

Cortex Software release 24.0.0

December 2008

Important Installation Notes

Since this a new major version release it cannot be installed as an update to V23. Instead V23 must first be uninstalled (using MS Windows Add/Remove Programs utility). Then V24 can be installed by running the appropriate install file.

Before uninstalling V23:

- 1. Back up your registration details and note down the Install Key Code since this will be required by the V24 installation process.*
- 2. Backup your database as a precautionary measure and identify it as used for Cortex 23 before installing Cortex 24.*

Note features are those added since initial release of Cortex 23.0.0

New Features

Octal Digital Output Module

Support for Octal Digital Output Module added. This module has eight digital logic outputs.

Network Supervisor

Now detects and reports modules left in firmware loader mode.

Direction detector

This object allows various proximity/beam detectors to be used in a doorway. This version allows the direction to be detected and therefore which room becomes unoccupied/occupied.

Macro object

Macro object now supports IF, ELSE and ENDIF actions.

Macro can now execute shell command (23.1.1).

Copy and Paste of Macro objects retains content (23.1.8).

Structure View.

Structure view now has alternative 'List View'. List view lists all the objects by type rather than ownership. A filter allows groups of objects to be viewed. Specifically designed for increasingly large networks where hundreds of objects need to be worked on.

New features in the popup menus of objects include an easier to use method of moving the plan visible part of an object to appropriate rooms.

Also again for large systems the ability to find an object in plan view. It will automatically scroll plan view to make the object visible and blink it for identification.

Connections Menus.

Connection boxes now increased in size to allow larger names to be viewed more easily (particularly for languages other than English).

Reflex Tools.



The various tools for Reflex have now been combined into one menu. This menu may be accessed from the main menu, visit Main menu | Tools | Reflex tools.

History: Reflex is referred to as packets sent from one module to another to perform some form of control without the need for Cortex/PC to be present. Various reflex menus may be found for each module allowing the module to be programmed for reflex operation. Some reflex action may be combined with normal Cortex behaviour.

Under normal Cortex operation there is no need to be concerned with Reflex programming. However more installations are relying on Cortex and an IDRAnet to be the only method of controlling a property's Lighting and Heating etc. To this end should the PC, Windows or Cortex have problems then some alternative method of controlling the system is desirable. Reflex allows an alternative but simpler method of controlling the network devices. To date the Reflex programming menus in Cortex have provide the method of setting up such Reflex control programs.

With increasingly large systems the creation of Reflex programs has become an onerous task. To this end a new 'Auto-Reflex Generation' feature has been created. This release caters for lighting control, motor control, plus some system watchdog features. It also provides a very simple heating control.

This generation process relies on the Cortex connections have been set up correctly. For example a Light may be configured for automation within Cortex i.e. Light Level, Room Presence etc. A button on a panel may be assigned to Toggle the lights operation and override the automation. Using this kind of information the 'Auto-Reflex Generation' will configure a Reflex packet for the button and store it in the module. This Reflex packet will control the Relay or Dimmer to toggle its state. Therefore under Reflex control the button operates the light manually.

This method has been shown to create hundreds of reflex packets, more accurately than manual creation and thus saves large amounts of work. It does not prevent users creating their own reflex operations in parallel with the auto-generated creations. Furthermore the auto-generated packets may be edited or deleted.

Note: This new feature may require some module firmware upgrades.

An option has also been provided to allow a DBI (dual button module) or QBI (quad button module) to provide a system reset facility. Such a module will be programmed for Cortex and Reflex automatically. It allows button 1 to be assigned the task of resetting the system. If Cortex is running it will back off and revert to its paused mode. The power-supplies will sequence and cause all modules to revert to Reflex mode. Button 2 is assigned to tell Cortex

to resume control. The LED's reflect what mode the network is in i.e. Cortex or Reflex operation.

The system watchdog feature has been in place for many years. A module monitors Cortex operation and should Cortex fail to respond after a specified period the module sends out a network reset putting all modules into Reflex operation. The Auto-Reflex Generation facility now offers to configure a module (typically the IPS – power supply) to perform this operation.

Digital input object

Additional new icons have been added which this object may represent. These include storage tanks i.e. for oil or sewage.

Door object

Now includes an optional plan view icon for a roof door (loft hatch).

On/Off load object

Now has plan view icon for European style sockets. Also option provided to invert icon state. On/Off load object now has default state option on network start. This prevents momentary change of state on network start if different to current state.

Multiple Light Logic

This object is designed to allow control of multiple lights in a room with a single button. The lights in a room may be optionally overridden using menus or several buttons. At some point it may be desirable to turn them all on or all off with a single button. The logic provided by this object is such that if requested and the number of lights 'on' is greater than a specified amount, it will turn all the lights off. If however the number of lights on is less than a specified amount, then it will turn the remaining lights on. Further presses cause all the lights to go on or off.

Light object

Light object now records in History file conditional information.

The light object has the ability to be overridden by the user. This overridden state may be optionally cleared on external light level change or time of day. A new option has been added which will clear the override state after the room has become unoccupied for a user-defined period. For example you request the lights to come on in a room during the morning (when it would normally be off) to do some specific work. You finish the task and leave the room for say two hours. The lights would have turned off automatically. When you return you want the lights to have reverted to normal operation i.e. not to come on. This option allows for this behaviour.

Two new light operating modes have been provided. The light may now operate simply on its timer. By using the Start timer connection the light will be triggered 'on' for the user-defined period. Typically behaviour might be where a room or cupboard does not have a motion sensor, and then a door opening will cause a light to come on for a defined period.

Closing the door could turn the light off straight away. Whilst this behaviour was possible by other means this new feature should make it simpler.

Also the light can now operate purely on the light level dark state. That is, when it's dark, turn on and when bright, turn off. Note: Unlike other modes, no protection is made to prevent oscillation if the light level is affected by the switched light source. Both new methods may be overridden as normal.

Note: The Light object now has a connection which represents the override state has been changed. One use is, if this is connected to an LED (of a button), then the LED may represent the Light has been overridden.

If the 'Operation mode' is set to 'Manual control only' and the option 'Enable this light to come on occasionally to make it look occupied' is checked then the Timer will become operational. This is used by the 'All light object' to turn the light on for a short period.

All Light object

The simulate occupancy option now only uses those lights enabled for occupancy simulation to decide how many lights are on in the property. This allows external lights (and others) to be on at night and not counted in the number of lights to illuminate option.

Intelligent Power Supply

Cortex 24 now provides improved support for systems with multiple power supplies. IPS history logging detail updated.

DFP (Display Function Panel) LCD

The control of the LCD display and button illumination has been updated. The background colour of the LCD display is now user selectable and dependant on what mode the LCD is in i.e. Idle, Edit mode, Displaying status etc. If required the display may remain the same colour all the time. The colour may also be different between day and night modes.

Dependant on mode of operation, the illumination levels for the buttons may be changed. For example when idle the background level for the buttons may be dim. However on user interaction or presenting some new information the brightness may be increased. Again these levels can be dependant on day and night modes.

The buttons of the DFP can now have background illumination of one colour and a status indication of another. For example the background light may be blue and a status indication of red.

Alarm object

The alarm object has been updated. Connections menus have been re-organised. The concept of a PET alarm has been added. An option is provided (user can enable) such that when arming the alarm, the option to put it into a Pet friendly mode is offered. This mode simply uses a different group of connections for its intruder trigger sources. Therefore when armed in Pet friendly mode only those trigger sources that would not be triggered by a Pet are used. This may include rooms that are excluded from a roaming Pet or simply just doors and window sensors.

Two new continuously protected groups of connections have been added. When the network is running but the alarm is not armed these connections are monitored continuously.

One group is referred to a Tamper (as with a conventional alarm). Any change of state on this input will trigger the alarm.

The second group is referred to as 'Boundary breach alarm'. In this group window sensors are included in the connection list. If a window is opened the state of the room is checked for occupancy. If the room is occupied the opening of the window is considered acceptable. If however the window is opened and the room is not occupied it is considered to have been opened from the outside and is considered un-acceptable and will sound the alarm.

Entering the security code may disarm the alarm. An alternative Danger code is provided. When this code is entered it will disarm the alarm as normal. However it will also trigger a special connection. The idea is that if you are forced to disarm your alarm under duress you enter the danger code. The alarm disarms as normal but secretly trigger perhaps an email, text message or a relay to a security centre to inform them of a problem.

Specific connections are now provided to trigger on different alarm conditions. These include...

- On Intruder detect.
- On panic
- On over temperature.
- On boundary breach.
- On tamper trigger.
- On danger code entered.



HVAC

The HVAC has a menu that provides default-heating operations considered the most common. This will be created by default as part of the HVAC object. Existing database users may create it from an option in the HVAC behaviour menu. A connection is made from the HVAC menu object start input to typically an MFP or DFP button 7, which allows it to be started.

A new feature has been added which provides a more comprehensive HVAC menu. It is invoked by making a connection from say button 7 of a DFP to a new HVAC input called 'Configure and Activate menu'. When this input is triggered the HVAC re-builds the HVAC menu. It then starts that menu. The menu is built dynamically so that only options that are relevant are presented e.g. an offer to turn it off is only made if it is currently on etc.

This dynamically created menu allows the DFP to control not only the HVAC in the room it is used but also any other HVAC in the property, including rooms, which have no MFP or DFP.

The Set-point edit menu is now followed by a comprehensive list of options. These include the ability to make the change only temporary (as before) but also permanent and apply it to different parts of the HVAC profiles. This provides a possible 41 different options.

The HVAC object now includes a new timer called 'Demand run on period'. When a demand is made this timer ensures a minimum request period. This is primarily to ensure the Boiler has a minimum 'on' period for efficient operation.

A DFP has the ability to present status lines of text when it is idle mode. The HVAC status line previously presented if the HVAC was on or off. This has now been revised and provides both measured temperature and set-point temperature. In addition a small icon shows if the heating is on, off, disabled or in Fallback.

The run on period of the HVAC can now be made different for Day and Night. The run on period is the time after which a room becomes un-occupied that the HVAC still uses the 'Occupied profile'. In general, a room that is visited and left frequently remains in its occupied profile. The splitting of Day and Night run on periods typically allows for at Night a much longer period to be used in bedrooms. This is because a room may be come unoccupied by timeouts during the night due to lack of motion detection as a person sleeps.

Features also included in Cortex 23 updated revisions....

xAP

The xAP objects have been updated and include the following:

Humidity object now supports xAP

Un-initialised values now reported as ?

xAP Control value connections filter fix.

Temperature, Light Level and Humidity support.

Alternative input source support.

xAP schema type now allows an alias.

xAP message can now tolerate EOL character at beginning of message.

xAP controller copy schema port check fix.

Scheduler can now trigger xAP controller inputs.
Scheduler can now trigger general logic object.
General logic object can now trigger xAP inputs.
xAP inputs can now optionally trigger message sending.
xAP copy schema option added.
All Display object now xAP enabled.
xAP xAPBSC.query, block name indifferent.
xAP address case insensitive.
Prevention of loops when using virtual Cortex objects.

Camera

Camera has option to maintain play speed e.g. 8 times when skipping scenes.

Cortex Software release 23.0.0

8th March 2008

Important Installation Notes

Since this a new major version release it cannot be installed as an update to V22. Instead V22 must first be uninstalled (using MS Windows Add/Remove Programs utility). Then V23 can be installed by running the appropriate install file.

Before uninstalling V22:

- 1. Back up your registration details and note down the Install Key Code since this will be required by the V23 installation process.*
- 2. Backup your database as a precautionary measure and identify it as used for Cortex 22 before installing Cortex 23.*

New Features

HVAC object



Now supports a supplementary heat or cooling source. A new connection is provided which allows connection to a zone valve, boiler, AC unit, or a HVAC Slave control. The logic is such that (optionally) if the outside temperature falls below a specified value then when the HVAC is demanding heat the supplementary heat source is also turned on to provide additional heat

Also the supplementary heat source may be optionally activated if the internal temperature falls at a rate greater than specified. This option is also subject to a user specified delay to prevent the supplementary heat source from being activated on temporary temperature drops i.e. door opening.

This feature is expected to be used where the property uses both electrical and wet radiator or under floor heating.

Slave Control



Now allows plan view icons of Heating coil or radiator.

All HVAC object



The all HVAC object manages properties relating to all HVACs. The Graph feature, which displays usage of the various heat zones in the property, has been removed. A new Energy Analysis feature replaces this.

Energy Analysis

A new feature has been added which may be accessed via the main menu 'View | History. Energy analysis'.

Its purpose is analysing the energy used for heating and lighting over a period of time. You can specify which objects to analyse and the report can be graphical or textual. The light object now allows you to specify the wattage rating of the light. Knowing how long the light is switched on and its wattage, allows its energy consumption to be estimated.

A feature is also provided which allows the user to estimate how much energy would have been used had the light been operated manually.

An analysis may also be made on the heating usage and is similar to that provided previously by the All HVAC object, which showed the amount of time the zones were heating over a defined period. Because the load represented by a zone may be defined in the zone valve object, the energy used may be estimated by its load and duration. The energy used by the various zones may therefore be compared.

An analysis may also be done on HVAC's and their zones. This is a more in depth analysis and requires that there is an outside temperature sensor defined and connected to the HVAC. This analysis will estimate the rate of heat lost and what heat is required to maintain higher or lower values than actually used.

The analysis is best performed over months or even a year. The analysis will take into account the operation of the HVAC i.e. if it was using fall-back, unoccupied temperature profile or occupied profiles (or switching between them). It will also use the actual room temperature, the outside temperature, the zone load and the amount of time the zone was on during the period.

The analysis can work out the duration and energy used from the actual historic data. But it can also work out what would have been the energy used if the heating had been using a fixed profile or a typical thermostat control.

The reality is that the actual usage will be a mixture of switching between occupied and unoccupied profiles. The Cortex control therefore may be compared with:

- Fixed thermostat. This is typically expected to be the worst-case energy usage.
- Occupied profile. A more sophisticated profile controller, but with no knowledge of occupancy.

An estimate is therefore made with what actually happened and compared to a less sophisticated fixed thermostat system or a fixed profile system. By specifying the cost of the energy used during the period, an estimate of the savings that Cortex Automation has provided can be calculated.

Light Objects

The light object provides a sophisticated means of controlling a light, based upon time of day, light level, occupancy etc. However a means of temporarily 'Overriding' the automation is required to allow the user to change the state of the light either keeping it on or keeping it off. This has been previously provided with a property called 'Override' which was either on or off. As the Light operation has become more sophisticated deciding what the 'Override' state should be doing has become more difficult.

To this end the core Light object code has been re-written and introduces some fundamental changes to the Override property. However the objective is still the same, which is to provide automation but allow the user to change the state in a logical heuristic manner.

If the user is upgrading from a previous version of Cortex then it may be necessary to change some of the properties introduced in the new Light object.

The override property can now have the following states:

- **None.** That is, the automation is as expected i.e. the current operating mode.
- **Temporary off.** The light remains off. Requests from other objects to turn on will be acted upon. If the normal operation would bring on the light then the override mode reverts to 'None'. If not, the override mode changes to temporary on (and optionally using presence).
- **Temporary on using presence.** The light comes on when the room is occupied.
- **Temporary on.** The light remains on regardless of the operating mode. Requests from other objects to turn off will be acted upon. If the normal operation would keep the light off then the override mode reverts to 'None'. If not, the override mode changes to temporary off.
- **Keep Off.** Will stay off even if other objects request the light to turn on i.e. menus, macros, Scheduler cannot turn the light on. That is until the override property is changed.
- **Keep On.** Will stay on even if other objects request the light to turn off i.e. menus, macros, Scheduler cannot turn the light off. That is until the override property is changed.

For most users the Override mode would not be explicitly set. The simple 'Toggle' input connection typically connected to a button will work out which override state is the most appropriate to use. For more sophisticated control the Override input is now a 'value input'. This means macro's, menus and scheduler etc may explicitly set the override mode for more advanced logic. The Turn On, Turn Off and Toggle connections may use the first 4 states of the override property. The 'Keep Off' and 'Keep On' are only used by explicitly setting the override mode. For example, a home cinema macro might set the light to 'Keep Off' to prevent normal automation or override reset or other panel buttons from bringing the lights on during viewing.

In previous versions the override state could be cleared automatically to revert the lights operation back to normal automation on outside light level change or time of day. However an option is now provided which declares a default value for the override mode (rather than just none). So for example a light may be fully automated but you consider that its normal operation is to be off. In which case a default of 'Temporary Off' would be used. If the user triggers the toggle connection, the light (if the conditions are satisfied) will turn on and the override state reverts to none. If not satisfied the override will be set to Temporary On.

The Operating mode (not to be confused with override mode) determines what parameters are used for the automation. In this version of Cortex other objects may now change it for specialist behaviour.

Note: If the users database has previously used the Lights Override connection then it will perform the same action as 'Toggle' for backwards compatibility. However the user may

want to review the use of the override connection for the Heuristic ‘Toggle’ connection now provided.

Also added to the new Light Logic is the ability to clear or set the Lights timer via connections. Note that the light will remain on if its operating mode conditions are satisfied even if the timer is cleared.

Room object



New advanced options have been added to the Room object. These should be considered experimental and not simply enabled without due consideration. The reason why a room becomes unoccupied is now tracked and shown in the behaviour menu. These states are defined as: 'Unknown', 'Door closure', 'Pulled out', 'Pulled in', 'Time out', 'House clear'.

If for example a room becomes unoccupied because of time-out, but then becomes occupied again due to a motion sense then it suggests the timeout is too short. An option is provided to automatically increment the timeout value. This does require good sensor coverage to work.

Another option is ‘Auto learn light timers’. This again uses the reason why the room becomes unoccupied. If a situation is detected where a light turns off prematurely and is re-instated (turned on) by the occupant moving then it is possible to automatically increase the lights ON timer value.

Note that if the behaviour of the room occupancy and light timer duration is found to be satisfactory then these options should be turned off.

Another option provided is called ‘Allow pull only on active sensor’. This requires the motion sensors in two adjacent rooms to be positioned to have certain characteristics. For example a hall has various rooms that it connects to. A study is off the hall. As traffic passes in the hall it may pull out the person from the study into the hall making it unoccupied. However the person in the study is motionless and therefore the condition of study unoccupied is false. Normally this should not matter provided a motion in the study is detected soon after and say the lights and other timers are of suitable duration.

With the option ‘Allow pull only on active sensor’ checked for the study, the logic is now modified such that motion in the Hall does not now pull someone out of the study unless the PIR in the study is active (red). The rationale being that for someone to be detected leaving the study and being picked up in the hall, would also mean that at the point the hall detection was triggered, the study PIR was still active. This is only suitable for use in rare cases and if falsely used will result in presence remaining in a room when no one is actually there.

The person count is now recorded in the history file.

Home Automation Networks



Support has been provided in previous versions of Cortex for the xPL protocol. The xPL protocol is an open source standardised method for home automation devices to communicate. The Cortex HA Network object allowed connection to an Ethernet for connection to an xPL hub and through which it could communicate to other xPL devices. The facilities provided by Cortex were to expose IDRATEK modules via their Cortex objects as if they were native xPL devices. This allowed for example a Cortex temperature

sensor to report its reading as if it were an xPL device. Only specific objects enabled by the user are exposed on the xPL network. No provision for Cortex to control xPL devices was provided as this was expected to be provided by xPLHal.

For further details see <http://wiki.xplproject.org.uk/index.php/FAQ>

In Cortex 23 support has been introduced for a similar protocol called xAP.

For further details see <http://www.xapautomation.org>

New options in the HA Network object allow connection to the xAP network. It is within the HA Network object that the feature is enabled.

Support has been provided for a number of Cortex objects to expose themselves as xAP devices and may send xAP messages or respond to xAP messages. In doing so they currently only support a schema call xAPBSC which is an xAP standard for basic status and control. As such the specification defines a device should report its state as 'On' or 'Off'. This has been used for an object such as a room to report occupied 'On' and unoccupied 'Off' or Door open 'On' and Door closed 'Off' etc Full details are provided in the help.

The Cortex objects exposed as xAP devices are:

House, Floor, Room, Door, Motion, Light, Light Level, Intercom, Boiler, Zone valve, Infrared receiver, Infrared transmitter, Buttons, Sound, (includes speech), Temperature, Telephone, Clock.

XAP control

A new object has been introduced called xAP Control. This is an object in which you can define new schemas and trigger them from normal connections within Cortex. For example, if you know the schema for an xAP light device outside Cortex, you can turn it on or off by pressing a button on an IDRATEK module. You would do this by creating a schema in the xAP control device with a name-value pair. Each name value pair has a Cortex connection port. The button object would be connected to the new xAP control port. Therefore pressing the button sends a message to the xAP control object, which in turn triggers the sending of the pre-defined schema out to the xAP network.

Similarly xAP devices, which send out packets, can be received by Cortex. The xAP control object can match incoming messages with pre-defined schemas within itself. On detection these can trigger an output port of the xAP control object.

The xAP control object can hold many schemas, alternatively multiple xAP control objects may be defined depending on how the user wants to organise the control.


Infrared Receiver

The behaviour menu for the infrared receiver has been re-designed to bring together the fixed and extended Infrared frame detection. The fixed infrared frame detection is performed in a module with the resulting frame number being reported to the network. If a module does not recognise a frame it can relay it to the network. Cortex reads the frame and compares it with possible frames stored within the Infrared object in Cortex.

Additionally each frame, fixed or extended, can now have a handset and button function name allocated to each frame. Apart from allowing the user to remember which frames were learnt against which infrared remote control button, the new 'All infrared object' also uses it. Its intention is that the new style behaviour menu makes it much easier to use extended frames.



All infrared object

A new object has been added which may be accessed from an Infrared Receiver behaviour menu or an Infrared Handset (transmitter) behaviour menu. This feature may be accessed using the  button. Its purpose is to allow infrared remote control data to be collected, saved, retrieved and downloaded as required. Therefore previously learnt data may be arbitrarily downloaded into any IR capable modules.



Buttons LED's

The LED on a button may be controlled automatically (default) or by explicitly turning it on or off by making suitable connections. A new connection to the LED is now provided which is called 'Follow state'. If suitably connected the LED will follow the state of the connected object. For example the heating LED on a panel may be connected to the output of a zone valve. The LED will then turn on when the valve is open and heating is being applied.

To gain manual control over the LED the LED representation in the behaviour menu is clicked to reveal a yellow circle around it. Clicking a second time provides access to a new option, indicated by a blue circle. Here the LED is still under manual control but the logic is inverted. Therefore turning it on will mean it will be off and vice-versa. An example of use is for the Light LED on a panel to be under manual inverted control (blue circle). The LED is then connected to follow the state of a Light object. When the Light is on the LED will be off. When the Light is off the LED will be on allowing the button, which controls the light, to be seen in the dark.



Single Light Dimmer

Now has connections which are triggered on a failure. This includes Mains failure/Bulb fault and Over current detection.



Quad Light Dimmer

The Objects structure has changed. ***QLD objects in Cortex V22 databases must be removed and re-instated after upgrade to V23.*** The button sub objects have now been removed and replaced by a switch input object. This allows greater flexibility in the digital inputs usage.



All Light object

One of the functions of the 'All Light object' is occupancy simulation. This can be configured to operate during a specific time of day. Now it optionally allows a Light Level sensor to be connected so that the operating period also requires it to be dark before the simulation process becomes active.



Now has option not to report a missed call.

Analogue Signal

New equations provided for signal input object as used for Analogue input object. Allows for Thermistor equations.

On Off Load object

The on off load object connections previously described as 'Enable' and 'Disable' have been renamed 'Turn On' and 'Turn Off' to better reflect their operation.

Humidity object

Now records in the History file the 'Dew point' on each measurement. The Dew point may now be graphically plotted. A new user-defined threshold has been added which monitors the rate of change of humidity. This can be used for example to trigger a fan on a high rate of Humidity change rather than absolute levels.

Temperature Sensor

The temperature sensor now has the ability to have an offset applied to its normal measured reading to correct for its placement affecting the readings. A temperature sensor may be placed in such a position in a room e.g. the ceiling such that it typically reads higher than where the user experiences the air temperature. As an approximation, a fixed offset may be added (typically negative value) such that the reported temperature is less than measured. It is this corrected value which is logged in the history file, reported in graphs and used by other objects e.g. HVAC.

Another compensation may be applied which takes into account the heating or cooling affect of a wall on which the module/panel is mounted. A temperature reading module should ideally be placed on an inside wall. If however it is placed on an outside wall and that wall is poorly insulated, the sensor may on a cold day be reading less than the average air temperature in the room. The cooling affect on the sensor will be influenced by the temperature gradient between inside and outside temperatures, the thermal conductivity of the wall and some time constant for how long it takes the wall to warm up or cool down.

An option is provided to try and take into account this effect. It requires the declaration of another temperature sensor, a user defined gain factor and a filter time constant. A filter time constant is required to smooth any changes in the outside temperature, which will take many hours to propagate through the wall.

It must be strongly stressed that this feature is not a substitute for a correctly placed sensor and should only be used in unavoidable situations and with caution, as the readings could easily be worse than those actually measured if unsuitable values are chosen.

Pulse Count object

Now supports plan view icons of anemometer (wind speed).

Floor object

The floor object monitors the rooms it owns to determine if the floor is occupied. As people move around, motion sensors are triggered and rooms change status. It is possible for all the rooms on a floor to become unoccupied for a short period i.e. after a door is closed until the logic receives new input. For this reason a filter has been introduced which prevents the floor from reporting unoccupied until after a specified time. A default value of one minute has been chosen. This feature should be of no consequence unless actions are being triggered on a floor being un-occupied.

History plan view playback added.

A new option is provided which allows the playback of the History from a Plan view perspective. This option can be found via the main menu under ‘View | History. Plan view play back’.

This option can only be used when the network is stopped. When activated a menu will be placed to the left of the plan view. It will list the history for a particular day chronologically. Note: It may take some time to populate the list.

If you click on a particular timed event in the list, the plan view icons (and adjacent text where applicable) will be updated (from the beginning of the day) to reflect the state of all the plan view objects at that time. You can then step through the history events and watch how the activity changes. It allows you to watch as doors are opened, motion detectors trigger presence in rooms and lights are switched on/off etc. This feature may be found useful for analysis where a particular behaviour was not as expected.

User defined Menu

The ability to move a menu line up or down the tree is now provided by right clicking on the menu item and selecting the option in the popup menu.

Motor Control Object

The motor control object now allows separate time periods for a device to open and close. This is provided where the operating time due to the motor load in one direction is different to the other e.g. opening or closing a roof window requires more effort to open and is therefore slower than to close.

A fourth motor type has been introduced. This is described as ‘Pulse relay 1 to close, pulse relay 2 to open’. One example of its use is where a curtain is opened by the press of a button and closed by the press of another. The relays are wired in to simulate the button presses.

Verification object

Typically used among other things to verify the operation of motorised values. This object now records failures detected in the History file for more detailed examination.

Telephone



It is possible to dial into Cortex via the telephone object and control other objects via a touch-tone menu. The menu items were spoken and the user selected an option by pressing a key on the telephone keypad. This menu was dynamically created based upon the full structure of the database. It was therefore necessary to navigate down the structure to the object of interest. This could require many levels to be selected before the required item was found.

It has been decided that for most practical use, the user typically wants to set only a few options. For this reason the automatically generated structure for the 'Voice prompted menu' has been removed. In its place is 'User defined menu' which is automatically created and called 'Telephone spoken prompt menu'. It may be found under PC | Communications | Telephone spoken prompt menu in the structure view. This menu has a default button, which will generate what is considered the most likely features that the average user will want to access via the telephone. The user may create their own menu items or augment the default menu.

The voice prompt menu guide is now shown in the Cortex control tab.

General Logic



The general logic object is a way of manipulating behaviour between objects by interceding logic gates into connections between objects to provide more sophisticated logic than generally provided by normal connections. Two new gates have been introduced. These are

- **SR Flip-Flop.** This is a Set/Reset device whose output is maintained. It can be used to remember a fleeting event and hold it for examination at a later time.
- **Monstable.** This is a logic gate whose output is set for a specified period of time and then resets automatically. It may also be reset prematurely by triggering its reset input.

Intelligent Power Supply



It may be possible under some rare occurrences that a module on the network locks up. Re-powering the device or using its reset button may resolve this. For example the lockup may be caused by excessive electrical interference or a module fault. In which case the cause should be identified and rectified e.g. ensure adequate spacing between module low voltage electronics and switched load wiring, connect suppressor components to the load, replace faulty module, etc.

Nonetheless since such an event could potentially occur when the system is unattended an intelligent failsafe feature has been introduced. If when Cortex is controlling the network it fails to communicate with a module it will 'Fault' the node(s). The IPS can be informed of such activity. The IPS can optionally be made to determine which power spur the module is connected to and re-power that spur to initiate a reset. The IPS can warn the user (via spoken message to a panel) that such an imminent re-powering is about to take place so that the user may cancel the operation if it is not desirable. A time limit is provided to determine how frequent such an operation may occur.

This should be considered a feature of last resort and not a means of ignoring any underlying problems.

Web Server

The Cortex Web Server now allows (Secure Socket Layer) SSL to be used to encrypt messages between the server and browser. An example is provided on how to generate your own certificates using a free program called SSLBuddy provided by aranatech.com.

Cortex Software release 22.0.0

23rd August 2007

Important Installation Notes

Since this a new major version release it cannot be installed as an update to V21. Instead V21 must first be uninstalled (using MS Windows Add/Remove Programs utility). Then V22 can be installed by running the appropriate install file.

Before uninstalling V21:

- 1. Back up your registration details and note down the Install Key Code since this will be required by the V22 installation process.*
- 2. Backup your database as a precautionary measure and identify it as used for Cortex 21 before installing Cortex 22.*

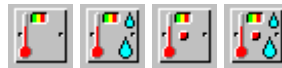
New Features

Intelligent Power Supply







Now has the ability, when a spur is shut down, to Fault the nodes that are connected to that spur. This prevents Cortex from trying to communicate with those devices until it has seen them power-up again. It can also attempt to reset a spur, which has tripped due to a fault e.g. over-current. Which nodes are powered from which spur, may now be found automatically at the request of the user via the behaviour menu. This does require a temporary power down of the spur.

Support for new sensor module variants.



Four new variants of the small modules have been introduced.

- **LTS**  Light Level, Temperature and 2x digital inputs.
- **LTH**  Light Level, Temperature, Relative Humidity and 2x digital inputs .
- **PLT**  PIR, Light Level and Temperature and 2x digital inputs.
- **PLH**  PIR, Light Level, Temperature, Relative Humidity and 2x digital inputs.

History graphical analysis tool



This allows two or more sources of data to be viewed on the same graph using, if required, separate x and y scales. Additionally the graph may be printed. For example the temperature profile of a sensor may be plotted for two separate date periods e.g. comparing two months of outside temperature on the same graph.

House, Floor

A separate connection trigger is now available for when the house or floor becomes occupied.

Room

A separate connection trigger is now available which becomes true on delayed occupied. This means that only after a user-defined period does this state become true. It may be used for example to only enable the occupied heating profile if the room is occupied for more than say 5 minutes.

Proximity detector

Logic is now provided for a proximity detector. Such a detector would register the presence of a person passing through an open door or an opening to a room. A digital input would typically receive a logic low on presence detect. The output connection of the proximity detector object is passed to the door object it is associated with. When proximity is detected it has the effect of clearing out presence in the rooms either side of the door. The system then relies upon motion detection in the appropriate room to locate where the person is now located.

Motor control and motor relay

A new logic object is available which provides automation for motors. Its primary use is for controlling motorised blinds, windows, projector screens etc. It allows optionally limit switches (digital inputs) to be used to control the motor or provide warnings of mechanical failure. It provides automatic default position (e.g. closure) by monitoring other objects e.g. House occupied, Rain sensor etc to close windows. Three different motor configurations (wiring types) are provided for. A buttons Led will reflect the active state of the motor. MFPs and DFPs also allow an annunciation report of the motor's activity. If attempts to operate the motors via a button cannot be fulfilled (because default actions are overriding the request) the button's LED will flash for 3 seconds.

Layers

This is a new system associated with making objects visible in plan view. Objects may now be associated with one or more layers. By turning on/off various layers the complexity of plan view may be increased or reduced. It may be of use during training or to simplify the view for normal users.

Light Synchronise

In general a light object can be automated to use light level and presence etc. The light level can be measured using a sensor located in the same room as the light being controlled (e.g. on an MFP), such that if the room is considered dark then the light can turn on. Once the light is turned on the light logic ignores the light level else the light could then be turned off because the room is no longer dark.

However, a room may have multiple lights individually controlled by light objects. If one light turns on, the light level in the room may increase sufficiently to prevent the other lights turning on (or the reverse).

For example, the user may have configured a button to toggle all the lights in a room with the expectation that if pressed, all the lights would come on (or all go off). However if the above scenario had happened i.e. one of the lights had initially come on for other reasons, then pressing the button would cause the first light to go off and the reset to come on.

The light synchronise facility provides a means of ensuring a group of lights can be made to come on or go off together from a single action despite other automation conditions affecting each of the lights differently. The logic consists of two new connections called 'Toggle and synchronise' and 'Synchronise (Output)'. The Toggle and Synchronise input of the first light is connected to the button. The Synchronise (Output) of the first light is connected to the 'Toggle and synchronise' input of the second. The Synchronise (Output) of the second light is connected to the 'Toggle and synchronise' input of the third etc. When the button is pressed the first light toggles. The second light receives a message from the first and sets itself to the state of the first light. The third light receives a message from the second and sets itself to the state for the second. Any number of individually controlled lights may be associated in this way. It is important to understand that this mechanism does not prevent the lights from being individually controlled but simply provides a way of allowing a single user action to set all the associated lights to be set to on or off.

Priority system introduced

This is internal to Cortex where packets to be sent to the network are queued for transmission. Now packets are assigned a priority depending on which object they originate from. This means packets to turn on/off lights have a higher priority than say turning the LED on/off on a panel. The result should be an improved user response to actions i.e. button presses. No action is required by the user.

DFP

On particular DFP types, the colour of the display can now be controlled. Also the Display and button lighting (and colour) can be automated depending on time of day, button press and other object state e.g. presence.

The DFP has buttons 11 and 12 labelled as a cross and tick. These are now used as the main Cancel and OK functions like buttons 1 and 2 on a MFP.

DFP Type 2

Provisional Support added for the new Horizontal style Display Function Panel (DFP) Type 2.

SLD

The Idratek Single Light Dimmer (SLD) has two buttons, which can turn on/off or dim the light using internal reflex actions. Cortex will now enable, as a matter of course, this internal mode when the network is shut down.

QLD

Provisional Support added for the new Quad Light Dimmer has been added.

Database save failure

If an error should occur during the saving of a database then previously a backup version was available (but not obvious to users). Improvements have been made but should an error occur, then on restarting Cortex, the previous state of the database will be restored.

Database analysis tool

A database analysis tool has been added to give users indications of possible errors in the way the database has been prepared. For example rooms are connected together using doors and stairs. However if not all the doors have been connected correctly it may be possible for a room or group of rooms to be isolated and give false presence as the person has no way to escape the house!

Camera

The amount of disk space used by a camera for its image files may now be automatically limited. Automated warnings to the user can be generated when disk usage reaches a specified amount. Also optionally old files may be deleted automatically on a daily basis to maintain a maximum disk space allowance for each camera. Also should the free disk space reach a user specified low, the camera will stop generating files to preserve system integrity.



The Camera object can now support the use of IP Cameras as its source (as compared to a video capture device as part of Windows). An IP camera is a network device with its own IP address. Although independent of a PC an IP Camera can typically stream data to PC disk and have its own built in web server.

Provided the IP Camera supports mjpeg streaming it should be possible for Cortex to use it as a source and perform all the features currently available to video capture devices. It also provide for automatic username and password logon where required. It therefore allows the Idratek network to integrate and provide triggers to and from such IP cameras.

The Camera image may now be rotated and/or mirrored. This however only works on digital video streams i.e. USB and IP Cameras.

User Defined Menu

You can now copy a branch of a menu tree and paste it to another part of the tree or another User Defined Menu. You can now move, using drag and drop, from one UDM to another.

Right clicking on a branch of the UDM tree can access these new features. An HVAC  object and Skype  object now have sub user defined menus created automatically to help the user prepare the database. They may be edited as required.

Button Menus

A single common style menu for all 1,2,4,8 & 12 button objects has now replaced the original separate Button Behaviour menus.

Node Firmware Loader

The Node Firmware Loader within Cortex performs the firmware update of modules. Certain firmware updates may now be downloaded from Idratek as part of the Cortex update process. The firmware loader now has the ability to scan the IDRANet and decide if appropriate firmware updates are available for modules used. The user is then offered the option to update the specified node.

Adding New objects

A new menu has been introduced when adding new objects to the database. This menu allows (where appropriate) that sub-objects can be made visible in a different location to where the module resides. For example a Din-rail Relay module may be located in a cupboard in a hall. However the Light objects associated with the relays would be placed in the appropriate rooms they control. This new menu allows such changes to be made at add time rather than visiting appropriate pop-up menus later (which can still be done). Also more connections between objects are made automatically at add time. These changes are designed to speed up the commissioning process for installers.

Alarm Object.

Now has an option in connections menu to create default connections automatically. It would typically be used at the end of the commissioning process after all the modules are in place on the network. The connections made should be checked for accuracy as certain assumptions are made (i.e. external doors have been defined as such).

On/Off load object

Now has a new toggle connection that can Pulse the load on for a specified duration and on second request set it to off (if it has not already timed out). Similarly another connection can Pulse the load off for a specified duration and on second request set it to on. An example of its use would be a button allocated to control a fan. If the fan were off, on button press, the fan would come on for 'n' minutes. However if the user pressed the button whilst it was currently on it would turn it off.


Defibrillator

This is a new separate program to Cortex. Its purpose is to serve as a watchdog to Cortex. When Cortex is running IDRANet it will send messages regularly to Defibrillator. If Defibrillator detects Cortex has stopped sending messages for more than 5 minutes it will reboot the PC. Either Defibrillator or Cortex may be started first. Typically Defibrillator will be placed in the start up directory for windows (as is Cortex). It resides in the system tray.

The rationale behind this watchdog is that Cortex interacts with many third party programs and Windows drivers. Should any of these fail it is possible to bring down Cortex. By using Defibrillator it will restore Cortex and any third party programs and drivers.

It should be stressed that a watchdog is a last resort safety measure and that it is always recommended that any triggering of the watchdog should be investigated and attempts made to discover and remove the root cause.

Defibrillator uses the new Cortex API. As part of the new Cortex 22 install, a link will automatically be placed in the start-up directory (as will a link to automatically start Cortex).

The world object  will log in its History if Cortex was abnormally terminated i.e. PC Rebooted whilst Cortex was running network or use of Task Manager termination.

Cortex API

The interface that 'Defibrillator' uses is exposed in the form of a Cortex API (Application Programmers Interface). Multiple third part programs can currently use the interface. It allows a third party program to interact with Cortex and provide functions not currently supported by Cortex. Specific objects may be exposed to the interface allowing messages to and from those objects to be used. See Cortex Help (How to ? advanced topics | Application Programmers Interface | Cortex API for detailed explanation.

Sample programs written in Borland Turbo Delphi, Microsoft C# 2005 Express Edition and Microsoft Basic 2005 Express edition, are provided to illustrate the interface. These can be found in the Cortex subdirectory 'Cortex API'.