short time interval can result in huge files with slow access times.

Many processes require high-speed logging while they are running but they don't run very often. There is no need for many data samples when the process is not running. Variable rate logging can handle these situations.

Suppose a process employs a kiln that is used to heat material that flows through it. Logged tags include:

Material Temperature – Ambient to 400 degrees F

Material Tons per Hour – 0 to 20.0 tph

Stack Temperature – Ambient to 600 degrees F

Draft – 0 to 10 inches water column

While the kiln is in use data is desired at a 15-second interval. It is desired to see the complete ramp-up at the beginning of the run and the ramp down at the end. A combination of a variable logging rate combined with pretrigger data, a posttrigger sample, and logging on change will work for this system. A contact that shows that the combustion blower is running controls a Corsair indicator tag.

The first step is to set up a c:\clogs\ folder for storing Corsair log data. The next step is to create a log record and go to the 'Triggers' zoom. Check 'Always' and enter '8h' for the Always interval. This will result in a minimum of 3 samples on days that the kiln does not run. Check 'Tag' on the fifth trigger tag. The 'PTag' button can be used to select the combustion blower running indication into the Trigger Tags column. Set the corresponding interval to '15s'. Check the 'Post' checkbox and accept the window.

Next go to the Action zoom. The Trigger Action is 'Make File'. The file is a 'Tag List'. The file name is 'Monthly Auto'. The prefix is 'C:\clogs\kiln_data_'. The suffix is '.CSV'. For options pick 'CSV File', 'Timestamp', 'GMT', and 'Pretrigger Data'. If the pretrigger samples are at a 15-second interval a 'Max Samples' value of 20 would produce 5 minutes of pretrigger data. Accept the Action window entries.

The next step is to go to the 4 tags that are going to be logged. Enter the minimum and maximum scale for the plot of the value. Enter a sample time of '15s' on each tag. Enter a trend period of '2h' for 2 hours. All the tags on a log with pretrigger data need to have identical sample times.

The material Tons per Hour tag should have the units label set to 'tph'. The integration period should be set to '1h'. The Totals Units label should be set to 'Tons'.

Next the Logs 'Tags' zoom is used to list the tags on the log. Each tag gets the desired pen color on the graph. Each tag should have the 'On Change' set to 'Yes'. Set the 'Change Delta' to 20.0 for material temperature, 1.0 for tons per hour, 30.0 for stack temperature, and 1.0 for draft.

The 'View' zoom is used to check 'Operator View' and 'Use for TBT and History'.

2-hour trends can be created for each tag if desired.

Now the log will always trigger every 8 hours. If the combustion blower comes on the previous 5 minutes of pretrigger data will be logged at 15 second intervals. All tags will log every 15 seconds as long as the blower is running and for one extra sample after the blower stops. The log is also triggered whenever a variable changes significantly.

Ethernet Objects

Ethernet Objects are database records that are entered by the developer to be shown as objects on the Communications Architecture printout.

There can be as many as three IP addresses entered for an Ethernet object. These are addresses for the 'PLC', 'MIS', and 'Other' networks. If they are left at 0.0.0.0 the object is not on that network. If 255.255.255.255 is entered the address is displayed as 'Auto'. Other values are shown as normal IP addresses.

Up to three notes can be entered on the Object to be printed on the architecture.

SQL Paths

There is a single-record editing window for SQL Path records.

SQL Path

Path Name: CIC SQL Path

Database Type: Old MS SQL Server

Server Clock GMT

Use DSN? DSN: [] Driver: SQL Server

Server: Dell_Laptop Database: EventDatabase Schema: dbo

Table: EventTable Trusted? User: []

Password: [] Extra: [] File: []

Buttons: F6 Mode, 2 Uses, Print, OK, Accept, Cancel

XXXXX

FTP Paths

There is a single-record editing window for FTP Path records.

FTP Path

Name: Events FTP Path

Server: []

Port (Dec): 0 Anonymous Active Use Proxy ASCII

User: []

Password: []

Proxy: []

Bypass: []

Buttons: F6 Mode, 4 Uses, Print, OK, Accept, Cancel

XXXXX

Driver Paths

Xxxx

Local and Global Database Views

Corsair uses a treelike family structure for database records.

Driver 1

PLC A

Tag A1

PLC B

Tag B1

Tag B2

Driver 2

PLC C

Tag C1

Tag C2

There are two driver records. The first has two PLC records titled A and B. The second has a single PLC record titled C.

A developer may be looking at the Driver 1 record. If he zooms on the data source zoom field he sees PLC A and PLC B. If he goes to the Driver 2 record and does the zoom he sees the PLC C record. The data source sheets that he sees are 'Local' views that are under the driver. They only show the data sources for the driver that started the zoom.

If the developer chooses the 'Edit' 'Data' 'Sources' menu option he sees PLC A, B, and C. This is a 'Global' view of the data sources.

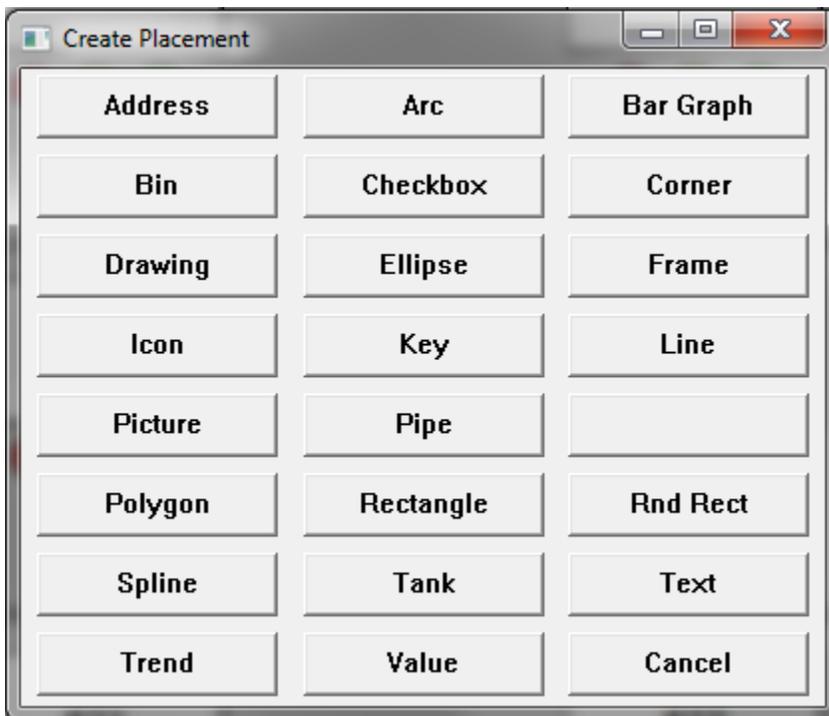
The same system of local and global views of data sources also applies to local and global views of tags.

When development data is being shown with a local view a menu option is available to switch the display to a global view. Another option switches it back to local.

Creating Graphics

The 'Edit' 'Graphics' 'Screens' menu option opens the development sheet that creates screens. Zooming on a screen record opens graphic development of that screen. The same thing is true for drawings.

Objects that are shown on Corsair screens or drawings are called 'placements'. These are the 'graphics primitives' of the Corsair system. Placements are created by moving the cursor to a location on the screen and pressing the F4 'Create' key. This opens the placement type selection window. The 'type' of a placement determines what the operator sees when the placement is shown on the screen.



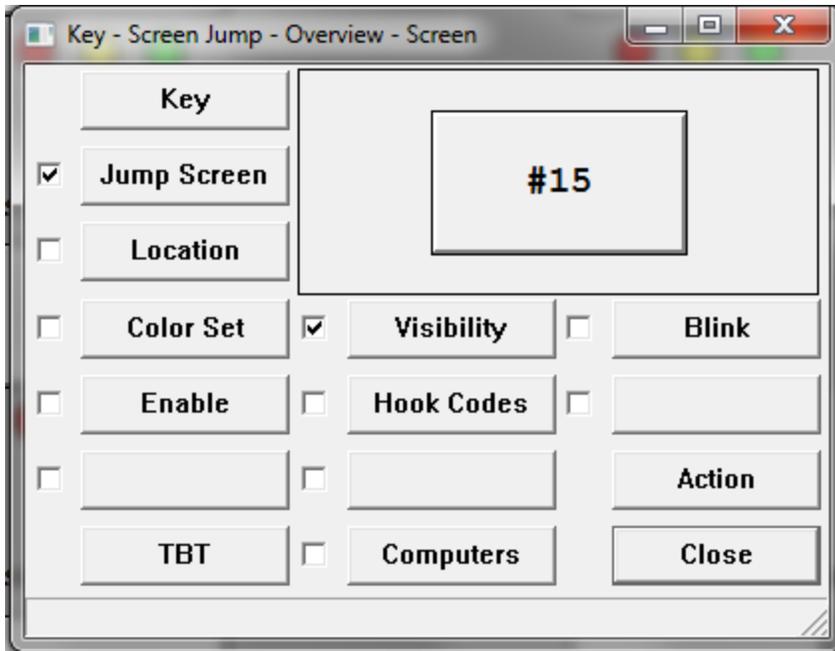
The developer must select a placement type. This type cannot be changed after the placement is created. If a wrong type is selected the placement must be deleted. The next step is the placement action window.



The developer must select the action for the placement. The 'action' determines what happens when the operator clicks on the placement. The available actions are different for screen and drawing placements.

The type of a placement and its action are independent. Each type can have any allowed action.

The developer should select an action. Now he sees the placement editing window. The appearance of this window changes based upon the placement type and action.



The updated appearance of the placement is shown on the upper right part of the window. It may be miniaturized if needed to fit in the space. The window is used to open other windows that change characteristics of the placement or a group of placements. After it is closed the placement can be moved and sized using a mouse and the keyboard.

Every placement has one or more handle points depending on the type. A text placement has one handle point. A rectangle has eight. To move a placement the developer clicks near one of the handle points. A second click on the same point causes the placement to start flashing. The developer moves the mouse cursor to a new location and clicks again to locate the placement. The movement can be canceled with the 'ESC' key.

The size of some placement types can be adjusted by rubber-banding. This is started by holding down the shift key while clicking on the handle point. This makes it a rubber-banding move instead of a whole placement move.

Placements frequently overlap each other on the screen. When a placement is selected there is a 'Front' and a 'Back' menu option. These options can be used to create any desired overlap pattern.

After one placement has been selected another can be added to the selection by pressing the Control key while clicking on the second placement. Placements can be moved as a group.

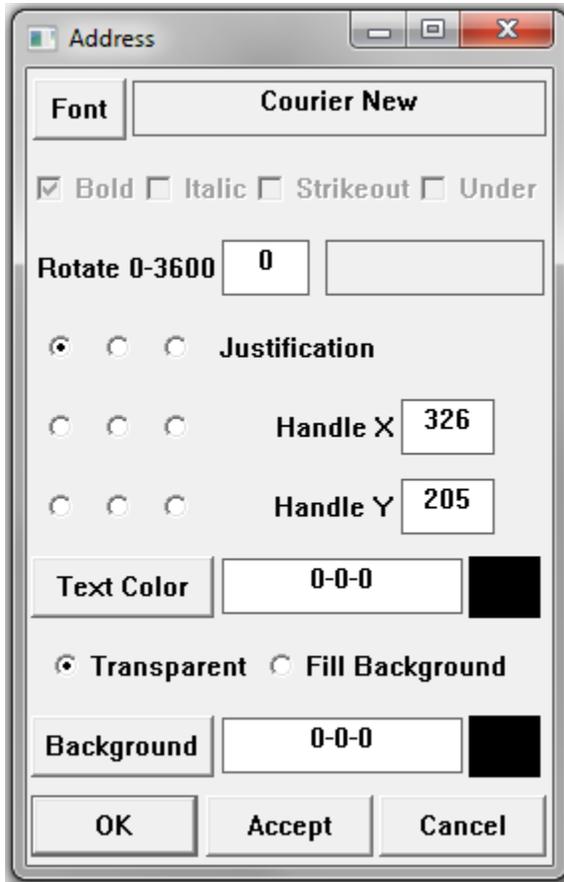
Pressing F2 opens the placement editing window for the selected placement(s).

Placement Types

The type of a placement corresponds to what the operator sees when the placement is on a screen.

Address

Xxx



Arc

Xxx

Arc

Center X Center Y

Start X Start Y

End X End Y

Width Height

Thickness Solid Dash

Dot Dash Dot Dash Dot Dot

Arc Color 

No Fill Chord Fill Pie Fill

Fill Color 

Bar Graph
Xxx

Bar Graph

Left X Top Y Width Height

Vertical
 Horizontal
 Don't Show Breaks
 Show Breaks

No Hover Text
 Hover Text
 Single Break
 Break Array
 Array Size

Font Hover 

Bold
 Italic
 Strikeout
 Underline

Value Color  Minimum Maximum

Background 
 No Border
 Border
 Color 

No Break 1
 Break 1
 Value Color 

No Break 2
 Break 2
 Value Color 

Label Element #

OK Accept Cancel

Bin
Xxx

Bin

Left X Top Y Width Height

Top Height Spout Width Overhang Bands

Color 
 Transparent
 Fill Top

Color 
 Transparent
 Fill Background

Border Color  Thickness

Top Line Color  Thickness

Seams Color  Thickness Shrink

Top Horizontal Vertical
 Stagger
 No

OK Accept Cancel

Checkbox

Xxx

Checkbox

Left X Top Y

Width Height

Default Colors

Check Color

Background

Top Outside

Top Inside

Bottom In

Bottom Out

Corner

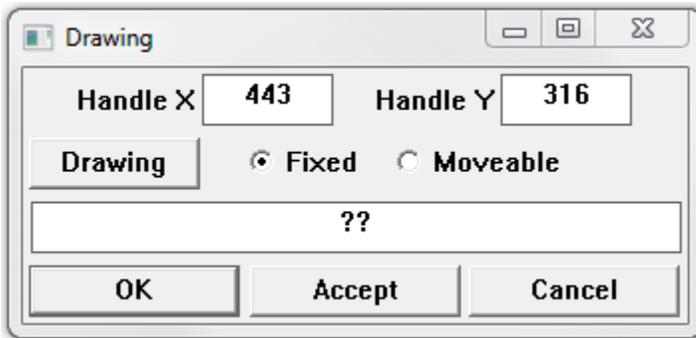
Corner placements are not implemented at this time.

Corner

Center X Center Y

Drawing

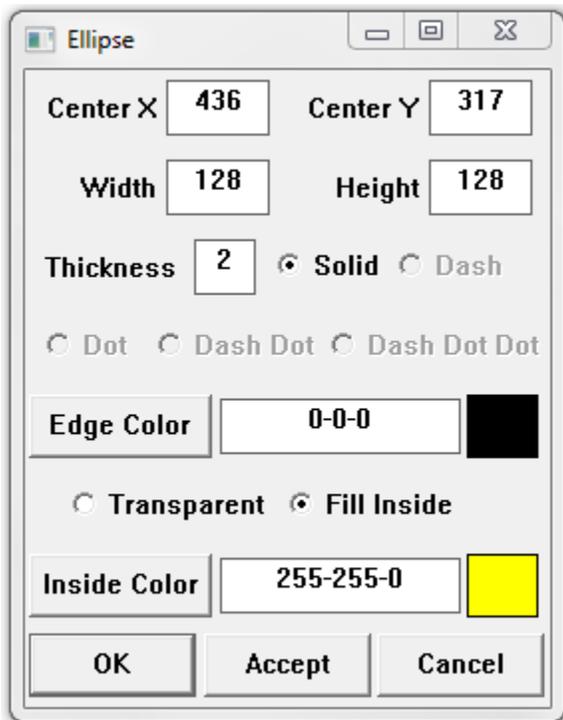
Drawing placements are used on screens to show Corsair drawings. Each placement needs to be linked to a drawing.



If the drawing is 'Fixed' it is shown on the same position on the screen as it was developed in the drawing window. In that case the handle point of the drawing and the handle point of the placement are not used. If it is 'moveable' the handle point of the drawing is placed on top of the handle point of the placement.

Ellipse

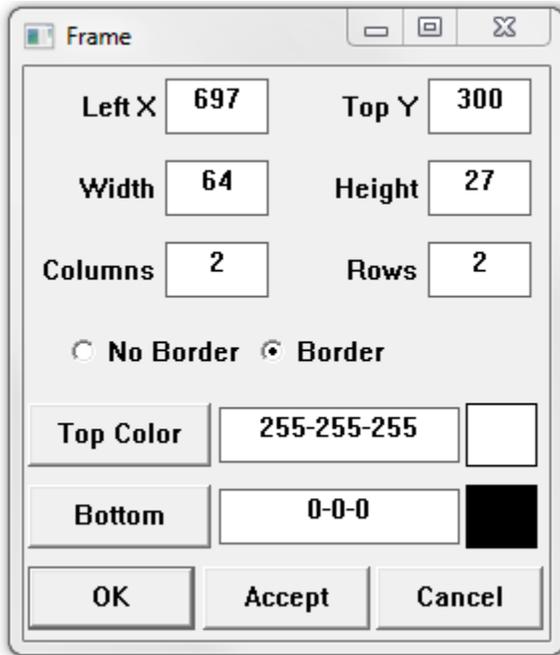
Ellipse placements can be used to draw circles if the height and width are the same.



The primary handle point of an ellipse placement is the center.

Frame

A Frame placement is used to mark divisions of other placements into rectangular frame 'cells'. The number of cells is defined by the frames row and column count values. A row count of 2 and a column count of 3 defines a 9-cell frame.



The primary handle point of a frame is the top left corner.

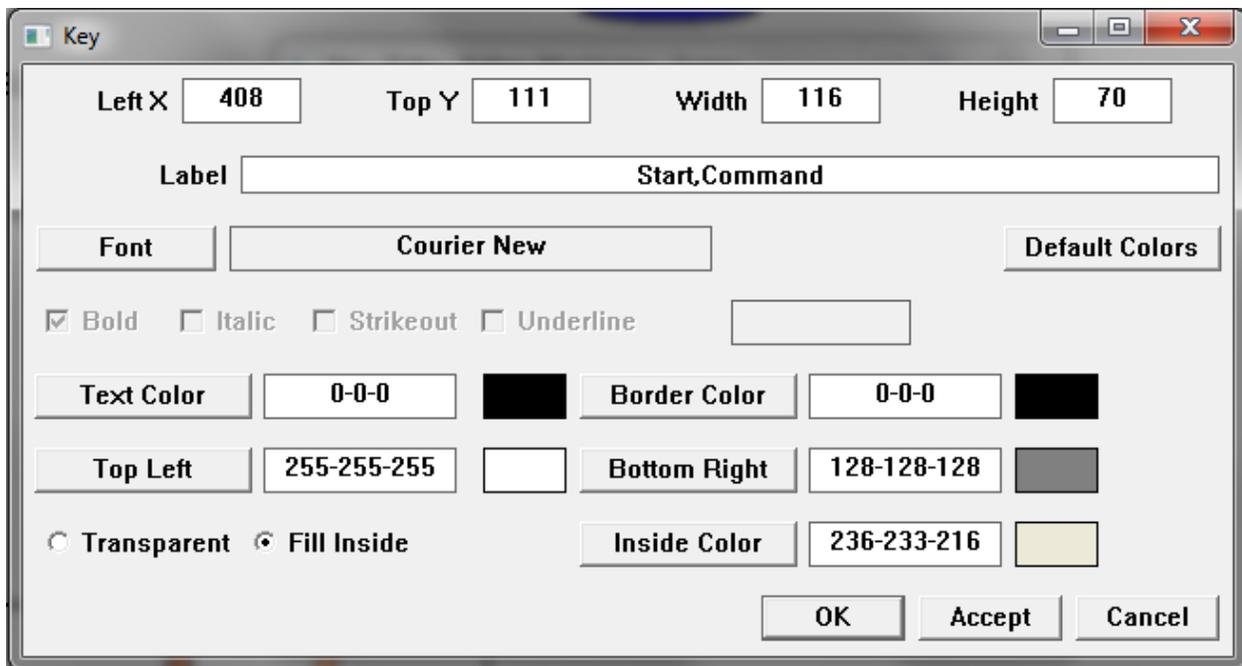
Icon

Xxx



Key

A Key placement is designed to look like a Windows button.



The label text can appear on multiple lines. The simplest method is comma separation. The developer types in

First Line,Second Line

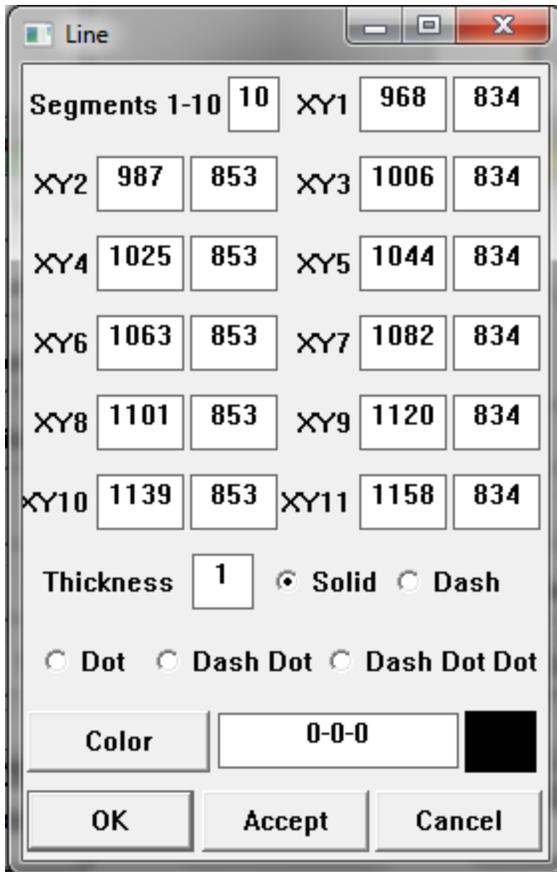
If a comma is to appear on the button the line can be marked with quotes.

“First Line, Continued”, “Second Line, Continued”

The Font button opens the font selector. The Default Colors button is used to initialize a suggested color combination.

Line

A line placement can consist of up to 10 segments. 10 segments have 11 handle points.

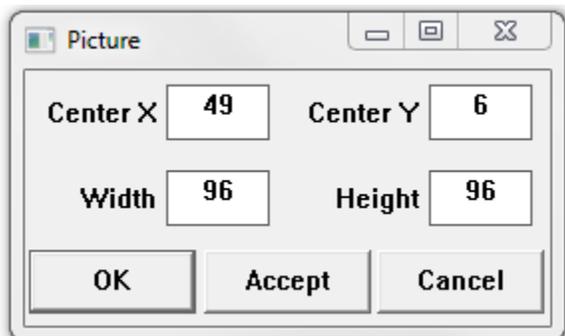


The X and Y positions are the handle points of the line. They may be adjusted here but it is usually easier to adjust them by rubber-banding with the mouse cursor.

A line with a thickness of 1 can have any of the style options from Sold to Dash Dot Dot. Thicker lines can only have the Solid style.

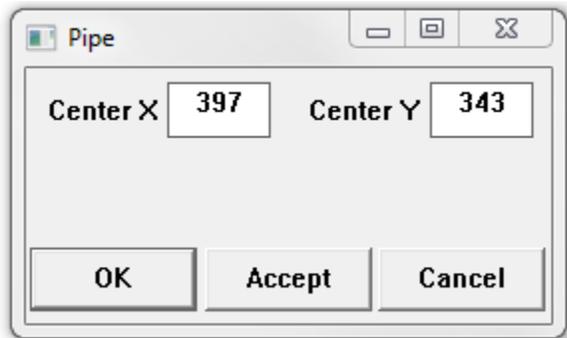
Picture

Picture placements are used to show a picture corresponding to the current value of a tag. The picture selection depends upon the type of the tag. Currently picture placements are only defined for HOA tags.



Pipe

Pipe placements are not implemented at this time.



Point Name

Xxx

Polygon

A polygon is a multi-sided placement that can be created in any position. It can have as many as 10 sides. A 10-sided polygon has 10 handle points.

Polygon

Sides 3-11 XY1

XY2 XY3

XY4 XY5

XY6 XY7

XY8 XY9

XY10 XY11

Thickness Solid Dash

Dot Dash Dot Dash Dot Dot

Line Color 

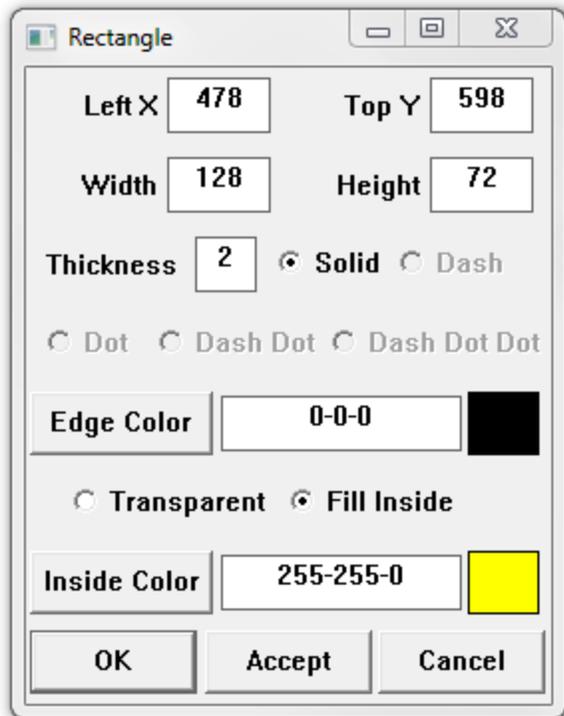
Transparent Fill Inside

Inside Color 

OK Accept Cancel

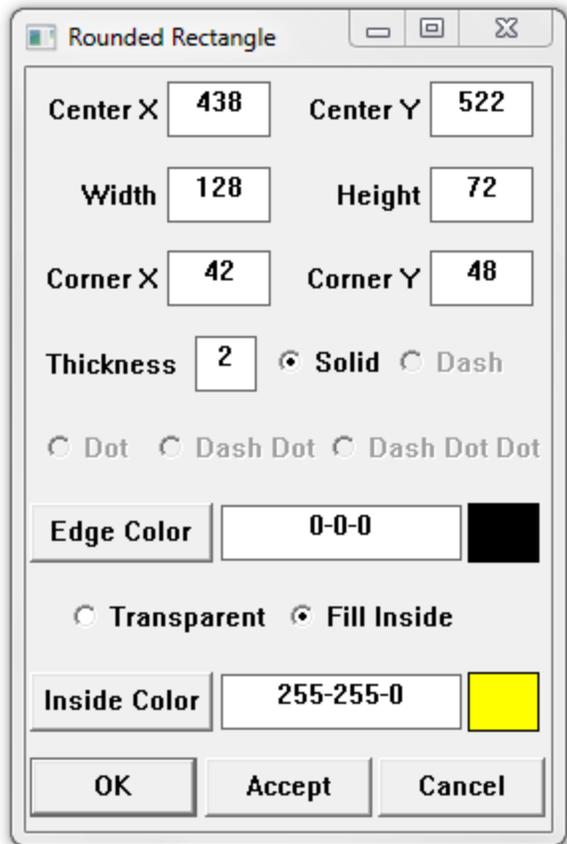
Rectangle

A rectangle is a 4-sided placement that is aligned straight up and down on the computer screen.



Rounded Rectangle

Xxx



The primary handle point of a rounded rectangle is the center.

Spline

Xxx

Spline

Start X Start Y

Ctrl X1 Ctrl Y1

Ctrl X2 Ctrl Y2

End X End Y

Thickness Solid Dash

Dot Dash Dot Dash Dot Dot

Color 

OK Accept Cancel

Tank

Xxx

Tank

Left X Top Y Width Height

Ring Thickness Ellipse Height Frame Thickness

Vertical Horizontal Left End Right End No Border Placard Border

Placard Width Height Color 

Liquid Color  Ring Color 

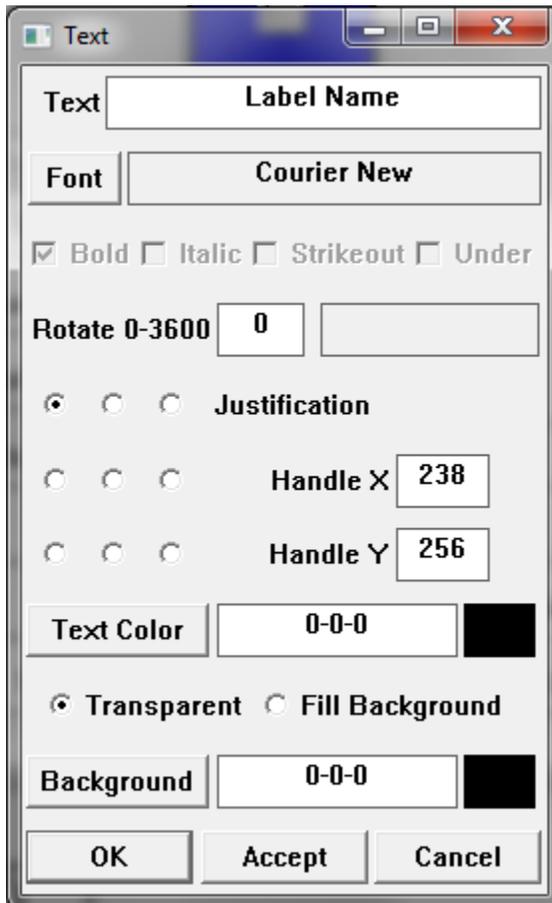
Frame Color  Background 

Minimum Maximum

OK Accept Cancel

Text

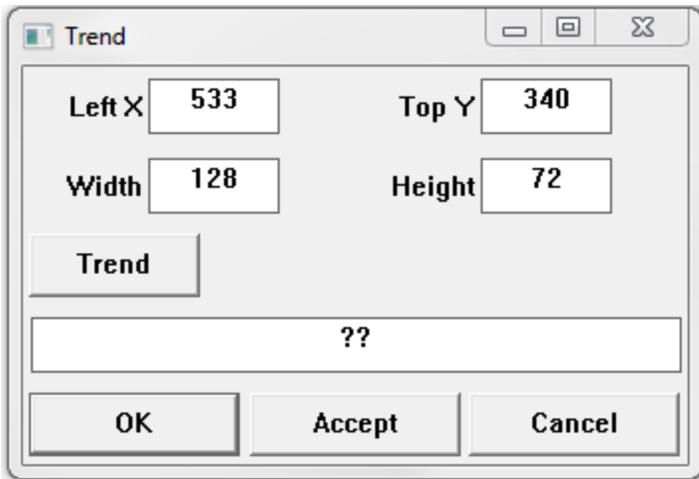
A text placement shows an unchanging text label on the screen.



The nine justification radio buttons control how the text label is arranged around the handle point. If the upper left button is selected the text is shown to the right and below the point.

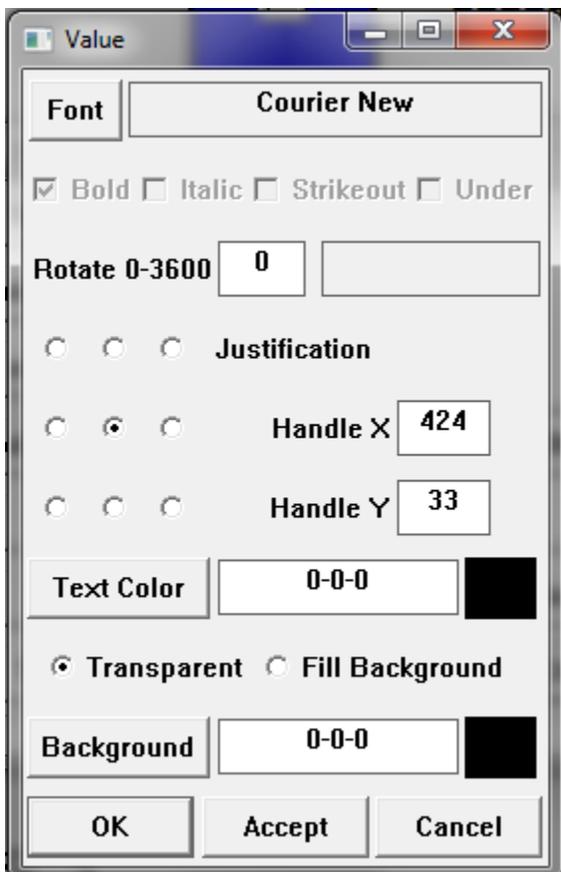
Trend

A trend placement specifies a rectangular region for the computer to display a data trend.



Value

A value placement shows varying text on a screen. The text shows the value of a tag.



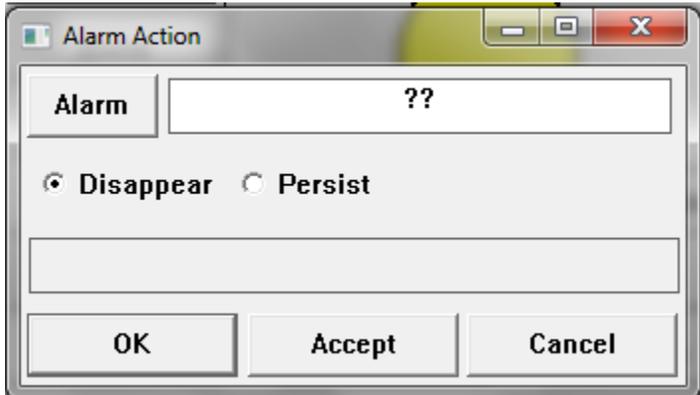
The nine justification radio buttons show how the value text is arranged around the handle point. If the upper left button is selected the text is shown to the right and below the point.

Placement Actions

A placement's action determines what happens when the operator clicks on it with a mouse or touches it on a touchscreen. The action may also determine when the placement is visible and when it blinks.

Alarm Action

The Alarm action must be linked to an alarm.



The status of the alarm controls the placement's visibility and blink properties. If the alarm is clear the placement is invisible. If the alarm is unacknowledged the placement blinks.

Call Action

The Call action works just like the alarm action except that it must be linked to a call.

Display Action

The display action is the simplest action. The placement is shown if the visibility conditions permit. It cannot be clicked on by the operator.

Entry Action

xxxx

I/O Jump Action

xxxx

Log Jump Action

xxxx

Source Comms Action

xxxx

Quick Trend Action

xxxx

Screen Jump Action

The Screen Jump action is used to create a placement that the operator can click on to change screens.

Sheet Jump Action

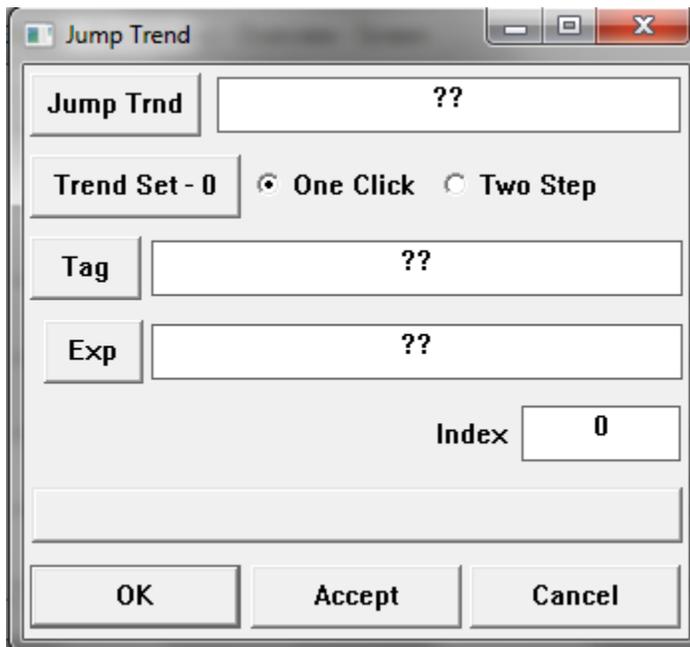
The Sheet Jump action is used to create a placement that the operator can click on to open a sheet.

Task Action

The Task action specifies a task that the computer will perform when the operator clicks on the placement. There are several options for the task. Each one is described in the Designer manual. Some tasks require specifying a tag. Some require a script. Some require a constant value.

Trend Jump Action

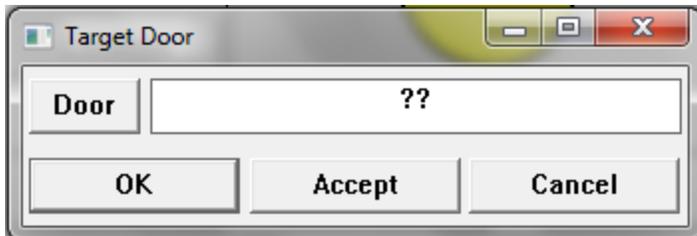
The Trend Jump action causes the computer to open a trend viewing window when the operator clicks on the placement. The developer must specify the jump destination.



In simple situations the 'Jump Trend' is the object of the jump. It is possible to select the object from a set of trends using the value of a Corsair triplet.

Door Action

This action links a door to the placement. It is only available on corrections versions of the program.



Clicking on the placement 'hooks' the window to the placement. At that time the placement's hook code values are transmitted. The operator can see his F1-F4 keystroke options on the status bar on the bottom of the screen.

When this action is used for a placement like an ellipse a color list can be specified. If no tag is tied to the list the color selection comes from the status of the door. When it is unsecure the 0 (zero) colors are used. When it is secure the 1 (one) colors are used. The same thing is true for drawing and icon lists. In each case if the list is indexed by a tag the value of that tag is used instead of the door status. This makes it possible to have more than two items in the list.

Door Command Action

This action enables the operator to execute a command for a door. It is only available on corrections versions of the program.



The developer must specify a door and which command to execute. The available commands emulate the F1, F2, F3, F4, or 'E' key.

Developing User Sheets

The 'Edit"More" Sheets' menu option opens the development sheet that creates user sheets. After a sheet record is created a number of rows and columns must be entered. Zooming on the sheet record opens an operator view of the sheet. If development is turned on the developer can start to define the contents of the sheet's cells.

The first set is to put a label on top of each column. Arrow to a cell on the column and press the 'L' key.

Discrete Input Conditioning - Column 1 Width

Text: Point

Tag: ??

Exp: ?? Index: 0

Visibility: ??

Invert Exp: ?? Index: 0

Blink: ??

Invert Exp: ?? Index: 0

OK Accept Cancel

Labels can be static or dynamic. A static label is fixed text that does not change. A dynamic label comes from the value of a tag.

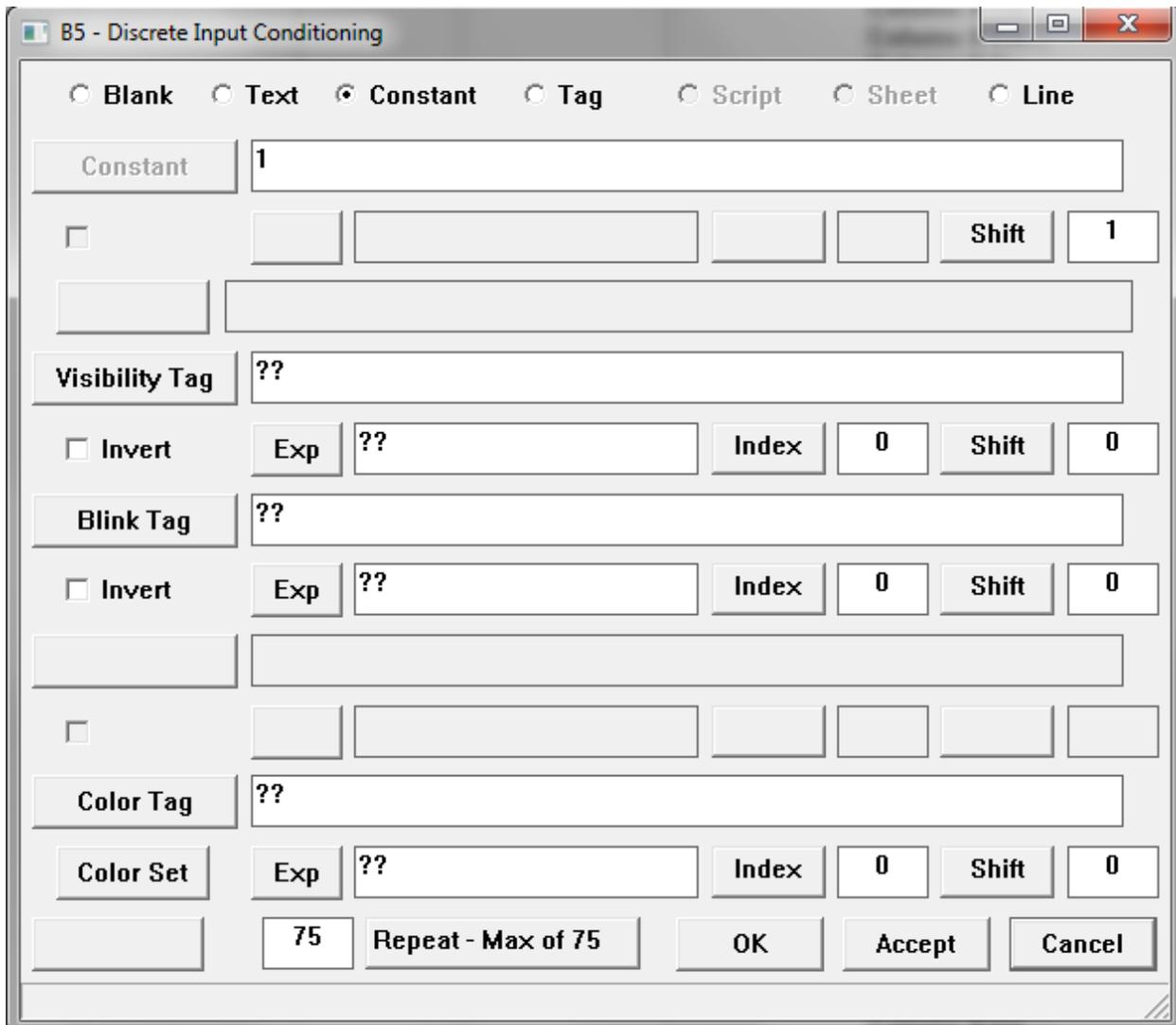
[Note – as of this writing labels can only be static]

Sheet cells are identified with designations that use a letter for the column and a number for the row. Starting at the top left of a sheet the cells are:

A1	B1	C1	D1
A2	B2	C2	D2
A3	B3	C3	D3
A4	B4	C4	D4

If the 'Development' menu option on a sheet is checked the cell designations are shown for all blank cells.

The contents of a cell are edited by arrowing to the cell and pressing 'F7' or 'Z' to zoom.



The definition of a cell includes a type. The radio buttons on the top of the window set this type. It also includes a place near the bottom to enter a repeat count number. If the repeat count is zero the definition only applies to that cell. If it is nonzero the definition also applies to that many cells immediately below it. A definition with a repeat count of 4 applies to a total of 5 cells.

A 'blank' cell has no contents. Blank cells cannot have a repeat count.

A 'text' cell requires the developer to enter a text label. This label is what will appear in the cell. If there is a nonzero repeat count each of the repeat cells will show the same label.

A 'constant' cell requires the user to enter a number. That number will appear in the cell. If there is a nonzero repeat count numbers will appear in the repeat cells. Each cell uses a number that is equal to the cell above it with the shift value added. If the constant is 2, the repeat count is 4, and the shift is 3 the 5 cells from top to bottom will read 2, 5, 8, 11, and 14.

A 'tag' cell is used to show the value of a tag.

Insert more stuff here

A 'Line' cell contains a horizontal line. Line cells are used to divide sections of the sheet.

Creating I/O Modules

Xxxxxxx

Developing with Corsair Blocks

Note – custom program blocks are not implemented at this time!

The Corsair program uses program blocks to perform calculations on tag data. A block is defined in one place. Each time that the block is used is an 'instance' of that block. A change in the definition of a block applies to all instances of that block.

Blocks come in two types, standard and custom. A standard block comes with the Corsair program. Standard blocks cannot be changed by the Corsair developer. Custom blocks are blocks that can be defined and changed by the developer.

Standard blocks must be installed into Corsair application before they can be used. When the development level is set to 'administrator' the Tools/Expert/Block menu options leads to a window with two lists. The left-side list has the currently installed standard blocks. The right-side list has the currently configured custom blocks. Standards are installed with the Standard/Install menu option. The standard block list must be empty before this installation can occur. This can be done from the 'Standard/clear and Install' menu option. The CAP Corsair application file must be saved after standard blocks are installed.

When a new version of the Corsair program is released the database checking utility will check to see if the installed standard blocks match the new version. If they do not match a system error will prompt the developer to clear and install the standard blocks again. This does not require any other changes to the users application file.

Custom blocks are created using normal sheet-style editing of the custom block database. This is done with the Edit/More/Custom Blocks menu option. Pressing the 'F4' (create key) inserts a new block record into the database. There are 5 zoom fields associated with each custom block. They are used for

block variable definition, block parameter definition, block initialization code, block forward code, and block reverse code.

The block variable definition window allows the developer to enter a purpose for the block. Each block can have up to 10 internal variables named 'VA' through 'VJ'. These variables are local to each instance of the block. They are not global. Each time a block instance is created it has its own copy of these variables that no other block can access. Variable definition includes a size for variable. If it is assigned a size of ten it is an array with ten values using indexes from 0 through 9. Each variable also has a type, which are the same as those used by Corsair tags. A data format is used for some tags and a string length is used for some tags.

There are a maximum number of bytes of data that can be used for block variables. It is shown on the bottom of the block variable definition window along with how many total bytes the configuration uses. If the total exceeds the maximum the variable configuration must be changed.

The block parameter definition window allows the developer to define parameters that are used by the block. The 'main' parameter is the tag that the block is assigned to. It can get a label. The 10 parameters that are assigned with a block instance are called 'PA' to 'PJ'. A descriptive label can be put on each parameter. Each parameter also needs a type and sometimes a format.

Sometimes a block may need more data space than what is available with internal block variables 'VA' through 'VJ'. Tags can be sized as very large arrays and then assigned as block parameters to solve this problem. A different tag may need to be created for each instance of the block. Use of the block variables whenever possible simplifies the development task and avoids unintended data collisions.

The code for each block is divided into 3 different sections, Initialization, Forward and Reverse. Initialization code is executed once a short time delay after the Corsair program starts interface operation. After the initialization code has been executed for each block the computer continuously executes the Forward code. The Reverse code is executed once each time that the operator enters a value into a tag that has a block assigned to it.

Most standard blocks do not have any code that the developer can see. Their code is contained in the Corsair program in a compiled form.

The window that is displayed from the Tools/Expert/Block menu option allows the developer to zoom in on any of the 5 zoom fields for any standard or custom block. Configuration of standard blocks cannot be changed. The Standards/Clone to New Custom menu option on this window is used to clone a standard block to a new custom block. It can then be modified just like any other custom block.

Standard or custom blocks can be assigned to a tag, alarm, or call tags. The tag must be a memory tag – it cannot be resident on a PLC or on a device(dave read this line). Memory tags can be displayed by picking the Edit/Data/Alarms followed by View/Local shows alarms that can be the result of block calculations. The same thing is true for calls.

Let's assume that a PLC has a 5 temperature values that are in Centigrade degrees. The interface needs to display them in Fahrenheit. The 'MC Plus B' standard block was designed for this purpose. The first thing to do is install standard blocks into the database. The Tools/Expert/Blocks window should show that the standard has been installed. The next step is to create a communication Driver record that is the proper type for the PLC. This is done under Edit/Data/Drivers. Creating the driver gives the developer a zoom option for PLC's. Zooming into this allow creation of 5 tags. These tags represent the raw vales in Centigrade degrees. Each of these tags should get a PLC address that is usable by the driver to read in the data.

The 'MX Plus B' block is not installed on the PLC raw value tags. 5 more tags must be created for the scaled Fahrenheit values. This is done by going to Edit/Data/Devices and creating the tags. These tags are computer-memory and not PLC resident. Near the right side of the development sheet there is a field named 'Block ID' The tag 'MX Plus B' should be typed in here. If this name exactly matches the name of either an installed standard or custom block an instance of the block is created. The two zoom fields to the right of the Block ID field pertain to this instance of the block. They are for block parameter assignment and block variable monitoring.

The Block Parameter Assignment window is used to define the tags that one instance of the block uses. Labels for the 10 parameters are shown. Each parameter can be assigned either a constant or a tag. In this example the PA parameter corresponds to the input to the block. It gets the tag of the raw centigrade value. The PB parameter sets a constant value of 1.8. The PC parameter gets a constant value of 32.

The block parameter assignment window has a 'Details' button that lets the programmer see an unchangeable summary of the configuration of the block parameters.

The block variables monitor window allows the developer to see current values of the variables inside a block instance. The top of the window has 10 radio buttons for block variables VA through VJ. Variables that have been defined to have a size of zero in the block configuration are grayed-out. Selecting a radio button causes a listing of the values of the indexes of the variable to appear. The type, data format, and string length of the variable are also shown.

The block variables monitor window has a 'Details' button that lets the programmer see an unchangeable summary of the configuration of the block variables.

The Corsair Select and Replace System

Corsair includes a system that the developer can do to speed up many of the tasks that are involved with making a model file. The Select and Replace system is a way of manipulating the ID text on database records. It is especially valuable when there are similar elements in the database, like a facility with several identical buildings. If the developer understands how to use Select and Replace and does the proper planning, he has very little work to do after he has finished the first building. The key is to make sure that the labels used for record ID are compatible with Select and Replace.

The system usually starts with the developer entering in 3 pairs of Select and Replace entries. Each pair is called a 'rule'.

A Select entry describes a pattern for a label. When a label is compared to a valid Select entry the computer can decide if the label matches that entry. A data source may have 100 tags on it. 73 of them might match the Select entry. The remaining 27 wouldn't match it.

A Replace entry describes a pattern to change a label. For each tag that matches the Select entry of a rule the computer will change the tag according to the pattern of the Replace entry. A complete operation usually involves 3 rules.

Rule 1	Rule 2	Rule 3
Select 1	Select 2	Select 3
Replace 1	Replace 2	Replace 3

For each label that is in the specified set the computer first goes to Rule 1. It checks for a valid Select 1 entry. If the Select 1 entry is invalid it goes on to Rule 2 and checks for a valid Select 2 entry.

If the Select 1 entry is valid the computer checks to see if the label is a match for Select 1. If it is not a match it goes on to check the validity of the Select 2 entry.

If the Select 1 entry is a match the computer checks to see if the Replace 1 entry is valid. If it is not valid that label is done and left unchanged. If the Replace 1 entry is valid the computer does the replacement specified by the rule and that label is done.

Most of the windows that use the Select and Replace system show the developer a 'Can' and 'Can't' count before he activates the rules. This allows him some opportunity to evaluate if the rules will do what he wants them to do.

All Select and Replace rule entries consist of at least one section. An example Select is the string 'abc'. This Select has one section containing 3 characters. It is an example of a 'literal' section. The label must match the literal section. Corsair uses case-insensitive selection. That means the following labels will match this Select.

'abc' 'Abc' 'ABC'

The system uses 6 special characters in the Select and Replace entries. They are:

Vertical Bar	
Asterisk	*
Question Mark	?
Number Sign	#

Caret ^

Tilde ~

It is possible to use these characters in Corsair ID labels but the practice is strongly discouraged.

The first special character is the bar. It is used as a section divider. Every entry has at least one section. The Select 'abc' is equivalent to the Select 'a|b|c'. The first select has one section with three characters. The second select has three sections which each have one character.

If the Asterisk character is the first character in a section it serves as a 'wild-card' that represents a group of characters. The group could be empty or it could be quite long. The Select 'A|*' matches any label that begins with the letter 'A'. A single 'A' by itself matches this Select.

Rules match up sections between the Select and Replace entries to see how the replace action is going to work. A possible rule could be:

Select 'A|*' Replace 'B|*'

This rule would select any label beginning with the letter 'A' whether it had anything else after it or not. It would change the leading 'A' to a 'B'.

The Question Mark is used as a single-character wildcard. It matches exactly one character. A rule could be:

Select 'A|?' Replace 'B|?'

This rule selects any label beginning with the letter 'A' that is exactly two characters long. It changes the leading 'A' to a 'B'.

This is another possibility:

Select 'AB|*|?' Replace 'C|*|DE'

This rule selects any label that begins with 'AB' and has at least one more trailing letter. The leading 'AB' is replaced with 'C'. Any characters in the middle are unchanged. The trailing single character is replaced with 'DE'.

The Asterisk and Question Mark characters only count as wild cards if they are the only item in the section. The Select 'AB*|?E' has no wild cards. It will only match the string 'AB*?E'. A special case occurs when the developer wants to have a literal section consisting of only an Asterisk or Question Mark. This is signified by a repeated character.

Select 'AB|**|CD|?'

This select matches labels beginning with 'AB*CD' followed by exactly one more character.

The L, U, M, and I can be in upper or lower case. Upper is the preferred syntax.

For the Lower, Upper, or Mixed case commands only the Caret and the letter appear in the section. The computer does the desired case correction on that section.

An insert command section consists of the Caret, the I, and some literal text. The text is inserted between two other sections. This is an exception to the rule that the number of sections in the Select entry must equal the number of sections in the Replace entry. Insert sections do not have corresponding sections in the Select entry.

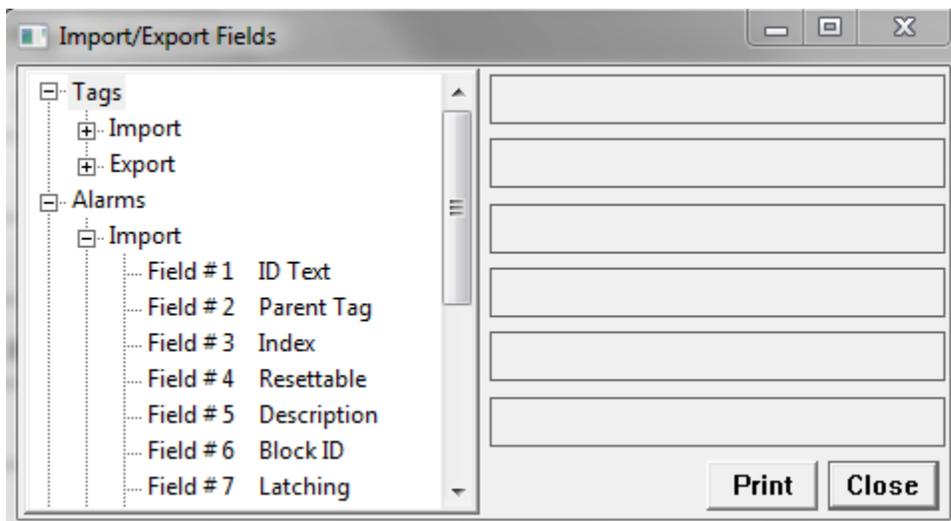
Numbers may be inserted as text like in the section '^I2'.

The Tilde character is reserved for future use.

Database Record Import and Export

A Corsair developer may need to create a large quantity of tags, alarms, calls, or doors. It may be more efficient to create these database records outside of Corsair using a spreadsheet program. The data can be saved in a CSV (comma separated variable) formatted file. The file data can then be imported into Corsair. The opposite operation is also possible where data is exported from Corsair to a CSV file.

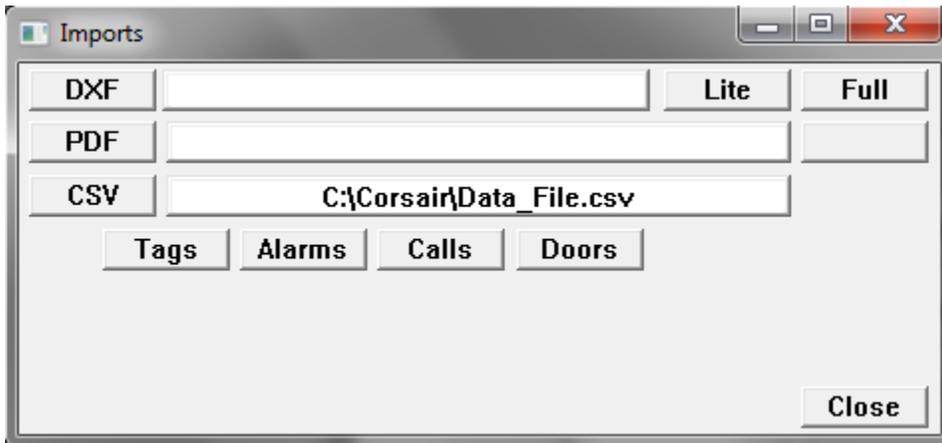
The format of the file will vary based upon the record type. A file of tags has different column types than a file of doors. The developer can see the format for the files from the 'Help' Program Information 'CSV Fields' menu option. This opens the Import/Export Fields window.



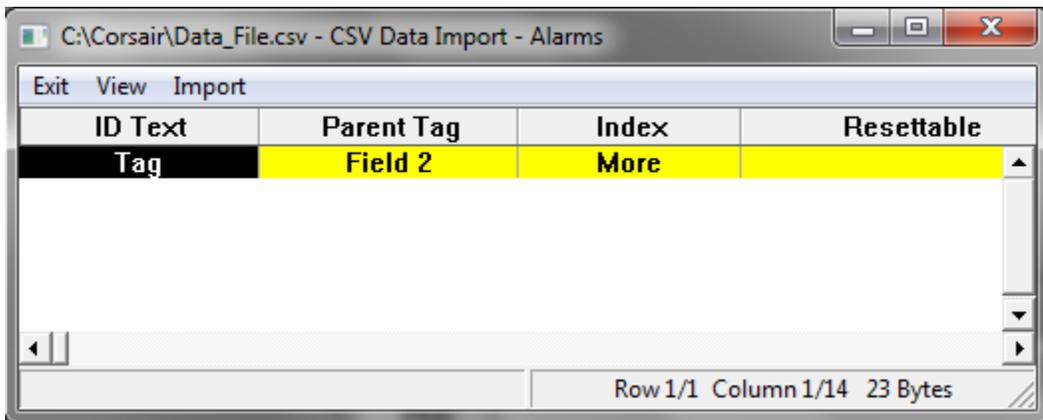
Every field of each type of file is described by a tree view here. The tree can be printed for paper documentation.

Corsair reserves the right to change the file formats as new features appear in the program. A data file produced for one version of the program may be incorrect for another version. In each case the Import/Export Fields window will show what is correct for that version of the program.

Importing a data file involves the 'Edit''Model Data''Imports' menu path.



The CSV button on the left side opens a file name picker control that can be used to put the file name in the edit control on the center of the window. At that time the buttons for Tags, Alarms, Calls, and Doors will appear. The developer must select the button that is correct for the data in the file. It opens the CSV data import window.



This window allows the developer to see the data from the file before he elects to create the records. Fields that have an error are shown in color to warn him of potential problems with the file. Ideally all of these errors are resolved before the import happens. Often the answer is to edit the file in the spreadsheet software and save the results. Closing and reopening the CSV Data Import window will cause it to try the new version of the file.

Some potential import errors are avoided by changing data in the Corsair model database. The developer may be trying to import a tag named 'Tank Level' when Corsair already has a tag by that name. He cannot edit the file data from the window but he can leave it open while he changes the name of the existing tag to 'Old Tank Level'. The 'View''Recalculate Errors' menu option will cause Corsair to recheck for the name collision.

When the developer wants to import the record data he picks the 'Import' menu option. If there is no problem the CSV Data Import window will close automatically. Problems will result in an error message.

Corsair's default behavior is to not do the import if any problem exists with the file data. Advanced developers can override this and import from imperfect data. This is done by adjusting two menu options that are under the 'Import' menu item.

The first option is the 'Warnings' option. It defaults to 'Abort Import' where nothing is imported if any kind of warning exists in the data. The 'Discard Line' option causes Corsair to not import from any line that has a warning on any of its fields. The 'Ignore' option instructs Corsair to accept every line and do the best that it can with the imperfect data.

One type of warning is when there is an item (Tag, Alarm, Call, or Door) already in the Corsair model database that has an ID name identical to a line of the import file. Special control options apply to these 'Duplicate' warnings. The 'Abort Import' default behavior is to not import anything if a duplicate exists anywhere in the file. The 'Discard Line' option causes Corsair to not import from any line that has a duplicate. The 'Replace' option causes Corsair to import the new record, change all links from the old record to the new one, and then delete the old record. The 'Append' option causes Corsair to import the new record. The developer will then have to manually resolve the name collision.

Exporting data to a file involves the 'Edit' > 'Model Data' > 'Exports' menu path. Exported files will contain more fields than what are required for import.

CSV Data Import

Data is frequently kept in disk text files in a CSV – comma separated variable – format. The file consists of a number of 'rows' which may also be called 'records'. Each row has one or more columns which are also called 'fields'. Typically each item is surrounded by double quote marks and the columns are separated by commas. An example of the contents of a CSV file is:

```
"R1C1","R1C2","R1C3",
```

```
"R2C1","R2C2","R2C3",
```

```
"R3C1","R3C2","R3C3",
```

CSV data import is a tool within the Corsair program that is used to import data from a CSV file into Corsair tags. The tags can be memory tags in the Corsair database or they can be located on a data source like a PLC. The tool can do import for as many as three tags at one time. The operation can be performed multiple times on different columns of the same CSV file to load many tags.

The first step before doing CSV import is to verify that the developer has administrative privileges. The status bar on the main Corsair windows should show 'Dev: Admin' on the lower right corner. If it does not, the level can be changed through the Users/ Change Level menu option.

The next step before data import is creation of all the necessary tags and locating them on the correct data sources with correct addresses. Most types of data sources can be used but frequently these tags will be created on a Modbus Memory Map data source. The imported data for a memory map is packed into a binary file where Corsair can access it far more efficiently than it can access a CSV file.

After the tags are created the developer must run the Corsair interface. This means that the interface check box is checked so that Corsair can perform whatever communications tasks are required by the import.

Now the developer must view the contents of the CSV file using the Corsair CSV file viewer. Pick View/ Files/ CSV File Viewer from the menu. A dialog box appears that allows selection of a file. When the file is open the developer can see the data items that are in it and how many rows and columns are present. One data item is shown in inverse (reverse) colors. This item is in the 'current' row and column position. The arrow keys can be changed to the current position. Typically the developer will arrow to the first item that he wants to import. He must not have the View/ Auto Refresh menu option checked.

The Edit/Import menu option is used to open the CSV data import window. This window has group boxes containing controls for the import of data into 3 different tags. Below them on the lower left is a button that is labeled 'Import' if an import is ready. If it is not ready it shows an error number that you can click on for an explanation of the error. The current row and column number of the CSV Viewer and the row and column totals are shown in boxes along the bottom of the CSV Import window.

Data Import Tag Controls

The first step to set up a tag for import is to check the 'Tag' check box. Next the 'Find' key is used to select the name of a tag into the next box. This box starts out with two question marks. It changes to the tag name as soon as one is found. The button to the right of the tag name is blank unless there is an error in the import setup of the tag. If this is the case an error number is shown and clicking on the button brings up an explanation of the error.

The 'Current' checkbox is only offered on the first tag of the three. It is checked if the data import of the tag is to begin at the current row and column position on the CSV file viewer. The row and column boxes for the tag cannot be entered if this option is checked. Unchecking it allows the row and column to be entered like with the other 3 tags.

The 'Index' number is the index of the first element of the tag that is to receive import data. If the tag has an array size of 20 valid index values can range from 0 to 19. Most imports will use zero values for the Index entry.

The 'Count' number is how many consecutive indexes of the tag will be imported starting at the entered row and column position and working down the column.

Assume the following entries:

Row 2 Colum 3 Index 4 Count 3

Corsair will do the following import:

Row 2 Column 3 to index 4 of the tag

Row 3 Column 3 to index 5 of the tag

Row 4 Column 3 to index 6 of the tag

Corsair will check all row, column, index, and count values for validity before an import can begin with errors showing on the buttons on the CSV Import window.

Data Import Type Conversion

The raw data coming in from the CSV file has a data type of string. This is converted to the proper type and format for the tag. In some cases this conversion may not yield the proper results. The 'Input Type' check box on each tag group allows the developer to select a tag type and format for an intermediate result. When it is used the data is converted from the string that is in the CSV file to the type and format of the intermediate result and then converted to the type that is required by the tag.

CSV Data Import for the GPS Markers Block

Data import is frequently needed for the GPS Markers Block. The CSV file data may be in three columns – latitude, longitude, and a name for each marker. This data is imported into three tags that are located on a Modbus Memory Map file. If there are 100 markers the latitude and longitude tags should have a size of at least 100 and the name string tag should have a size of at least 101. The Latitude and Longitude tags should be 64-bit integers with Latitude and Longitude formats. The Marker names tag should be a string with a length greater than or equal to the longest string to be imported. Each of the tags should have a proper Modbus address and the Memory Map data source should be set to 'Real? Yes' before the import is attempted. Since imports must be performed with the Interface running Corsair will write the data to the memory map file.

It must be noted that the GPS marker name tag uses index 0 for the marker name if no valid marker is found. Index 1 of the name tag corresponds to the first latitude and longitude position. Because of this the Index entry for the import of the latitude and longitude tags should be zero and one for the marker names tag.

The latitude and longitude may be in decimal degree format in the CSV file. If this is the case the Input Type option needs to be checked with a suggested type of Double Float and a format of -3.7. Corsair will convert the string from the CSV file to a Double Float and then to its internal latitude and longitude format.

CSV Data Import for the GPS Areas Block

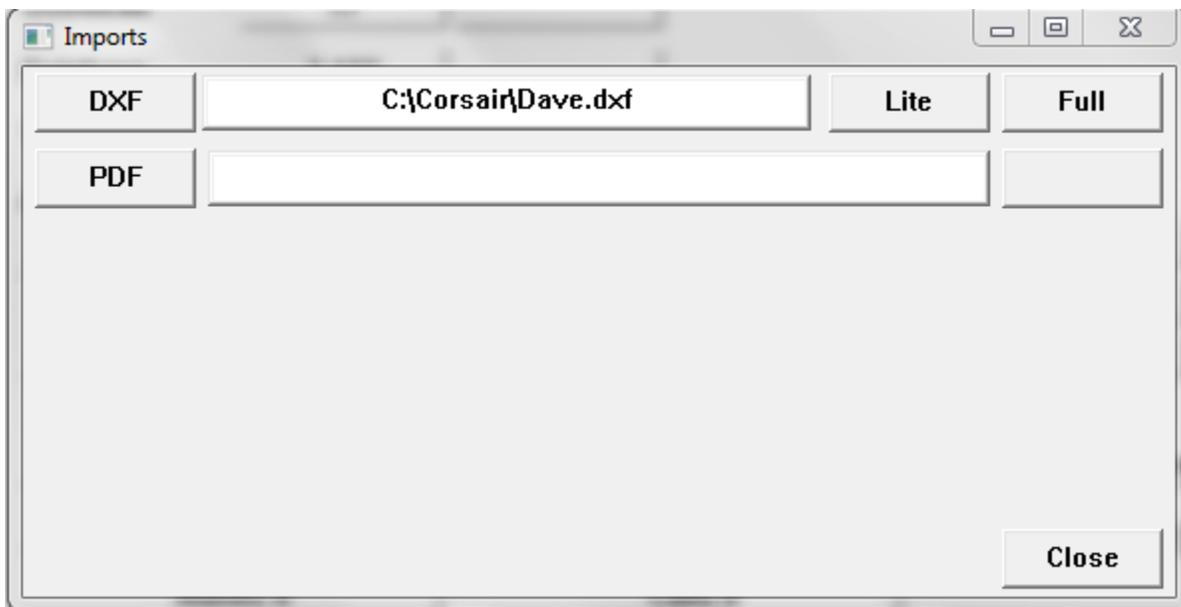
The data import requirement for the GPS Areas block differs from the GPS Markers block since the latitude and longitude tags are grouped into 4 indexes for each area. If there are 100 areas the latitude

and longitude tags would have a size and an import count of 400 starting at index 0. The string tag would have a size of 101 and an import count of 100 starting at index 1.

DXF Import

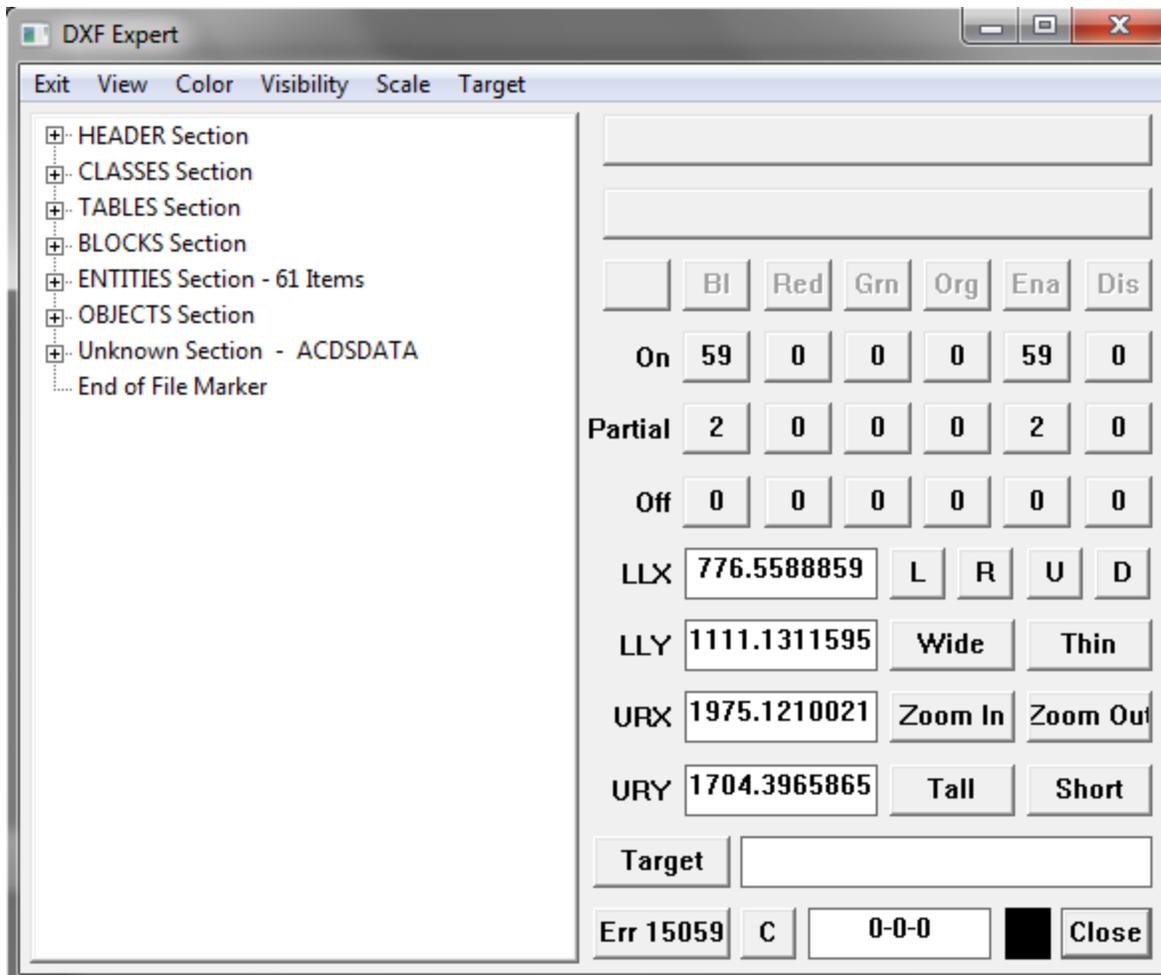
DXF files are generated by computer drafting programs. The Corsair program has a limited ability to read .DXF files and translate them to drawing placements. The developer imports the DXF into a DXF tree format. He can then isolate sections of the drawing to create drawing placements. The tree data can be saved in a separate file to avoid having to import a second time at a later date. DXF import works best on a computer with more than one monitor.

DXF import starts at the import window.



The DXF button is used to browse for DXF files to be put in the edit box. If a file name is entered the 'Lite' and 'Full' buttons will open the file for import. The 'Lite' button opens the DXF expert without loading the tree control. The 'Full' button opens the DXF expert using the tree control. DXF import can take a lot of time with several minutes possible for large files. For an initial import of a large file the Lite option is recommended. The data can then be saved as DXF tree data.

If the developer already has a DXF tree data file he can click on 'Lite' or 'Full' with no name in the edit box. After he gets to the DXF expert window he can then load the tree file.



The DXF expert window consists of a tree control on the left side and other controls on the right side. The tree control will be empty if the window was opened with the 'Lite' option.

Every DXF file consists of a number of elements called 'Entities'. They are listed in the Entities section of the tree control. Corsair cannot work with all the possible types of DXF entities. An entity that Corsair cannot use is an 'Undrawable' entity. Each drawable entity can be 'Enabled' or 'Disabled'. Enabled entities are used to create Corsair drawing placements. Disabled entities are kept in the tree data but are not used for placements.

The developer can preview imported DXF data using the 'View' menu options. If he picks 'View' All' he can see all of the drawable entities that fit onto the preview window. Four numbers are used to scale the preview window. They are LLX (Lower Left X), LLY (Lower Left Y), URX (Upper Right X), and URY (Upper Right Y). Based upon these 4 values entities can be On-Screen and seen in the preview window, Off-Screen and not seen, or Partial and partially seen. The DXF expert window shows totals for each type.

On the preview window disabled entities will show in a light gray. Other entities will show in one of the 4 import grouping colors. These are four colors that can be used by the developer to mark different

groups of entities. They are Black, Red, Green, and Orange. These colors are not used for Corsair drawing placements that are created from the DXF expert. They are only used to segregate entities as the developer works with the DXF tree to create drawings. The color that is normally used for created placements is shown on the bottom right side of the expert window. The 'C' button is used to open a color selector to aid the developer in finding the color that he wants.

While the developer is looking at the preview window he can use the L(ef), R(igh), U(p), and D(own) buttons to change the scaling numbers that define what he sees on the window. The Zoom In and Out buttons perform those functions. Wide, Thin, Tall, and Short should be used with some more caution as they are more likely to distort the horizontal and vertical aspect ratio of the drawing entities. With most Corsair systems the drawings are not to scale and the developer is free to adjust them for the best appearance.

The Preview window comes in 6 different versions. The Result version shows what the proposed Corsair drawing will look like. It does not show disabled elements. The other versions of the preview window show disabled elements in light gray. The developer can preview all of the grouping colors or only one of them as desired.

The Target menu item controls how the drawing placements are created. The developer can select using all entities, or just the entities from one of the color groups. In every case undrawable and disabled entities are not used. The expert can create a new Corsair drawing. He can also replace the current contents of an existing drawing or append to the contents of that existing drawing. The line color for the imported placements will be the color that is entered at the bottom of the window or (Future) color information from the DXF file.

The Target button is used to open a Corsair selector to enter an existing drawing for a replace or append operation. The button on the bottom below the target button is used to perform the operation. The View/Target menu option lets the operator see the drawing before and after the placements are created. The View/Drawings menu option opens editing of drawing records. The View/File Text menu option opens the original PDF file to allow the operator to explore it's contents.

The develop can select an entity from the tree control by clicking on it. He can then eliminate (delete) it, set its grouping color, and enable or disable it from the buttons on the right side. The selected entity will flash on the preview screen. Another option is to flash all entities on the preview screen that are on the same DXF layer as the selected element.

Many menu options are available to set options on multiple entities at the same time. Corsair does not allow drawing placements to be created if any of them are off-screen. The 'Visibility''Disable''Off-Screen' menu option can take care of these situations.

The 'Scale' menu has options to change the scaling numbers to fit groups of entities to the Corsair screen. When a drawing is first loaded it is scaled using the drawing extents values from the DXF file. A menu option can return the scalings to these values. There are orientation options for the imported drawing.

If a developer is looking at a portion of a drawing with a desired scaling and orientation he can mark that position with the 'Remember As' options. The values will be returned with the 'Go Back to' options.

The 'Tiny Moves' menu item is to switch the image moving and zoom buttons to a much finer resolution for precise positioning of the imported image.

Corsair Micro-CAPs

A CAP file is a CorsairHMI application database. It may contain several inter-linked databases of drivers, screens, tags, and so on. Most CAP files contain sufficient information that they provide a complete database for the Corsair program to run a customer's application. Micro-CAPS are also CorsairHMI application database files. They use the exact same file format and file extension as any other CorsairHMI CAP file. They may be complete runnable applications but generally they contain far less information. A Micro-CAP may contain only a single screen and its placements, or just a drawing, or just a PLC with a number of tags on it. The definition of a Micro-CAP is that it is a CAP that was 'extracted' from another CAP. Extraction is the process of removing database records to create a smaller file that contains a sub-set of the data that was in the original file. The primary purpose of Micro-CAPS is to provide the CorsairHMI developer with a library of database components that he can combine to serve as a basis for his application.

Database Extraction

Database Extraction is the process of deleting records out of a CorsairHMI CAP to prepare it to be saved as a Micro-CAP for library purposes. One way that this can be done is manually with the developer deleting records as desired. He then saves the data using the 'Save As' menu option. This approach only works well with small databases. The developer must be careful not to use the 'Save' menu option as that would result in the extracted database being also stored in the original file. There is a developer preference to enable automatic saving of the CAP file when the CorsairHMI program exits to the operating system. This preference should be shut off when doing manual database extraction. Corsair can be directed to do the reduction with the entry of an extraction specification. This specification tells the program what to keep (and what to eliminate) in the database. Database Extraction is done using the CorsairHMI application transfer window.

The Application Transfer Window

The application transfer window is used to move data between CorsairHMI applications and the computer's disk drive. Transfers can be across CorsairHMI memory, from memory to disk file, from disk file to memory, or from disk file to disk file.

To get to the application transfer window CorsairHMI must be set for administrator development. The developer clicks on Setup/Application List. He must then decide if the source application is CorsairHMI memory or a disk file. If the source application is memory the developer must arrow up or down to highlight the desired source memory. He then clicks on Edit/Transfer from Memory. If the source application is a disk file the developer clicks on Edit/Transfer from Disk.

The top group box of the transfer window deals with the source application. If the source is memory the number and nickname of the source application is shown at the top. If the source is a disk file a browse button is shown to help the developer to find the file.

Once everything is correct on the source application the developer moves on to specify the destination application. The three options are Same Place, Memory, and Disk. Same Place means that the result of the transfer will end up in the same place (memory or disk file) where it started. This option is frequently used for screen resolution changes. Memory means that the transfer results will go into Corsair application memory. A memory number from 1 to 100 must be entered by the developer. Disk means that the transfer results will go to a disk file. The developer can type in a file name. The browse button is an aid to find files.

The type of transfer action must be specified. The options are Verify Blank Destination, Replace Destination, and Merge with Destination.

Verifying a Blank Memory Destination means that the destination application memory must be empty or the transfer will not occur. Verifying a Blank Disk Destination means that the destination file must not exist on the disk or the transfer will not occur.

Replace Destination means that the destination memory contents will be replaced with the results of the transfer. If it is used with a disk file the old file contents will be lost. If the disk file does not exist a new one will be created.

Merge with Destination means that the transfer results will be merged in with what is already in the destination. Merging with a memory destination requires a valid application in that memory. Merging with a disk destination requires an existing valid disk destination file.

Transfer Option: Display Resolution

The developer may wish to change the display resolution of the source CorsairHMI database as part of an application transfer. Micro-CAP library files containing graphical database records (screens or drawings) will have been developed assuming a particular screen resolution. This resolution may have to be changed during Micro-CAP import to match the target computer system.

Display resolution changes will always require manual adjustment of graphic items to make the images look right at the new resolution.

Transfer Option: ID Prefix

A developer may wish to prefix all ID names in the source CorsairHMI database as part of an application transfer.

Transfer Option: Database Extraction

Database Extraction can be done during application transfer to form a Micro-CAP.

Corsair Database Extraction Functions

The following automatic extractions are possible:

- Reduce to all screens without tags
- Reduce to all screens keeping the tags
- Reduce to a single screen with or without tags
- Reduce to just drawings
- Reduce to a single drawing
- Reduce to just the icon database
- Reduce to a single icon
- Reduce to a single driver with its PLCs and tags
- Reduce to a single PLC with its tags
- Reduce to a single tag

Other extractions must be done manually by the developer.

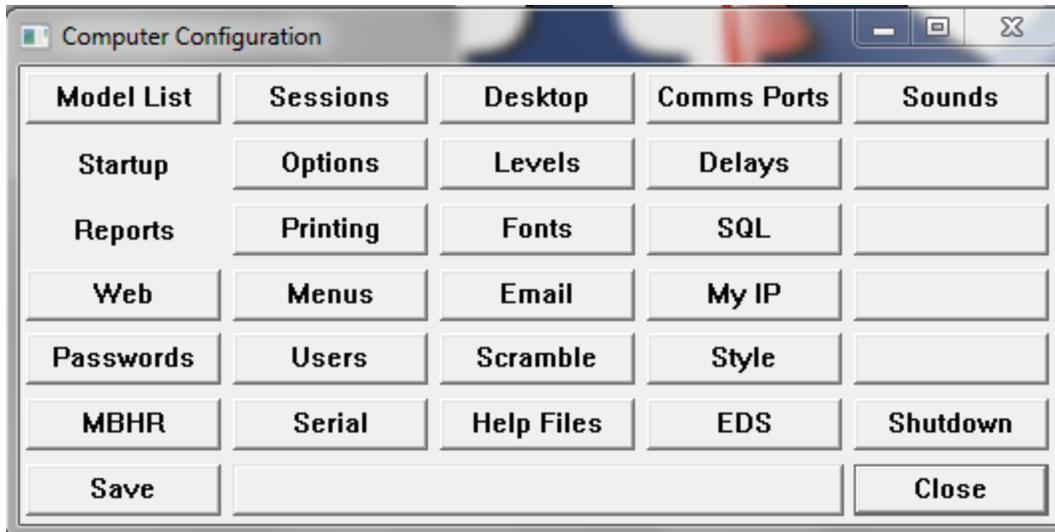
Micro-CAP Import

A Micro-CAP is imported into a larger application through the Corsair application transfer window. Usually the Micro-CAP is a disk file. The source is the disk file. The destination is CorsairHMI memory. The destination action should be set to 'Merge with Destination'.

A Micro-CAP can be imported into an application multiple times. Usually different ID prefixes should be done with each import to help keep the data properly sorted and prevent duplicate tag errors.

Computer Configuration

Computer Configuration is a file containing a collection of information about what the Corsair program is to do. Multiple Corsair programs on separate computers frequently use the same Model file(s). Each computer uses a different Computer Configuration file. This enables each computer to know what it can and cannot do with the Model data. This is why the Configuration file is an essential component of Plantwide interface. The default name for this file is 'Corsair.cfg'. The full name for this file is usually 'c:\corsair\corsair.cfg'.



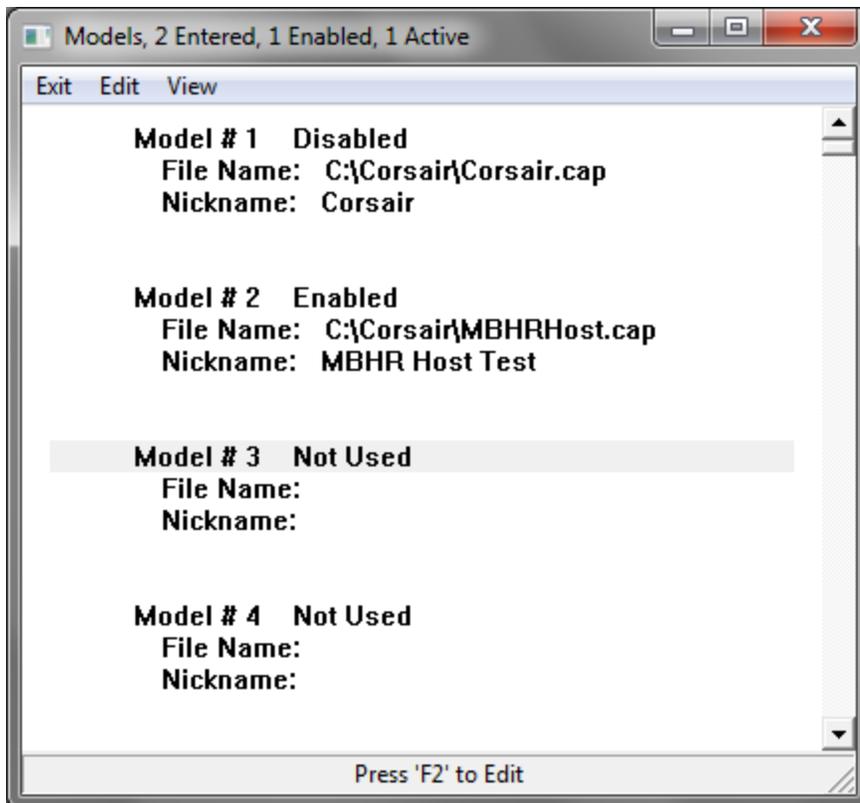
The computer configuration window has several buttons representing different groups of configuration items. The edit control on the bottom of the window tells when the configuration data has been changed and needs to be saved using the button on the lower left side of the window.

Model List Group

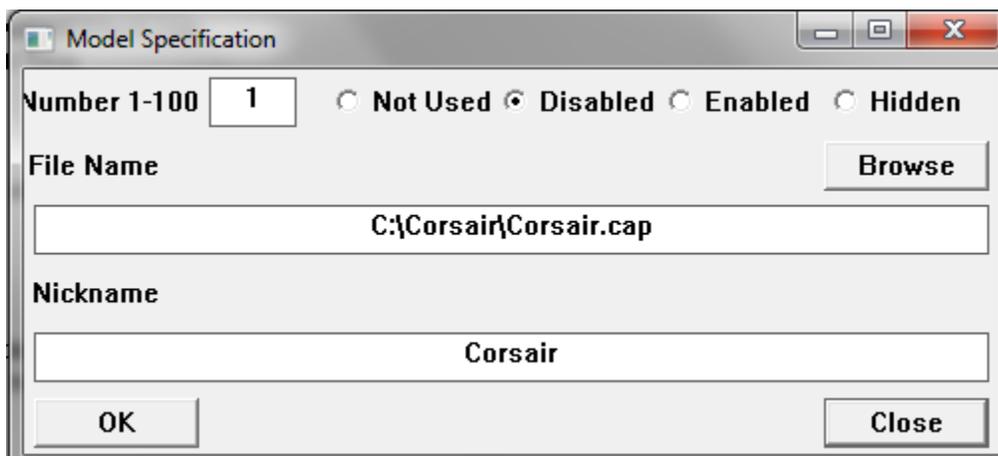
The 'Model List' menu option opens the window that is used to configure what model files are used by the CorsairHMI program. A model file contains an interface database with tags, graphic screens, sheets, alarms, and all the information needed for a complete operator view. These files typically have file names ending with a '.cap' extension. The normal default file name is:

C:\Corsair\Corsair.cap

If the license allows for Multiple-Model operation Corsair can use up to 100 model files at a time. The Model list window has a scrollable list of these files.

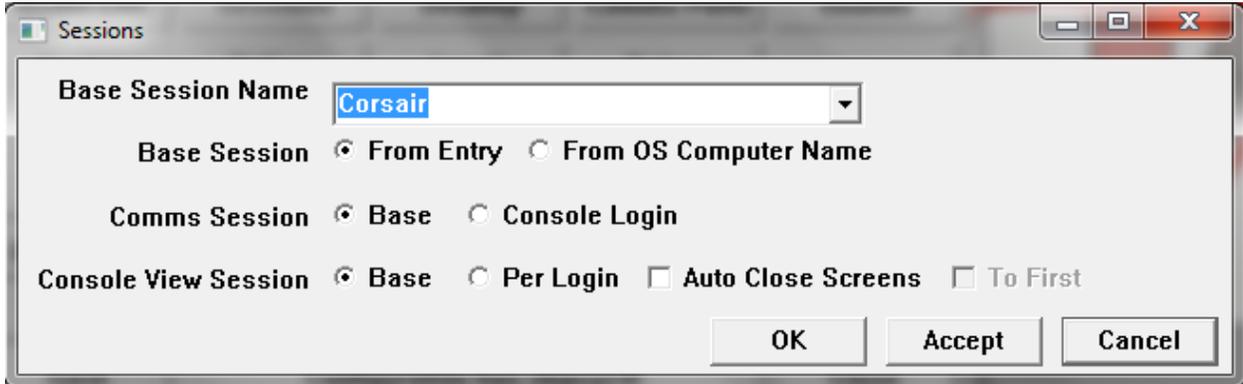


Each file has a number from 1 to 100. The Edit menu option or the F2 key opens the window that specifies a model.



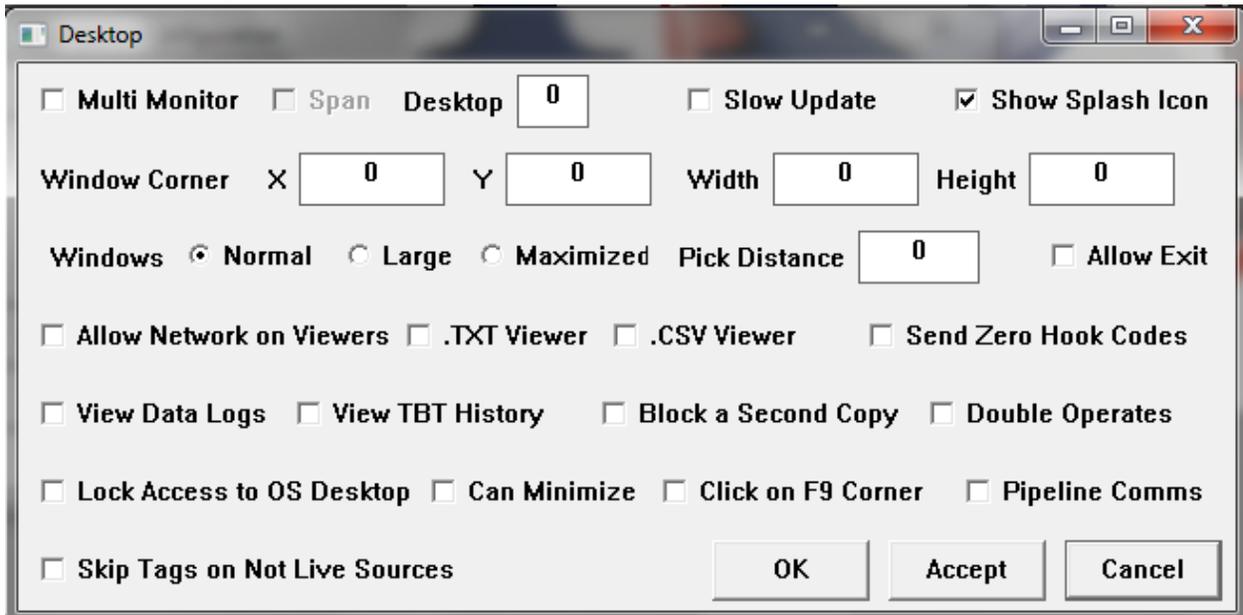
Each model is specified with a file name and a nickname. The nickname is what the operator uses when he selects which model he wants to see. If a model is 'Not Used' the file name and nickname are forced to be blank. If it is 'Disabled' the file and nickname entries remain but the program does not use the file. Normal operating models are 'Enabled'. A 'Hidden' model is loaded and operated by the program but it cannot be selected by the operator. Administrator-level development is required to view a hidden model.

Sessions Group



Xxxx

Desktop Group



The Corsair program can be used on computers with multiple monitors without any special settings. The operator can drag each window to the monitor that he wants it to appear on. Windows will tend to appear on the operating system's #1 'primary' monitor.

More options are available when a setting is made on the Computer Properties window Interface tab. There is a 'Desktop Monitor' number. This number is the number of the monitor that the Corsair Desktop gray screen is to appear on. In single-monitor systems this number is left at zero. The Corsair

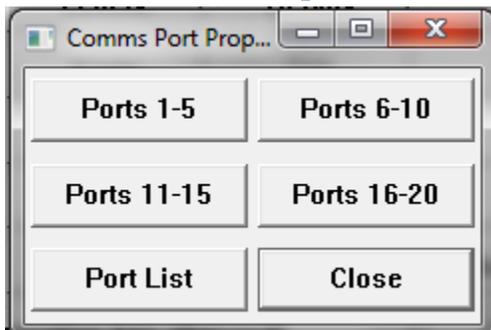
Multi-Monitor features will be activated when it is set to a nonzero value. Corsair keeps track of up to 4 monitors. If a value of 2 is entered for the Desktop Monitor and the computer properties are saved the next time that Corsair is started it will appear on monitor #2.

Under the Edit/Data/Sessions menu option the developer can enter a screen for each of the 4 monitors on the session record. When the operator of that session clicks on the 'Run' button these screens will appear in the proper places.

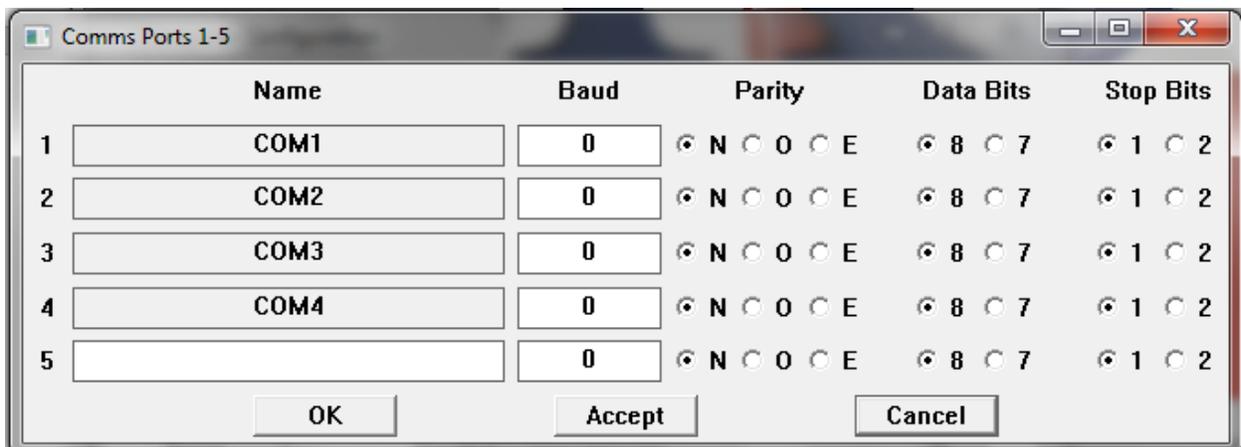
The Help/About window contains a button that can be used to access a window giving information about the monitors that are present on a system.

The 'Show Splash Icon' checkbox is used to display a bitmap in the center of the Corsair desktop window. It must be a file named 'corsair.bmp' in the same folder as the Corsair program. After turning on the checkbox and saving the computer configuration it may be necessary to restart the Corsair program for the icon to appear.

Comms Ports Group

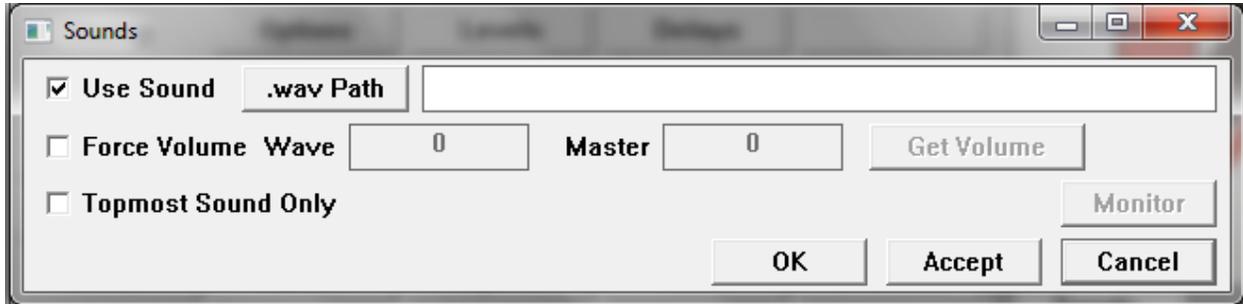


The CorsairHMI program can use a maximum of 20 serial communications ports. In the future the 'Port List' button will be used to display what ports are available on the system.



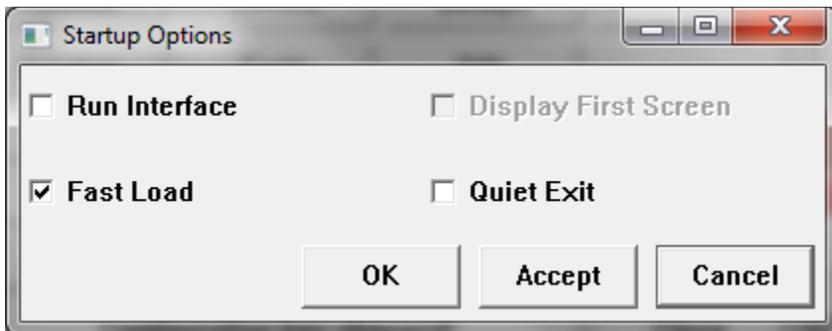
The first 4 ports use fixed standard names. Ports 5 through 20 have developer enterable names. Each port is configurable for baud rate, parity, data bits, and stop bits. Parity options are none, odd, or even.

Sounds Group



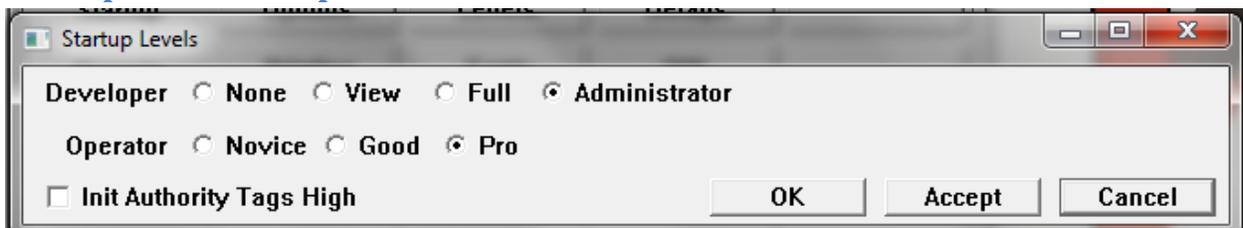
Xxxx

Startup Options Group



Xxxx

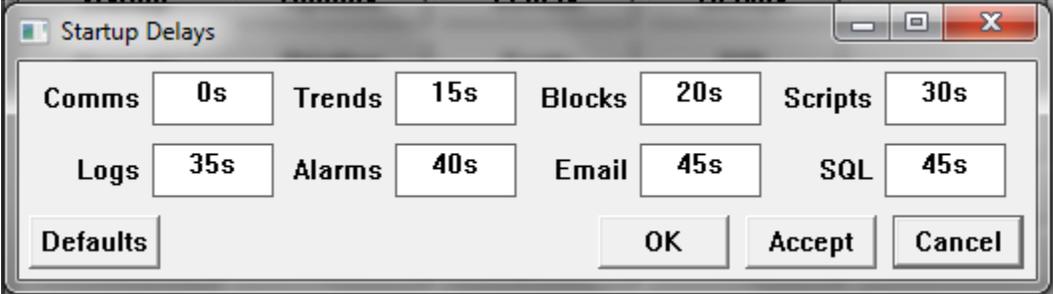
Startup Levels Group



The Startup Developer level can be adjusted from 'None' to 'Administrator'. Typically, it is left at Administrator during project development and set to None for normal interface operation after that.

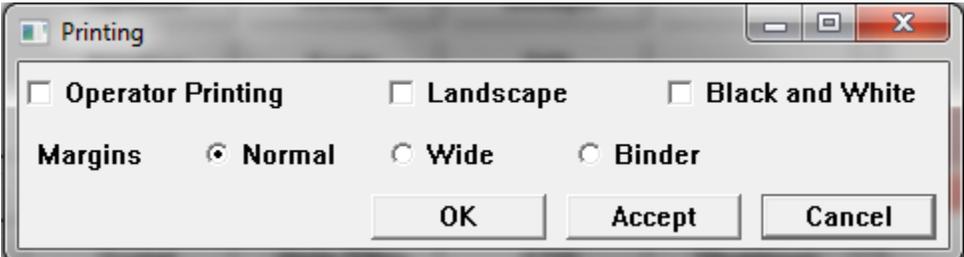
The Startup Operator level can be adjusted from 'Novice' to 'Pro'. This group only determines the levels when the Corsair program starts running. After that point the levels can be adjusted from the 'Users' 'Change Levels' menu option.

Startup Delays Group



Xxxx

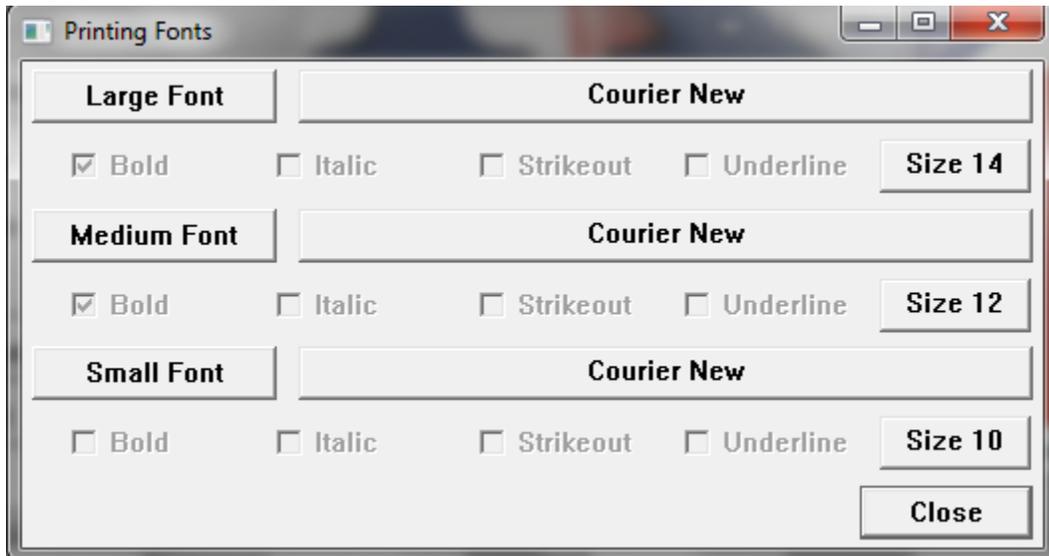
Reports Printing Group



When development is shut off the computer looks at the 'Operator Printing' checkbox to see if it should allow printing. Default options for Landscape vs Portrait printing and Black and White versus color are set here.

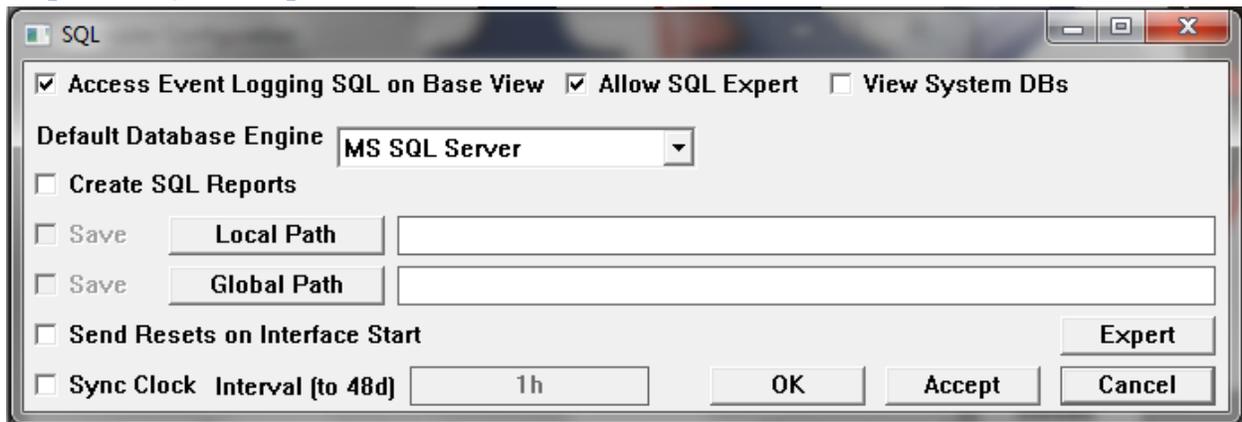
Normal margins leave some margin on each side of the page. Wide printing leaves narrow margins on each side. Binder printing leaves some margin on the left side of the sheet and very little margin on the right side.

Reports Fonts Group



If the three printer fonts are not selected printouts may have very tiny print that is unreadable. The illustration shows a possible suggestion for a starting point.

Reports SQL Group



The 'Create SQL Reports' checkbox is checked if this computer is to be allowed to create reports. They can optionally be saved to a local or global folder location. The folders are specified as paths. A Windows path should end with a slash, like 'C:\corsair\local_reports\'.

The 'Access Event Logging . . .' checkbox must be on for the operator to be able to view logged events.

Web Group

The screenshot shows a 'Web' configuration window with the following settings:

- Host Mode:** No Login - Operate Only
- Host URL:** http://+:80/
- Show Users:**
- Use Ajax:**
- Use REST:**
- REST PUT:**
- View Session:** Base, Fixed, Default, Per Login
- Buttons:** Status, OK, Accept, Cancel

The 'Host Mode' selector is what turns on hosting. There are several options depending on how the developer wants the web client login operation to work.

'No Web Host Operation' is exactly that.

'Login to Operate and to View' means that a login is always required. Operating capability depends upon the authority system.

'Login to Operate, Bypass to View' means that anyone can get in without a password but they cannot change anything. If someone logs in the authority system decides what they can do.

'Login – Bypass to Operate' means that anyone can get in without a password or by logging in. In either case the authority system decides what they can do.

'No Login –View Only' means that no login is asked for and there is no operating capability.

'No Login – Operate Only' means that there is no login and the authority system decides what can be done.

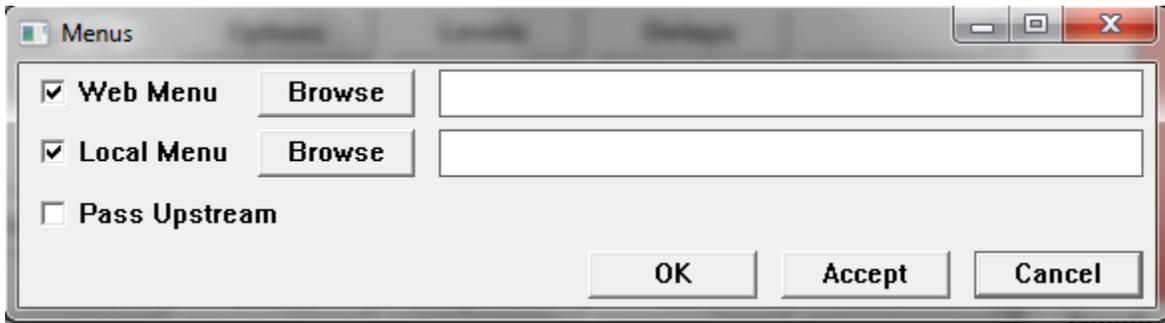
The 'Show Users' checkbox allows User names to be shown as selectable items on the Web Browser's login page.

The 'Host URL' specification is normally left blank. A sample entry for a Windows system is shown near the entry box. It is used to apply a specific URL name to the host or to configure the host for a port other than the default port 80.

The 'Use Ajax' checkbox is always checked. It is only left off for special diagnostic situations where the browser user wants to see the HTML code that Corsair is sending.

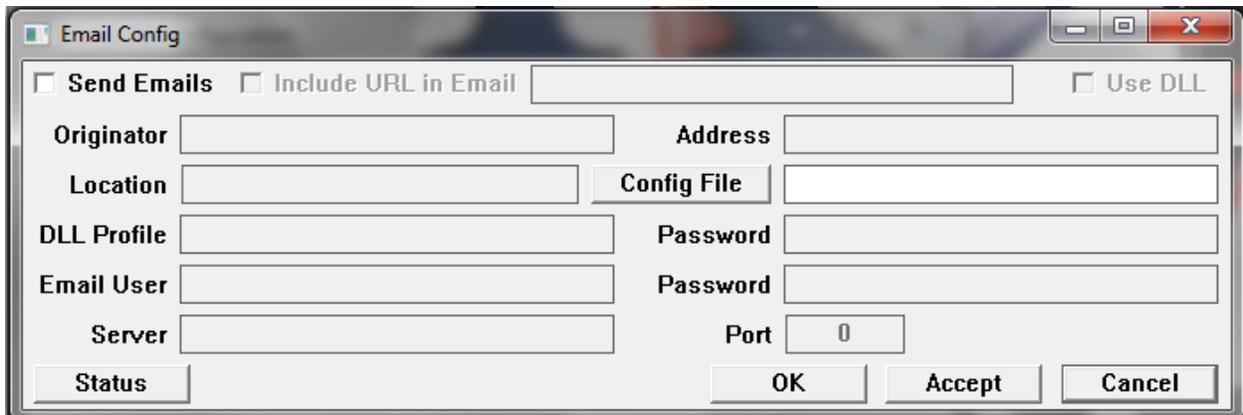
The next selections determine what image the web browser sees on the initial login page. The first choice is the default Corsair airplane automation. The next choice is a png file that the developer must specify. The third choice is no image. The fourth choice of an html file is not available at this time.

Menus Group



These options are for a future capability that is not complete at this time.

Email Group

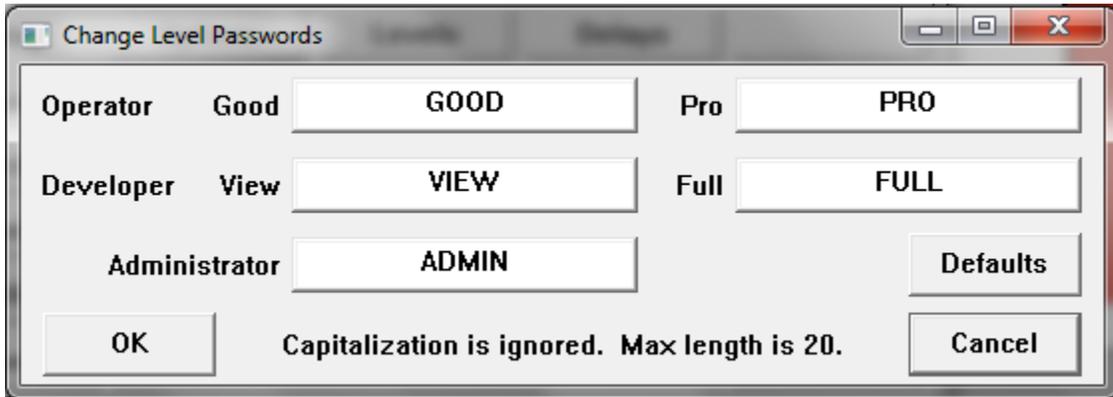


Xxxx

My IP Group

A Corsair computer that is connected to the Internet can determine what IP address is used for its Internet connection. This configuration group is used to set up how this determination is done. It is described in the 'My IP' section of this manual.

Passwords Group



The 'Change Level Passwords' dialog box contains the following fields and controls:

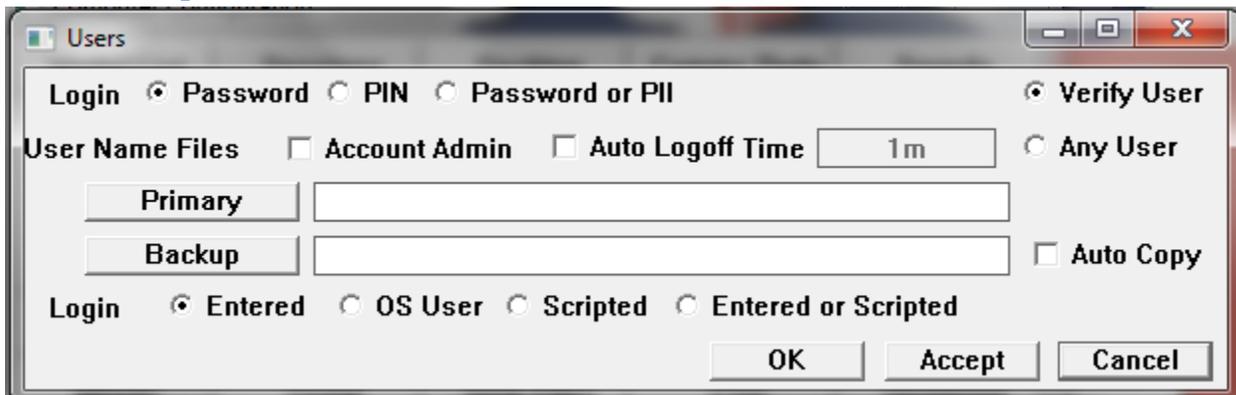
Operator	Good	<input type="text" value="GOOD"/>	Pro	<input type="text" value="PRO"/>
Developer	View	<input type="text" value="VIEW"/>	Full	<input type="text" value="FULL"/>
Administrator		<input type="text" value="ADMIN"/>		

Buttons: OK, Cancel, Defaults

Text: Capitalization is ignored. Max length is 20.

There are passwords that are used to set operator and development levels.

Users Group



The 'Users' dialog box contains the following fields and controls:

Login: Password, PIN, Password or PII, Verify User

User Name Files: Account Admin, Auto Logoff Time: Any User

Primary:

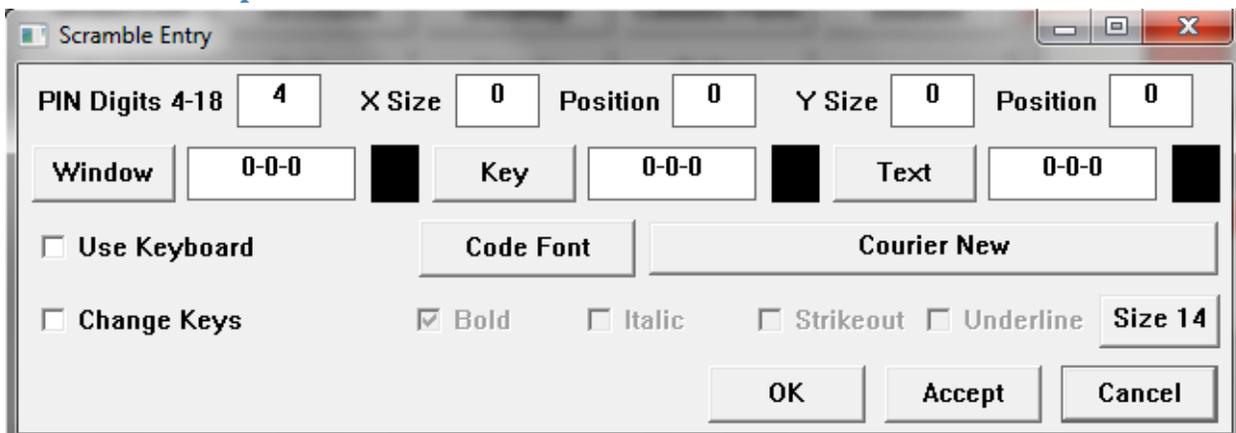
Backup: Auto Copy

Login: Entered, OS User, Scripted, Entered or Scripted

Buttons: OK, Accept, Cancel

Xxxx

Scramble Group



The 'Scramble Entry' dialog box contains the following fields and controls:

PIN Digits 4-18: X Size: Position: Y Size: Position:

Window: Key: Text:

Use Keyboard Code Font:

Change Keys Bold Italic Strikeout Underline Size 14

Buttons: OK, Accept, Cancel

Corsair can display a specialized numeric keypad on the screen in some high-security applications. The positions of the numbers on the screen can change every time the keypad is used so that someone cannot recreate the password sequence by watching the touchscreen from a distance.

Style Group

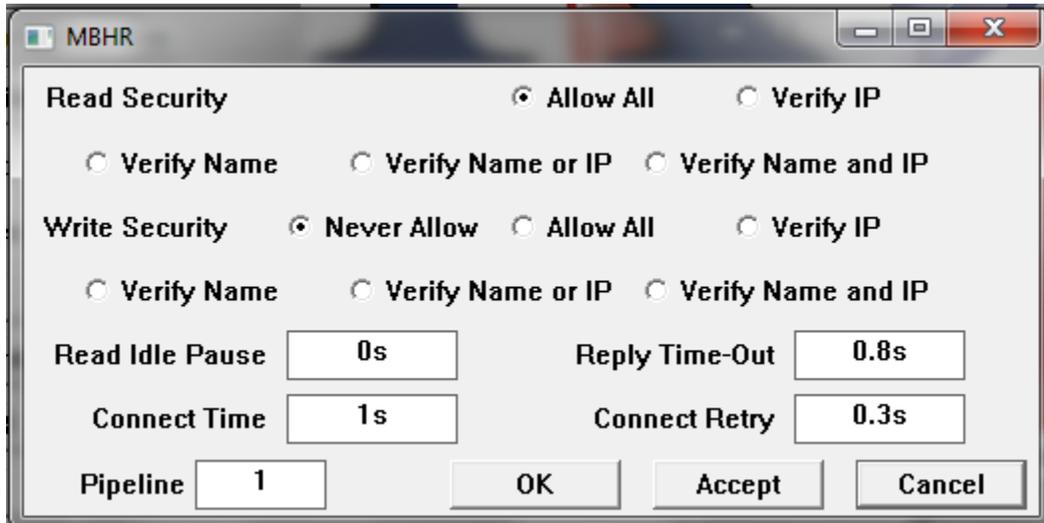


The 'Use Small Windows' checkbox is used when the Corsair program is running on a small monitor like a Raspberry Pi touchscreen. Some development functions are shown differently on small monitors.

Status Height is the height in pixels of the status bar on the bottom of a Corsair graphic screen. It normally uses a text font that comes from the operating system. The Custom Font checkbox permits the developer to select another font. The Custom Color checkbox permits the developer to select the color of the status bar.

The template font is the font that is used for text on I/O module templates. The illustration shows a suggestion.

MBHR Group



The screenshot shows the 'MBHR' configuration dialog box. It has a title bar with a green icon and the text 'MBHR'. The dialog contains several sections of radio button options and input fields. The 'Read Security' section has 'Allow All' selected. The 'Write Security' section has 'Never Allow' selected. There are input fields for 'Read Idle Pause' (0s), 'Reply Time-Out' (0.8s), 'Connect Time' (1s), 'Connect Retry' (0.3s), and 'Pipeline' (1). At the bottom are 'OK', 'Accept', and 'Cancel' buttons.

The 'Read Security' and 'Write Security' options are for when the computer is operating as an MBHR host. Write Security defaults to 'Never Allow'. If it is enabled with 'Allow All' the MBHR remote computer can write data to the host which can then write it to a PLC.

The remaining parameters are for MBHR remote operation. Read Idle Pause determines the time between requests from the remote. Putting a time of 0.05s here helps to keep from flooding the master with requests.

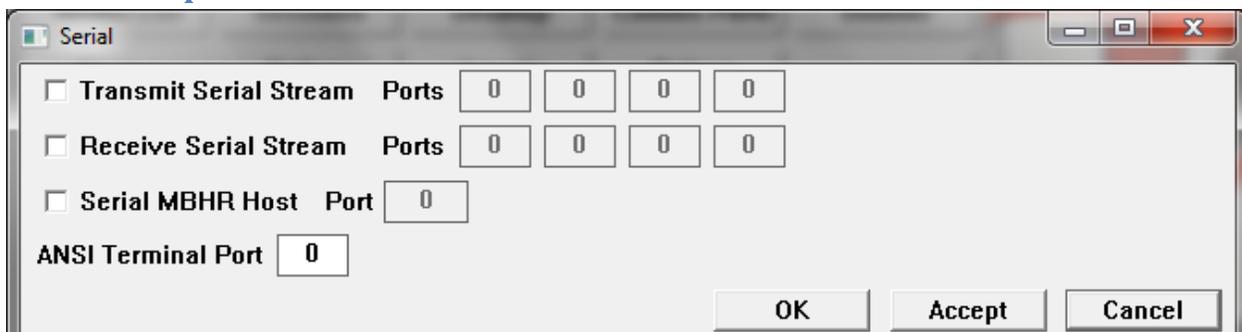
The Reply Time Out time is how long the computer allows for the host to reply. A host over the Internet may need a longer time-out than one on a local computer. 10 seconds works well in many cases.

The Connect time is how long the computer allows for the host to respond when it initially requests a connection. It should be as long or longer than the Reply Time Out. 20 seconds is a possibility.

Connect retry is the time after a failed connect request before another connect is tried.

The pipeline count can vary from 0 to 16. 0 or 1 are for no pipelining of communications requests. Excessive pipeline counts can cause performance problems with the Corsair host, especially with Corsair extended Modbus protocol.

Serial Group



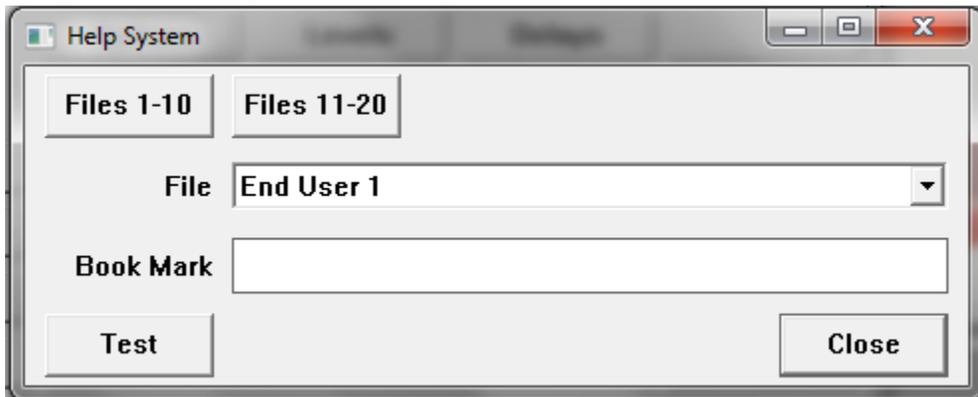
The screenshot shows the 'Serial' configuration dialog box. It has a title bar with a green icon and the text 'Serial'. The dialog contains several sections of checkboxes and input fields. The 'Transmit Serial Stream' section has a checkbox and four 'Ports' input fields, all set to 0. The 'Receive Serial Stream' section has a checkbox and four 'Ports' input fields, all set to 0. The 'Serial MBHR Host' section has a checkbox and a 'Port' input field set to 0. The 'ANSI Terminal Port' section has an input field set to 0. At the bottom are 'OK', 'Accept', and 'Cancel' buttons.

This group is used to assign serial ports for some configuration options. If Corsair is using Streaming Serial data security either transmit or receive serial ports are specified here. A maximum of 4 ports can be operated at the same time.

Corsair can operate in a serial MBHR Host mode using the Modbus RTU protocol. It requires one serial port.

Corsair can act as a ANSI-compatible serial terminal if it is started with a special command line parameter. This requires a serial port.

Help Files Group



Xxx

Help Files 1-10

	File	Prefix	Suffix
End User 1	<input type="text"/>	<input type="text"/>	<input type="text"/>
End User 2	<input type="text"/>	<input type="text"/>	<input type="text"/>
End User 3	<input type="text"/>	<input type="text"/>	<input type="text"/>
End User 4	<input type="text"/>	<input type="text"/>	<input type="text"/>
Overview	<input type="text"/>	<input type="text"/>	<input type="text"/>
Operator	<input type="text"/>	<input type="text"/>	<input type="text"/>
Developer	<input type="text"/>	<input type="text"/>	<input type="text"/>
Designer	<input type="text"/>	<input type="text"/>	<input type="text"/>
Experts	<input type="text"/>	<input type="text"/>	<input type="text"/>
Corrections	<input type="text"/>	<input type="text"/>	<input type="text"/>

OK Cancel

Xxx

Help Files 11-20

	File	Prefix	Suffix
Battery Mon	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reserved	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reserved	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reserved	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reserved	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reserved	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reserved	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reserved	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reserved	<input type="text"/>	<input type="text"/>	<input type="text"/>
Glossary	<input type="text"/>	<input type="text"/>	<input type="text"/>

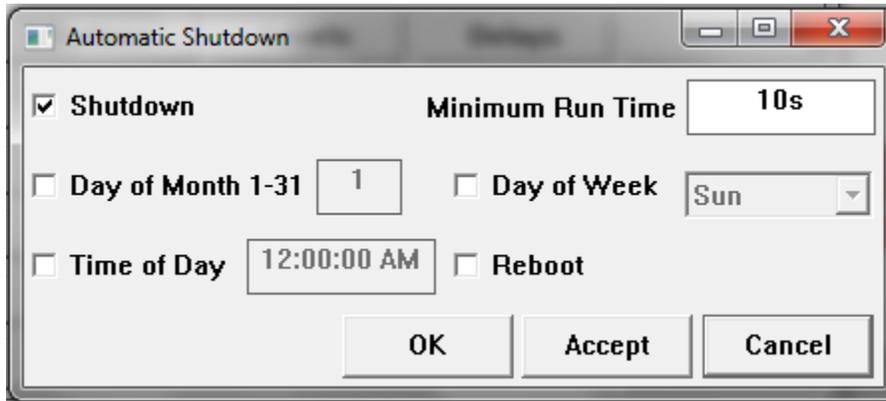
OK Cancel

Xxx

EDS Group

This is a future option that is not active at the present time.

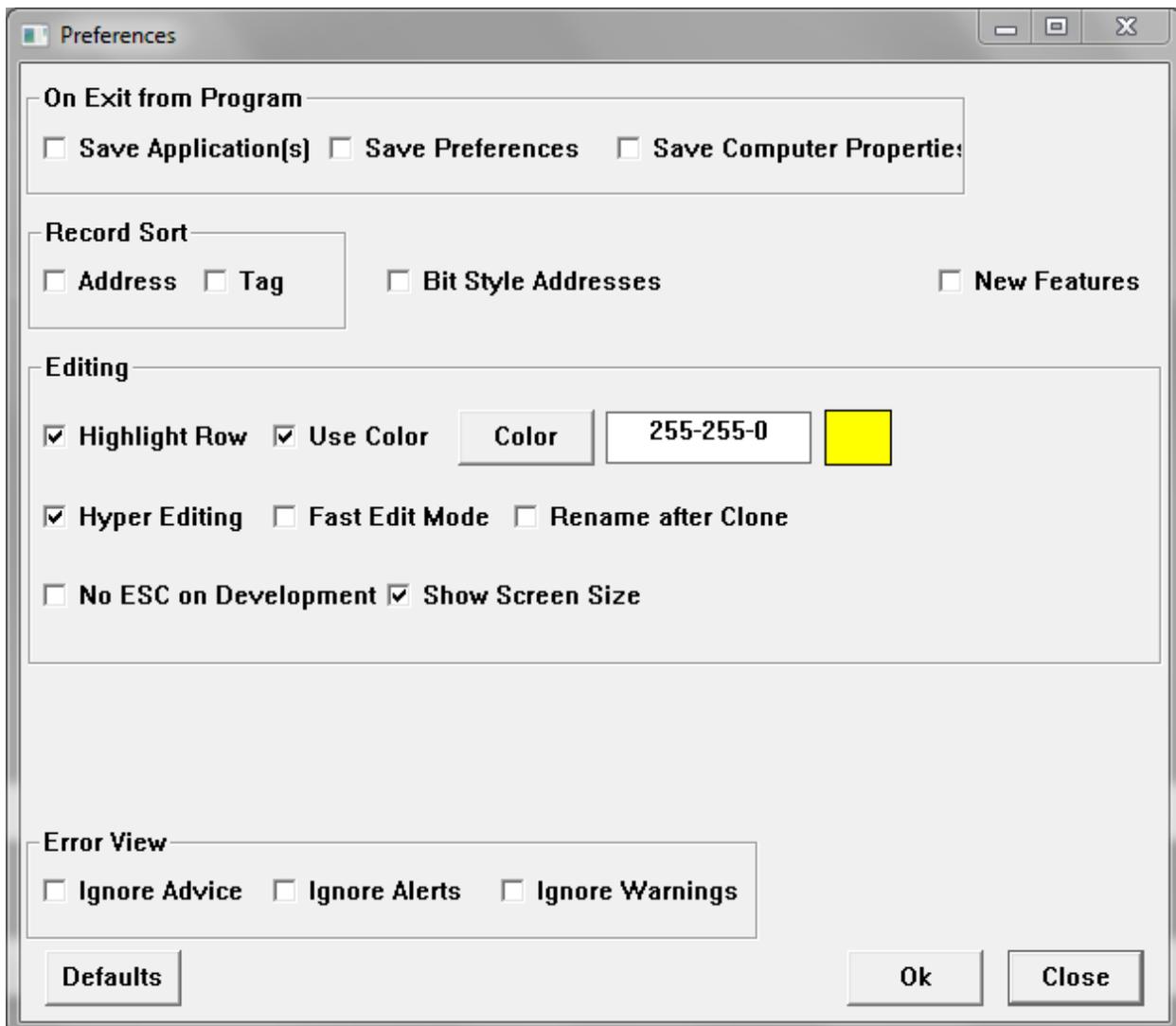
Shutdown Group



Windows Corsair can shut the computer down and optionally reboot it automatically. It will run for at least the specified minimum run time. If no additional conditions are checked it will shut down after that time. Up to 3 additional conditions can be added as requirements after that time. Windows may have to run Corsair 'As an administrator' for this option to work.

Developer Preferences

The 'Setup' 'Preferences' menu option opens the developer preferences entry window.



The program can be individually configured to automatically save Models, Developer Preferences, and Computer Properties when it is closed.

The 'Record Sort' options enable a background sortation process for records like tags, alarms, calls, and register blocks. Some developers may prefer to leave it shut off so that records do not 'jump around' as data is entered. One option is to turn both of these items on, do a complete thinker cycle, and then turn them both off.

The 'Highlight Row' checkbox is used for sheet-style Corsair database development. It causes the computer to highlight the entire record, not just the current field. The color is adjustable. The illustration has a suggestion.

Database error checking goes on constantly as a background process. Advice, Alerts, and Warnings can selectively be ignored if the developer desires.

Printing the Manual

The Corsair manual is a collection of printouts that are generated by the Corsair program. Each printout is considered to be a chapter of the manual. The Corsair developer can select what chapters are to be included when he prints the manual. He can select many options for the manual including whether or not to include a main index of the chapters, whether to print indexes with the chapters, page numbering options, and so on.

Manuals are printed for different purposes with different audiences. Engineers need to review the design of a project as a part of submittal review. The operators that use the Corsair interface are one audience. The electricians that maintain the control system are another. The PLC programmer has other needs. The Corsair developer needs other chapters. Usually different versions of the manual are printed for different people.

Printing the Manual

The manual is printed from the Corsair program's main window by picking File/Print/Manuals from the menu. A window appears that allows the developer to select a printer and a number of copies. Clicking on the 'Print' button on this window causes the computer to go on to the 'Print Manual' window. This window allows the selection of manual printing options. Clicking on its 'Print' button starts the printing process.

Sometimes when the Corsair program is first run with a large database some of the chapter options do not appear in the Print Manual window. This may be because the program needs more time to complete its background processing tasks. Usually the problem is resolved by closing the Print Manual window for a few minutes while waiting for the background processing to complete. The Memory Summary window can be accessed from the main window through the About/Memory Summary menu option. It gives information about the results from the background processing. Sometimes the interface may need to be shut off for a few minutes to allow the processing to be completed more quickly.

Engineer's Review Chapters

Specifying engineers on large projects sometimes require submittal documentation to verify that the contractor understands the plans and specifications. Some chapters of the Corsair manual can be used as a portion of the submittal.

The Computer List chapter tells the engineer what interface computers will be provided by the contractor.

The Alarms and Calls chapters tell the engineer information about the projects scope of work.

The engineer may also be interested in the Listed Drawings and PLC I/O chapters.

The interlock schemes chapter is used with corrections versions of the Corsair program. The engineer can use it to verify that the Corsair developer has the same understanding of door interlock logic that he has.

Operators Chapters

The Computer List chapter tells the operators what computers are running the Corsair program.

The Alarms and Calls chapters give a listing of alarms and calls that have been programmed into the system. The printout can be of all the alarms and calls in the database or only those that are seen on a particular computer.

The Listed Drawings chapter may include information that is of interest to the operators.

Electrician's Chapters

Electricians are sometimes interested in the TCP addresses chapter. They may be called in when a Corsair computer cannot communicate to a PLC. One thing that they can do is use the operating system 'Ping' utility to verify that the network cable and hardware going to the PLC are working. This chapter tells the electrician what address to use to ping a PLC on the network

The Computer List chapter gives a list of what computers are on the system using the Corsair database. In a large installation it may be a help to tell the electricians where the interfaces are.

Listed Drawings can be used to include wiring diagrams and other documentation about a system.

The PLC I/O chapter tells the electrician where PLC inputs and outputs are located and information about their addresses, module types, and signal ranges.

PLC Programmers Chapters

The PLC programmer is typically interested in the TCP addresses chapter. It documents the IP address of each PLC on the network that the Corsair program communicates with. The PLC programmer must set up the address into each piece of equipment using his programming software. The chapter helps to verify that each IP address on the network is different.

The Driver and PLC Configuration chapter documents how the communications configuration of the Corsair program is set-up. The PLC programmer may need this information to help configure his equipment.

The Door Bits chapter is only available with the corrections version of the Corsair program. It is used to co-ordinate PLC data addresses between the PLC program and the Corsair database.

The interlock schemes chapter is used on corrections projects to aid the PLC programmer with writing door interlock logic.

The Tree chapter is by far the most important chapter for the PLC programmer. It is the document of the data interface between him and the Corsair developer. He should keep an updated copy of the tree with him at all times when writing his PLC program and during start-up.

The CI Device Addressing chapter is used with Corsair computer-indexed device tags. It correlates PLC addresses with Corsair computers in systems where multiple computers are used. This speeds up the writing of the PLC program and helps avoid errors in it.

The ASCII table chapter is useful for the PLC programmer in when he needs to do some ASCII work in the PLC code. The Corsair programs ASCII expert windows are designed to give him more help in these cases.

Corsair Developers Chapters

The Corsair developer is interested in the TCP addresses chapter since it documents the IP address of each of the Corsair computers on the network. These addresses must be set into the computer through the Windows control panel network set-up features.

The device use counts chapter can be an aid to the Corsair developer. It shows how many times each device tag in the database is linked to something else in the database. Normally each device should be used at least once. This chapter is extremely valuable when a project has multiple PLCs with identical programs on identical pieces of equipment. The developer can check this chapter to see if the use counts of the devices on each PLC match. If they do not match there is a development error.

The Alarms and Calls chapters can be used by the developer to check what he is doing against project specifications.

The Blocks chapter is used to document the calculation blocks that are available for the developer to use.

The Computer Properties chapter provides documentation as to how the properties on the Corsair computer are configured. This record shows what the computer is allowed to do, how it is configured for printing, and other items.

The Driver and PLC Configuration chapter documents how the communications configuration of the Corsair program is set-up.

The tree chapter can provide a visual way for the developer to check that PLC addresses in the database are in the correct place.

The password rules chapter helps the developer to understand the relationship between authority device values and password levels. This can be difficult to understand without the help of the chapter.

The Templates chapter provides information to the Corsair developer about how to use each of the standard templates that are within the Corsair program. This includes the required point names, devices, device sizes, and sometimes suggested addresses.

Reference Chapters

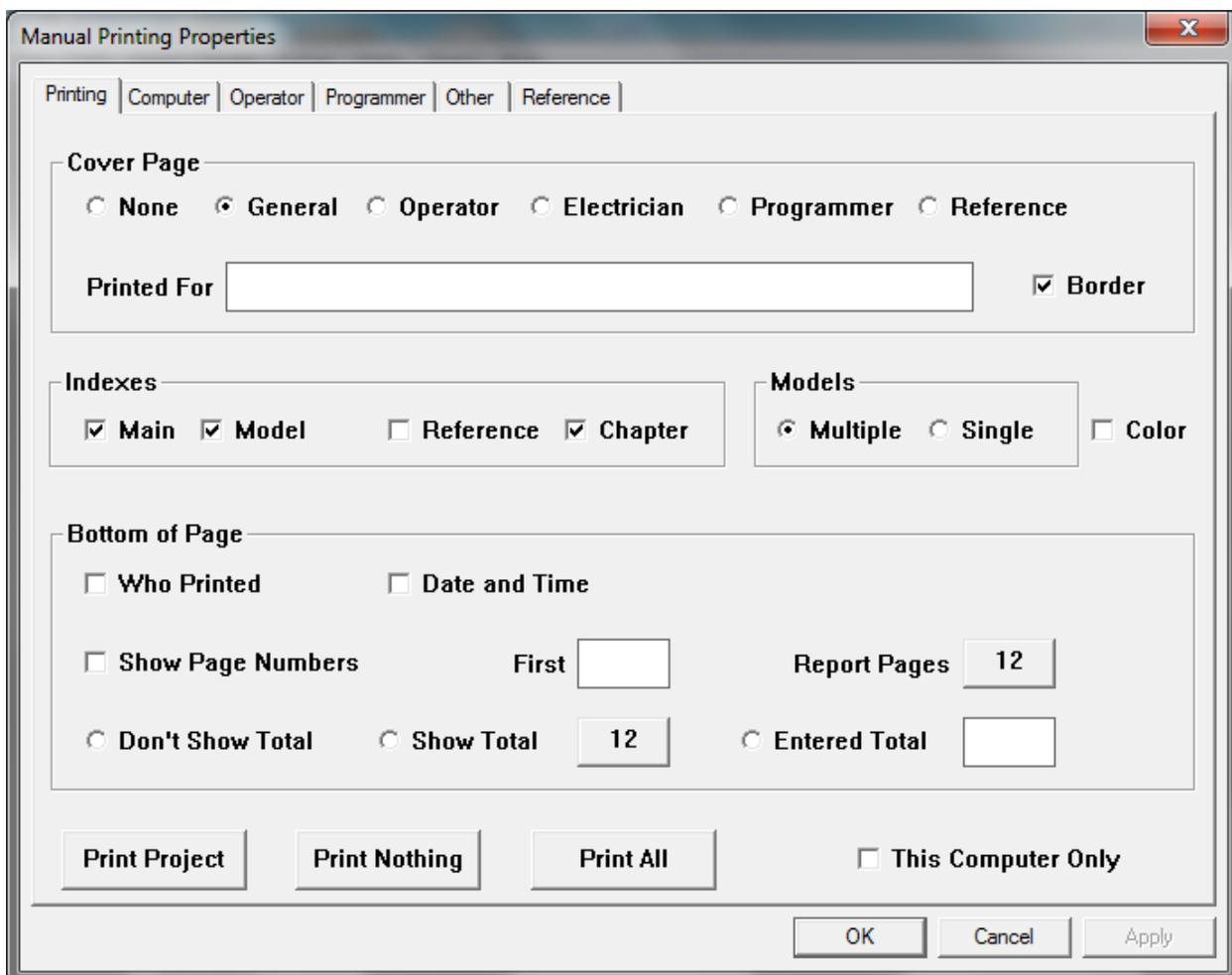
Some chapters are for reference purposes. They serve as an addition to other documentation for the Corsair program.

The Help chapter is used as a quick keystroke reference for the Corsair developer.

The Error Codes chapter is a printout of the error code numbers that are used by the Corsair program. It is helpful for the developer when he sees an error code number on the screen.

The ASCII Table chapter prints out a detailed chart of the ASCII codes ranging from 0 to 127.

The Templates chapter tells how the Corsair program uses its standard data templates.



The My IP System

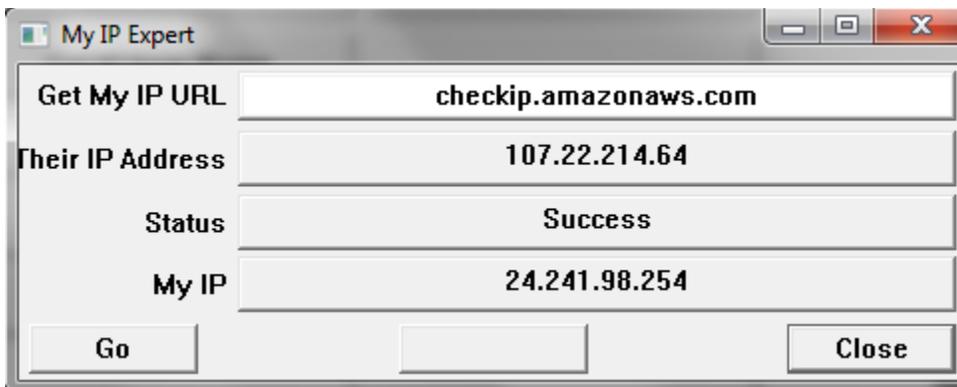
All versions of CorsairHMI contain a built-in system for the Corsair computer to determine its address on the Internet. This is generally not the same as any of the IP addresses of the computers Ethernet ports. The system is not needed for installations that have a fixed known IP address. It may be important when a fixed IP is too expensive, when it is not available, or for temporary installations.

Corsair must access a web service that provides reporting of IP addresses. The developer must determine what URL address he is going to use for a service. Two possibilities are:

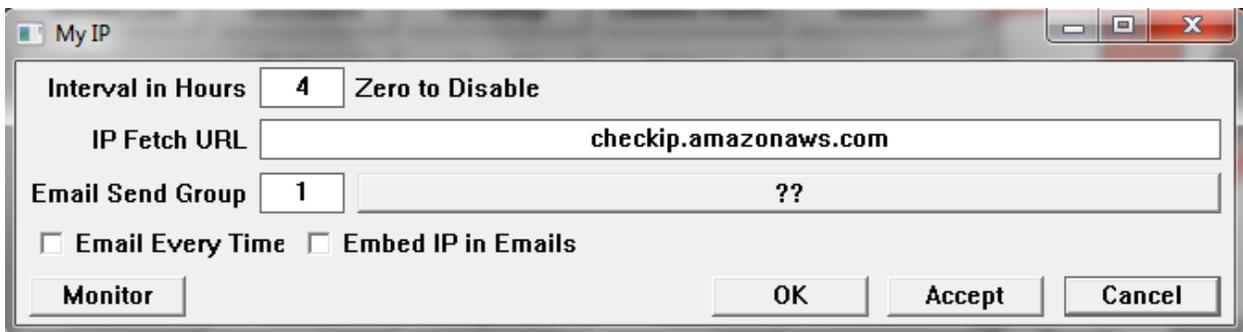
checkip.amazonaws.com

whatismyip.akamai.com

These and other URLs can be verified for proper My IP operation using the TCP Expert's My IP Expert.



When proper operation of a URL has been verified with this window it must be entered into the My IP group on computer configuration.

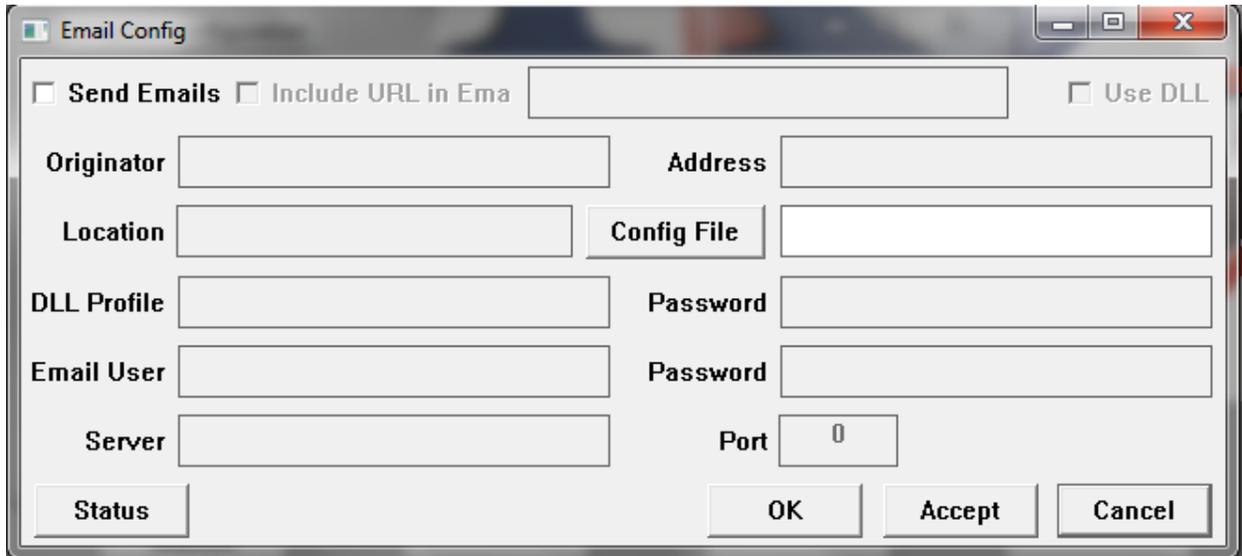


The Hours entry is set to a nonzero value to activate the automatic My IP system. The first check will be 5 minutes after the CorsairHMI program is started. Additional checks will occur at intervals of the number of hours that is entered here. The service's URL is entered into the edit box on the right.

The Send Group number is the key number for an email send group for IP address notifications. It must be nonzero for messages to be sent. The normal action is to send one message when the address is checked the first time. After that the address is rechecked at the IP Hours interval but messages are not sent unless it changes. The 'Always' check box is used to make Corsair send messages with every check

of the IP address. If the hours are set to 24 and Always is checked Corsair will send an IP address notification at daily intervals.

The next step may be to embed the My IP address in alarm email messages. This is done from the computer properties email tab.



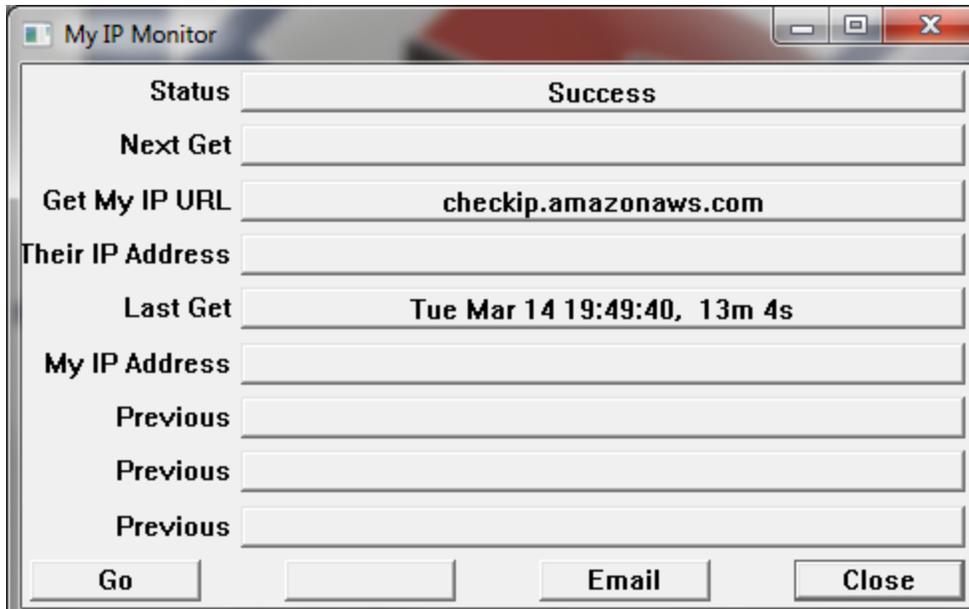
The option to 'Include URL in Email' must be checked. Normally Corsair uses the URL that is entered into the edit box. If 'Embed IP' is also checked it will put its current My IP into each email. It will send 'http://' followed by the IP followed by whatever is entered into the edit box. This may be something like ':83' if Corsair is acting as a web host on nonstandard port 83.

If the following things are true:

- Corsair is configured to act as a web host.
- The My IP system is properly configured.
- The local router is properly set up with port forwarding.
- The Internet Service Provider is cooperative.

then when Corsair sends an alarm message by texting to a cell phone the cell phone user can click on the URL in the message to open his browser and view Corsair.

The Help/About window has a button to open the My IP system monitor window.



This window shows a summary of what is happening with the automatic My IP system. The Go button can be used to start a read of the IP. The Email button provides a review of the IP Address reporting email message.

New Versions via Email

CorsairHMI may wish to email a new version of the corsair.exe program to a customer. Many email systems will not permit receiving a .exe executable program. The customary way to handle this is to email the program disguised as a Microsoft Word document. The file name that is used is 'Mike.doc'. When you receive this file do not open it. It must be renamed to 'corsair32w.exe'. The operating system may issue a warning about changing the extension of a file name. This warning can be ignored.

Using CorsairHMI with Linux

Versions of the CorsairHMI software can be purchased for use with computers that are running the Linux operating system.