EST3 Series
Student Workbook

P/N 3100041 • Rev 6.0 • 25FEB11

DOCUMENT HISTORY

Date	Revision	Reason for change
7JAN00	1.0	Initial release
12MAR01	2.0	Upgrade to support 3SDU release 2.0 and 3.0.
30JAN08	3.0	Dropped Installation part of course. Upgrade to support 3SDU release 3.61 or greater.
30DEC10	4.0	Updated branding to reflect UTC Fire & Security.
11FEB11	5.0	Modified materials to include EST3X panel.
25FEB11	6.0	Incorporate pilot course feedback.

Table of Contents

Section 1
Agenda

Introduction

This section contains the course agenda for the five day EST3 Series Technician Certification course.

UTC Fire & Security

A United Technologies Company

EST3 Series Technician Certification Agenda

Day 1:

Introduction
Review Logical Addressing
Review Front Panel
Operate Front Panel – LAB
Develop Labels and a Labeling Plan - LAB
Access the 3-SDU Help Utility
Introduce System Definition Utility (SDU)

Day 2:

Customize the 3-SDU - LAB
Create a New Project and Configuring Project – LAB
Configure System Cabinets – LAB
Configure Local Rail Modules (LRMs) – LAB
Configure Time Controls – LAB
Configure and Label Non-SIGA Devices – LAB
Configure and Label Main Bldg SIGA Devices – LAB

- Enter Devices By-Zone and In-Mass
- Capture Serial Numbers – Scanning Method
- Upload Actual Data
- Reconcile Actual vs. Expected Data
- Download Configured SIGA Data
- Utilize SIGA Status feature

Configure and Label Maintenance Building SIGA Devices - LAB

- Enter Devices In-Mass
- Capture Serial Numbers – Mapping Method
- Upload Actual Data
- Reconcile Actual vs. Expected Data
- Download Configured SIGA Data
- Utilize SIGA Status feature

Day 3:

Scan Serial and Number Barcodes – LAB
Reconcile Actual and Expected Databases – LAB
SIGA Convert and Download to 3-SSDC1s
DB Convert and Download to CPUs – LAB
Configure Logical Groups - LAB
Program Rules – LAB

Day 4:

Program Rules – LAB
Configure 3-ASU and Generate Audio– LAB
Download to the 3-ASU – LAB
Evaluate Project and Grade Practical

Day 5:

Administer Final Exam
Review Final Exam
Award Classroom Certification

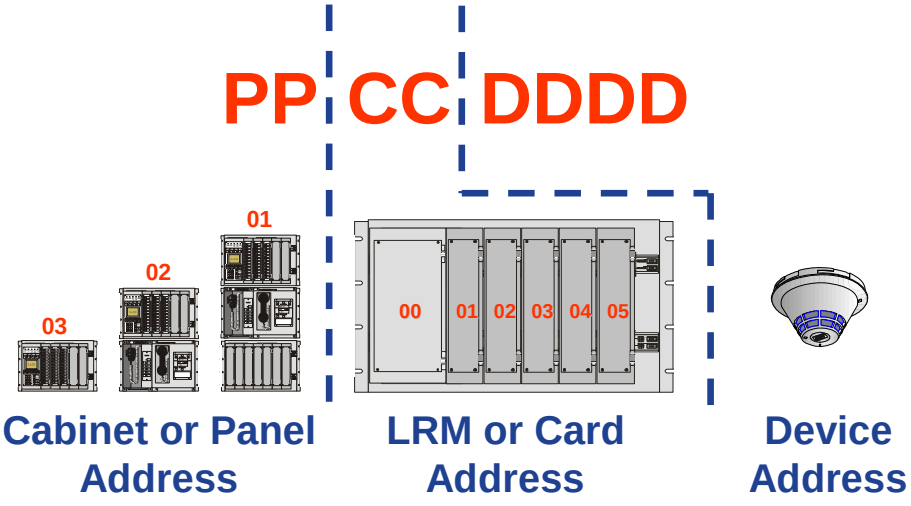
EST3 Series Technician Certification Course

Logical Addressing

UTC Fire & Security
A United Technologies Company

Logical Addressing

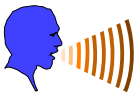
- **A Logical Address consists of up to eight digits**



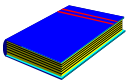
Device addresses

Device

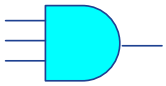
Audio Message



Text Message



Logical Group

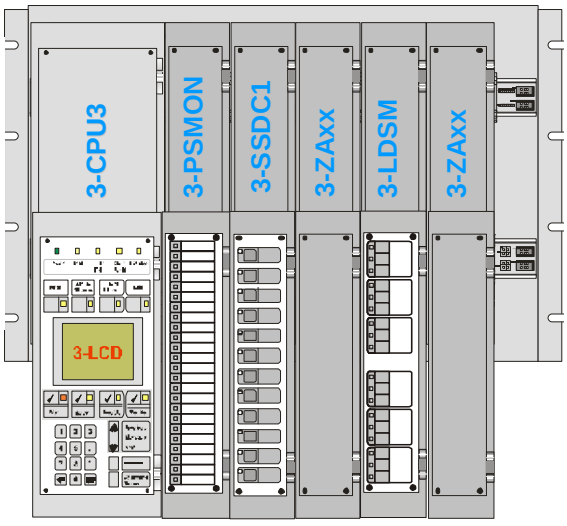


Pseudo Point

LRM Addressing

Hardware Layer Logical Addressing → **00 01 02 03 04 05**

Each Operator Layer Module's address is determined by adding 32 to the host LRM address



Operator Layer Logical Addressing → **32 33 34 35 36 37**

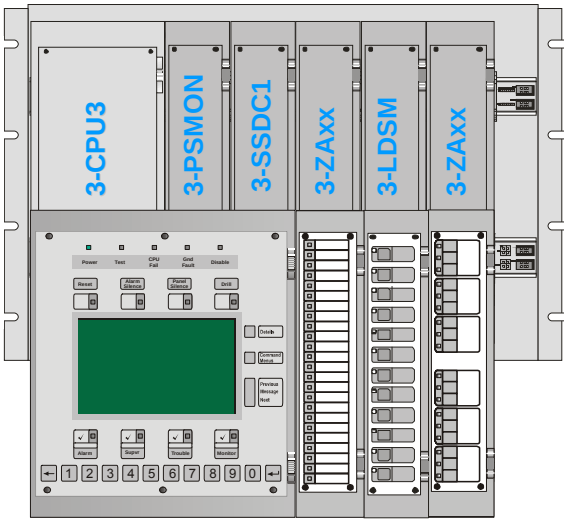
Note:
Each LRM has a logical location (address and a physical slot location)

The 3-PSMON in this chassis is address 01 and located in slot position 03

LRM Addressing with 3-LCDXL

Hardware Layer Logical Addressing → 00 01 02 03 04 05

Each Operator Layer Module's address is determined by adding 32 to the host LRM address

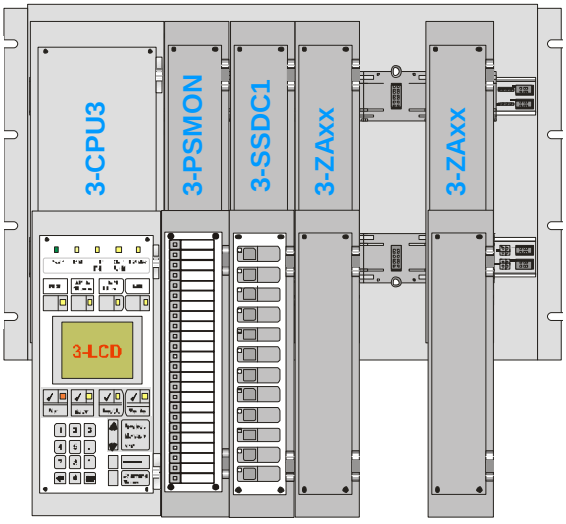


Operator Layer Logical Addressing → 32 35 36 37

LRM Addressing w/ empty slot

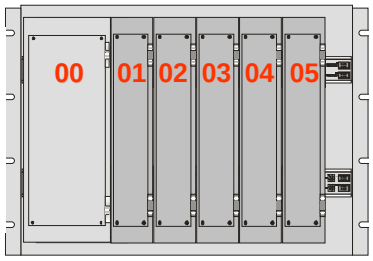
Hardware Layer Logical Addressing → 00 01 02 03 04 05

Note: Logical addressing is tied to slot locations regardless of an LRM

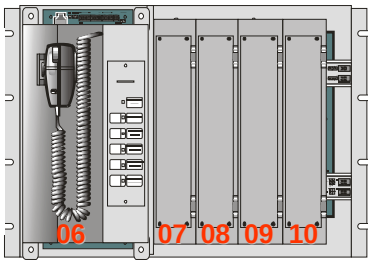


Operator Layer Logical Addressing → 32 33 34 35 36 37

3-ASU/3-CHAS4



3-CHAS7



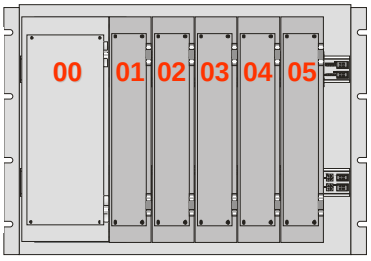
3-ASU/3-CHAS4

3-ASU/3-CHAS4 chassis consists of: an ASU assembly mounted on first footprint of back of chassis, an ASU cover panel and a four slot rail assembly for optional LRMs

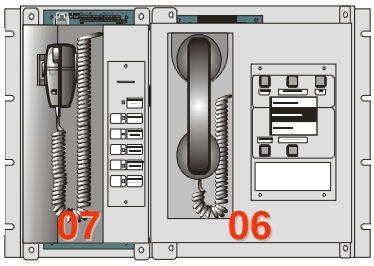
3-ASU/3-CHAS4 Chassis addressing is assigned from left-to-right

Note: 3-CPU3 and SDU see each chassis or remote annunciator as seven slot locations. The ASU uses three slot positions. ASU with address of 06 is located in slot 08 and the LRM 07 is located in slot 11.

3-ASU/FT



3-CHAS7



3-ASU/FT

3-ASU/FT chassis consists of: an ASU assembly mounted on first footprint of back of chassis, an ASU cover panel, 3-FTCU assembly mounted on second footprint and FTCU cover panel

3-ASU/FT chassis addressing is assigned from right to left. This rail has been replaced with ribbon cables. The 3-CPU3 electrically sees 3-FTCU first and then the 3-ASU.

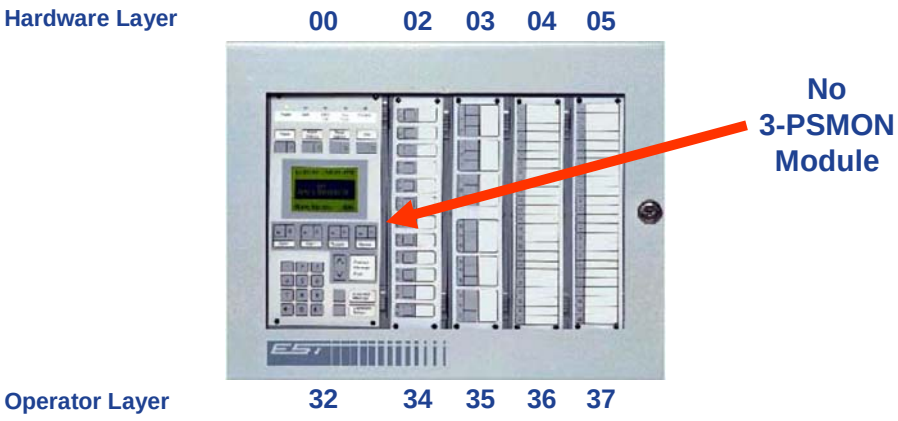
In this case, ASU 07 is considered to be located in slot 08 and FTCU 06 is considered to be located in slot 11

Logical Addressing

- **Each logical and physical object (device) has an assigned address**
- **Preconfigured STARTUP system event**
  - Active object when system is powered up
- **Eight integer address for this object is: 0N 00 0001**
- **Addressing and slot locations become complex**
- **Program system using the System Definition Utility (SDU)**
- **3-SDU is development tool assigning and tracking addressing**
- **Each object is assigned a naming convention called a LABEL**

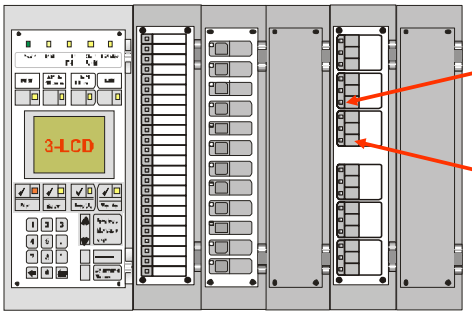
Additional addressing example

- **3-6ANN Remote annunciator includes hardware and operator layer**
- **Applications software sees as seven slot enclosure**
- **Enclosure does not have power supply**
- **Power distribution via daughter board on 3-ANNCPU module**
- **Software configures 3-6ANN without slot 3, making address scheme**



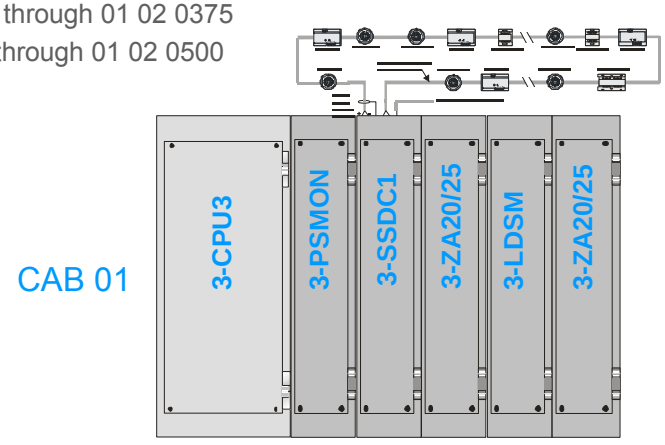
Additional addressing example

- **Device addresses for operator panel's LEDs starts at 0129**
- **Device addresses for operator panel's switches starts at 0001**
- **If this is cabinet 1, what's the eight integer address for 6th LED on 3x3x6 Control/LED display?**
- **What's the eight integer address for 9th switch on 3x3x6 Control/LED display?**



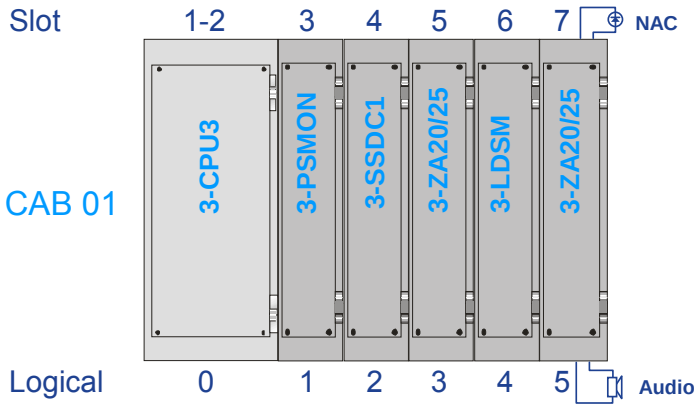
Signature addressing example

- **Address range for SIGA detectors on 3-SSDC or 3-SSDC1 SIGA data riser shown below is 01 02 0001 through 01 02 0125**
- **Address range for SIGA modules on 3-SSDC or 3-SSDC1 SIGA data riser shown below is 01 02 0126 through 01 02 0250**
- **If this were 3-SDDC or 3-SDDC1, address ranges for second loop would be:**
  - SIGA detectors 01 02 0251 through 01 02 0375
  - SIGA modules 01 02 0376 through 01 02 0500



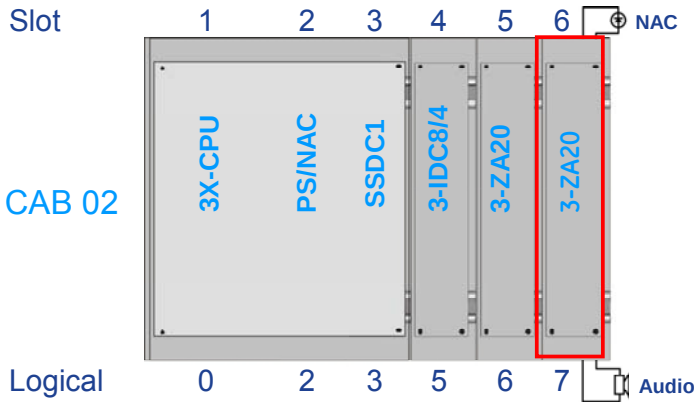
Amplifier addressing example

- **NAC Output of this 3-ZA amplifier is 01 05 0001**
- **Audio Output of this 3 ZA20/25 amplifier is 01 05 0000**
- **Device address of audio out is the amplifier**



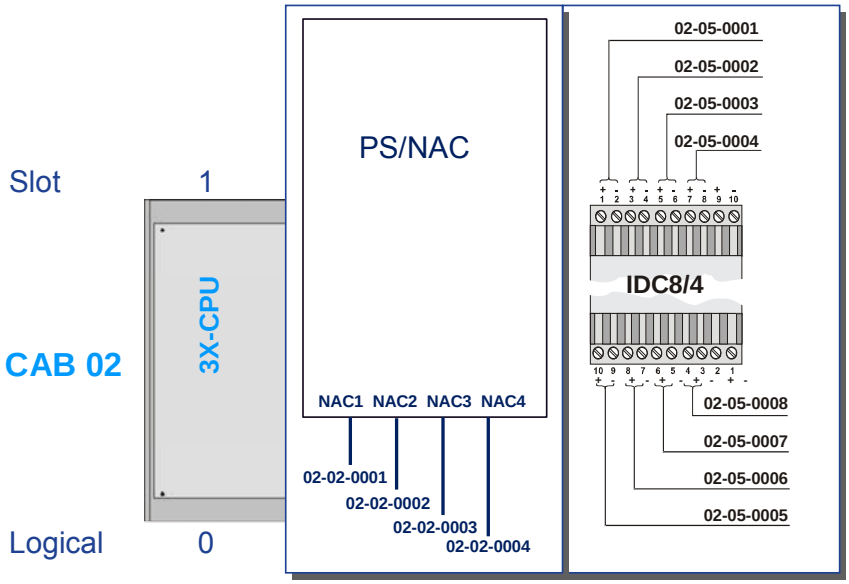
Amplifier addressing example

- **NAC Output of this 3-ZA amplifier is 02 07 0001**
- **Audio Output of this 3 ZA20/25 amplifier is 02 07 0000**
- **Device address of audio out is the amplifier**



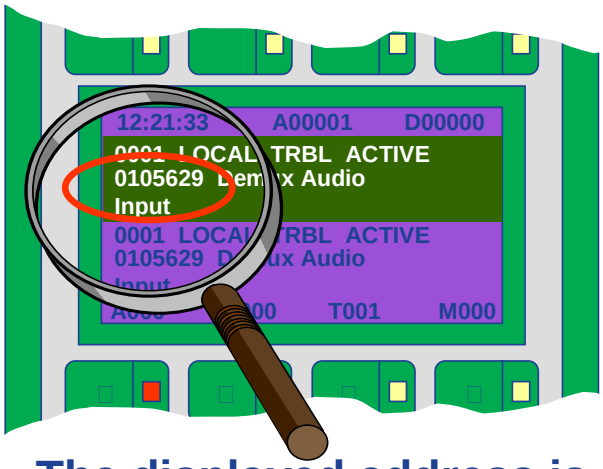
IDC8/4 addressing

- IDC8/4 LRM is installed in slot 4 of cabinet 2
- PS/NAC is installed in slot 3 of cabinet 1, NAC1 is address one of that card. The power supply is mounted to the chassis.



Displayed addressing

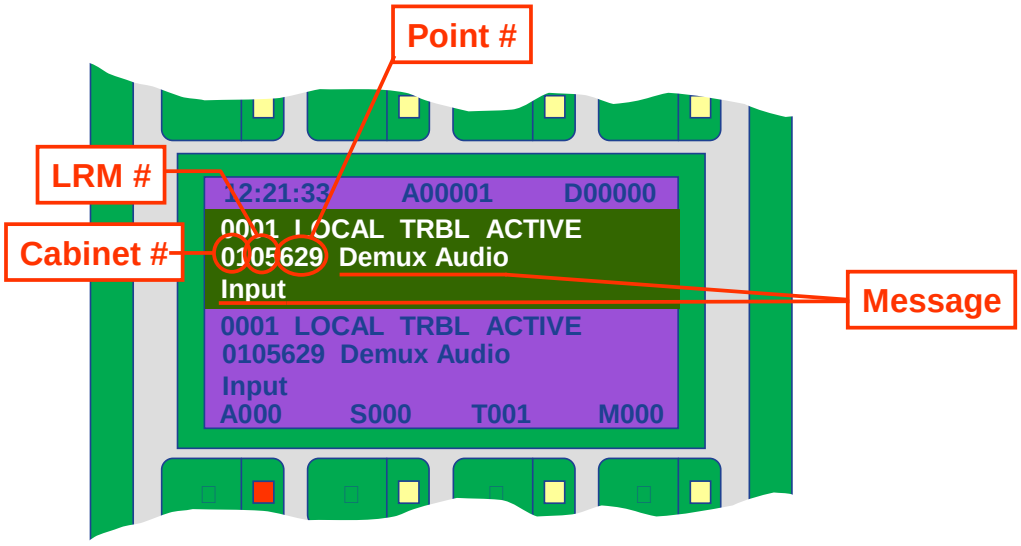
Display when Demux Audio Input failure occurs for an amplifier is shown



The displayed address is
0105629

Displayed addressing

Display when Demux Audio Input failure occurs for an amplifier is shown

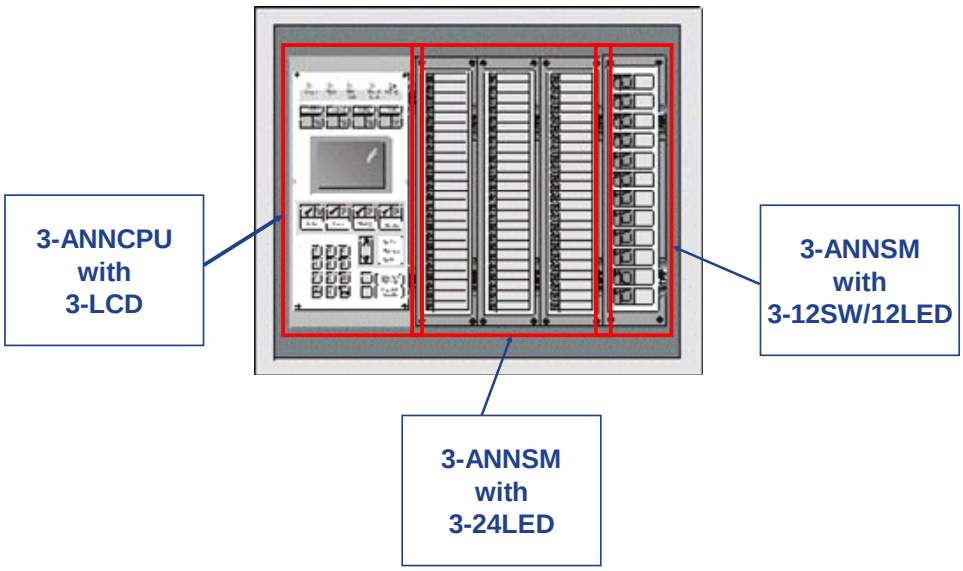


EST3 Series Technician Certification

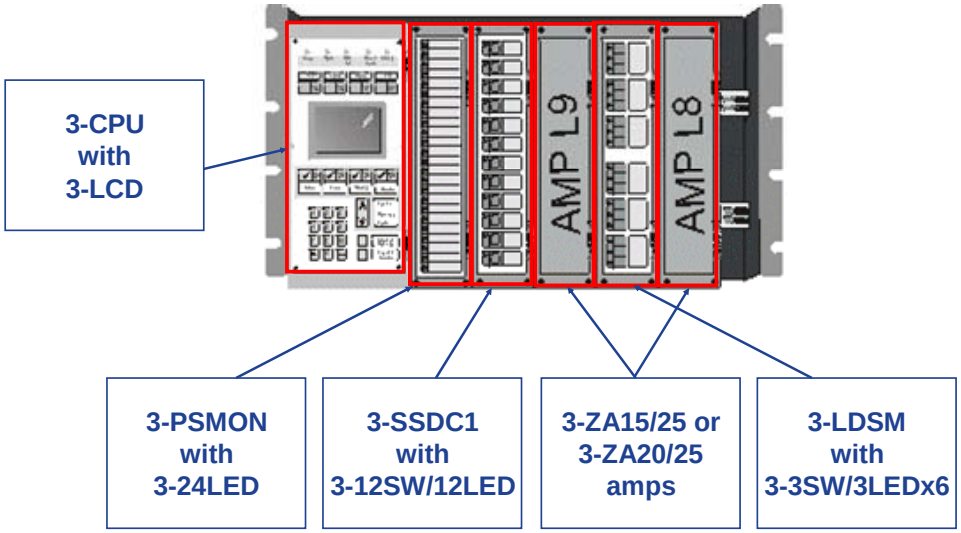
Lab Station Configuration

UTC Fire & Security
A United Technologies Company

CAB #3 (panel 03) - 6-ANN Remote Annunciator Security Desk

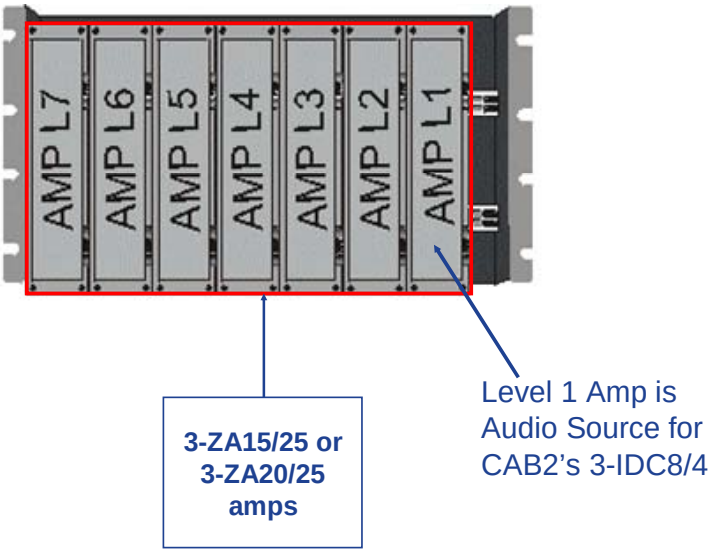


CAB #1 (panel 01)
Top Chassis of the 3-CAB21

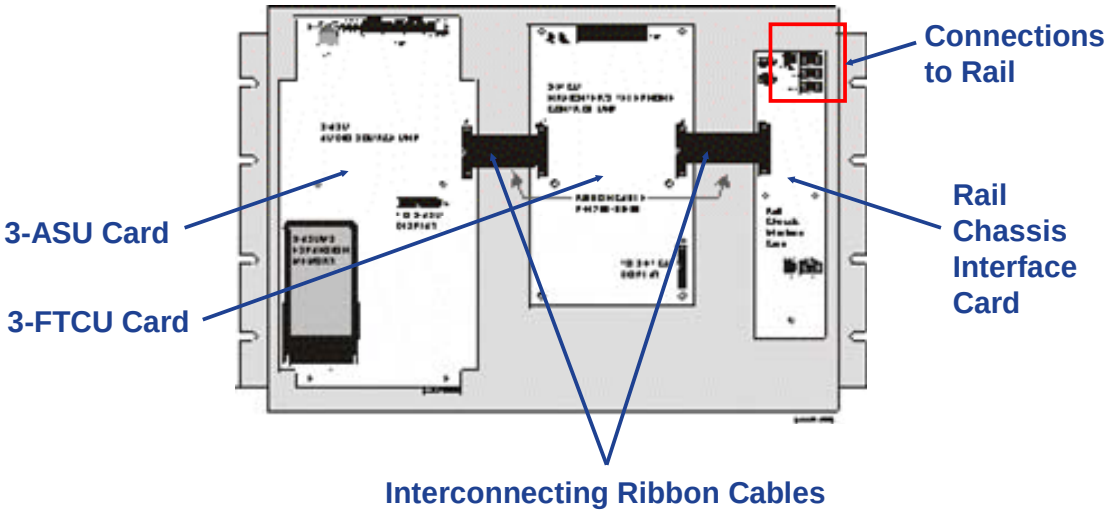


Check amplifier types in your station

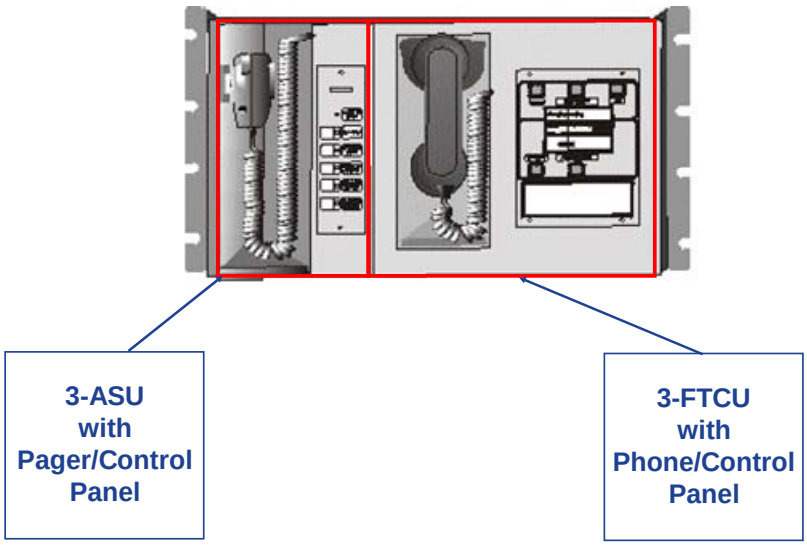
CAB #1 - Middle Chassis
of the 3-CAB21



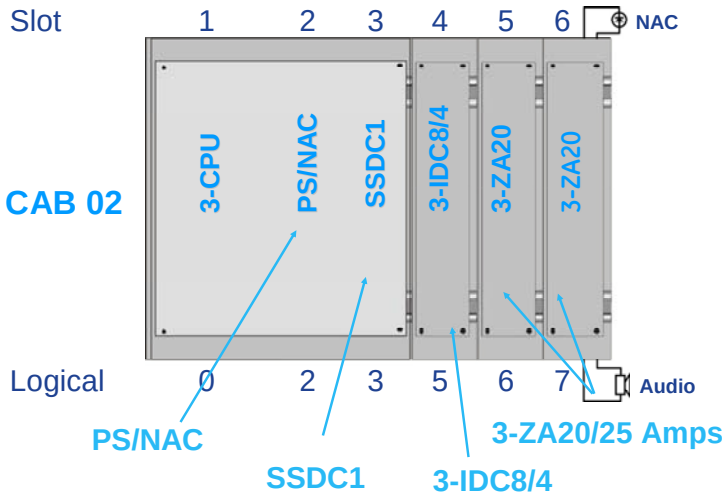
CAB #1 – Bottom chassis backplane of the 3-CAB21



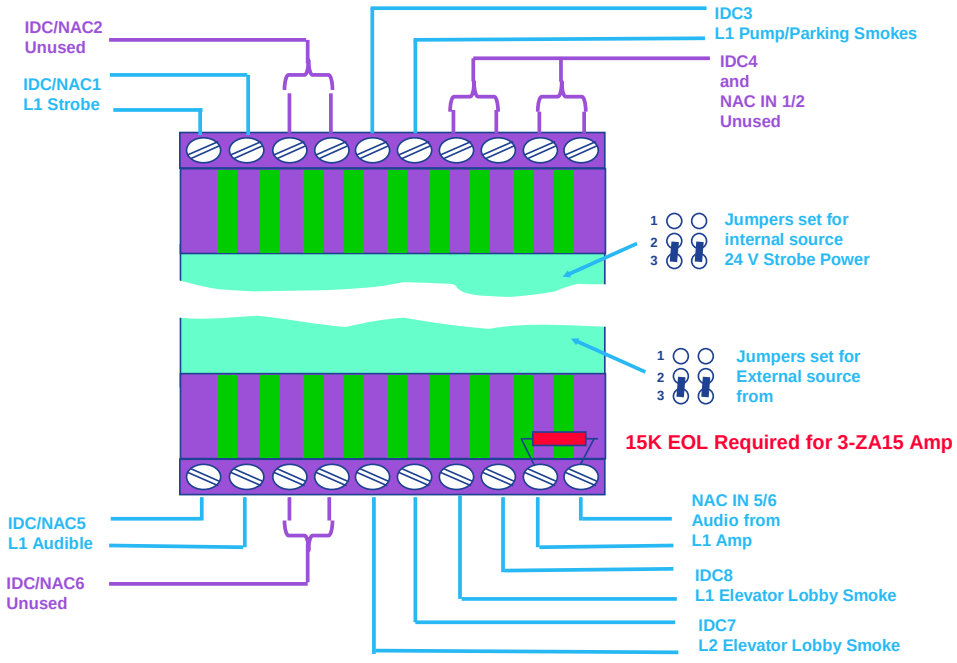
CAB #1 - Bottom Chassis of the 3-CAB21



CAB #2 - CAB6
Maintenance Building



3-IDC8/4 Configuration



Objectives

- **Differentiate between U.S. Local and U.S. Proprietary Mode**
- **Identify the features of the EST3 Front Panel**
- **Discuss the EST3X Front Panel operations**

U.S. Local Mode

- **In U.S. Local Mode, the panel does not require acknowledgement of each event**
- **Events may be acknowledged and restored by pressing reset**
- **View and acknowledge events in the event messages window**
- **Trouble and monitor events are self-restoring when the off-normal condition is resolved**
- **Alarm and supervisory events must be manually restored**

U.S. Proprietary Mode

- **Used in applications requiring minimal operator intervention when life safety events are handled by facility personnel**
- **Each event must be individually acknowledged and viewed by pressing the respective alarm, supervisory, trouble or monitor event queue's acknowledge button**
- **The restoration of each event is posted and must be individually acknowledged**
- **The Previous/Next buttons are disabled from scrolling the message queues**

EST3 Front Panel Operations

Main Menu

- Display the EST3 Main Menu by pressing the front panel's Command Menus Button
- The complete Main Menu displays to all operators
- Menu privileges are controlled by 4 levels of password access

Selections

- Scroll to select the desired menu item and press Enter or select the menu item by number on the numeric keypad
- Note: Press the Next button to scroll to view additional Main Menu items

Status command

- The Status command enables Level 1 through 4 operators to view or print a listing of active alarm, trouble, supervisory, monitor or all points
- This enables operators to view or print a list of points tested or disabled, outputs turned on, and the status of security partitions or holdup devices

Enable command

The Enable command allows Level 2 through 4 operators to enable a previously disabled logic group, switch, device, card, LED or time control

Disable command

- **The Disable command allows Level 2 through 4 operators to disable a logic group, switch, device, card, LED or time control**
- **Actual enable and disable operator privileges vary**
- **A Level 2 operator is permitted only to enable or disable a zone group**
- **Level 4 operators are permitted to enable or disable a card**

Activate command

The Activate command allows Level 2 through 4 operators to:

- Activate alternative sensitivity or message routing
- Start a guard patrol route or a check-in period
- Turn on a relay, an LED, an audio amplifier or an audio message

Restore command

The Restore command allows Level 2 through 4 operators to:

- Restore to primary sensitivity or message routing
- Stop a guard patrol route or a check-in period
- Turn off a relay, an LED, an audio amplifier or an audio message

Reports command

The Reports command allows Level 1 through 4 operators to generate and review or print reports regarding:

- Signature detector sensitivity
- Detectors requiring maintenance
- Panel event history
- Panel software and hardware revision levels
- MODCOM compliance

Program command

The Program command allows Level 1 through 4 operators to:

- Toggle the displayed language (if configured)
- Set system time or date
- Create and edit the holiday list
- Change operator passwords
- Restart a panel or the system
- Clear the system's event history and alarm counter

Test command

- **The Test command allows Level 1 through 4 operators to perform a front panel lamp test sequence**
- **This command permits Level 4 operators to start, stop or cancel a service group test sequence**

Security command

- **The Security command allows Level 1 through 4 operators to arm or disarm a security system partition or reset latched security alarm events**
- **Permits Level 2 through 4 operators to bypass or remove the bypass of a security device**
- **Present only on the Main Menu if the system has been configured for fully integrated security operations**

EST3X Front Panel Operations

Controls and indicators

- **The rotary controller is used to enter data or to select menu functions or commands**
- **Press the rotary controller to access the Queue menu**
- **To select a menu function or command, scroll to the desired function or command and press the rotary controller again to enter**
- **To correct errors, scroll to the Delete arrow < and press the rotary controller**

Controls and indicators

- **Active alarm, supervisory, trouble and monitor event messages are posted in separate queues**
- **Alarm, supervisory and trouble message queue's LED flashes when an active event message is posted in its respective queue and has not been acknowledged**
- **Monitor event message activates Acknowledge button LED and posts in queue**

Controls and indicators

- **In the US proprietary mode, the restoration of each event must also be acknowledged by pressing the Acknowledge button**
- **In the US local mode, all active events can be globally acknowledged by pressing the Reset button**

Front panel Queue Menu

- If your system is off-normal, use the Queue Menu to scroll to Alarm, Supervisory or Trouble
- To access the Main Menu, press the rotary controller and scroll to select the Main Menu

Front panel Main Menu

- Choose a command or function from the front panel Main Menu display by rotating the rotary controller and pressing again to select
- The complete menu displays for all operators
- Menu privileges are controlled by four levels of password access

Back command

- **The Back command takes you back to the Queue Menu**

Exit command

- **The Exit Menu command takes you back to the System Normal display**

Status command

- **The Status command enables Level 1 through 4 operators to view or print a listing of active alarm, trouble, supervisory, monitor or all points**
- **This enables operators to view or print a listing of points tested or disabled, and outputs turned on**

Enable command

- **The Enable command allows Level 2 through 4 operators to enable a logic group, switch, device, card, LED or time control previously disabled**

Disable command

- The Disable command allows Level 2 through 4 operators to disable a logic group, switch, device, card, LED or time control
- The actual enable and disable operator privileges vary
- For example, a Level 2 operator is permitted only to enable or disable a zone group, while Level 4 operators are permitted to enable or disable a card

NEW Activate commands

- Remote Read Lock: prevents programmers from communicating with the panel remotely
- Remote Write Unlock: prevents programmers from writing to the panel remotely
- Sensor Bypass: bypasses the photo portion of a PHS if configured for supervisory without disabling the heat
- Gas Accelerate Response: a feature to speed up testing of CO detectors
- Drill is now found under the Activate Menu (no Drill button on the LCD)

Restore command

The Restore command allows Level 2 through 4 operators to:

- Restore to primary sensitivity or message routing
- Turn off a relay, an LED, an audio amplifier or an audio message
- Like the enable and disable commands, the actual operator privileges vary for both the activate and restore commands
- Restore Remote Read Lock and Remote Write Unlock
- Restore Drill
- Sensor Bypass
- Gas Accelerate Response

Reports command

The Reports command allows Level 2 through 4 operators to generate and review or print reports on:

- Signature detector sensitivity
- Detectors requiring maintenance
- The panel's event history
- The panel's software and hardware revision levels
- 3MODCOM compliance

Program command

The Program command allows Level 1 through 4 operators to:

- Toggle the displayed language
- Set system time or date
- Create and edit system's holiday list
- Change operator passwords
- Restart a panel or the system
- Clear the system's event history and alarm counter

The actual operator privileges vary for programming system parameters

Test command

- **The Test command allows Level 1 through 4 operators to perform a front panel lamp test sequence**
- **This command permits the Level 4 operators to start or cancel (stop) a service group test sequence or a device test**

Security command

- The Security command allows Level 1 through 4 operators to arm or disarm a security partition
- This menu has to be enabled via programming the SDU for it to appear
- This option is only available if your 3X is on an EST3 network
- 3X does not support security by itself

Object

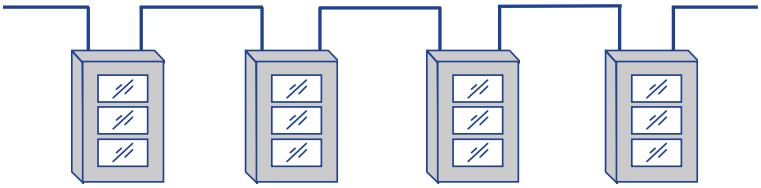
What is an object?

Any addressable device, circuit, message, pseudo point or other entity within the EST3 system database used to initiate an input event within a rule or is the subject of a rule's output action

EST3 objects

What are some objects found within the EST-3?

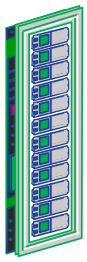
Each cabinet or remote annunciator node within the system may be considered an object



EST3 objects

What are some objects found within the EST-3?

Each EST-3 Local Rail Module (LRM) and Operator Layer LED/Switch Panel



3-SSDC(1) or 3-SDDC(1) Loop Controller
3-ZA20 or 3-ZA40 Amplifier
3-PS/MON
3-IDC8/4
3-AADC
3-OPS
3-LDSM

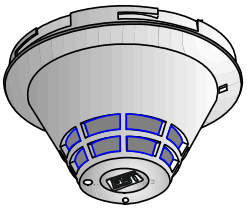
3-24x
3-12/S1x
3-12/S2x
3-6/3S1x

Each LED and switch is an object in the database

EST3 objects

What are some objects found within the EST-3?

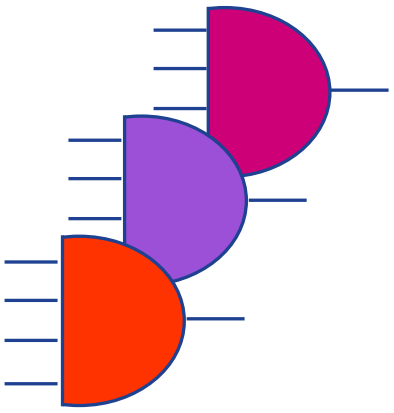
Each smoke detector or module is an object in the database.



EST3 objects

What are some objects found within the EST-3?

Logical Groups are objects in the database.

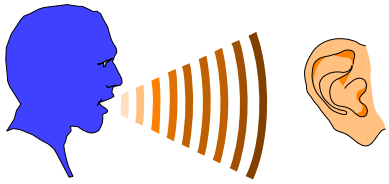


AND
ZONE
MATRIX
SERVICE
Instruction TEXT
Partition

EST3 objects

What are some objects found within the EST-3?

Audio and text messages are objects in the database



EST3 objects

What are some objects found within the EST-3?

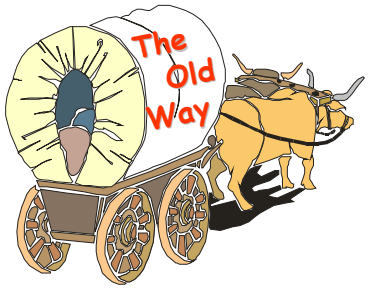
600 series Pseudo Points are objects in the database

Object Label

Any descriptive word or words (a naming convention) created by the system designer to identify a specific object within the EST3's or 3-SDU's database

What and where is this device?

01020134



New Object Labels

Labels contain component Modifiers

New terminology

More descriptive!

Object Label Parameters

Must have an object label (from a LED to the project)

40 characters in length

Not case sensitive

A = a

Must be unique (Compiler error if duplicated)

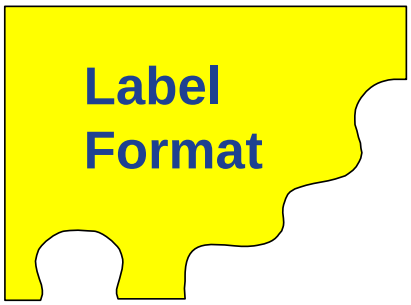
Arbitrary (At the programmer's discretion)

Do NOT use { , }, [,], #, <, >, *, %, spaces, ', ", :, or ; in object labels

ASCII characters are acceptable

Object Labels

When developing labels and your labeling plan consider:



Object Labels

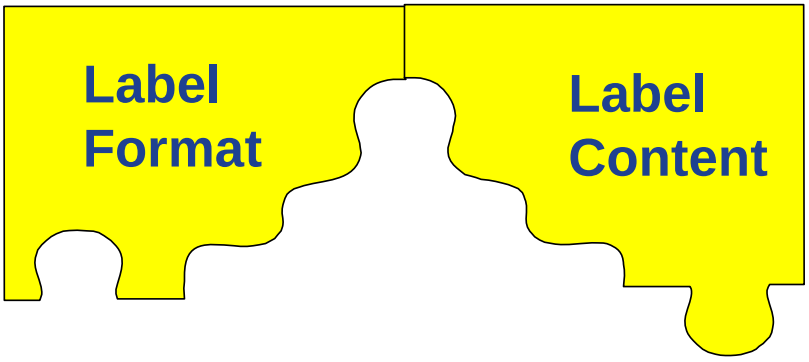
Make your object labels readable and understandable!

- **Viewed online, on printed reports and possibly on the 3-LCD or 3-LCDXL display**

Which label is easiest to read and understand?

BUILDING1_CABINET1 **BLDG1_CAB1** **B1_C1**

Object Labels



Object Labels

Describe location

BLDG1_FLOOR1_LOBBY

BLDG1_FLOOR2_ELEVATOR_LOBBY

Describe function

BLDG1_CAB1_DMPR_CNTRL

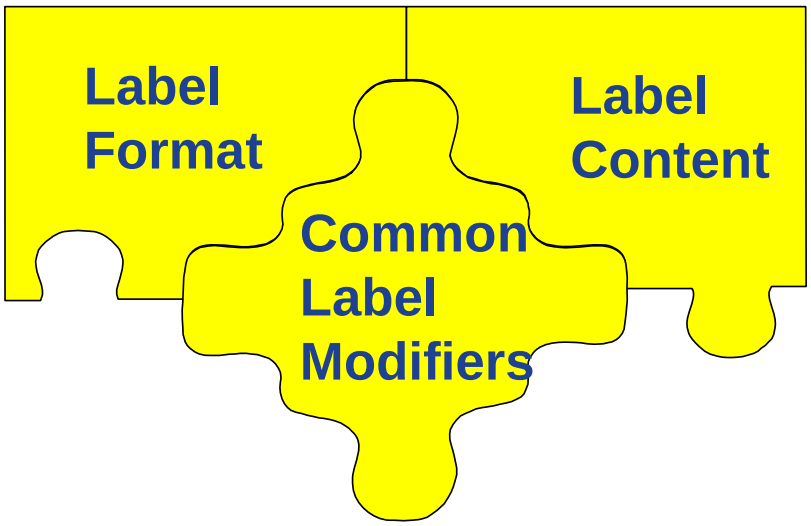
BLDG5_LEV7_EVAC_AMP

Describe device type

BLDG1_LEVEL3_PULL

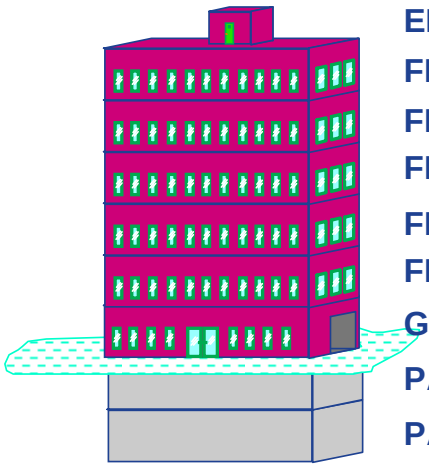
BLDG3_MECH_RM_SMK

Object Labels



Object Labels

Floor and level designators may be awkward in rules programming.

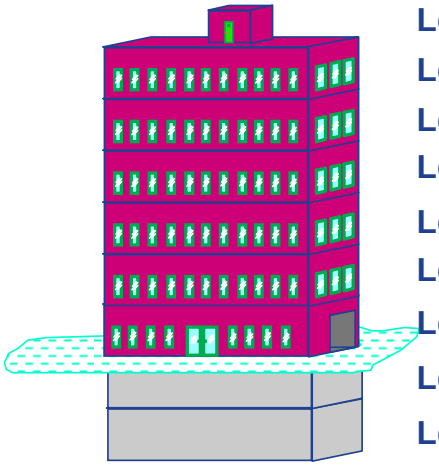


ELV_EQ _ROOM
FLOOR_6
FLOOR_5
FLOOR_4
FLOOR_3
FLOOR_2
GROUND_FLOOR
PARKING_LEVEL1
PARKING_LEVEL2

Object Labels

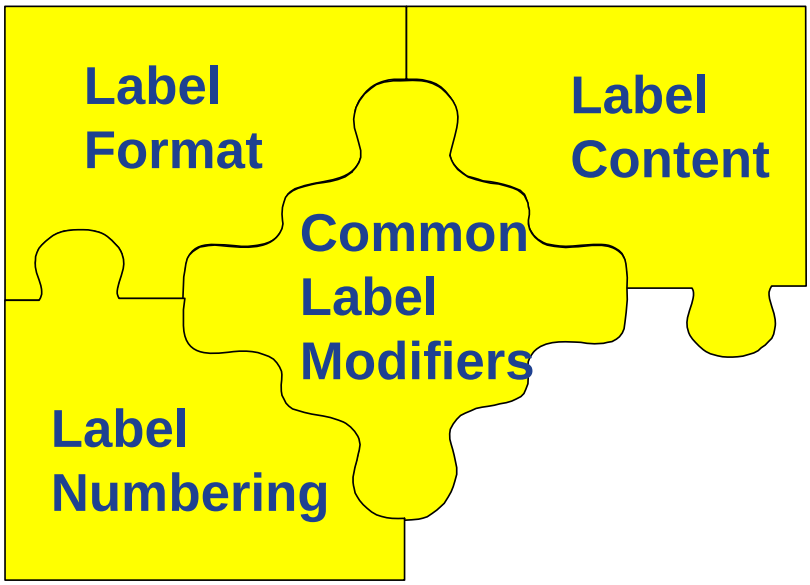
Labeling like this makes rules programming much easier.

Location messages display like this:



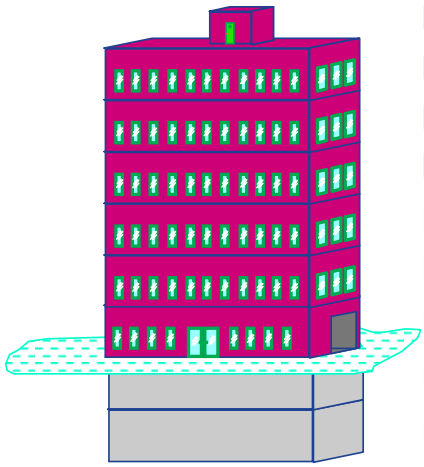
Level9	ELV_EQ _ROOM
Level8	FLOOR_6
Level7	FLOOR_5
Level6	FLOOR_4
Level5	FLOOR_3
Level4	FLOOR_2
Level3	GROUND_FLOOR
Level2	PARKING_LEVEL1
Level1	PARKING_LEVEL2

Object Labels

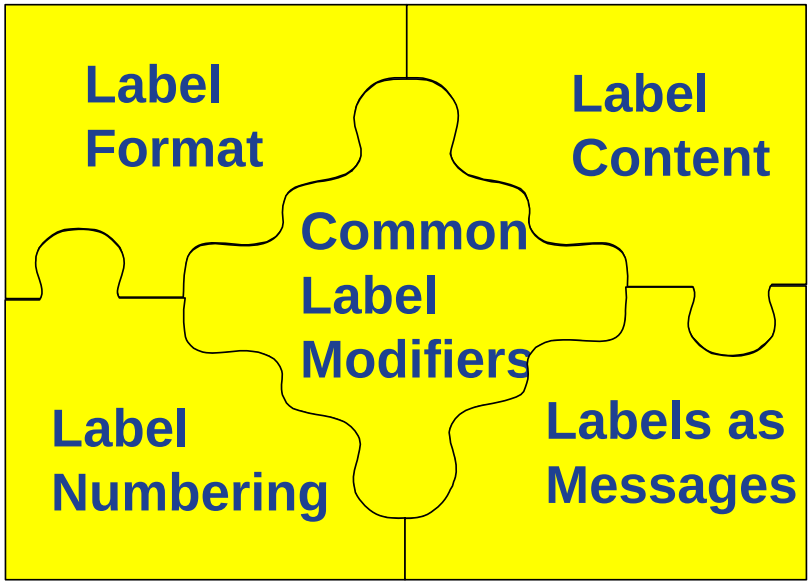


Object Labels

Number modifiers uses the power of the SDU Programming tools.
{Level1_Smk, Level2_Pull}



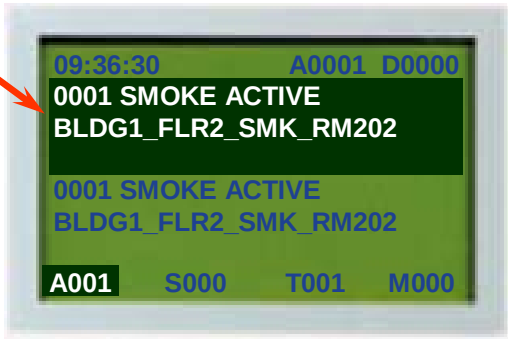
Object Labels



Object Labels

Use the object label as location messages

BLDG1_FLR2_SMK_RM202



Each location message line is 21 characters

Object Labels

Each object in the database has:

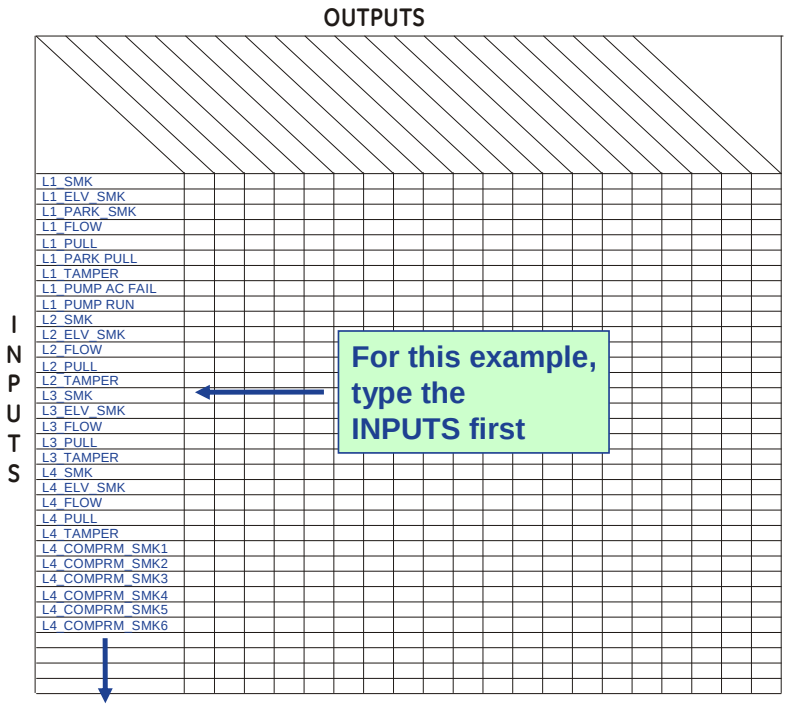
An Address used by the system (i.e. 01020111)

A Label used for programming (e.g. L2_ELEV_LOB_SMK)

A Location Message used for display
(e.g. Ground Floor Elevator Lobby Smoke)

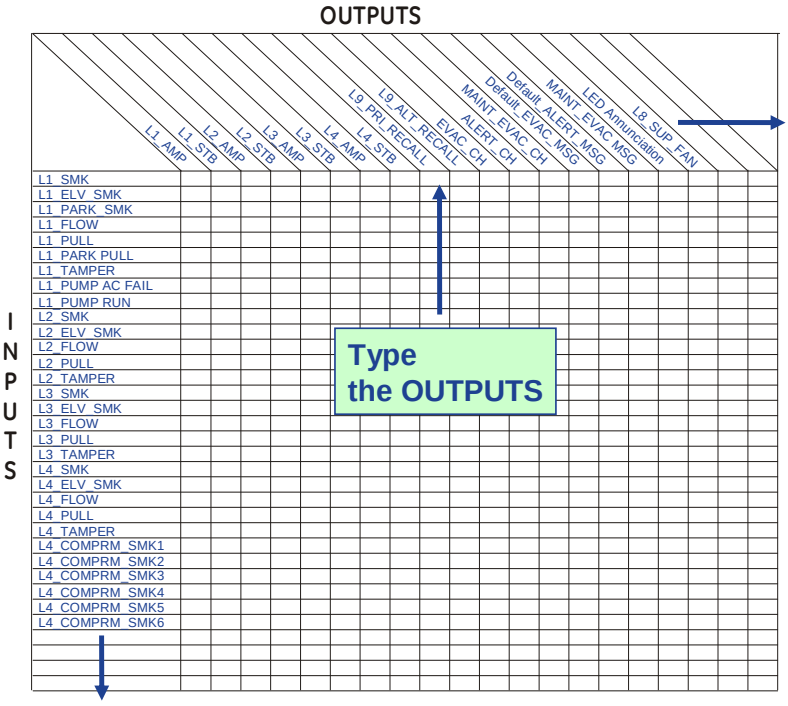
Label plan

Develop a labeling plan with the I/O Matrix Worksheet or Excel on your PC



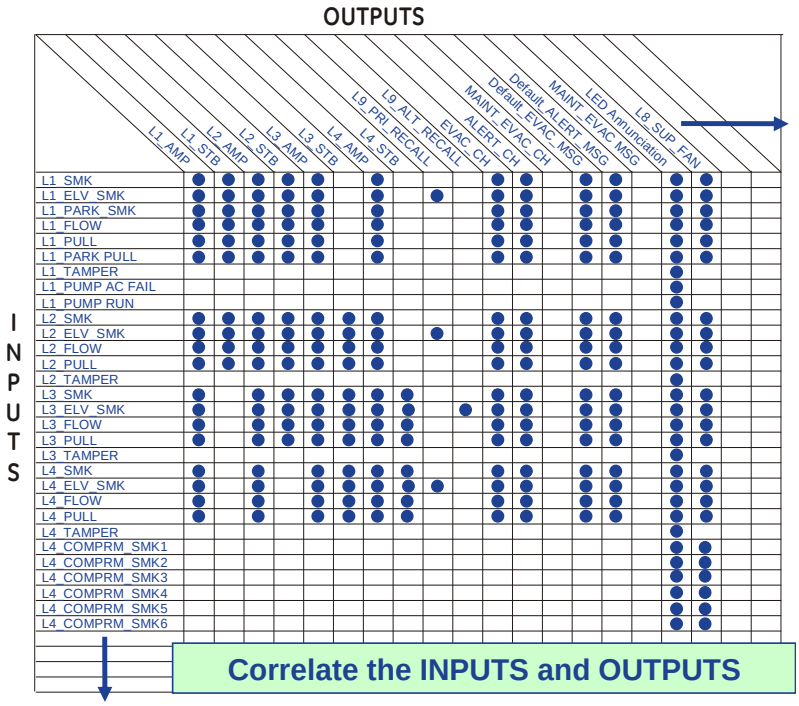
Label plan

Developing a labeling plan with the I/O Matrix Worksheet or Excel on your PC

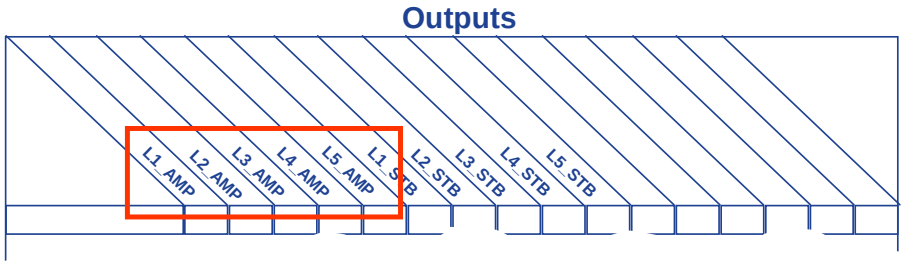


Label plan

Developing a labeling plan with the I/O Matrix Worksheet or Excel on your PC



Label plan



Amplifiers are listed as the outputs

Audio can be tricky:

- Up to eight default and/or custom channels which are objects needing labels
- Default messages and/or custom messages sent over the channels to the amplifiers need labels

Output

What type of object is a switch?

- **INPUT**
  OUTPUT

What type of object is a LED?

  INPUT
- **OUTPUT**

Output

What type of object is a SIGA-CC1 configured as a Firefighter's Phone?

The firefighters' phone must be activated or connected. In this case, consider the CC1 as an OUTPUT.

When a firefighters' phone is plugged in, it initiates a call-in to the panel.
In this case, it is considered an INPUT.

The firefighters' phone operations are built into the microcode. Custom rules may be written to LED annunciate firefighters' phone call-ins and activations. In this case, firefighters' phones are listed as INPUTS during the planning process.

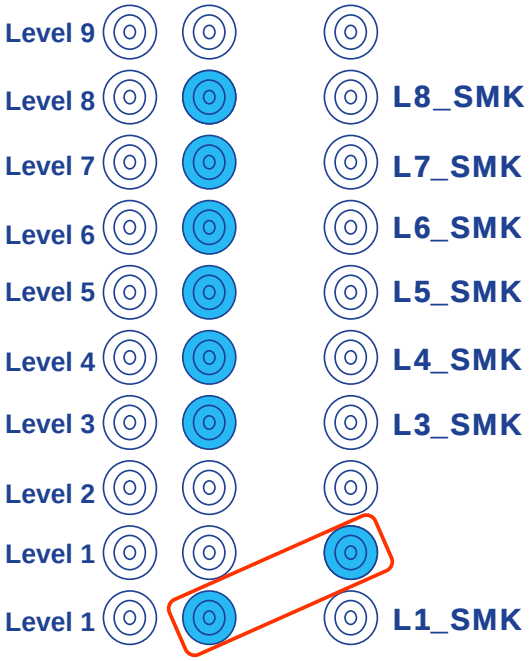
In the following example, we opted to make the middle modifier the variable

L1_PARK_SMK
L1_PUMP_SMK
and so on

We could have made the last modifier position the variable

L1_SMK_PARK
L1_SMK_PUMP
and so on.

Smoke Detector Labels

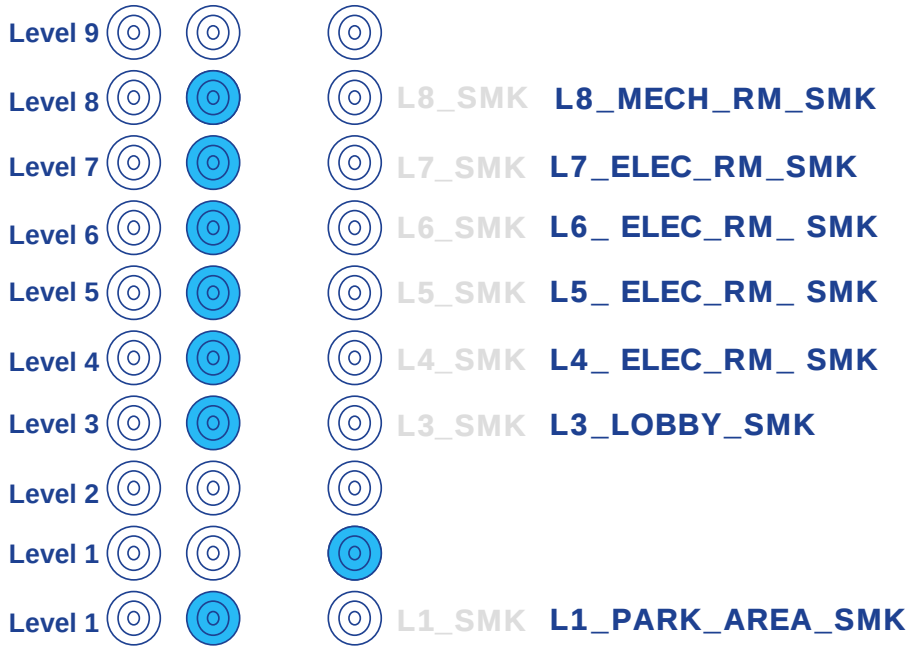


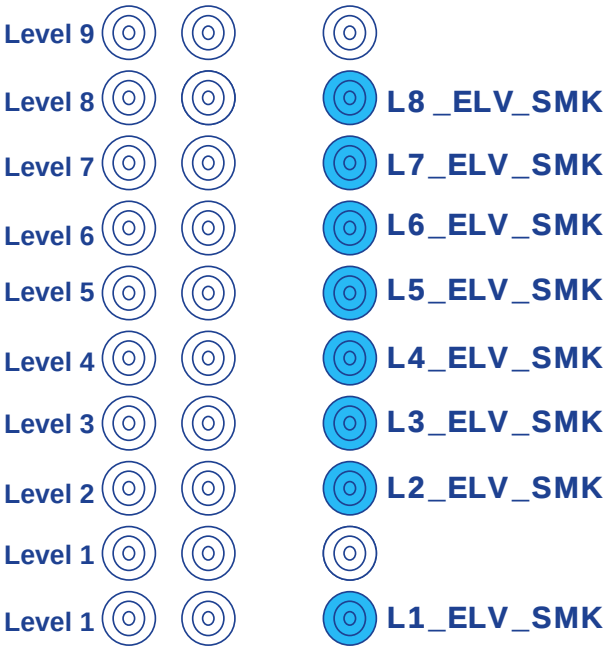
General purpose type smoke detectors on levels 1, 3, 4, 5, 6, 7 and 8 individually initiate alarms

Use L and the level # for our first modifier

Use _SMK for the last modifier

Note: The two L1_SMK detectors are conventional smokes connected to IDC circuit 3 on 3-IDC8/4 Module, where the object label is assigned to the circuit.





Elevator lobby smokes on levels 1 through 8 individually initiate alarms and capture the elevators

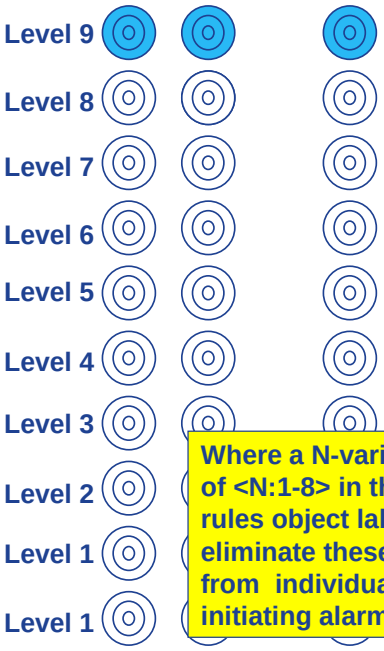
Use L and the level # for our first modifier

Use _ELV for the second modifier

Use _SMK for the last modifier

To be more descriptive:

Ln_ELV_LOBBY_SMK



Label the three level 9 smokes, which are in an AND group and **DO NOT** individually initiate alarms.

Exclude these smokes from individually initiating alarms. This is accomplished using a N-variable when programming rules or by making the labels unique.

For this example let's label these smokes:

Where a N-variable of <N:1-8> in the rules object label will eliminate these smokes from individually initiating alarms

L9_SMK1
L9_SMK2
L9_SMK3

Or by adding numbers to the end of the _SMK modifier, which eliminates these smokes from individually initiating alarms.

These nine level 4 smokes are in a MATRIX group and **DO NOT** individually initiate alarms.

Exclude these smokes from individually initiating alarms. Because two other smokes have an L4 modifier, we can not use a N-variable. However, we can accomplish this by making the labels unique.

L4_COMPRM_SMK1
L4_COMPRM_SMK2
L4_COMPRM_SMK3
L4_COMPRM_SMK4
L4_COMPRM_SMK5
L4_COMPRM_SMK6
L4_COMPRM_SMK7
L4_COMPRM_SMK8
L4_COMPRM_SMK9

Where the numbers added to the _SMK modifier eliminate these smokes from individually initiating alarms

Or we could reposition the modifiers to make the labels dramatically unique:

COMPRM _L4_SMK1
COMPRM _L4_SMK2
COMPRM _L4_SMK3
COMPRM _L4_SMK4
COMPRM _L4_SMK5
COMPRM _L4_SMK6
COMPRM _L4_SMK7
COMPRM _L4_SMK8
COMPRM _L4_SMK9

Placing the middle COMPRM modifier first eliminates these smokes from individually initiating alarms

Project Smoke Detector Labels

A N-variable [<N:n-n>] is used to specify a specific range of objects labels

A wildcard [*] is used to specify all variable modifiers in objects labels at a modifier position

Project Smoke Detector Labels

Ln_SMK
Ln_ELV_SMK
L9_SMKn
L4_COMPRM_SMKn

The object label in our rule's input statement to initiate an alarm when an individual smoke goes off normal would be:

ALARM SMOKE 'L<N:1-8>_*SMK'

The 1 through 8 N-variable excludes the three level 9 smokes

Project Smoke Detector Labels

Ln_SMK
Ln_ELV_SMK
L9_SMKn
L4_COMPRM_SMKn

The object label in our rule's input statement to initiate an alarm when an individual smoke goes off normal would be:

ALARM SMOKE 'L<N:1-8>_*SMK'

The * includes the variable modifier used to specify locations, even with 0 characters at the modifier position

Project Smoke Detector Labels

Ln_SMK
Ln_ELV_SMK
L9_SMKn
L4_COMPRM_SMKn

The object label in our rule's input statement to initiate an alarm when an individual smoke goes off normal would be:

No wildcard

ALARM SMOKE 'L<N:1-8>_*SMK'

The number part of the level 4 computer room _SMK modifier makes these labels unique and excludes these smokes for individually initiating alarms

Project Smoke Detector Labels

For our example project, let's assume we labeled the smokes

Signature Sensors
L9_EAST_SHAFT_SMK
L9_WEST_SHAFT_SMK
L9_ELV_EQ_RM_SMK
L8_MECHRM_SMK
L8_ELV_LOBBY_SMK
L7_ELEC_RM_SMK
L7_ELV_LOBBY_SMK
L6_ELEC_RM_SMK
L6_ELV_LOBBY_SMK
L5_ELEC_RM_SMK
L5_ELV_LOBBY_SMK
L4_ELEC_RM_SMK
L4_ELV_LOBBY_SMK
L4_COMPRM_SMK1
L4_COMPRM_SMK2
L4_COMPRM_SMK3
L4_COMPRM_SMK4
L4_COMPRM_SMK5
L4_COMPRM_SMK6
L4_COMPRM_SMK7
L4_COMPRM_SMK8
L4_COMPRM_SMK9
L3_GND_LOBBY_SMK
L3_ELV_LOBBY_SMK
IDC ZONES
L2_ELV_LOBBY_SMK
L1_PARKING_SMK
L1_ELV_LOBBY_SMK

Rule Object Label
'L<N:1-8>_*SMK'

Project Smoke Detector Labels

We could easily have labeled the smokes

Signature Sensors	Signature Sensors
L9_EAST_SHAFT_SMK	L9_SMK _EAST_SHAFT
L9_WEST_SHAFT_SMK	L9_SMK _WEST_SHAFT
L9_ELV_EQ_RM_SMK	L9_SMK _ELV_EQ_RM
L8_MECHRM_SMK	L8_SMK _MECHRM
L8_ELV_LOBBY_SMK	L8_SMK _ELV_LOBBY
L7_ELEC_RM_SMK	L7_SMK _ELEC_RM
L7_ELV_LOBBY_SMK	L7_SMK _ELV_LOBBY
L6_ELEC_RM_SMK	L6_SMK _ELEC_RM
L6_ELV_LOBBY_SMK	L6_SMK _ELV_LOBBY
L5_ELEC_RM_SMK	L5_SMK _ELEC_RM
L5_ELV_LOBBY_SMK	L5_SMK _ELV_LOBBY
L4_ELEC_RM_SMK	L4_SMK _ELEC_RM
L4_ELV_LOBBY_SMK	L4_SMK _ELV_LOBBY
L4_COMPRM_SMK1	L4_SMK1 _COMPRM
L4_COMPRM_SMK2	L4_SMK2 _COMPRM
L4_COMPRM_SMK3	L4_SMK3 _COMPRM
L4_COMPRM_SMK4	L4_SMK4 _COMPRM
L4_COMPRM_SMK5	L4_SMK5 _COMPRM
L4_COMPRM_SMK6	L4_SMK6 _COMPRM
L4_COMPRM_SMK7	L4_SMK7 _COMPRM
L4_COMPRM_SMK8	L4_SMK8 _COMPRM
L4_COMPRM_SMK9	L4_SMK9 _COMPRM
L3_GND_LOBBY_SMK	L3_SMK _GND_LOBBY
L3_ELV_LOBBY_SMK	L3_SMK _ELV_LOBBY
IDC ZONES	IDC ZONES
L2_ELV_LOBBY_SMK	L2_SMK _ELV_LOBBY
L1_PARKING_SMK	L1_SMK _PARKING
L1_ELV_LOBBY_SMK	L1_SMK _ELV_LOBBY
Rule Object Label	Rule Object Label
'L<N:1-8>_*SMK'	'L<N:1-8>_SMK_*'

Project Smoke Detector Labels

We could easily have labeled the smokes

Signature Sensors	Signature Sensors
L9_EAST_SHAFT_SMK	L9_SMK _EAST_SHAFT
L9_WEST_SHAFT_SMK	L9_SMK _WEST_SHAFT
L9_ELV_EQ_RM_SMK	L9_SMK _ELV_EQ_RM
L8_MECHRM_SMK	L8_SMK _MECHRM
L8_ELV_LOBBY_SMK	L8_SMK _ELV_LOBBY
L7_ELEC_RM_SMK	L7_SMK _ELEC_RM
L7_ELV_LOBBY_SMK	L7_SMK _ELV_LOBBY
L6_ELEC_RM_SMK	L6_SMK _ELEC_RM
L6_ELV_LOBBY_SMK	L6_SMK _ELV_LOBBY
L5_ELEC_RM_SMK	L5_SMK _ELEC_RM
L5_ELV_LOBBY_SMK	L5_SMK _ELV_LOBBY
L4_ELEC_RM_SMK	L4_SMK _ELEC_RM
L4_ELV_LOBBY_SMK	L4_SMK _ELV_LOBBY
L4_COMPRM_SMK1	L4_SMK1 _COMPRM
L4_COMPRM_SMK2	L4_SMK2 _COMPRM
L4_COMPRM_SMK3	L4_SMK3 _COMPRM
L4_COMPRM_SMK4	L4_SMK4 _COMPRM
L4_COMPRM_SMK5	L4_SMK5 _COMPRM
L4_COMPRM_SMK6	L4_SMK6 _COMPRM
L4_COMPRM_SMK7	L4_SMK7 _COMPRM
L4_COMPRM_SMK8	L4_SMK8 _COMPRM
L4_COMPRM_SMK9	L4_SMK9 _COMPRM
L3_GND_LOBBY_SMK	L3_SMK _GND_LOBBY
L3_ELV_LOBBY_SMK	L3_SMK _ELV_LOBBY
IDC ZONES	IDC ZONES
L2_ELV_LOBBY_SMK	L2_SMK _ELV_LOBBY
L1_PARKING_SMK	L1_SMK _PARKING
L1_ELV_LOBBY_SMK	L1_SMK _ELV_LOBBY
Rule Object Label	Rule Object Label
'L<N:1-8>_*SMK'	'L<N:1-8>_SMK_*'

How might you label the:

Lets use:

Signature Sensors
L9_EAST_SHAFT_SMK
L9_WEST_SHAFT_SMK
L9_ELV_EQ_RM_SMK
L8_MECHRM_SMK
L8_ELV_LOBBY_SMK
L7_ELEC_RM_SMK
L7_ELV_LOBBY_SMK
L6_ELEC_RM_SMK
L6_ELV_LOBBY_SMK
L5_ELEC_RM_SMK
L5_ELV_LOBBY_SMK
L4_ELEC_RM_SMK
L4_ELV_LOBBY_SMK
L4_COMPRM_SMK1
L4_COMPRM_SMK2
L4_COMPRM_SMK3
L4_COMPRM_SMK4
L4_COMPRM_SMK5
L4_COMPRM_SMK6
L4_COMPRM_SMK7
L4_COMPRM_SMK8
L4_COMPRM_SMK9
L3_GND_LOBBY_SMK
L3_ELV_LOBBY_SMK
IDC ZONES
L2_ELV_LOBBY_SMK
L1_PARKING_SMK
L1_ELV_LOBBY_SMK

PULLS?

Signature Modules
L9_PULL
L8_PULL
L7_PULL
L6_PULL
L5_PULL
L4_PULL
L3_PULL
L2_PULL
L1_PUMP_PULL
L1_PULL

HEATS?

Signature Sensors
L8_HEAT
L7_HEAT

WATERFLOWS?

Signature Modules
L8_FLOW
L7_FLOW
L6_FLOW
L5_FLOW
L4_FLOW
L3_FLOW
L2_FLOW
L1_FLOW

FIREPHONES?

Signature Modules
L9_FIRE_PHONE
L8_ FIRE_PHONE
L7_ FIRE_PHONE
L6_ FIRE_PHONE
L5_ FIRE_PHONE
L4_ FIRE_PHONE
L3_ FIRE_PHONE
L2_ FIRE_PHONE
L1_ FIRE_PHONE

Light a LED on an operator layer panel to indicate the Floor of Incident, regardless of the type of device that went into alarm.

How would you construct the rule's object label using the labels of this example?

'L<N:1-9>*'

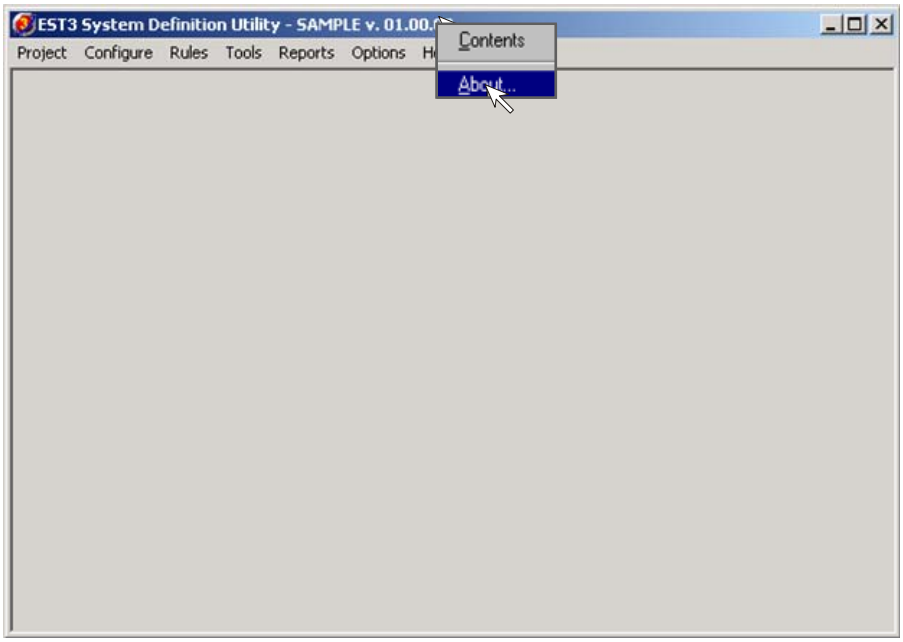
EST3 Series Technician Certification

Using the 3-SDU HELP Utility

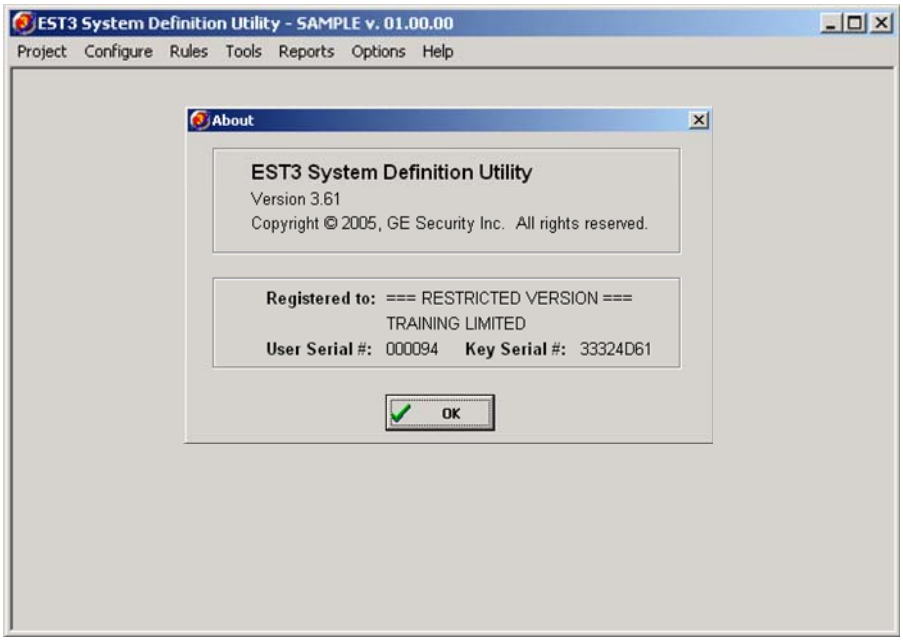
UTC Fire & Security
A United Technologies Company

SDU Help

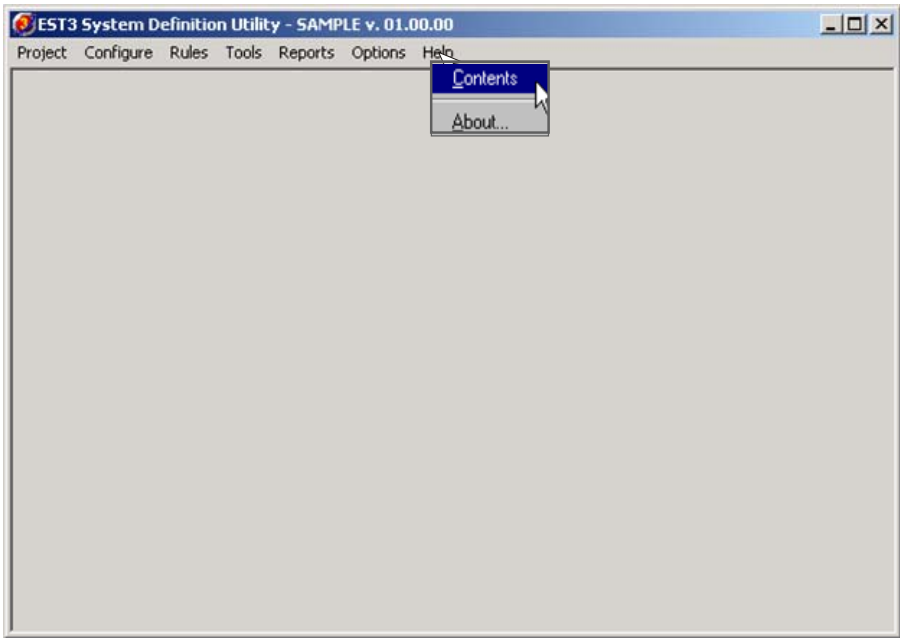
To see information about your current 3-SDU select Help and About



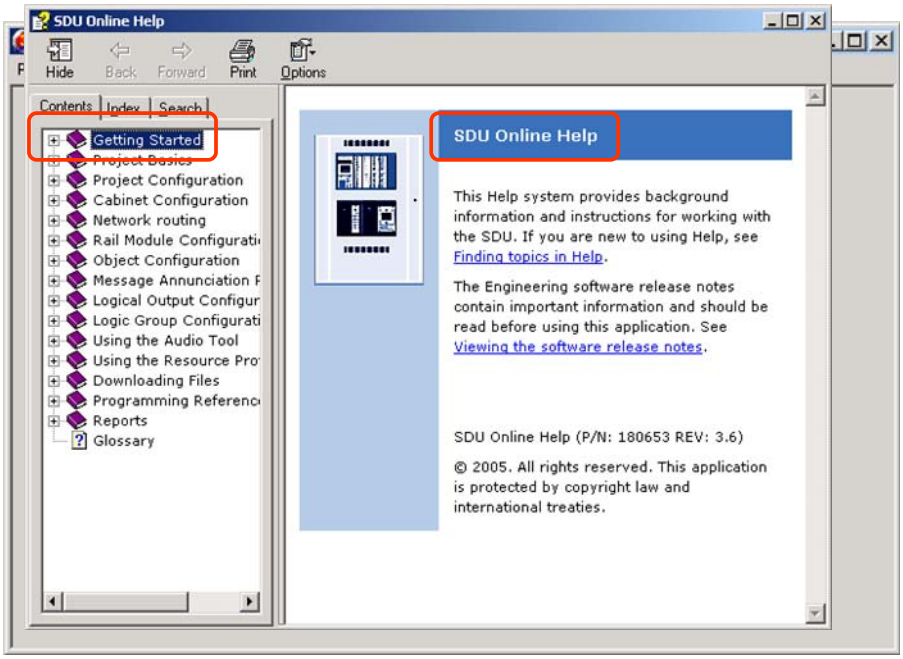
To see information about your current 3-SDU select Help and About



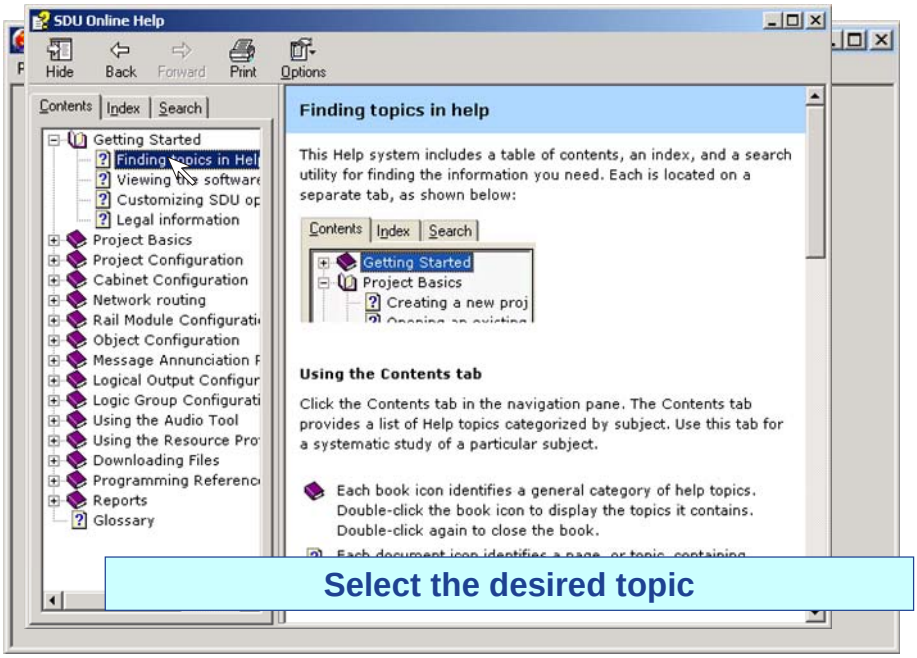
To use the 3-SDU HELP utility select Help and Content to launch the HELP utility



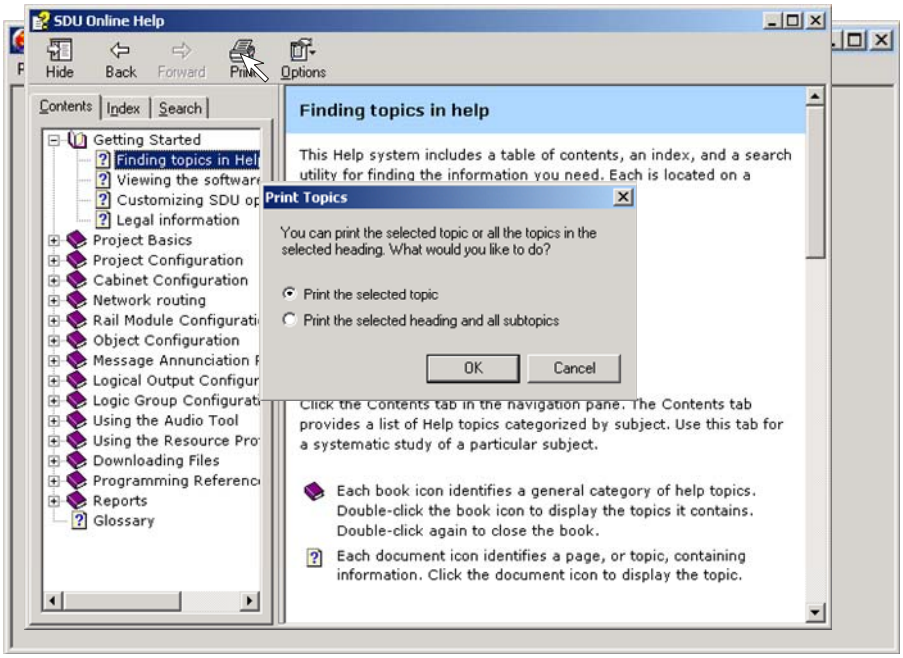
To use the 3-SDU HELP utility select Help and Content to launch the HELP utility



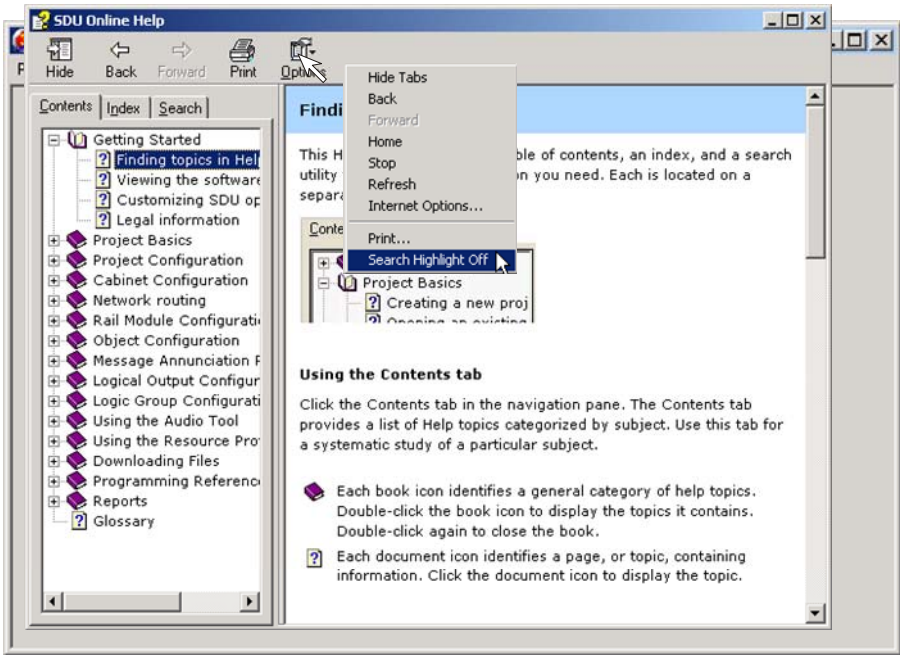
Click + for the desired section under the Contents tab (Getting Started in this example).



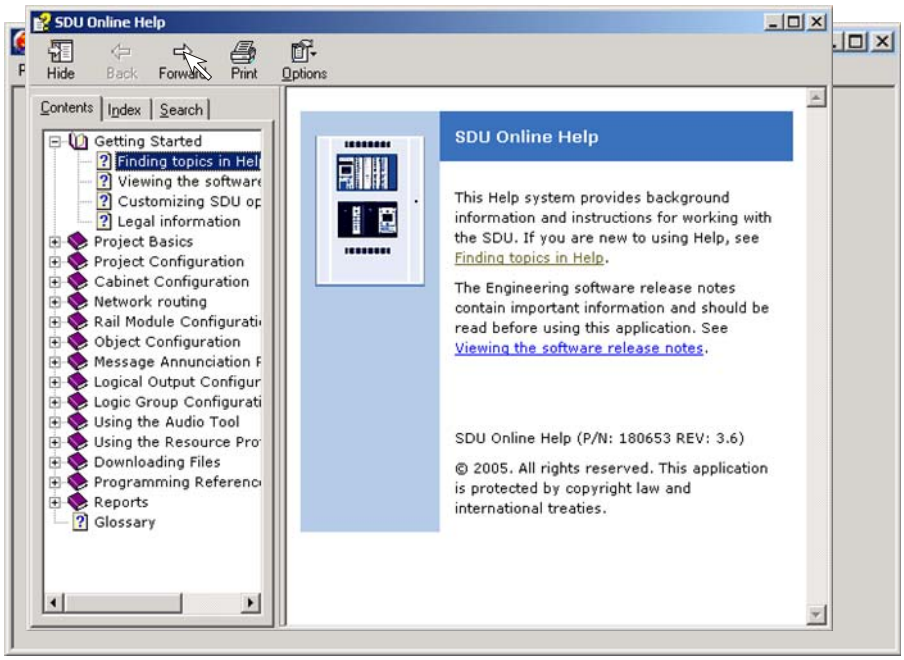
Print any selected topic or Section of topics by clicking on the Main Menu print icon.



The Main Menu contains an Options icon enabling you to select the HELP options.

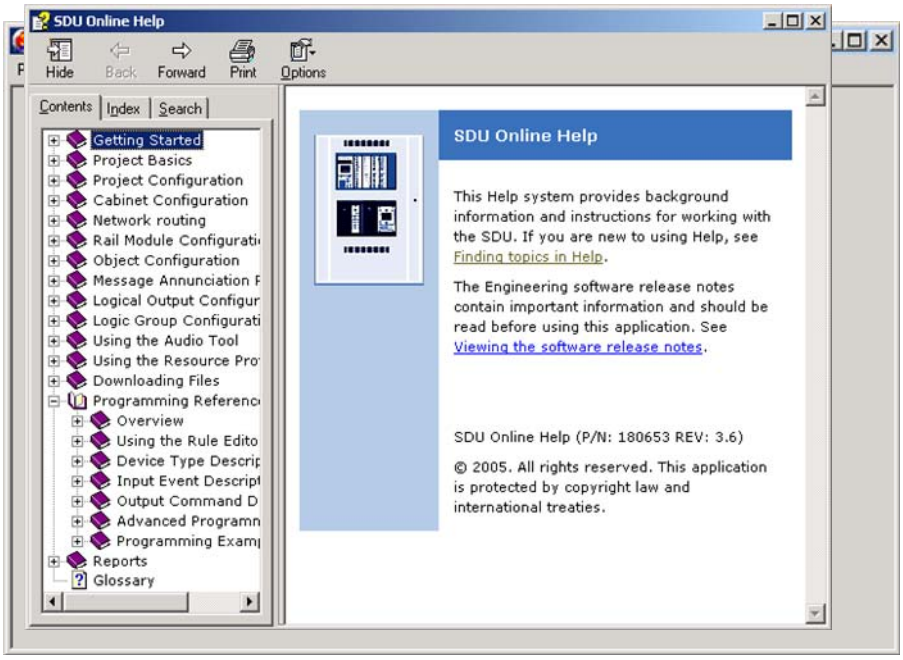


The Main Menu enables you to page Back and to page Forward

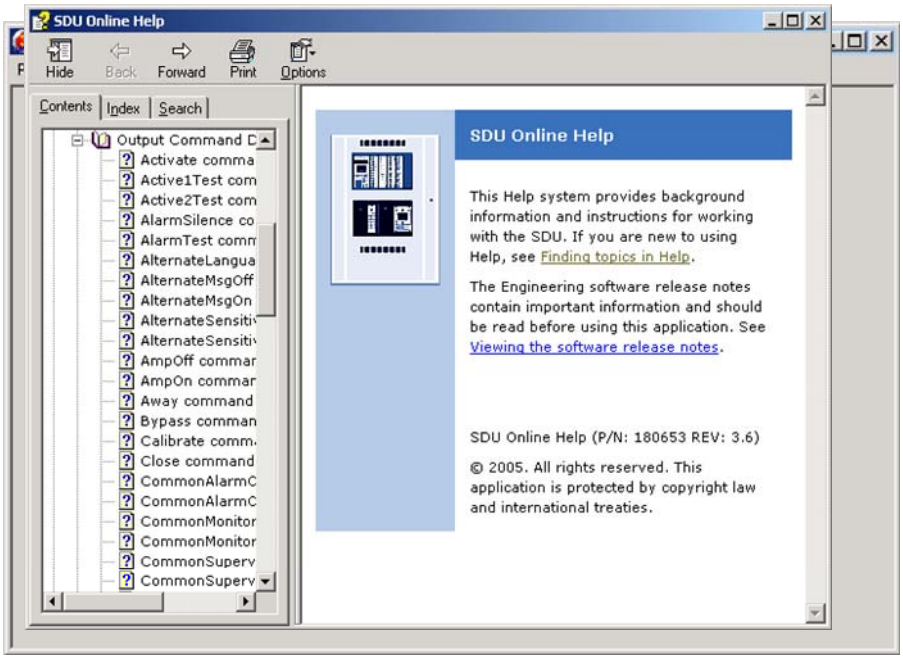


Let's use an example where we want to get help on the Message On (MSGON) output command by using the Contents tab.

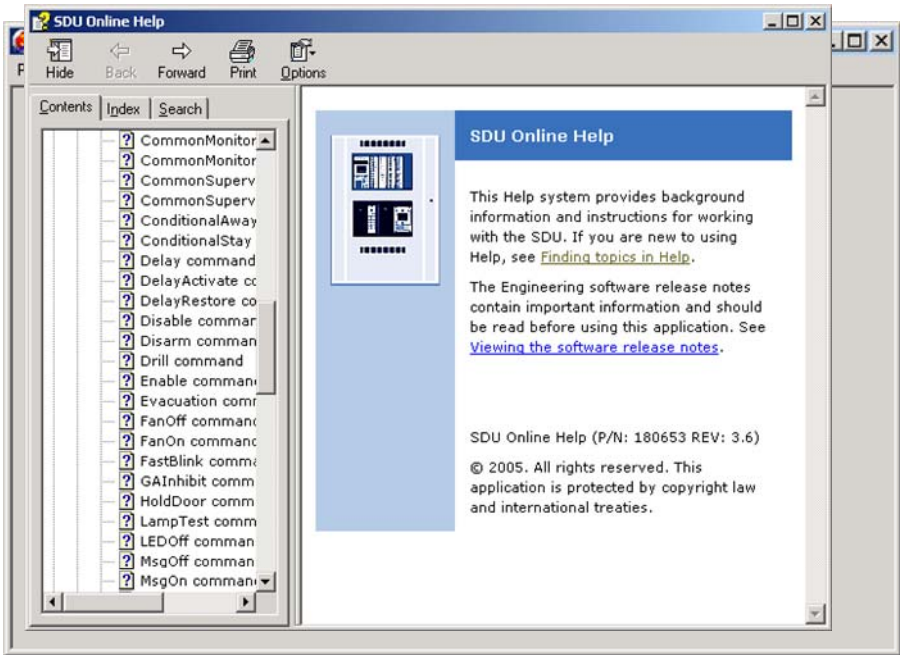
Select the + Programming Reference book icon from the Contents tab.



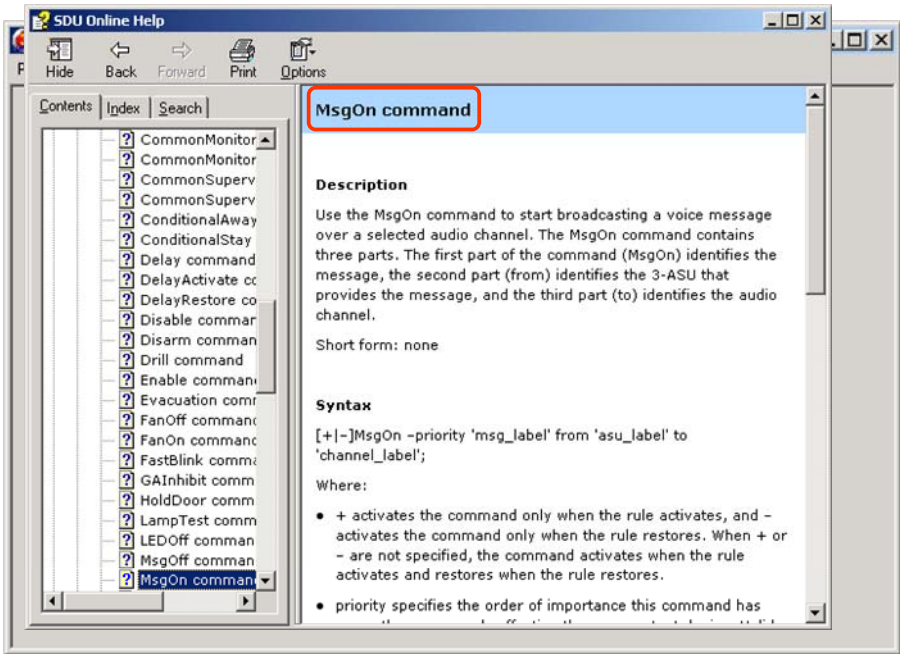
Select the + Output Command Descriptions book icon from the Sub-Section book icons



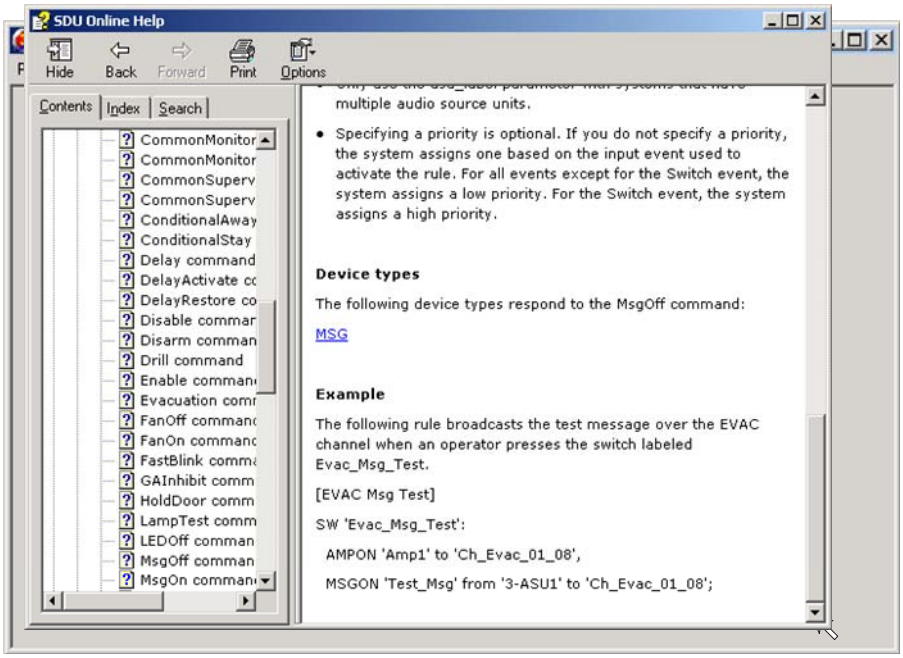
Scroll through the topics list until to find the MsgOn command



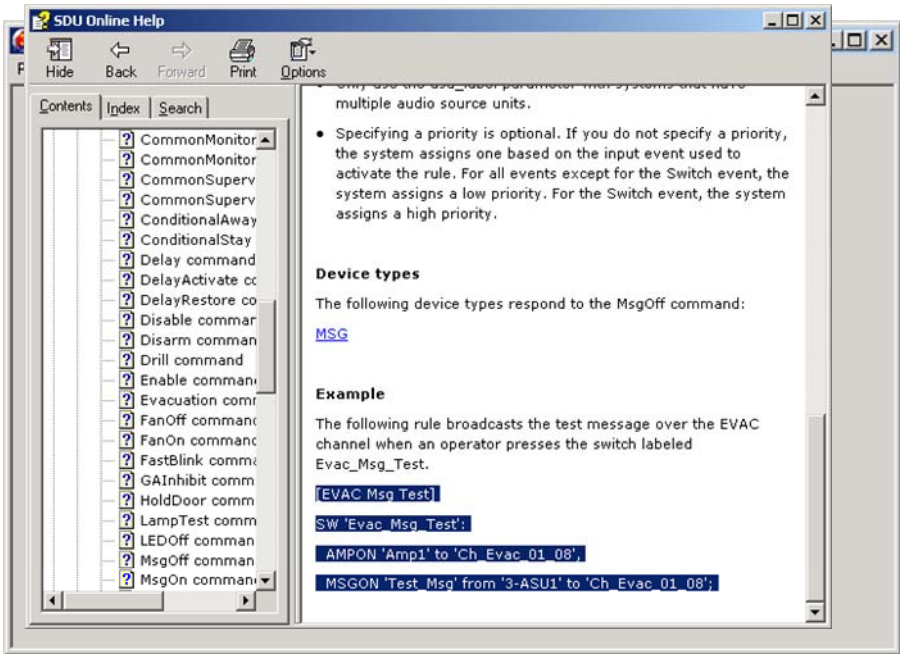
Click the MsgOn command and the MsgOn command topic displays



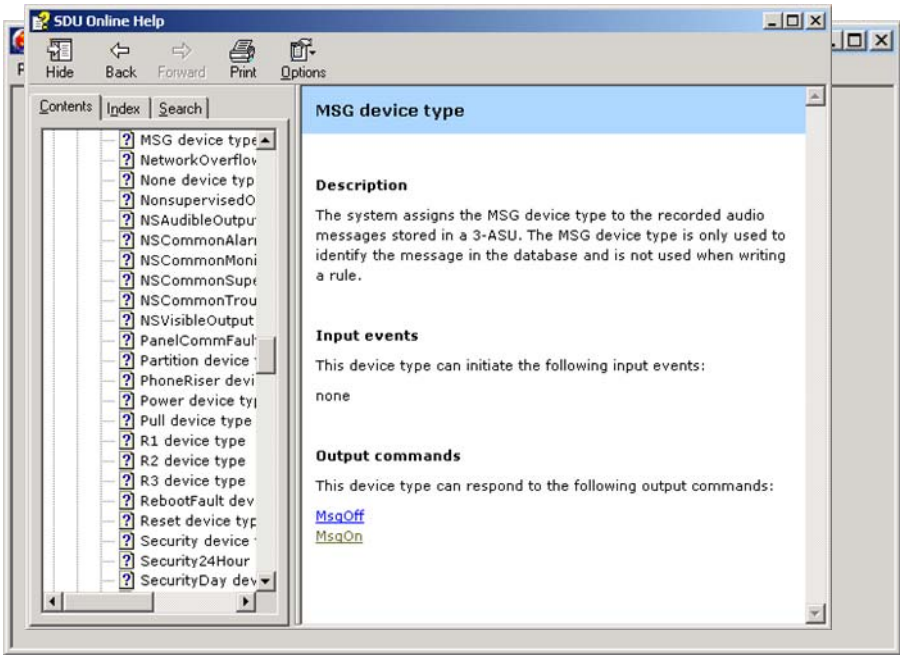
Scroll through the MsgOn command description to obtain the information you need



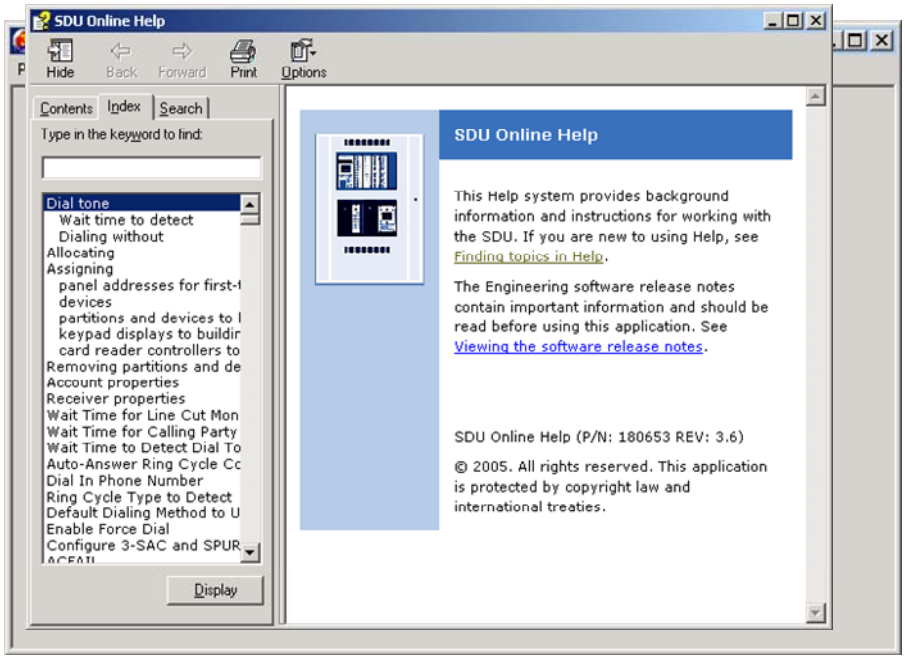
Highlight text and copy it (Control C or right mouse click) and then paste it (Control V or right mouse click)



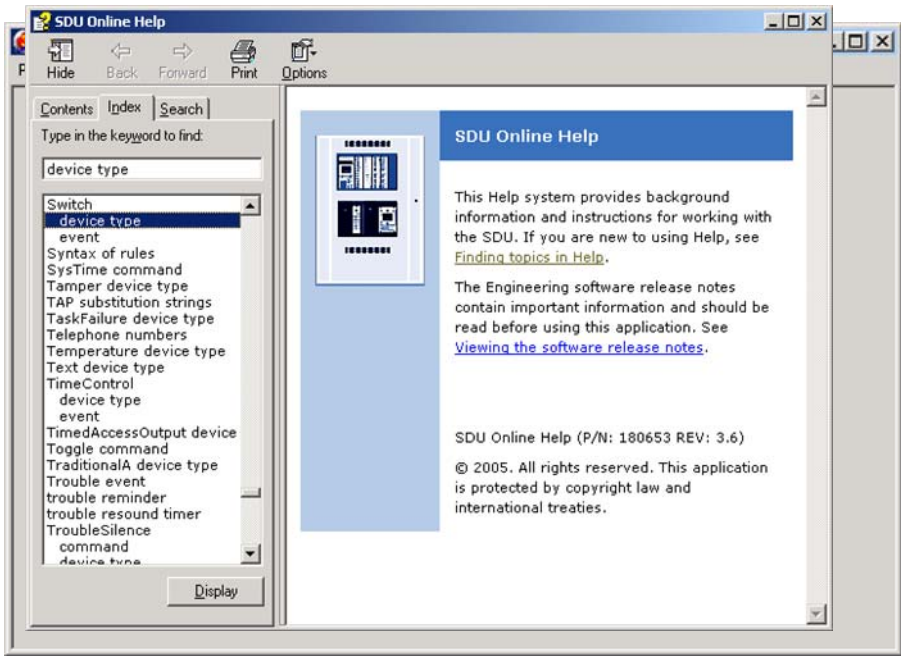
Whenever highlighted and underlined text is displayed (device type MSG for this example) click to branch to its description



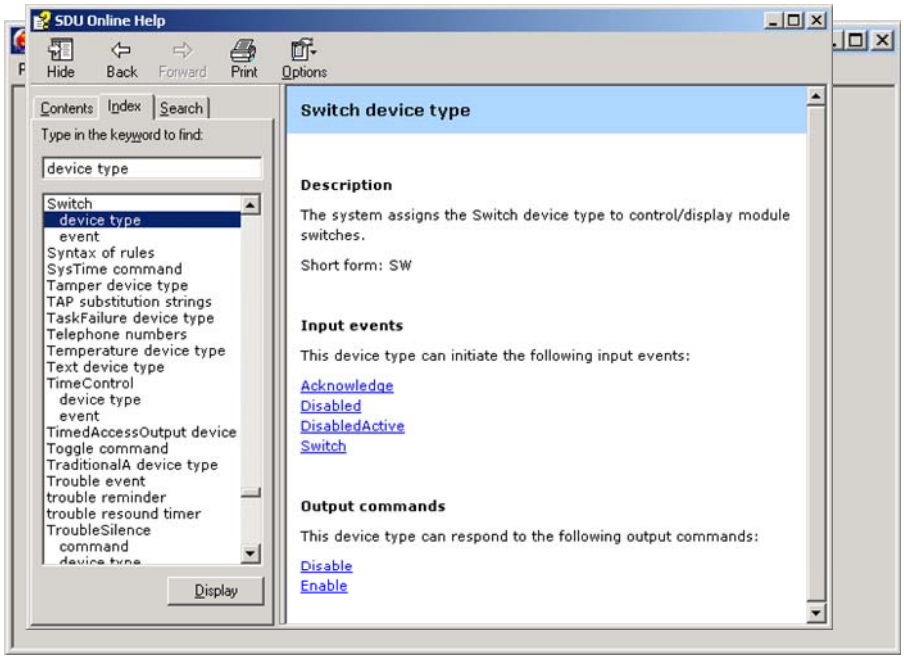
Index



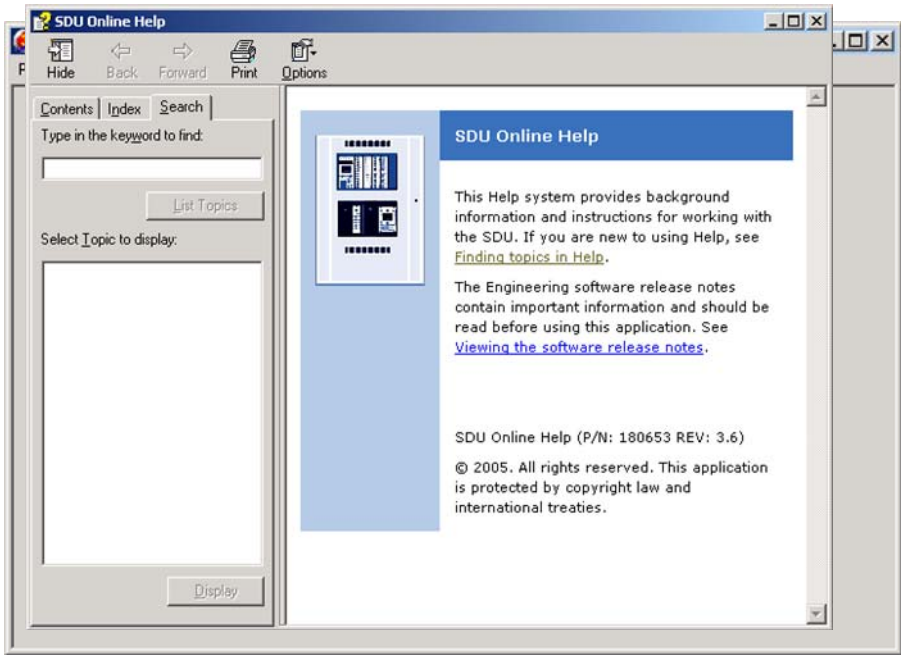
The Index tab provides a list of key words, terms and phrases. Type the word or topic you are looking for and scroll the list to select the desired topic.



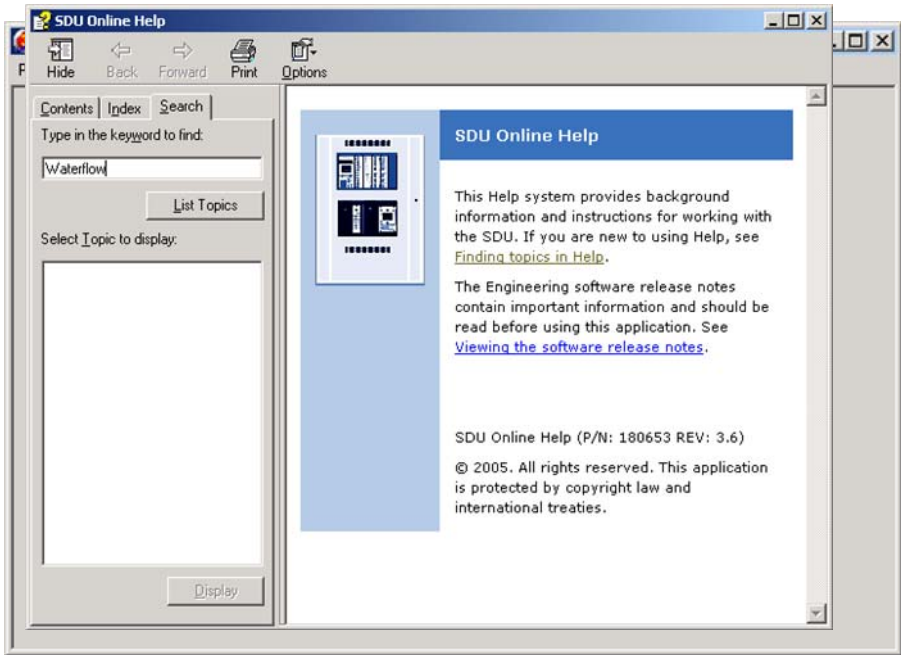
Click the Display button to display the desired topic's description



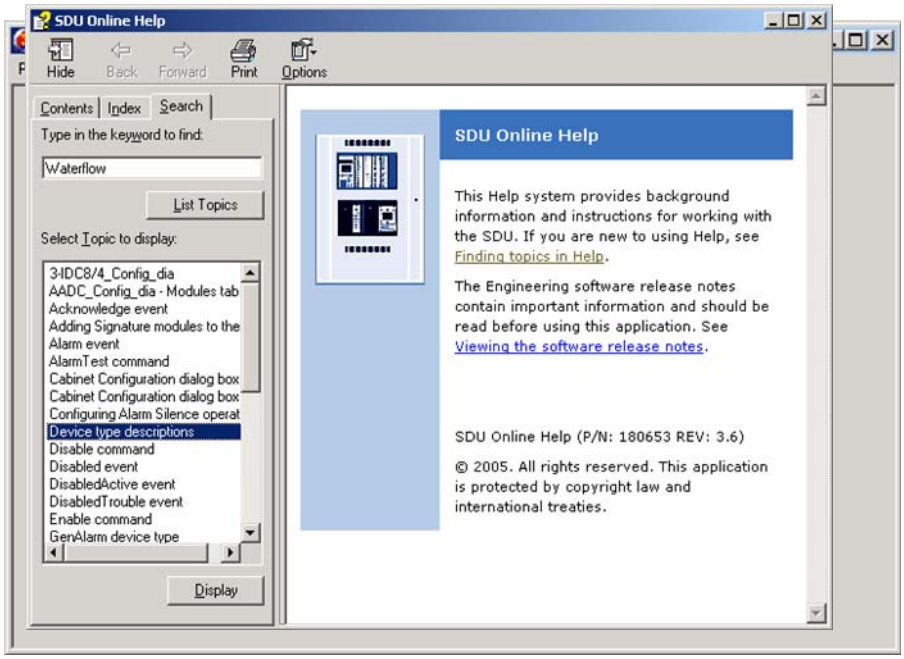
Search



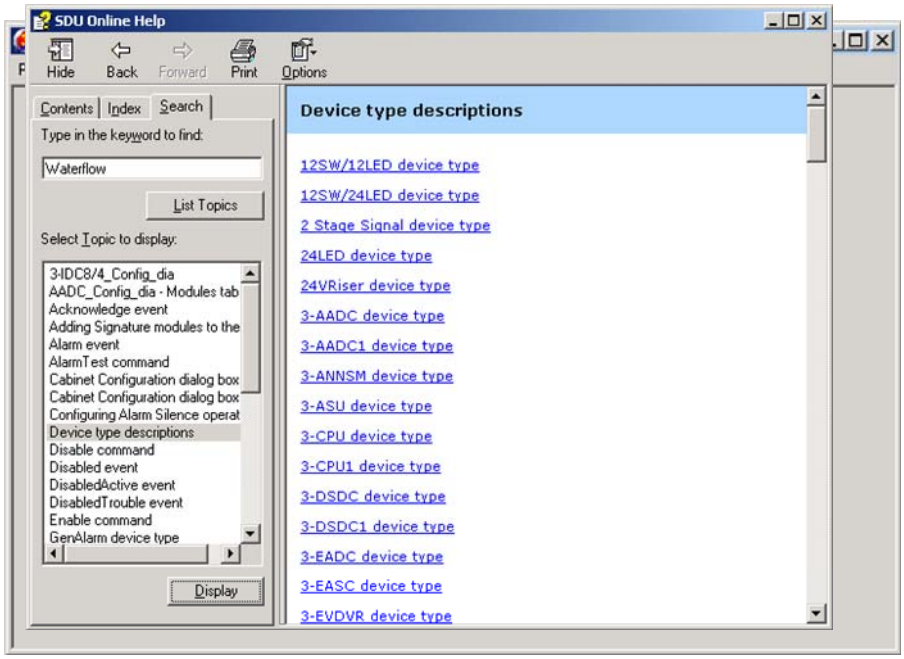
Type a word or topic and the List Topics button highlights



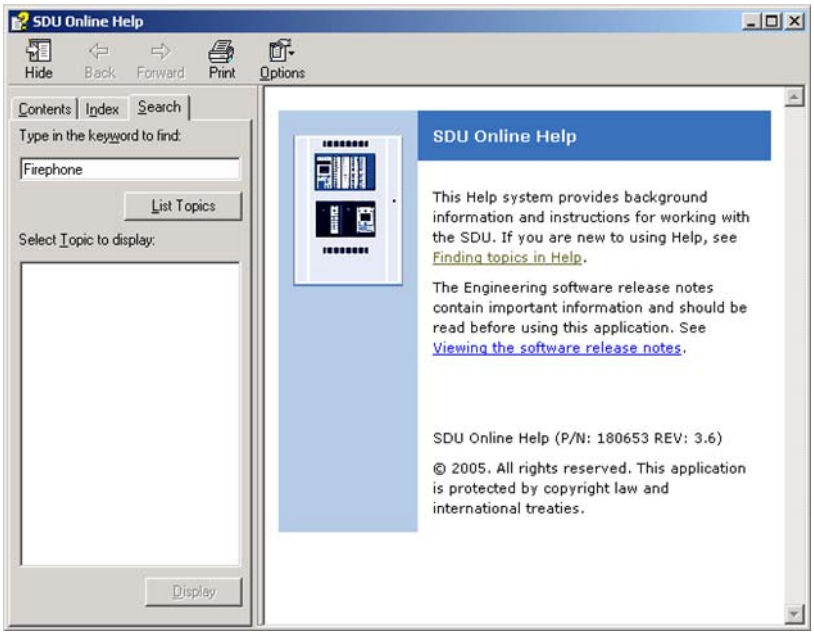
Click the List Topics button to display the desired list of topics, scroll the list and select the desired topic.



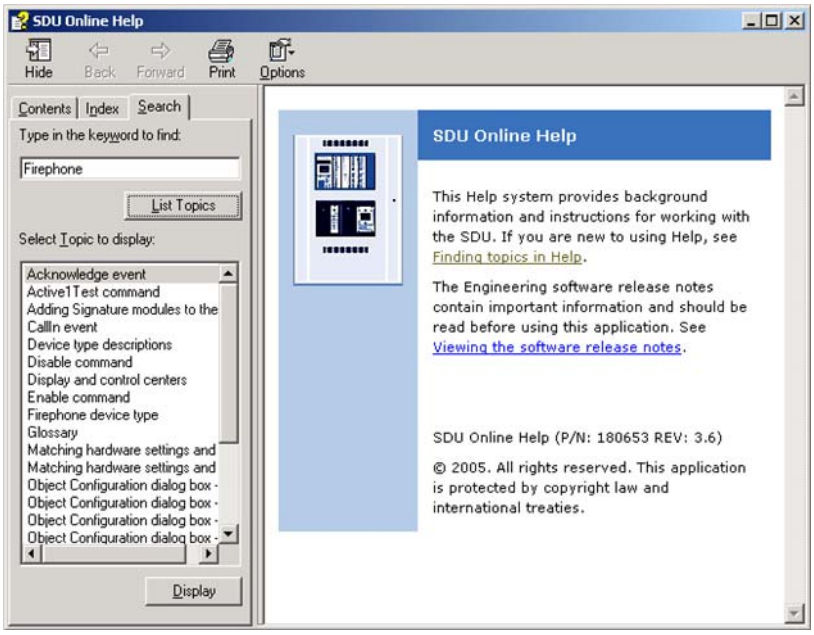
Click the Display button to display the desired topic's description.



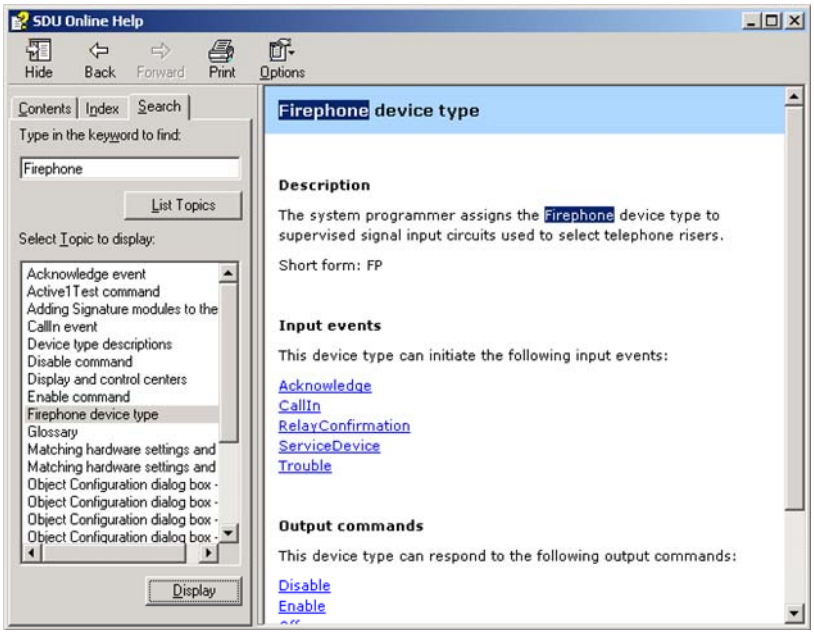
One way is to start by using the Search tab.
Type in the device type Firephone.



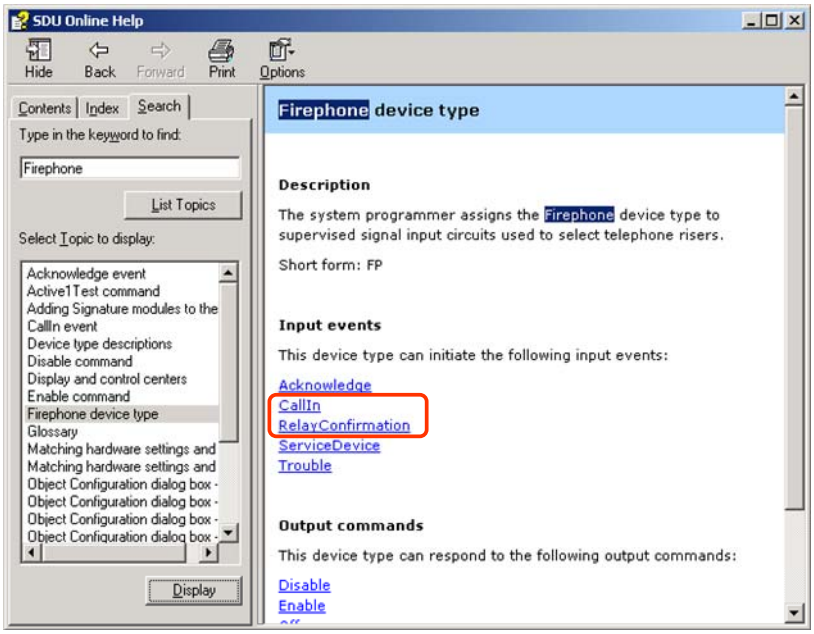
Click List Topics to display a list of topics related to the word Firephone.



Click on Firephone device type and Display to display the description for the Firephone device type.



CallIn is the input event type for the call in LED and RelayConfirmation is the input event type for the activation LED



EST3 Series Technician Certification

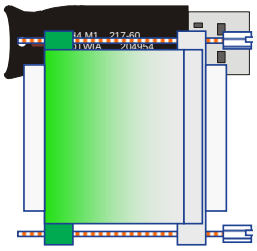
An introduction to the 3-SDU

UTC Fire & Security
A United Technologies Company

3-SDU software

A software key or hasp must be installed to use the 3-SDU

This 3-SDU software key is provided as a USB key and is also available as a key which installs on the parallel port of your pc



The 3-SDU checks and installs this hasp's software during SDU Installation

3-SDU software

The Borland Database Engine (BDE) software must be installed to use the 3-SDU

Cab1	3-LCD1	3-LCD	0	3-LCD1	
Cab1	3-CPU1	3-CPU1	0	3-CPU1	
Cab1	3-CPU1	LOCALTROUBLE	600	Annunciator_Supervision_01_01	01000600
Cab1	3-CPU1	LOCALTROUBLE	601	ClassA_Failure_01_01	01000601
Cab1	3-CPU1	GROUNDFAULT	602	Ground_Fault_Detection_01_01	01000602
Cab1	3-CPU1	LOCALTROUBLE	603	Audio_Supervision_01_01	01000603
Cab1	3-CPU1	LOCALTROUBLE	604	Internal_Fault_01_01	01000604
Cab1	3-CPU1	LOCALTROUBLE	605	Database_Supervision_01_01	01000605
Cab1	3-CPU1	LOCALTROUBLE	606	Code_Supervision_01_01	01000606
Cab1	3-CPU1	LOCALTROUBLE	607	Auxiliary_Port_One_01_01	01000607
Cab1	3-CPU1	LOCALTROUBLE	608	Auxiliary_Port_Two_01_01	01000608
Cab1	3-CPU1	LOCALTROUBLE	609	Panel_in_Download_Mode_01_01	01000609
Cab1	3-CPU1	LOCALTROUBLE	610	Network_Audio_Circuit_A_Fault_01_01	01000610
Cab1	3-CPU1	LOCALTROUBLE	611	Network_Audio_Circuit_B_Fault_01_01	01000611
Cab1	3-CPU1	LOCALTROUBLE	612	Unexpected_Card_01_01	01000612
Cab1	3-CPU1	LOCALTROUBLE	616	Network_ClassA_CircuitA_Failure_01_01	01000616
Cab1	3-CPU1	LOCALTROUBLE	617	Network_ClassA_CircuitB_Failure_01_01	01000617
Cab1	3-PS/M1	3-PS/M	0	3-PS/M1	
Cab1	3-PS/M1	LOCALTROUBLE	600	Annunciator_Supervision_01_03	01000600

The 3-SDU checks and installs this BDE software during SDU Installation

This is a 32-bit application

3-SDU software

The 3-SDU is provided on a thumb drive upon the completion of this course and updates may be obtained via our WEB site

The 3-SDU application is setup to auto run its installation

This is a 32-bit application

3-SDU software

The 3-SDU is provided on a thumb drive upon the completion of this course and updates may be obtained via our WEB site

The 3-SDU application is setup to auto run its installation

This is a 32-bit application

3-SDU software

Release 3.61 or greater of the 3-SDU may be installed without the software key by using a PIN of DEMO4C22

When installed in this manner, the 3-SDU may be used for demonstrations and for training purposes

When installed without the software key the 3-SDU will not export a project and will not communicate with the EST3 panel

You will use the 3-SDU installed without its software key to do homework later in this course

3-SDU software

Composed of the 3-SDU applications software and component software modules for the LRMs

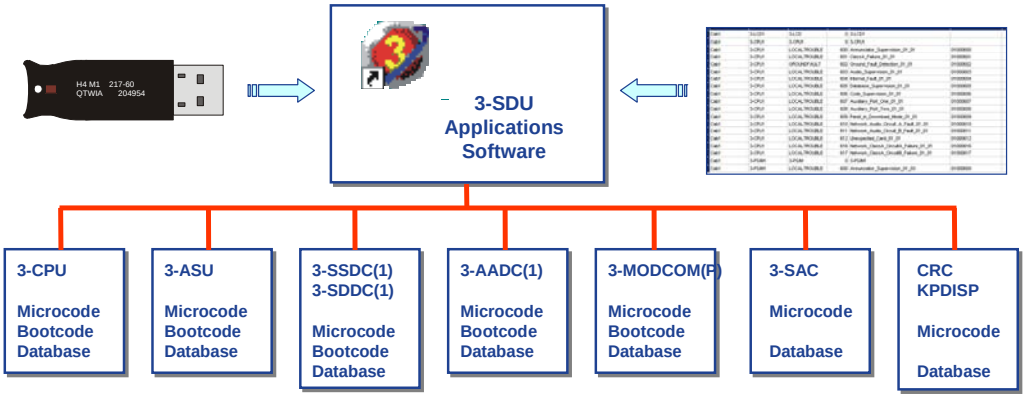
The component software modules for the LRMs are:

Microcode or applications code – the predefined default software for each LRM type for the operation of each marketplace's application

Bootcode or bootstrap – the predefined software for each LRM type to boot up its respective applications software

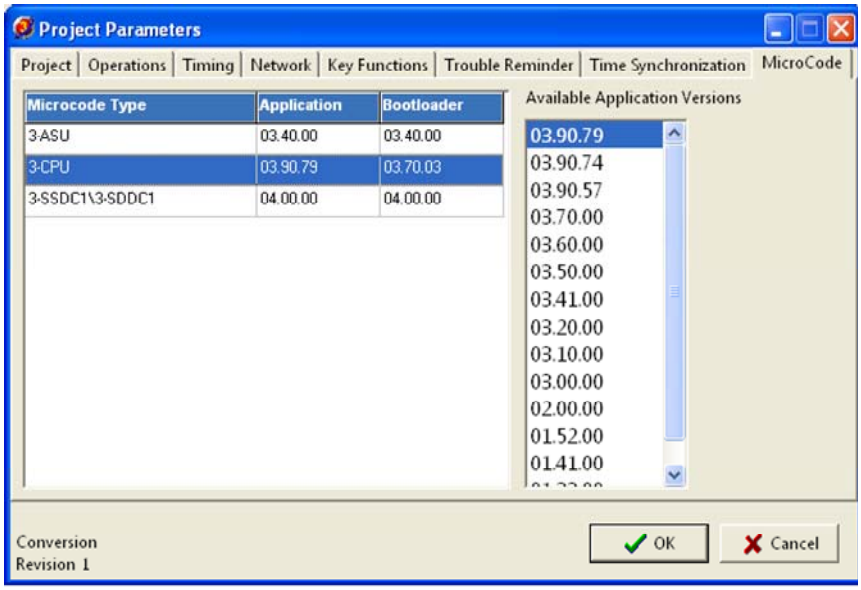
Database – Which is the customized version of the project's application.

3-SDU software



3-SDU software

Each of the component software modules may be at a different release level



EST3 Series Technician Certification

Scanning Barcode Serial Numbers

UTC Fire & Security
A United Technologies Company

Review

What did the Loop Controller do when we powered it up?

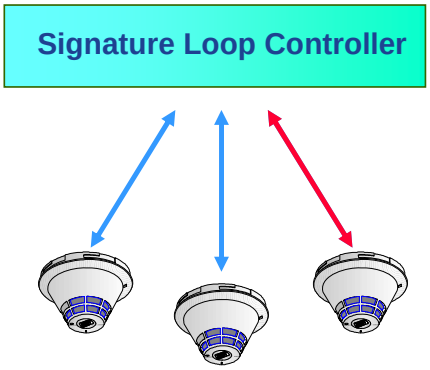
Mapped its circuit devices, giving each device a short address, creating the initial actual database.

Let's review the 3-SSDC(1) and 3-SDDC(1) mapping process

First Step

Identify all the devices on the data loop

The Signature Loop Controller asks for the highest serial numbered device that is in a New Start condition
(all Signature devices are shipped with a New Start Bit Set)



The loop controller communicates with each device until the device in a new start condition with the highest serial number is determined

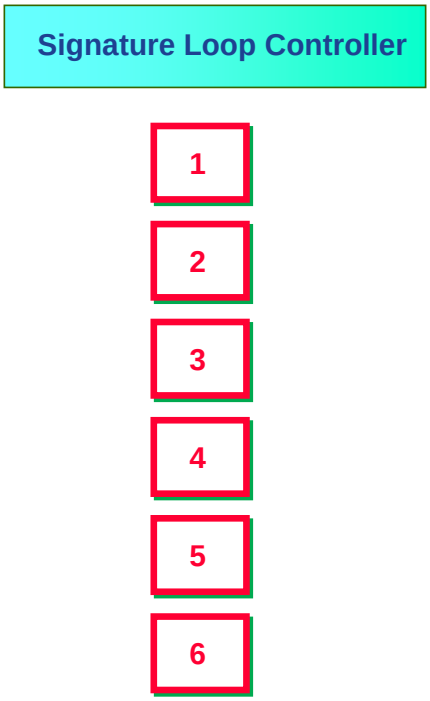
First step

When the device with the highest serial number is identified, the loop controller resets its New Start Bit and assigns the device a *short address*

The loop controller repeats this process for the next highest serial numbered device in a New Start Condition

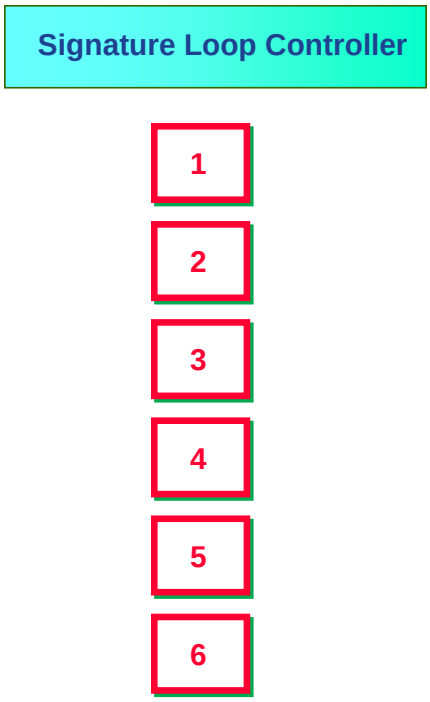
The loop controller resets the next device's New Start Bit and assign it the next *short address*

The loop controller continues this process until there are no further devices in a new start condition and has determined all the devices in its circuit



Second step

1. All of the devices are in a straight line
2. The loop controller knows what is on the line, but does not know the devices relationship with each other
3. When mapping is disabled, this is what the map looks like



Second Step
Loop Controller develops a Path List

A Path List is a list of all the devices located along the shortest electrical path between a selected device and the loop controller

The loop controller calls each device, in turn, and tracks the device's response back to the loop controller

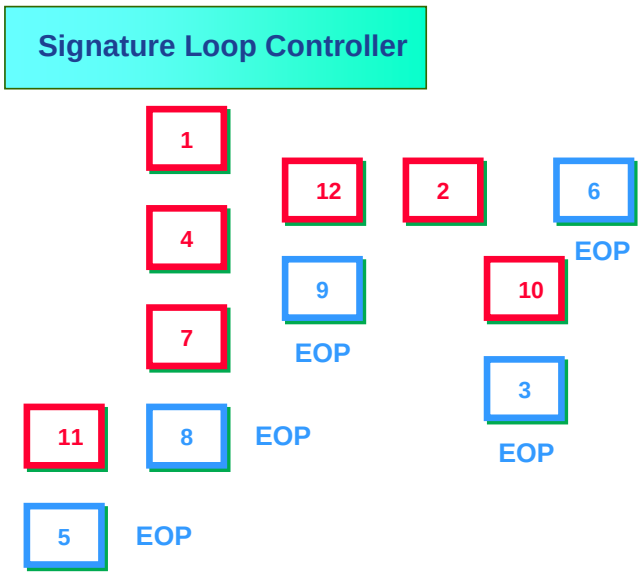
Second step

The loop controller performs this task by making each device, in turn, draw current. When a voltage drop is sensed at the device's dropping resistor, it responds to the loop controller that it is in the electrical path.

With a completed Path List, the loop controller begins to see the relative locations of each device on the loop.

The loop controller also can identify the End-of-Path (EOP) Devices.

Those farthest from the loop controller on any given electrical path.



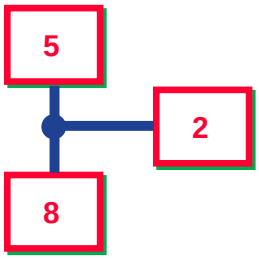
Third step

Loop controller develops a Signature List

A Signature List is a list of the series and branch connections along the shortest electrical path between each EOP device and the loop controller

A series Connection
is a single, untapped wire
between two devices

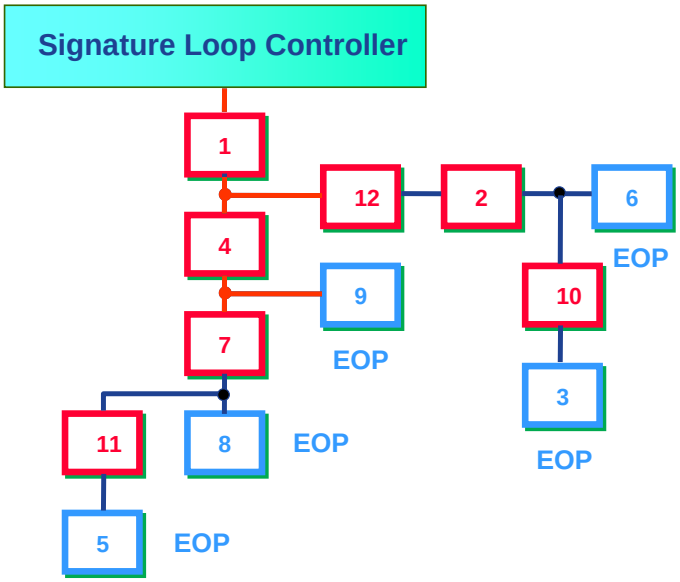
A branch Connection
is a simple T-tapped wire
between three devices



Third step

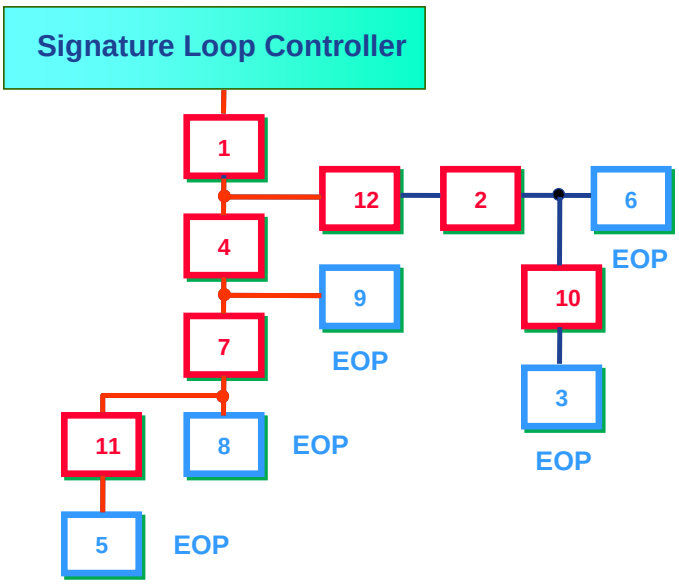
With the Signature List complete, the mapping procedure is done.

Each EOP Signature must be unique. A Signature list begins at the EOP device and ends at the loop controller.
For example, the Signature for Device 9 is branch, branch, series.



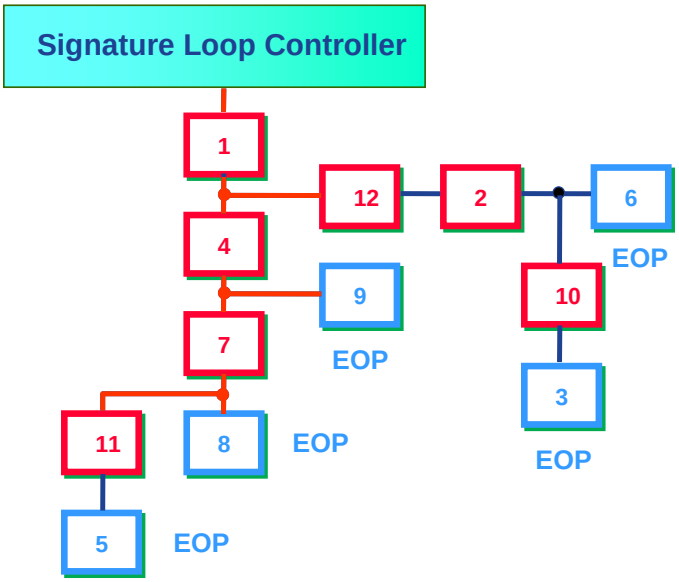
Signature of EOP device 5

Series, Branch, Branch, Branch, Series



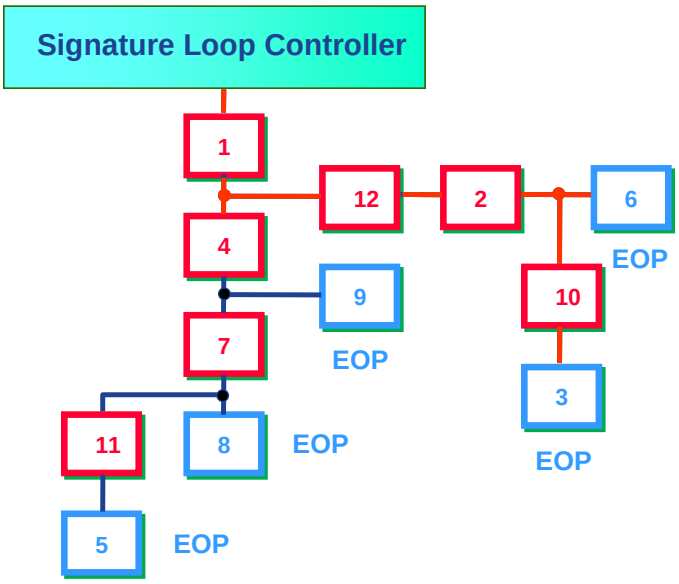
Signature of EOP device 8

Branch, Branch, Branch, Series



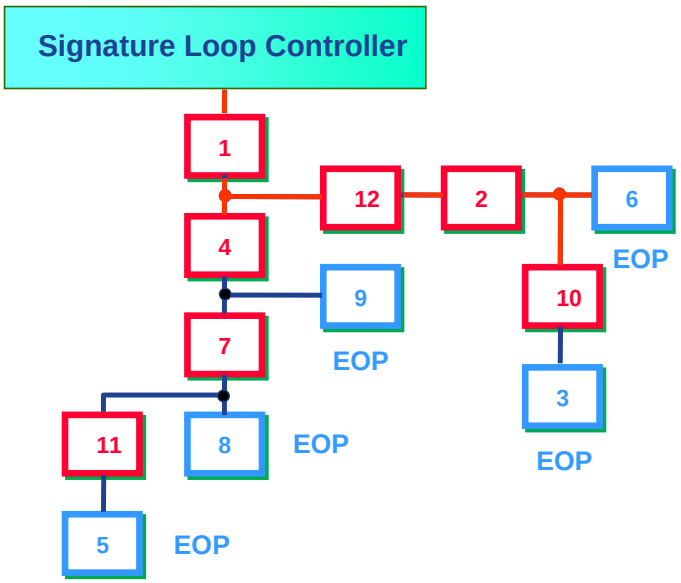
Signature of EOP device 3

Series, Branch, Series,
Branch, Series



Signature of EOP device 6

Branch, Series,
Branch, Series



Identical signatures

What is the term we use when we have identical signatures?

A Balanced Map

Are you permitted to have a balanced map?

Used to be NO!

It is now a recommendation not a requirement.

Unbalance the map by moving the T-Tap or adding a device to one of the balanced legs. If you simply assign addressing or reverse the device inputs and outputs you may achieve a green panel, but:

The system may not perform to manufacturer's specifications

Loop Controller

What happened when the Loop Controller was powered up?

It mapped the loop devices creating the actual database for the Signature devices on the loop

This actual database is resident in the loop controller

What specific data is contained for each Signature device?

Loop Controller's Actual Database

Each device's short address
Each device's serial number
Each device's model type
Each device's base (sensors only)
Each device's attributes

Short address for new installs or the existing address previously programmed

Loop Controller's Actual Database

Each device's short address
Each device's serial number
Each device's model type
Each device's base (sensors only)
Each device's attributes

Loop Controller's Actual Database

Each device's short address
Each device's serial number
Each device's model type
Each device's base (sensors only)
Each device's attributes

SIGA-IPHS, SIGA-PHS, SIGA-PS, SIGA-ION, SIGA-HFS, and SIGA-HRS for sensors
and
SIGA-270, SIGA-WTM, SIGA-CR for modules

Loop Controller's Actual Database

Each device's short address
Each device's serial number
Each device's model type
Each device's base (sensors only)
Each device's attributes

Standard, Relay or Isolation Base

Loop Controller's Actual Database

Each device's short address
Each device's serial number
Each device's model type
Each device's base (sensors only)
Each device's attributes

Such as:

- Alarm Verify (Sensors & Modules)
- Alt Alarm Verify (Sensors & Modules)
- Sensitivity (Smokes only)
- Alt Sensitivity (Smokes only)
- Pre-Alarm (Smokes only)
- Personality (Modules only)

3-SDU's Expected Database

Each device's 3-SDU configured address
Each device's 3-SDU configured model type
Each device's 3-SDU configured base (sensors only)
Each device's 3-SDU configured attributes
Each device's 3-SDU configured labels and messages

Assigned when added to the 3-SDU database, critical to application operations and must be programmed into each device on the Loop

3-SDU's Expected Database

Each device's 3-SDU configured address
Each device's 3-SDU configured model type
Each device's 3-SDU configured base (sensors only)
Each device's 3-SDU configured attributes
Each device's 3-SDU configured labels and messages

SIGA-IPHS, SIGA-PHS, SIGA-PS, SIGA-ION, SIGA-HFS, and SIGA-HRS for sensors
and
SIGA-270, SIGA-WTM, SIGA-CR for modules

3-SDU's Expected Database

Each device's 3-SDU configured address
Each device's 3-SDU configured model type
Each device's 3-SDU configured base (sensors only)
Each device's 3-SDU configured attributes
Each device's 3-SDU configured labels and messages

Standard, Relay or Isolation Base

3-SDU's Expected Database

Each device's 3-SDU configured address
Each device's 3-SDU configured model type
Each device's 3-SDU configured base (sensors only)
Each device's 3-SDU configured attributes
Each device's 3-SDU configured labels and messages

Such as:

- Alarm Verify (Sensors & Modules)
- Alt Alarm Verify (Sensors & Modules)
- Sensitivity (Smokes only)
- Alt Sensitivity (Smokes only)
- Pre-Alarm (Smokes only)
- Personality (Modules only)

3-SDU's Expected Database

Each device's 3-SDU configured address
Each device's 3-SDU configured model type
Each device's 3-SDU configured base (sensors only)
Each device's 3-SDU configured attributes
Each device's 3-SDU configured labels and messages

Not included in the loop controller's database
Eventually downloaded to all system 3-CPUs

3-SDU's Expected Database

What is missing from the 3-SDU database?

Each device's 3-SDU configured address
Each device's actual serial number
Each device's 3-SDU configured model type
Each device's 3-SDU configured base (sensors only)
Each device's 3-SDU configured attributes
Each device's 3-SDU configured labels and messages

The Serial Numbers!

Capture Serial Numbers

Capture the actual SIGA sensor and module Serial Numbers into the 3-SDU's Expected Database to correlate the two databases.

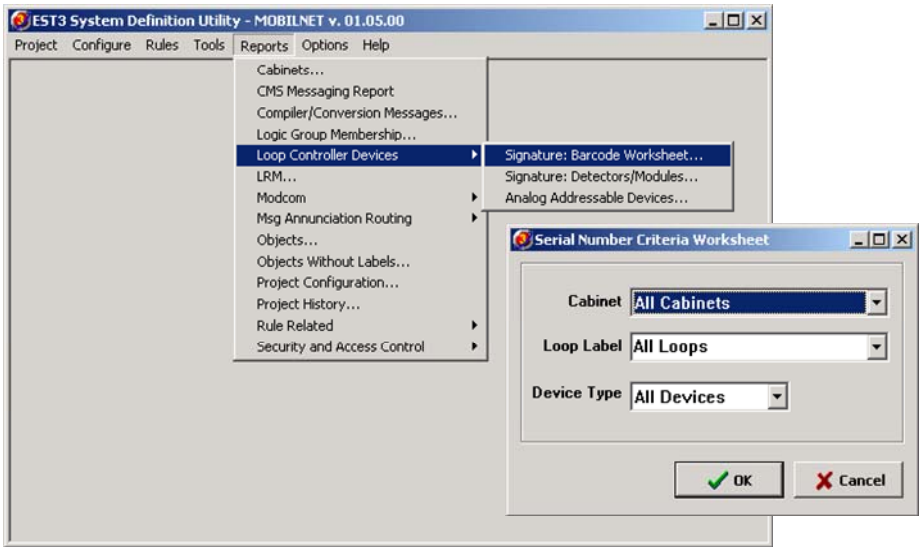
Recommended ways:

1. Scan the barcode for each device's Serial Number

2. Map devices by uploading the actual data and capturing each device's Serial Number when reconciling the two databases

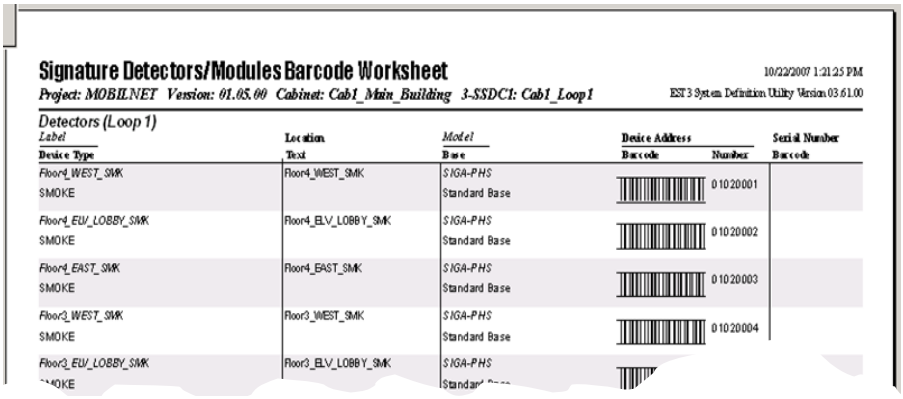
Scan Serial Numbers

Select Reports from Main Menu, Loop Controller Devices and Signature Barcode Worksheet from pull down menus, and then All Devices, Detectors or Modules from dialog box



Scan Serial Numbers

Print the displayed Barcode Worksheet from this screen using the print icon. These worksheets are sent to the field site where the system is installed.



Signature Detectors/Modules Barcode Worksheet

10/22/2007 1:21:25 PM

Project: MOBILNET Version: 01.05.00 Cabinet: Cab1_Main_Building 3-SSDC1: Cab1_Loop1

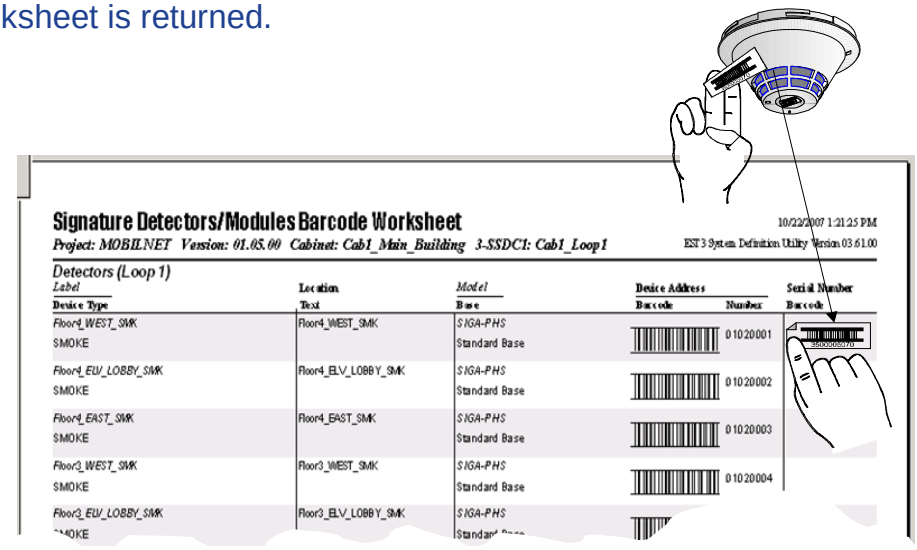
EST3 System Definition Utility Version 03.61.00

Detectors (Loop 1)

Label / Device Type	Location Text	Model Base	Device Address Barcode	Number	Serial Number Barcode
Floor4_WEST_SMK SMOKE	Floor4_WEST_SMK	SIGA-PHS Standard Base		01020001	
Floor4_ELV_LOBBY_SMK SMOKE	Floor4_ELV_LOBBY_SMK	SIGA-PHS Standard Base		01020002	
Floor4_EAST_SMK SMOKE	Floor4_EAST_SMK	SIGA-PHS Standard Base		01020003	
Floor3_WEST_SMK SMOKE	Floor3_WEST_SMK	SIGA-PHS Standard Base		01020004	
Floor3_ELV_LOBBY_SMK	Floor3_ELV_LOBBY_SMK	SIGA-PHS			

Scan Serial Numbers

At the field site each device's Barcode Label is attached to the appropriate position on the worksheet. The completed worksheet is returned.



Signature Detectors/Modules Barcode Worksheet

Project: MOBILNET Version: 01.05.00 Cabinet: Cab1_Main_Building 3-SSDC1: Cab1_Loop1

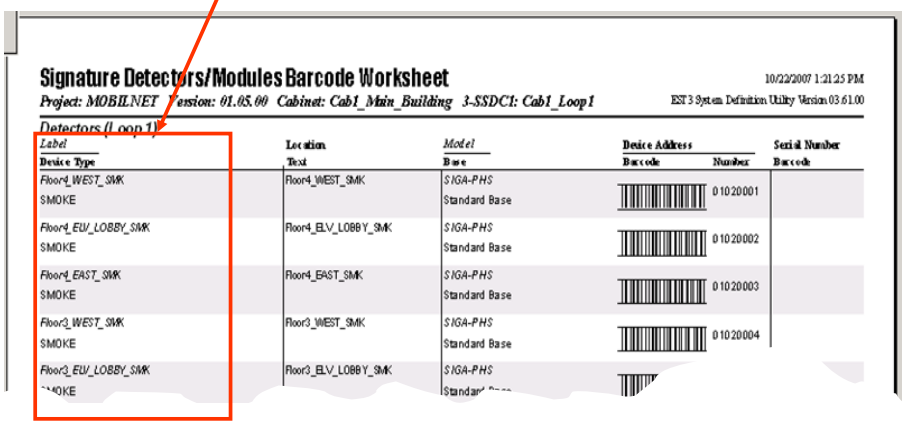
10/22/2007 1:21:25 PM
EST3 System Definition Utility Version 03.61.00

Detectors (Loop 1)

Label / Device Type	Location Text	Model Base	Device Address Barcode / Number	Serial Number Barcode
Floor4_WEST_SMK SMOKE	Floor4_WEST_SMK	SIGA-PHS Standard Base	01020001	
Floor4_ELV_LOBBY_SMK SMOKE	Floor4_ELV_LOBBY_SMK	SIGA-PHS Standard Base	01020002	
Floor4_EAST_SMK SMOKE	Floor4_EAST_SMK	SIGA-PHS Standard Base	01020003	
Floor3_WEST_SMK SMOKE	Floor3_WEST_SMK	SIGA-PHS Standard Base	01020004	
Floor3_ELV_LOBBY_SMK SMOKE	Floor3_ELV_LOBBY_SMK	SIGA-PHS Standard Base		

Scan Serial Numbers

Each detector or module device object Label and Device Type is listed



Signature Detectors/Modules Barcode Worksheet

Project: MOBILNET Version: 01.05.00 Cabinet: Cab1_Main_Building 3-SSDC1: Cab1_Loop1

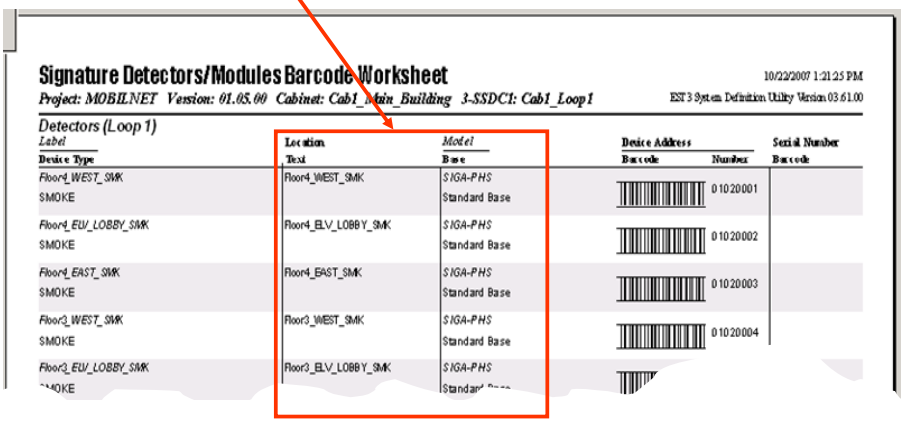
10/22/2007 1:21:25 PM
EST3 System Definition Utility Version 03.61.00

Detectors (Loop 1)

Label / Device Type	Location Text	Model Base	Device Address Barcode / Number	Serial Number Barcode
Floor4_WEST_SMK SMOKE	Floor4_WEST_SMK	SIGA-PHS Standard Base	01020001	
Floor4_ELV_LOBBY_SMK SMOKE	Floor4_ELV_LOBBY_SMK	SIGA-PHS Standard Base	01020002	
Floor4_EAST_SMK SMOKE	Floor4_EAST_SMK	SIGA-PHS Standard Base	01020003	
Floor3_WEST_SMK SMOKE	Floor3_WEST_SMK	SIGA-PHS Standard Base	01020004	
Floor3_ELV_LOBBY_SMK SMOKE	Floor3_ELV_LOBBY_SMK	SIGA-PHS Standard Base		

Scan Serial Numbers

Each device's Location Text Message, Model and Base (Detectors) are listed



Signature Detectors/Modules Barcode Worksheet

Project: MOBILNET Version: 01.05.00 Cabinet: Cab1_Main_Building 3-SSDC1: Cab1_Loop1

10/22/2007 1:21:25 PM
EST3 System Definition Utility Version 03.61.00

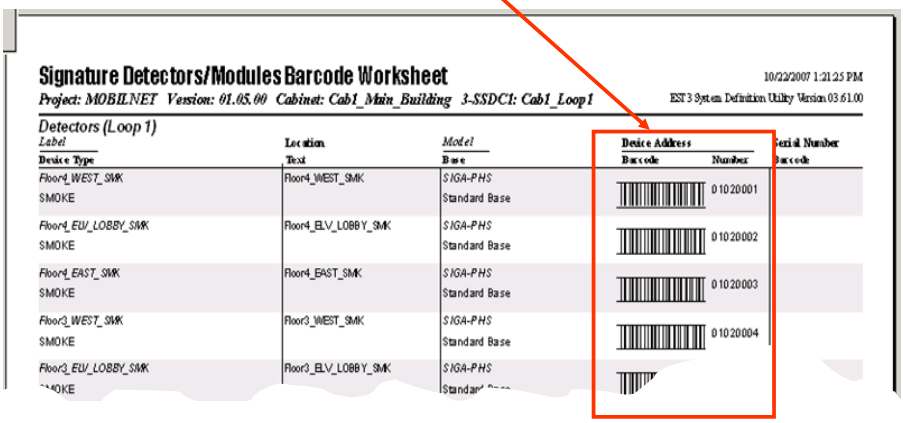
Detectors (Loop 1)

Label / Device Type	Location Text	Model / Base	Device Address Barcode	Device Address Number	Serial Number Barcode
Floor4_WEST_SMK SMOKE	Floor4_WEST_SMK	SIGA-PHS Standard Base		01020001	
Floor4_ELV_LOBBY_SMK SMOKE	Floor4_ELV_LOBBY_SMK	SIGA-PHS Standard Base		01020002	
Floor4_EAST_SMK SMOKE	Floor4_EAST_SMK	SIGA-PHS Standard Base		01020003	
Floor3_WEST_SMK SMOKE	Floor3_WEST_SMK	SIGA-PHS Standard Base		01020004	
Floor3_ELV_LOBBY_SMK SMOKE	Floor3_ELV_LOBBY_SMK	SIGA-PHS Standard Base			

For modules, the personality is listed

Scan Serial Numbers

Each device's Device Address Barcode and Number are listed



Signature Detectors/Modules Barcode Worksheet

Project: MOBILNET Version: 01.05.00 Cabinet: Cab1_Main_Building 3-SSDC1: Cab1_Loop1

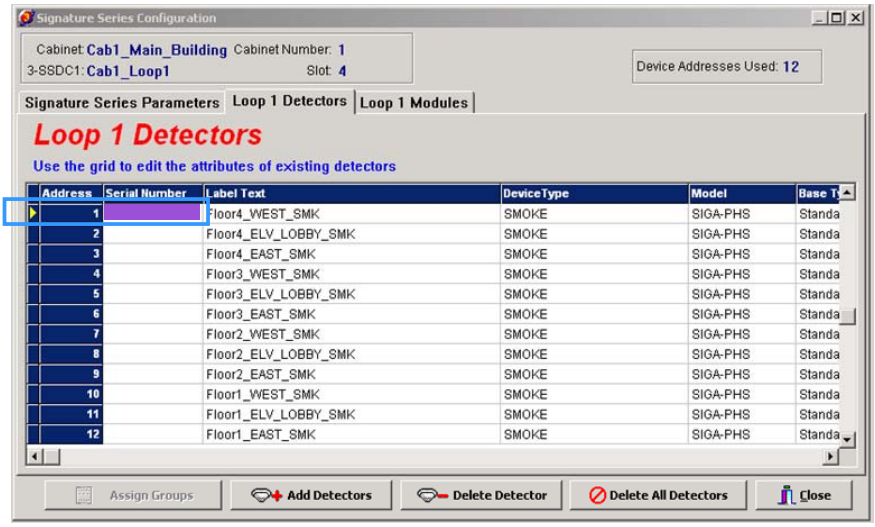
10/22/2007 1:21:25 PM
EST3 System Definition Utility Version 03.61.00

Detectors (Loop 1)

Label / Device Type	Location Text	Model / Base	Device Address Barcode	Device Address Number	Serial Number Barcode
Floor4_WEST_SMK SMOKE	Floor4_WEST_SMK	SIGA-PHS Standard Base		01020001	
Floor4_ELV_LOBBY_SMK SMOKE	Floor4_ELV_LOBBY_SMK	SIGA-PHS Standard Base		01020002	
Floor4_EAST_SMK SMOKE	Floor4_EAST_SMK	SIGA-PHS Standard Base		01020003	
Floor3_WEST_SMK SMOKE	Floor3_WEST_SMK	SIGA-PHS Standard Base		01020004	
Floor3_ELV_LOBBY_SMK SMOKE	Floor3_ELV_LOBBY_SMK	SIGA-PHS Standard Base			

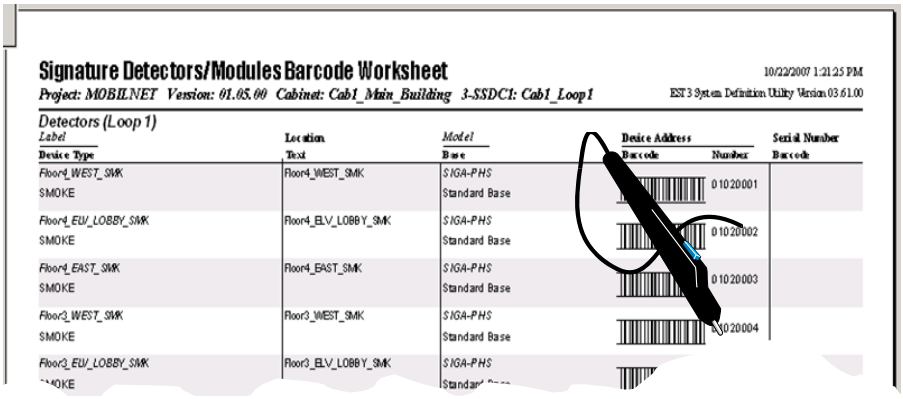
Scan Serial Numbers

When the selected loop's sensor or module configuration screen displays, the device with the first address (1 for sensors and 126 for modules) is at the top of the list



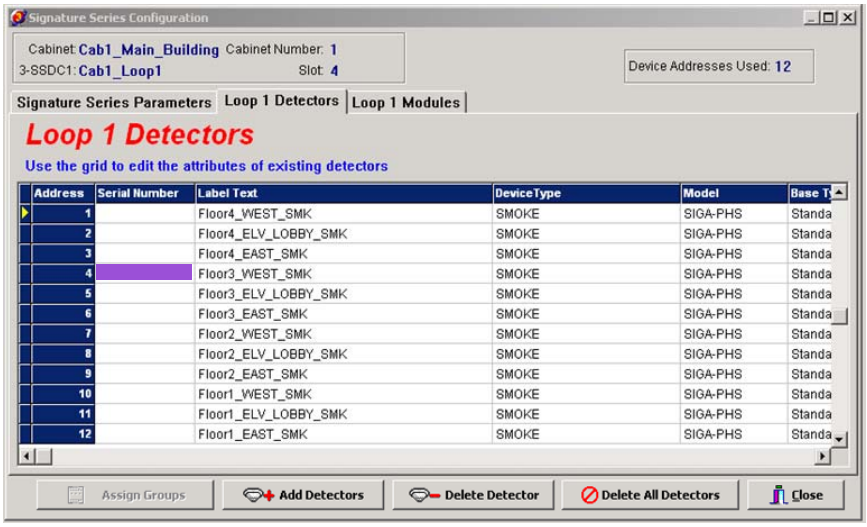
Scan Serial Numbers

Scan the desired Device Address Barcode (4 for this example) to select the desired device to the top of the screen



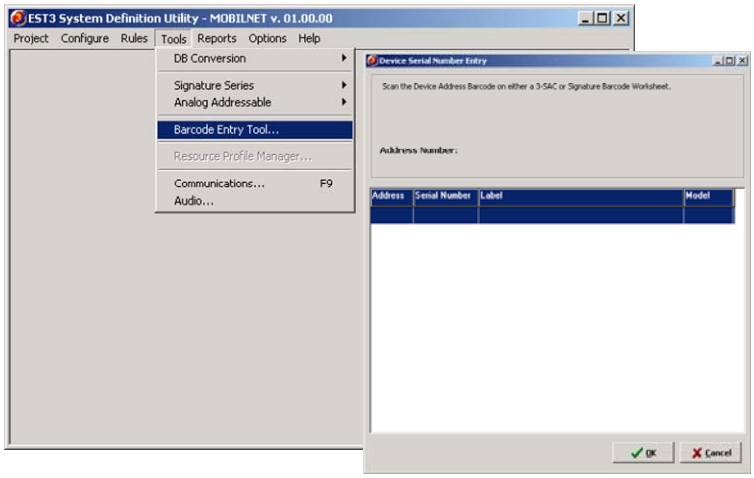
Scan Serial Numbers

Address 4's device is now selected. Scan the Serial Number as you select each device



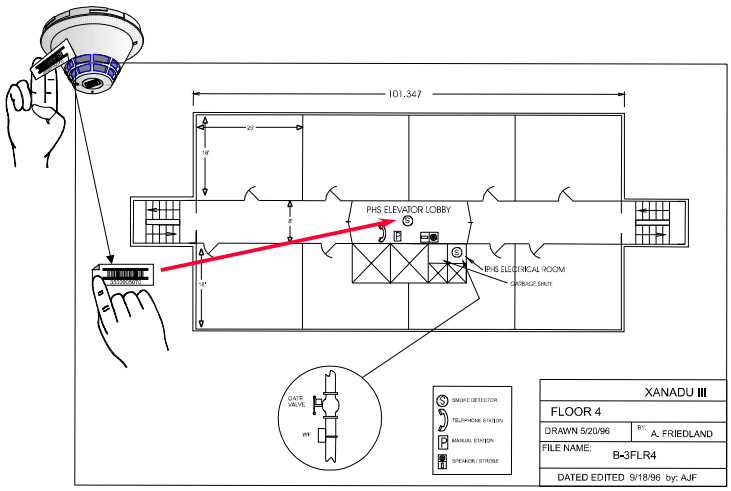
Scan Serial Numbers

The 3-SDU provides a short cut to scan barcodes using the Barcode Entry Tool. Select this tool and scan the address barcodes and related serial number barcodes.



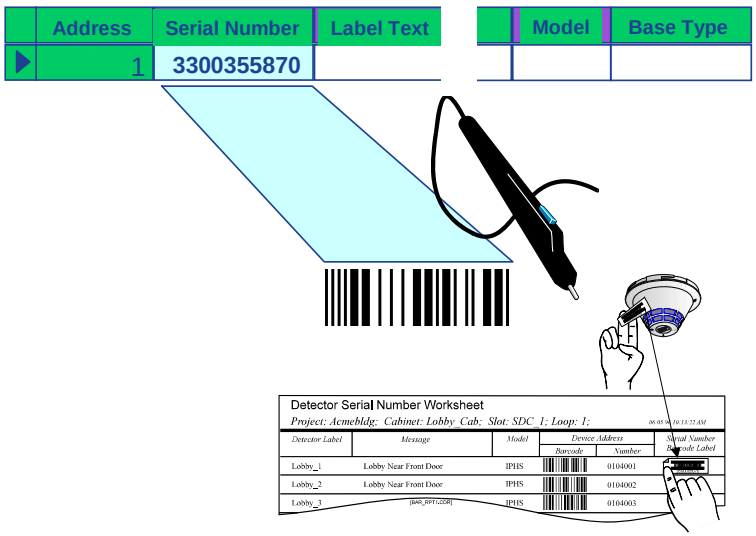
Scan Serial Numbers

Use the site drawings to achieve the same result



Scan Serial Numbers

Scanning Serial Numbers is easy



EST3 Series Technician Certification

Reconcile Actual and Expected Databases

UTC Fire & Security
A United Technologies Company

Actual vs. Expected

3-SSDC

The ACTUAL Database is resident in the loop controller and contains the undesired device addressing established at installation and the Maintenance building Serial Numbers missing from the expected data base

The EXPECTED Database is resident in the 3-SDU and contains the desired device addressing established during the configuration process but is missing the Maintenance Building Serial Numbers

Reconcile databases

Q: How do we resolve the differences between the development environment and the actual field installation?

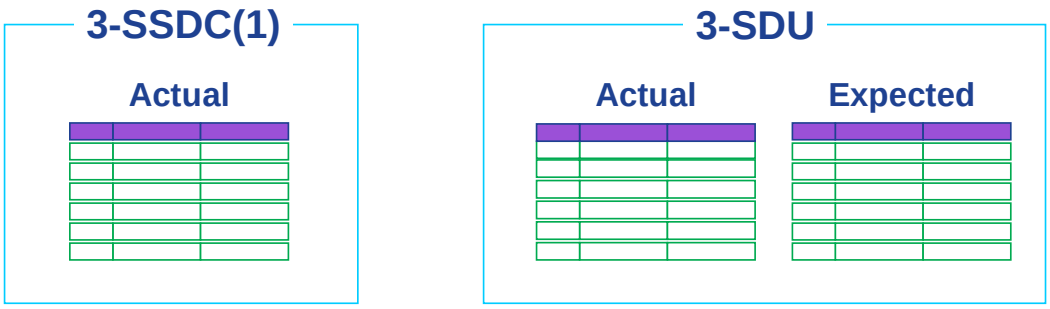
A: By RECONCILING the ACTUAL and EXPECTED Databases

Connections

Connect the upload/download cable between the 3-SSDC(1) or 3-SDDC(1) in standalone mode or any 3-CPU in network mode and Comm Port 1 on your computer

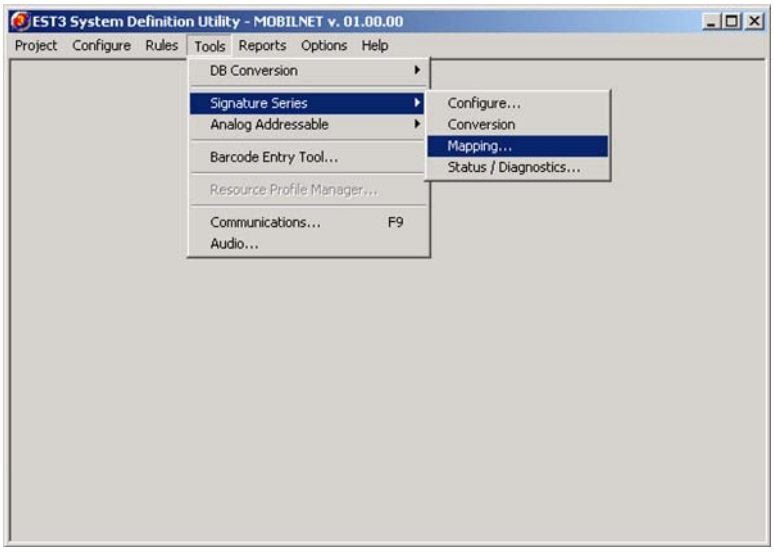
Upload Actual data

Upload the Actual data from the 3-SSDC(1) or 3-SDDC (1) into the 3-SDU and map the actual data vs. the expected data by matching serial numbers



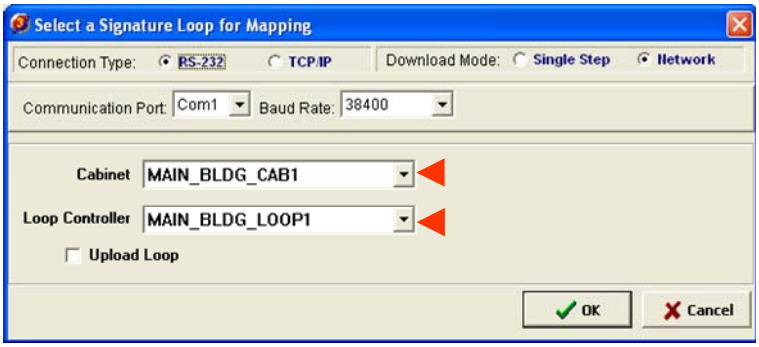
Upload and map Actual data

To upload and map the Actual data into the 3-SDU, Select Tools, Signature Series and Mapping



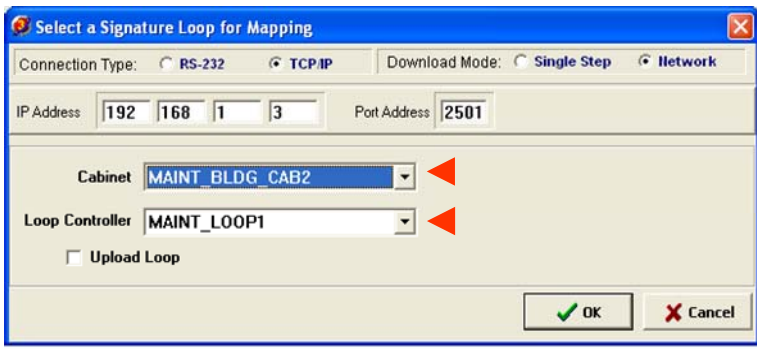
From the Select a Signature Loop for Mapping Dialog Box select:

1. RS-232 Connection Type
2. Network Download Mode
3. Comm Port and Baud Rate
4. Cabinet
5. Loop Controller
6. Upload Loop check box
7. OK

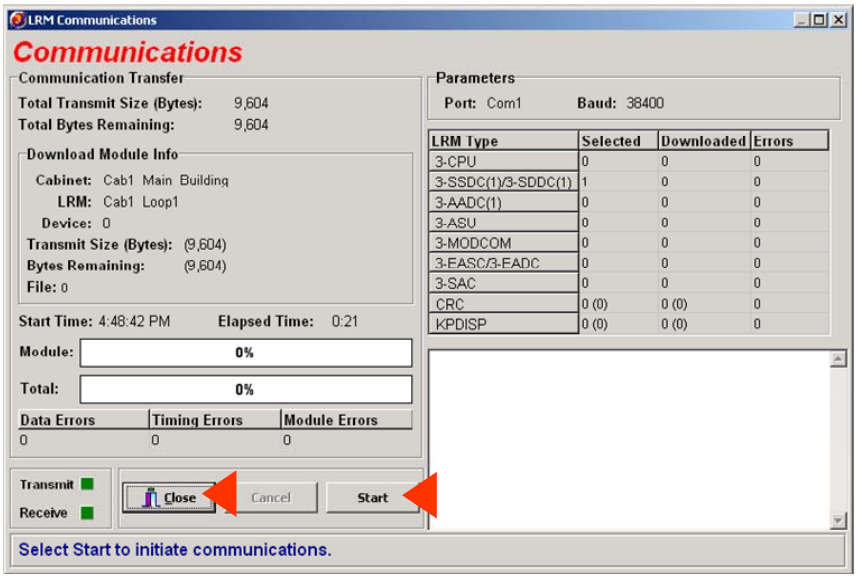


Another method of communication is TCP/IP:

1. Click TCP/IP
2. Click Single Step or Network
3. Type the IP address of the panel
4. Use default Port Address
5. Select Cabinet and Loop Controller
6. Select Upload
7. Click OK

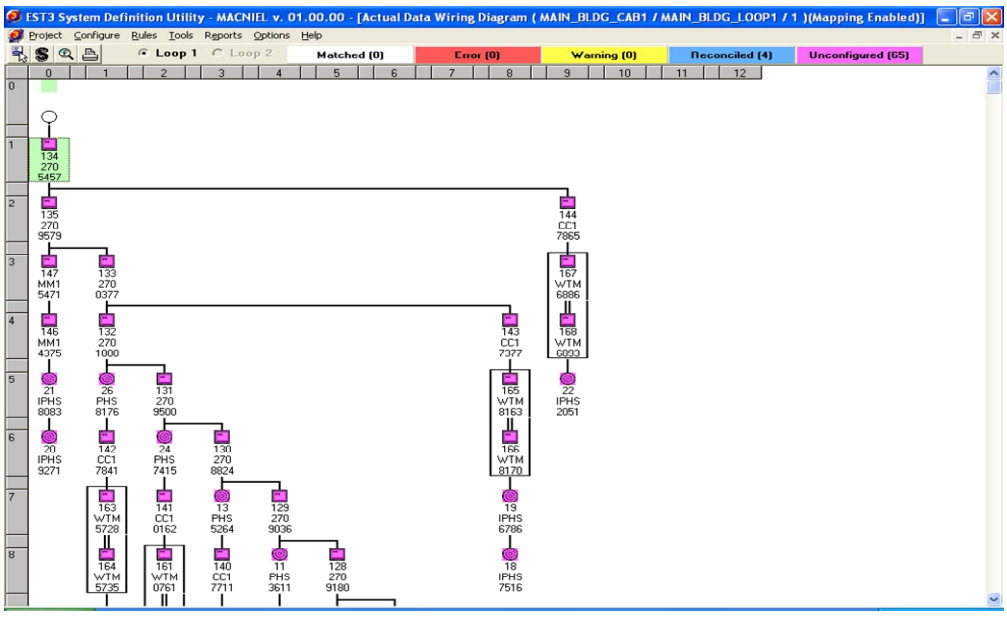


1. Click Start to initiate communications
2. Click Close when communications is complete



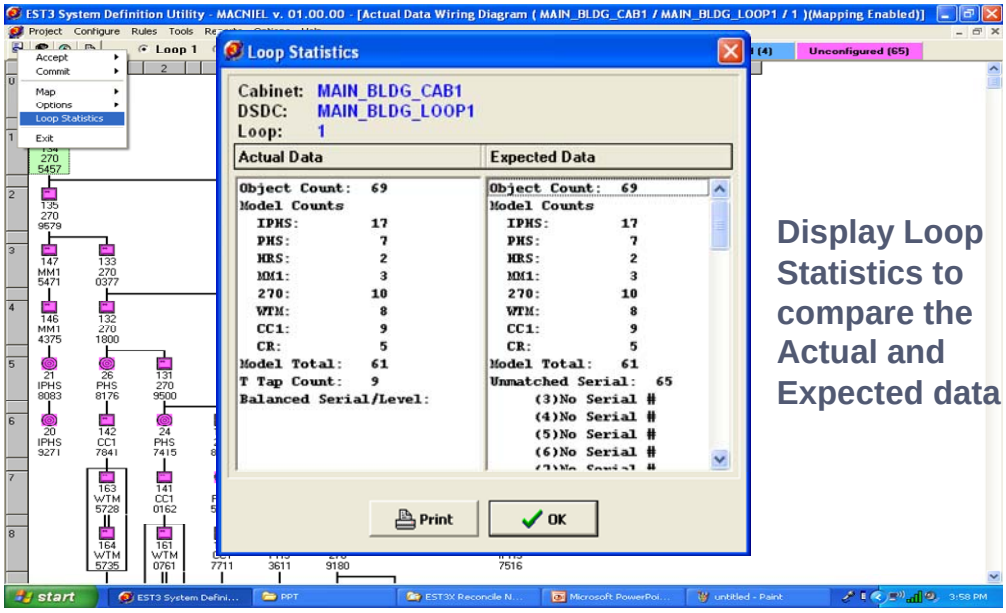
Actual Data Wiring Diagram

The map displays



Menu button

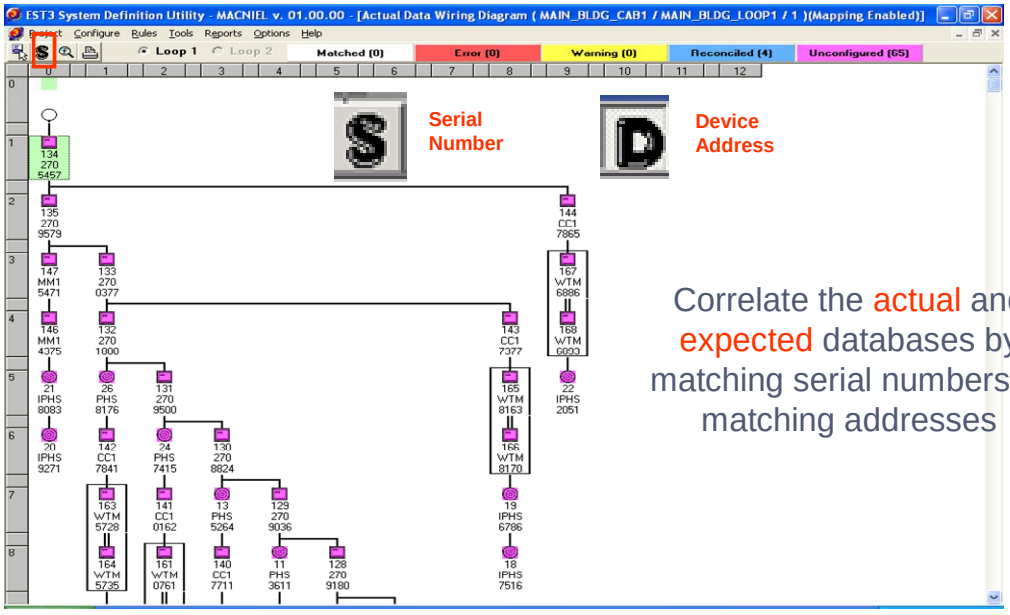
The first button is the Menu 



Display Loop Statistics to compare the Actual and Expected data

Serial number vs. address

Select a match by serial number or device address



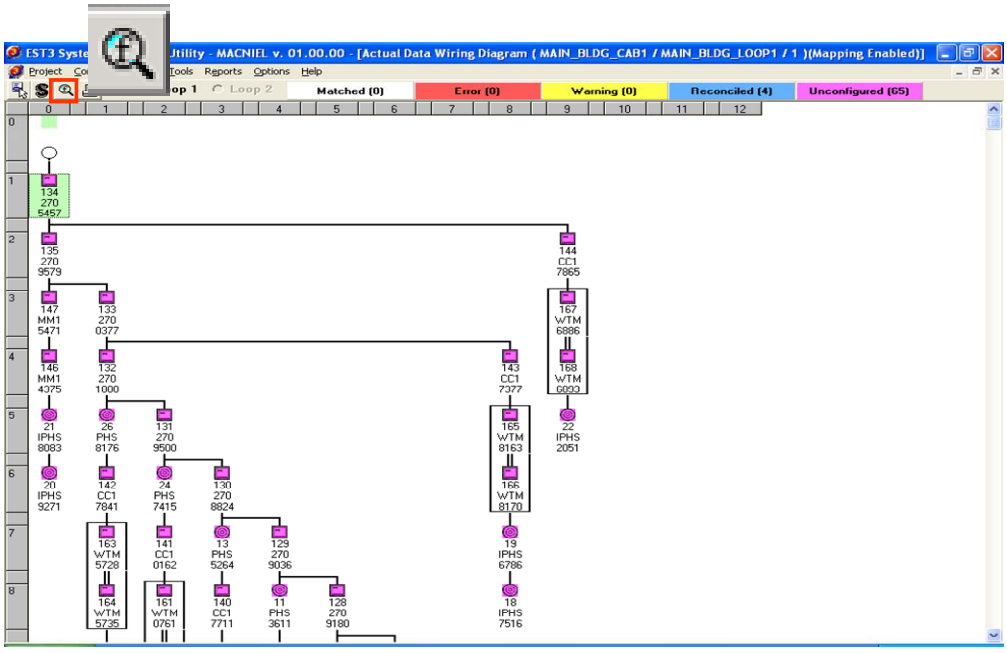
Correlate the actual and expected databases by matching serial numbers or matching addresses

Conflict between Actual and Expected

- With the default S selected, the actual vs. expected mapping is accomplished by matching serial numbers
- Select match by device address D to repair the expected database

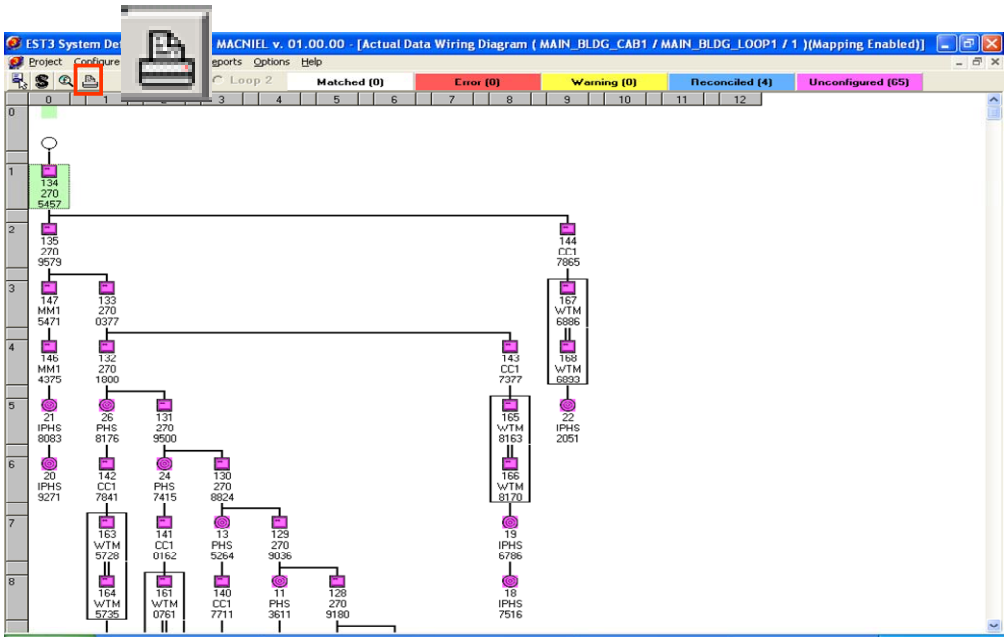
Search

Search data within the map



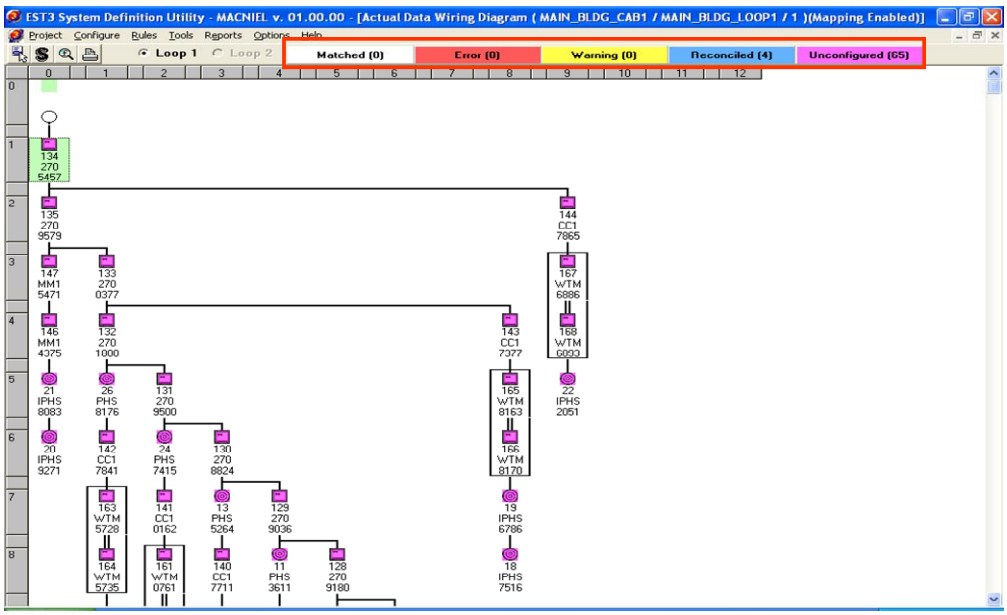
Print

Print the displayed view of the map



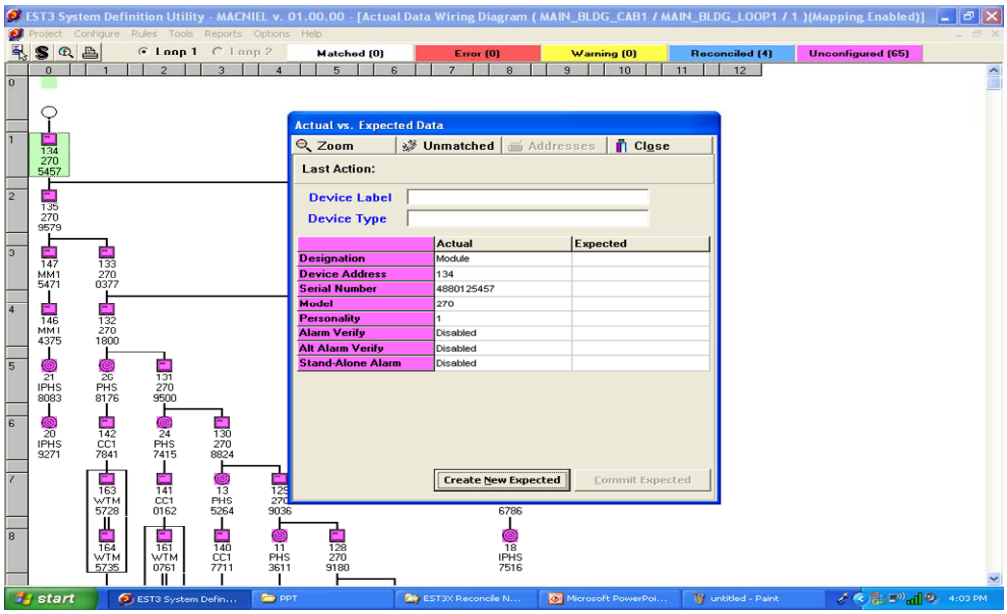
Conflict between Actual and Expected

White = match; Red = error; Yellow = warning;
Blue = reconciled; Pink = unconfigured



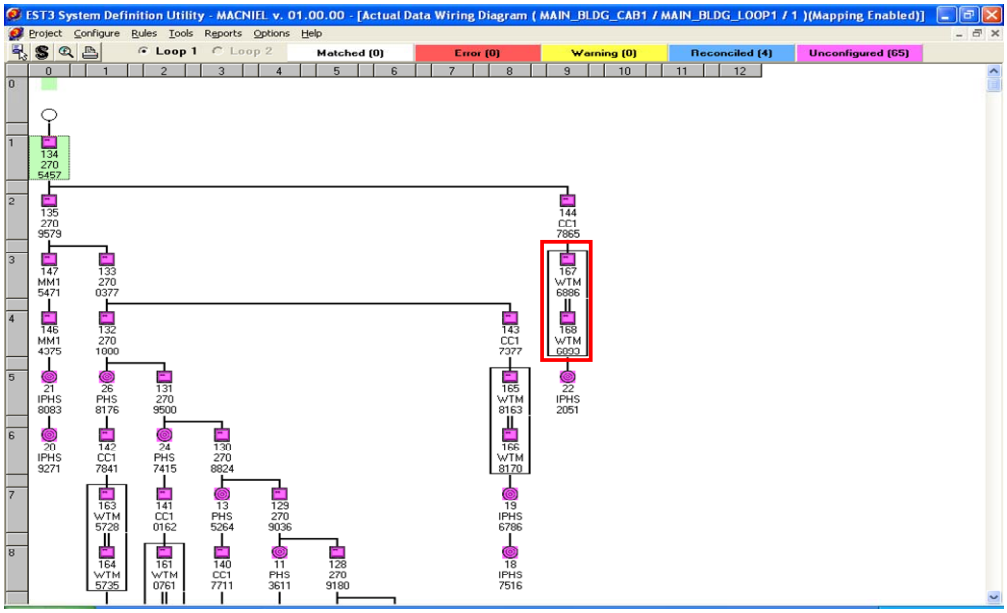
Actual/Expected Data

Double click the device to display the Actual/Expected Data Form

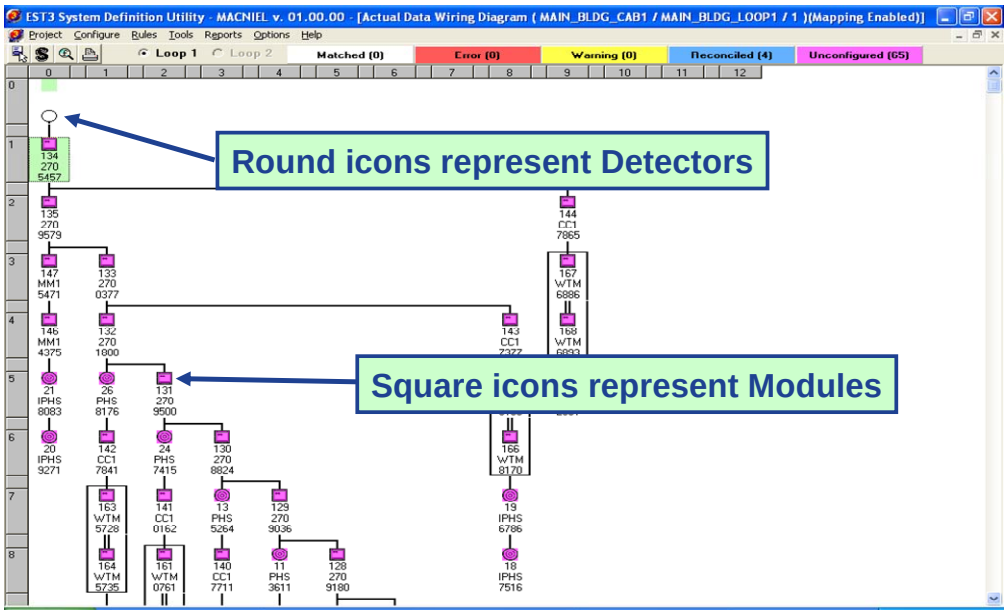


Double line

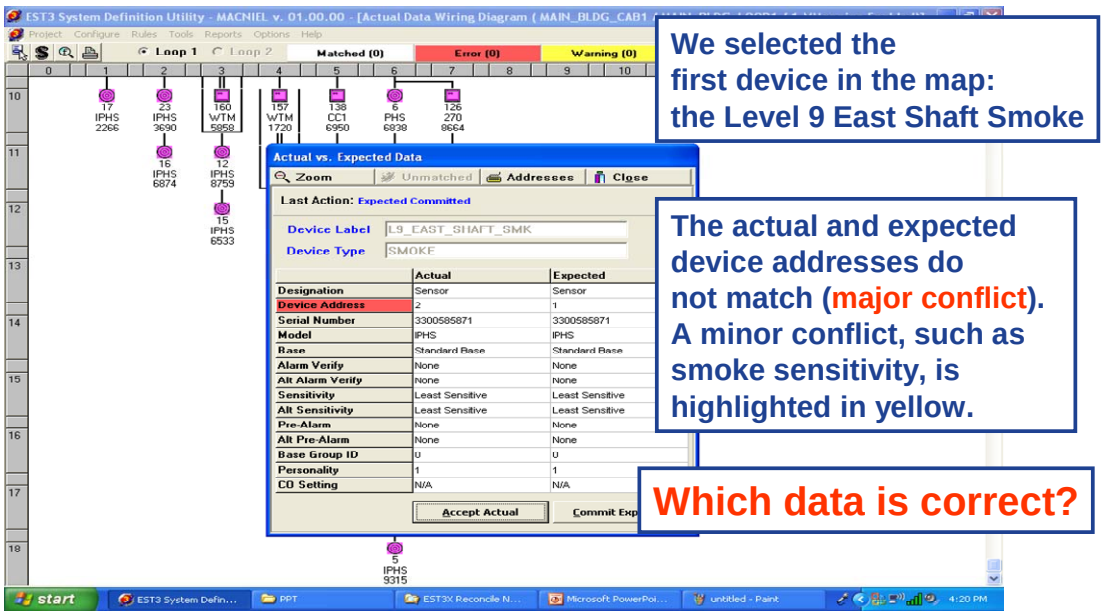
A double line indicates a dual address device: SIGA-WTM, SIGA-CT2, and SIGA-CC2



Device icon



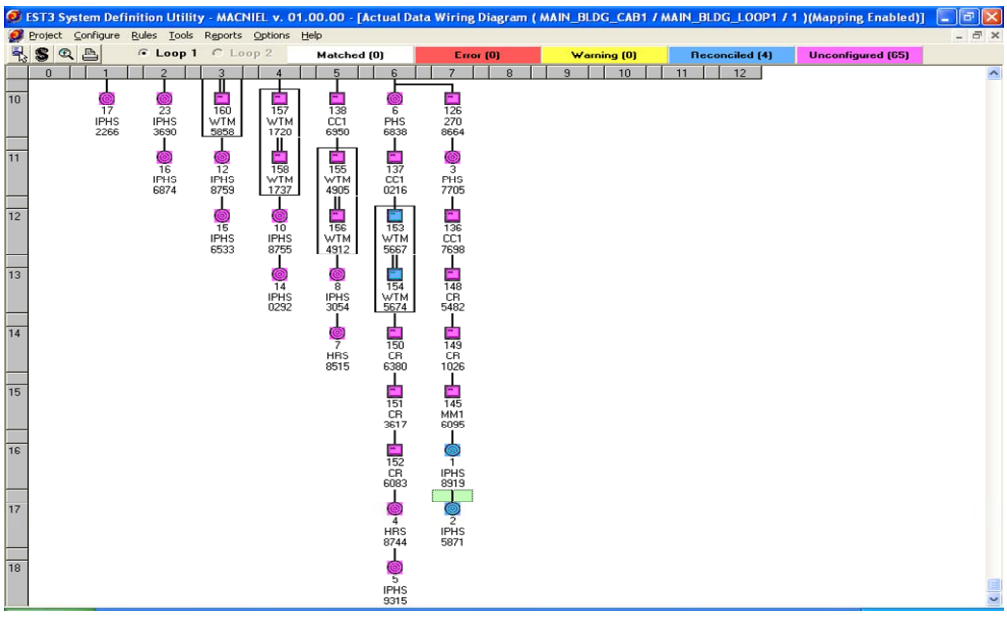
To individually reconcile each device, double click the desired device in the map (L9_EAST_SHAFT_SMK) and click Show Actual/Expected icon



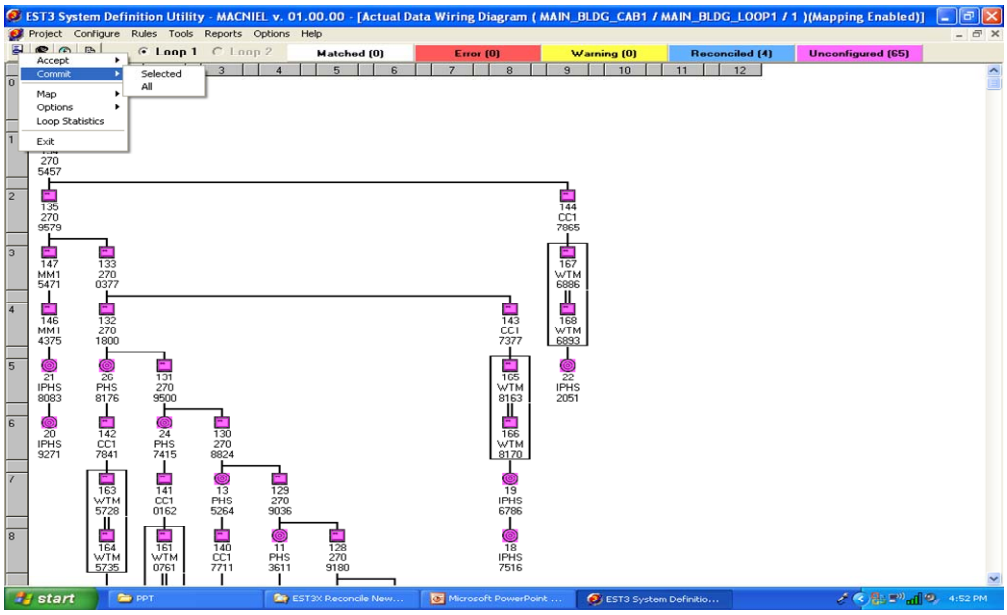
If you said the Expected Data you are...

Correct!

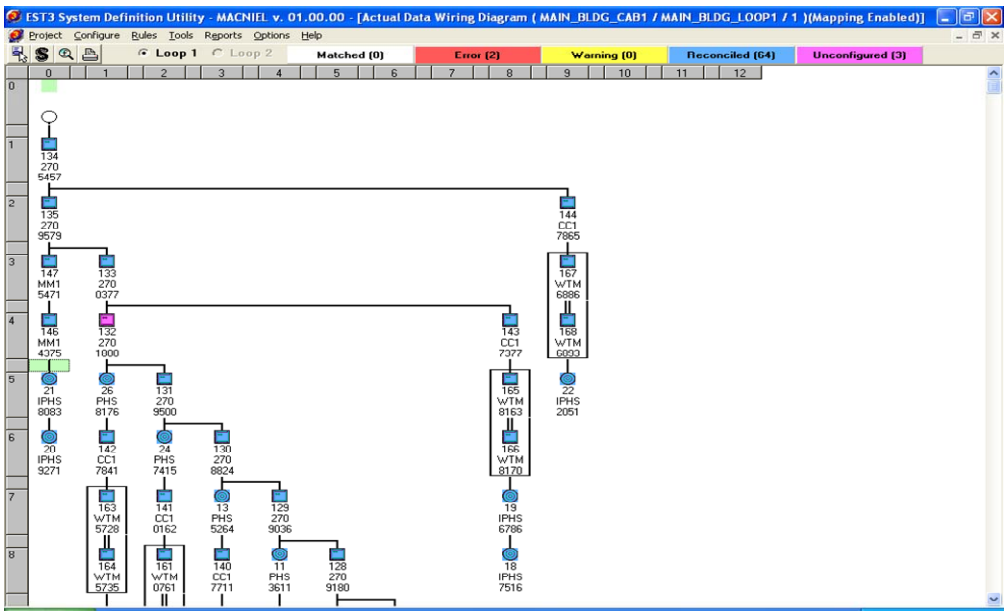
Click Commit Expected to reconcile for this device and the device icon turns blue indicating a changed state and a reconciled conflict



To Commit All, click the left Menu button, select Commit and select All or right click anywhere on the displayed screen

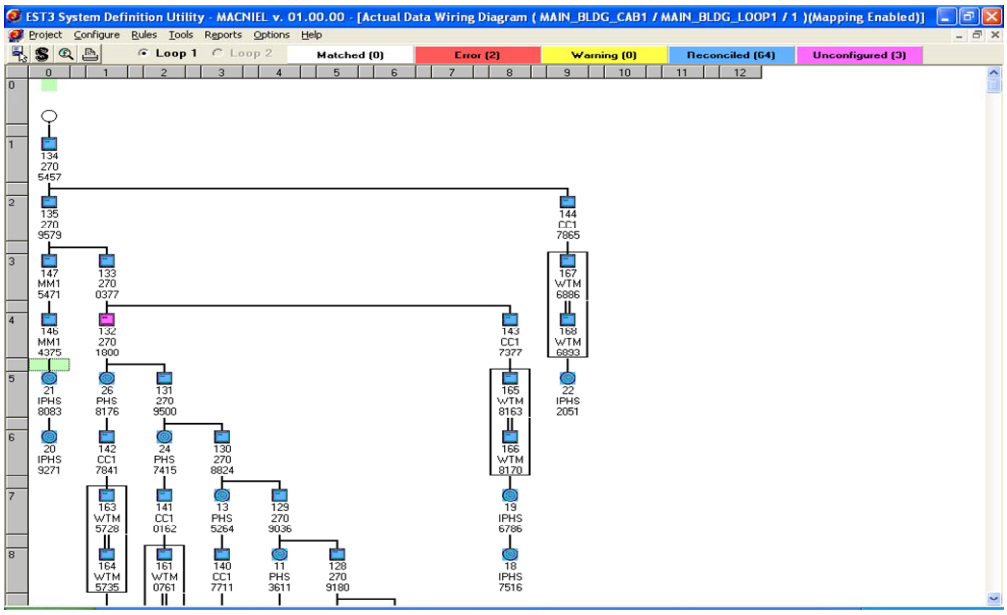


Let the SDU do the work for you



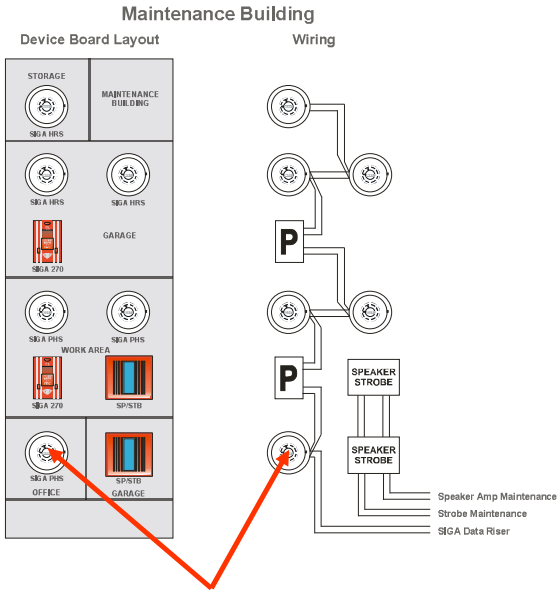
Conflict resolution

If major conflicts other than the device address exist (serial number or model type) the icon remains RED



Mapping method

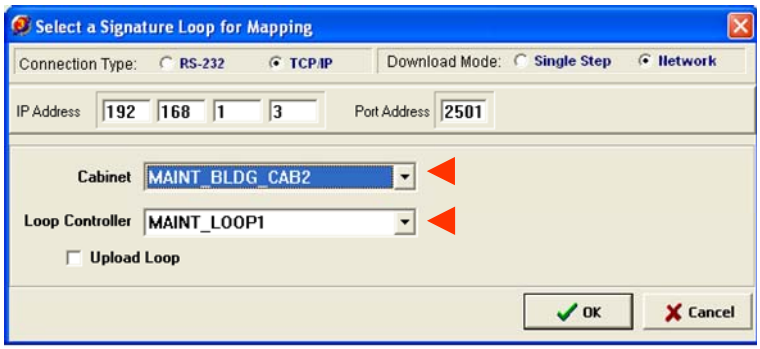
To reconcile using the mapping method, correlate specific device locations with your configured labels



The first device in the Maintenance Building wiring diagram is the Maintenance Office Smoke

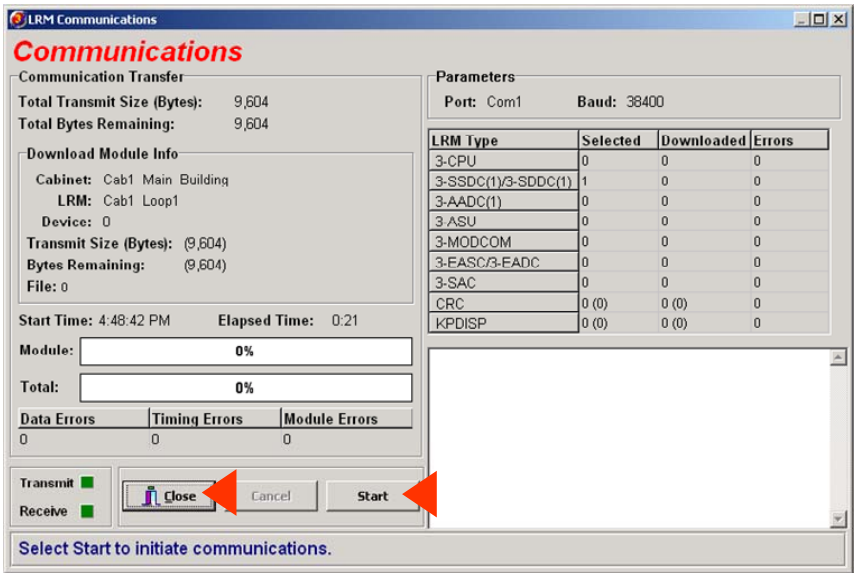
Mapping method

- Follow the same procedures used for the main building to upload and map the Actual data for the Maintenance Building into the SDU
- Select the maintenance building loop for mapping and initiate this upload



Mapping method

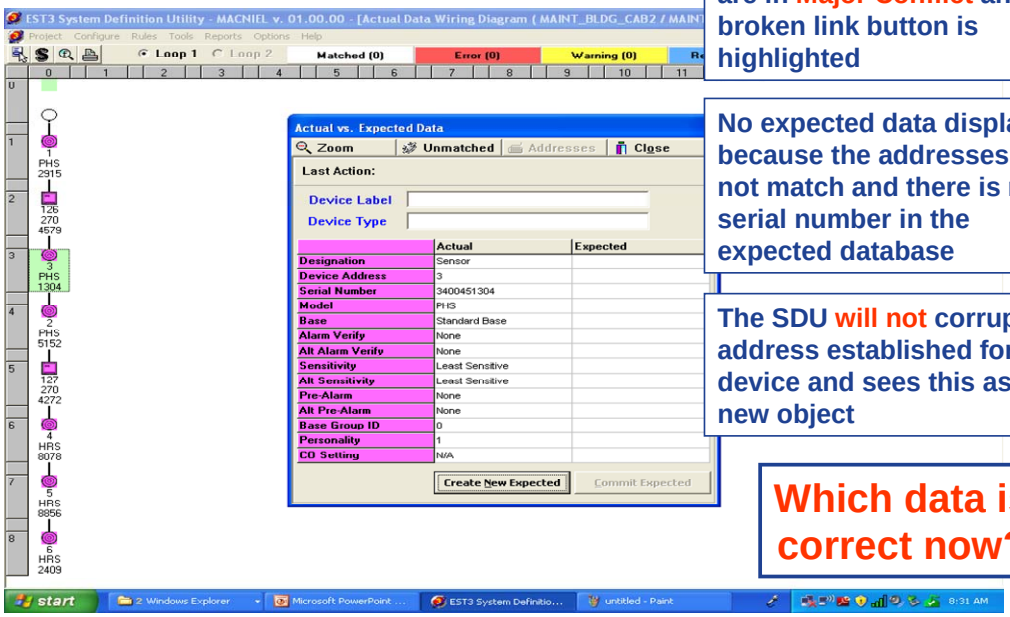
1. Click Start to initiate communications
2. Click Close when communications is complete



Actual data

Double click the first device in the map

The Show Actual/Expected and the Actual vs. Expected Data form displays



The Device Address, Serial Number and Model are in Major Conflict and the broken link button is highlighted

No expected data displays because the addresses do not match and there is no serial number in the expected database

The SDU will not corrupt the address established for this device and sees this as a new object

Which data is correct now?

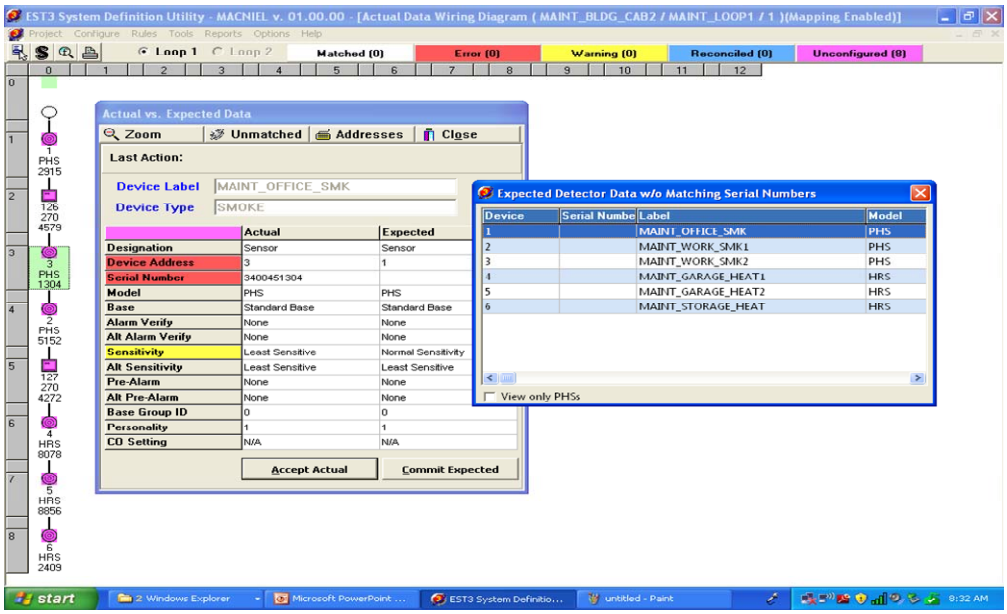
If you said the Actual Data you are...

Wrong!

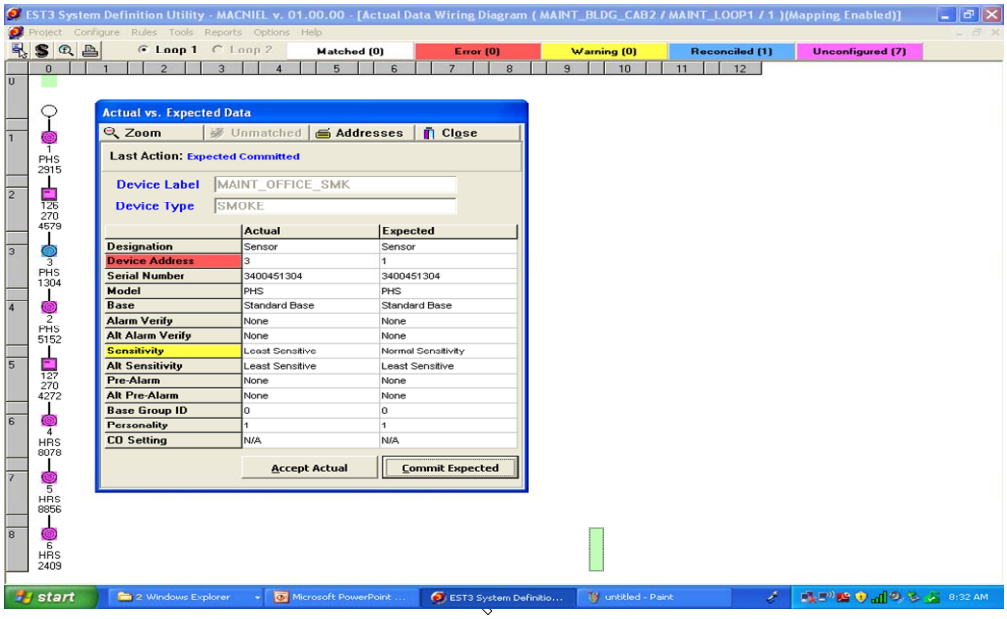
Do NOT Create New Expected

If you Create New Expected, you will corrupt the expected database adding devices to your programmed database

1. Click the Broken Link (Unmatched) Button
2. The Expected Detector Data without Matching Serial Numbers screen appears



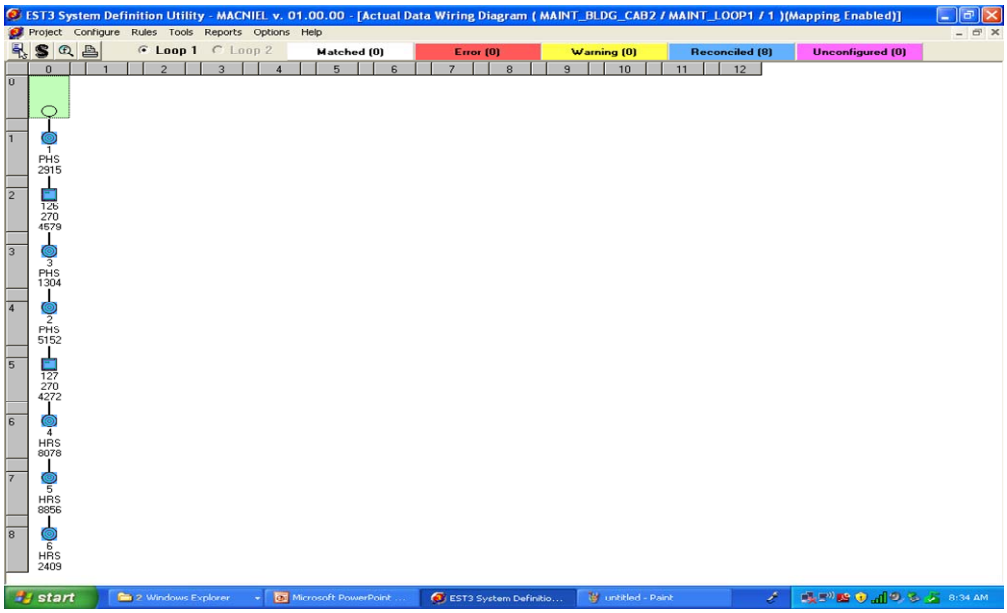
1. Click the label of the desired device
2. Click Commit Expected to reconcile devices w/ missing numbers
3. The serial number is imported into the expected database and the device icon for the Maint_Office_Smoke turns blue indicating a changed status



Remember

- The 3-SDU is designed to capture the serial number into the Expected Database in this manner
- Do not Create New Expected
- Continue this process until all devices have been reconciled and all the missing serial numbers are in the expected database

Device icons turn blue indicating a changed state



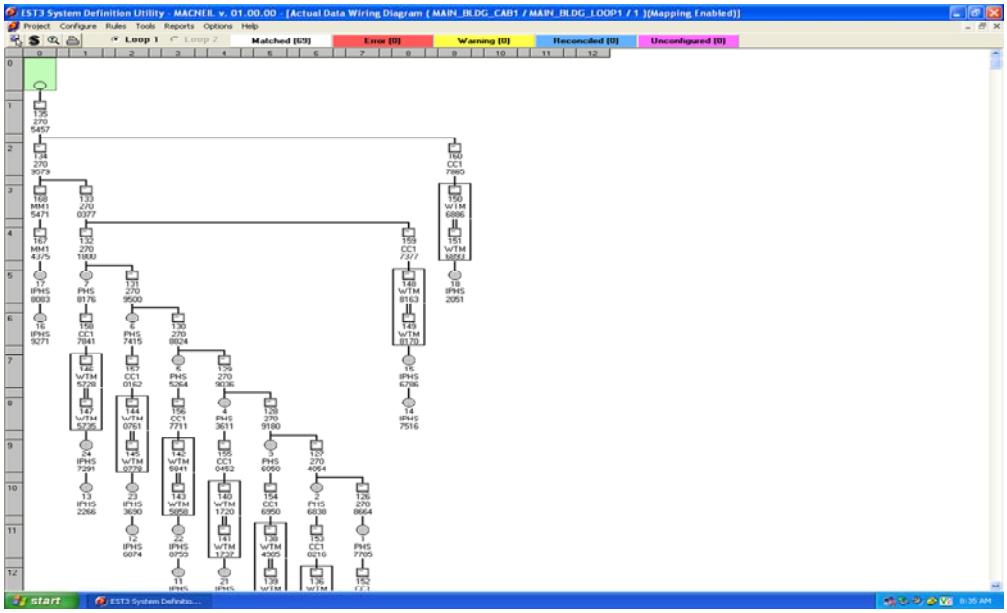
Reconciled databases

Regardless of the method used (scanning or mapping) we now have reconciled databases with expected device addresses, labels and the actual Serial Numbers

SIGA Convert

- SIGA Convert and download the RECONCILED Databases to the two Loop controllers making your programmed database the new actual database
- After this download to the Loop controllers is complete and mapping is complete, re-upload the actual data and map it in the SDU
- Verify white map
- Compile, DB Convert and download to the 3-CPU3s so they contain the same information

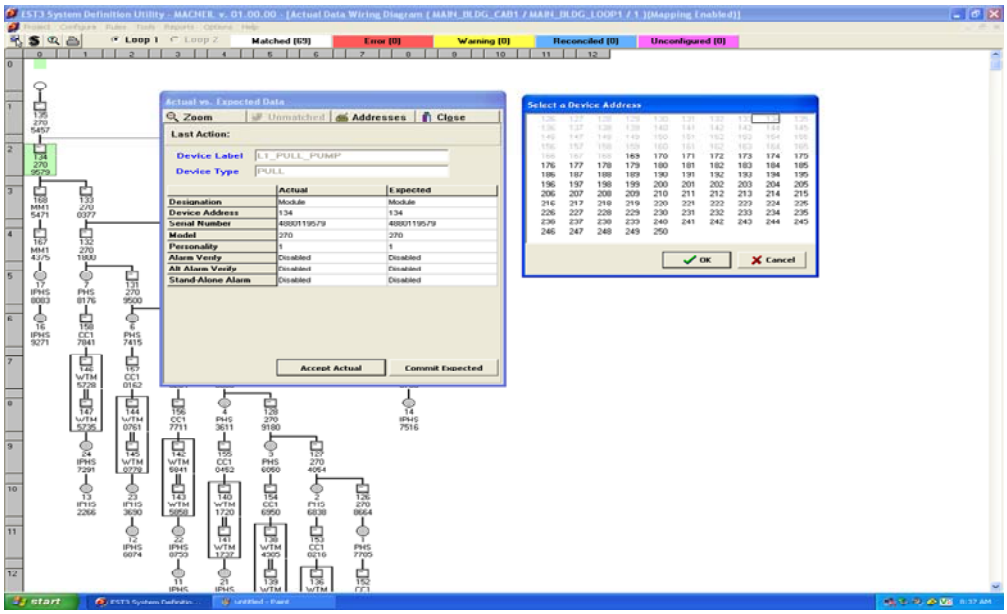
White map appears



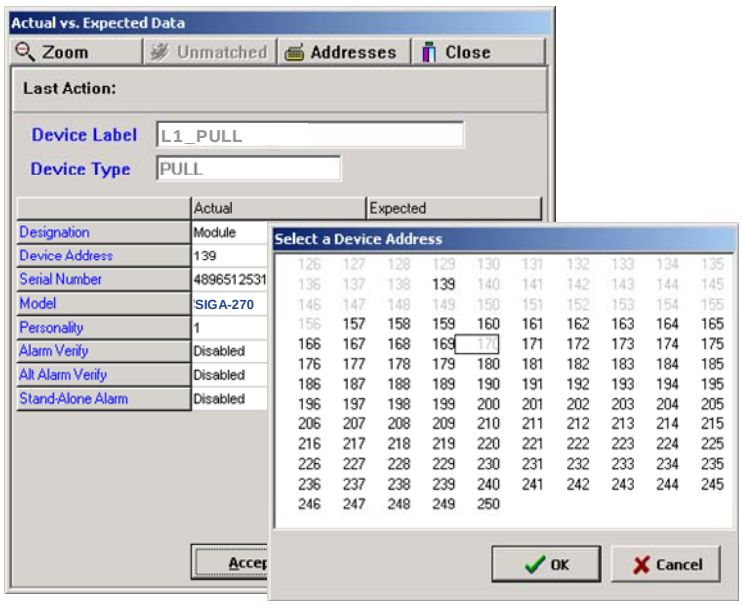
Mapping Utility

The 3-SDU mapping utility has the capability to assign expected database device addressing on a device-by-device basis

1. Select a device on the map and click the hand-on-list
2. To review this device's address, click address



1. The Select a Device Address dialog box appears displaying the address assigned
2. To assign a new address, click the desired address, click OK
3. The assigned expected address changes



EST3 Series Technician Certification

Rules Programming

UTC Fire & Security
A United Technologies Company

What is a Rule?

A VERY POWERFUL statement specifying action or actions (outputs) taking place when a specific, or group of objects (inputs) go off-normal or active

Rules consist of:

- A Rule Label
- An Input Statement
- An Output Statement
- Comments

(Optional)

[General Alarm]

```
ALARM  SMOKE :  STEADY  'LED_1_22';
{This is an example of comments surrounded by braces}
```

The Rule Label

- Must be enclosed in brackets
- May have spaces
- May have up to 40 characters
- Must be unique

The Input Statement

Contains up to 3 Parts:

The Input Statement

Contains up to 3 Parts:

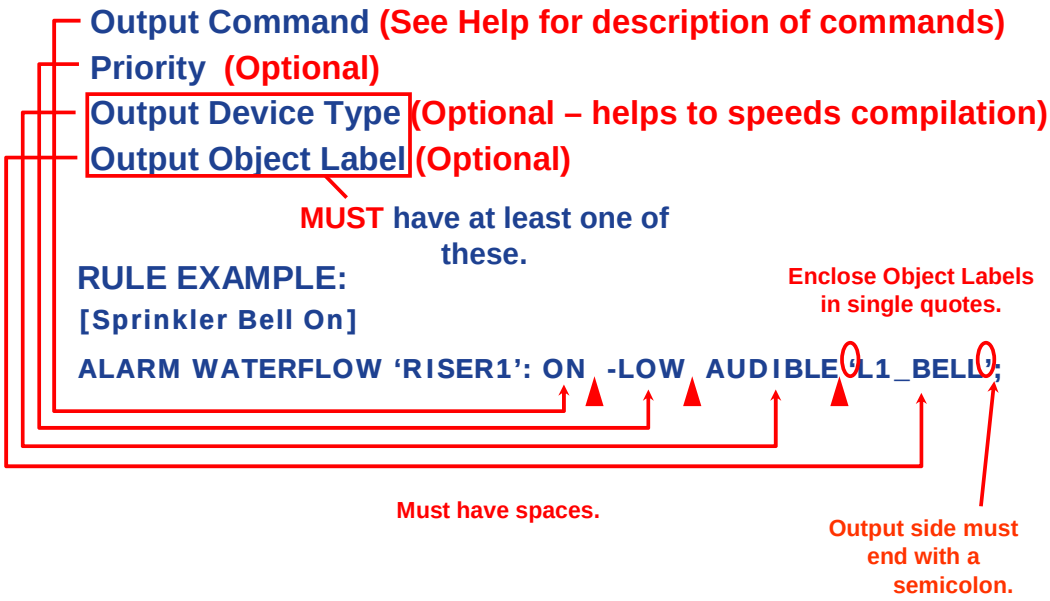
- Input Event Type (See Help utility for description of types)
- Input Device Type (Optional - helps to speed compilation)
- Input Object Label (Optional)

RULE EXAMPLE:

[Sprinkler Bell On]
ALARM 'RISER1':

The Output Statement

Contains up to 4 Parts:



The Output Statement

Contains up to 4 Parts:

- Output Command (See Help for description of commands)
- Priority (Optional)
- Output Device Type (Optional – helps to speeds compilation)
- Output Object Label (Optional)

RULE EXAMPLE:

[Sprinkler Bell On]

ALARM WATERFLOW 'RISER1': ON 'L1_BELL';

Single Output Statement

We now have a complete rule:

```
[Waterflow]
Alarm Waterflow 'Riser1' : On -low Audible 'L1_bell';
```

Which could be written as:

```
[Waterflow]
Alarm 'Riser1': On 'L1_bell';
```

If you want all Audibles to turn on, write it as:

```
[Waterflow]
Alarm 'Riser1': On AUDIBLE;
```

No object label needed.

The output side MUST end with a semicolon.

Multiple Output Statements

An input event must initiate several outputs:

```
[Waterflow]
Alarm 'Riser1' : On 'L1_bell',
                 Steady 'LED_1_22',
                 On VISIBLE,
                 AMPON 'L6_AMP' to 'EVAC';
```

Use commas to separate multiple output statements.

Turns on ALL circuits with device type of VISIBLE (no label needed).

Output Side MUST end with semicolon after last output statement.

What message is broadcast?

Multiple Output Statements

A rule where an input event must initiate several outputs:

```
[Waterflow]
Alarm 'Riser1' : On 'L1_bell',
                 Steady 'LED_1_22',
                 On VISIBLE,
                 AMPON 'L6_AMP' to 'EVAC';
```

Up to 32 Output Statements and only 1 Input Statement per rule

What message is broadcast?

Write a separate rule to manually restore the elevator and FACU LED from a momentary switch on the FACU after the elevator has been inspected.

```
[Fire alarm with primary elevator capture]
Alarm 'FLOOR<N:1,2,4-8>_Zone' :
                        AMPON 'FLOOR<N>_AMP' TO 'EVAC',
                        ON 'FLOOR<N>_STROBE',
                        AMPON '*' TO 'ALERT',
                        ON -LATCH 'FLOOR3_CAPTURE',
                        FAST -LATCH 'FLOOR3_CAPTURE_LED';

[Primary elevator capture restore]
Switch 'FLOOR3_CAPTURE_RESET' :
                        + OFF 'FLOOR3_CAPTURE',
                        + LEDOFF 'FLOOR3_CAPTURE_LED';
```

Momentary Switch

Command Qualifiers

Priorities in Rules

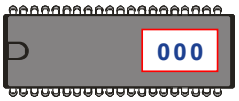
The priorities which are used in SDU applications are:

-Low does not override high or medium priorities. Unless specified, all output commands are **-Low** priority by default. A low priority counter is used to track low priority command activations (increments) and restorations (decrements).

-Medium overrides low priorities but not high priorities. Used in applications where a rule's output command overrides the action of another rules action of a lower priority. A medium priority counter is used to track medium priority command activations (increments) and restorations (decrements).

-High overrides medium and low priorities. Used in applications where a rule's output command overrides the action of another rules action. An input event of a switch has a built-in **-High** priority for all of its output statements. A High priority counter is used to track high priority command activations (increments) and restorations (decrements).

A counter is simply a dedicated register for each priority type (High, Medium and Low) within the CPU keeping track of each priority's output command activations and restorations.



An output command's initial activation increments the corresponding counter.

The output command's restoration decrements the corresponding counter.

When executing an output command with a priority, you must restore using the same priority.

-Set enables a command to change the state of an output object regardless of its pre-set priority controlled state without clearing the High, Medium or Low priority counters. -Set effectively holds the desired activated or restored state of the object until it is returned to the pre-set command state by the next command.

Set itself does not have a counter.

On execution of the next command, the output object returns to the priority counter remembered (activated or restored) state. For Example:

If an output object had been turned ON with a high priority, an OFF output command with a set priority would turn it OFF on its execution without clearing the high priority counter. When the next output command executes for this object, it would return to the remembered previous high priority state held in the counter (ON) regardless if the command is ON or OFF.

Note that if the object was previously turned OFF with a high priority, it would return to the OFF state.

Caution

Use caution when using the -SET priority. The counters are not cleared and your application returns to the remembered pre-counter controlled states when the next command is executed.

-Latch enables a command to change and latch the state of an output object regardless of its pre-latch priority controlled state. **-Latch** clears the High, Medium or Low priority counters. **-Latch** holds the desired activated or restored state of the object until the next command is executed.
Latch itself does not have a counter.

The latch command will not restore on a rules restoration. A separate event is required to latch the output object to the other state and the output object will not return to the pre-latch priority counter (activated or restored) state because counters have been cleared.

- If an object was activated ON by a -High priority within a rule, its corresponding counter would be 001
- If a separate rule now latches the object OFF, the high priority counter is cleared to 000
- When the object is latched ON, the counter stays at 000
- If the original high priority rule restores, the counter decrements indicating a count of -001
- The counter is now out of sequence

Latch priority example

A latch priority is used when the application requires personnel to verify an off normal condition before manually restoring the system.

Your application may require a rule to initiate an EVAC message and turn on visibles for the floor-of-incident, send an ALERT message to all other floors, capture the elevator to floor 3 during the alarm condition and annunciate the elevator capture on the FACP.

However, when the off normal condition is cleared and the system is reset, you want to hold the elevator captured and annunciated until it can be personally inspected.

Let's write this rule:

```
[Fire alarm with primary elevator capture]
Alarm 'FLOOR<N:1,2,4-8>_Zone' :
                                    AMPON 'FLOOR<N>_AMP' TO 'EVAC',
                                    ON 'FLOOR<N>_STROBE',
                                    AMPON '*' TO 'ALERT',
                                    ON -LATCH 'FLOOR3_CAPTURE',
                                    FAST -LATCH 'FLOOR3_CAPTURE_LED';
```

This rule initiates the desired audio and visual NAC response during an alarm condition and restores the audio and visual NAC circuits on the rules restoration sequence when system is reset.

The Elevator is held captured and the FACP LED continues to flash fast after the rule is restored due to the -LATCH priority.

Write a separate rule to manually restore the elevator and FACP LED from a momentary switch on the FACP after the elevator has been inspected.

```
[Fire alarm with primary elevator capture]
Alarm 'FLOOR<N:1,2,4-8>_Zone' :
                                    AMPON 'FLOOR<N>_AMP' TO 'EVAC',
                                    ON 'FLOOR<N>_STROBE',
                                    AMPON '*' TO 'ALERT',
                                    ON -LATCH 'FLOOR3_CAPTURE',
                                    FAST -LATCH 'FLOOR3_CAPTURE_LED';

[Primary elevator capture restore]
Switch 'FLOOR3_CAPTURE_RESET' :
                                    + OFF 'FLOOR3_CAPTURE',
                                    + LEDOFF 'FLOOR3_CAPTURE_LED';
```

Momentary Switch

Command Qualifiers

Wildcards “*”

A wildcard acts as a substitute for a single character or group of characters (modifiers), at a common modifier position, within a rule’s input or output object label

For example:

```
[HVAC1]
Alarm smoke ‘floor1_duct_smk’ : On ‘hvacrelay’;

[HVAC2]
Alarm smoke ‘floor2_duct_smk’ : On ‘hvacrelay’;

[HVAC3]
Alarm smoke ‘floor3_duct_smk’ : On ‘hvacrelay’;

[HVAC4]
Alarm smoke ‘floor4_duct_SMK’ : On ‘hvacrelay’;
```

We have four events turning on the same output device.

Inclusive.

By using the wildcard this can easily be written as:

```
[HVAC]
Alarm ‘floor*_duct_smk’ : On ‘hvacrelay’;
```

Another * example:

[Heat Detector Alarm]
Alarm 'Floor*_Heat_ Detector' : FAST 'Detector_LED';

[Floor 3 Alarm]
Alarm 'Floor3_ Detector_West ' : FAST 'Detector_LED';

[Lobby Alarm]
Alarm 'Bldg2_ Detector_East_lobby' : FAST 'Detector_LED';

As long as we have a common Label Modifier.

We can write these rules, as follows, by capitalizing on the power of the wildcard.

[Activate Detector LED]
Alarm '*Detector*' : FAST 'Detector_LED';

The bottom line: Better CONSISTENCY in labeling is needed.

N-Variable '<N:>'

The N-Variable is a programming function which replaces the numbers contained in an object label, within a rule, with a programmable variable N.

The proper syntax (input statement):

Use '<N:#>' in an input object label to replace a single number.

Use '<N:#-#>' in an input object label to replace a range of numbers.

Use '<N:#,#-#,#,#-#>' in an input object label to replace combinations.

A maximum of 255 characters can be used between ' ' including the < >.

'N:' must be followed by an index number between 1 and 32767.

A Wildcard and an N-Variable may be included in the same object label. However, Wildcards are not valid entries within N-Variables.

An N-Variable will not recognize a leading 0.

N-Variable '<N:>'

Let's look at an example using '**<N:#-#>**' to replace a range of numbers.

```
[Alarm1]
Alarm 'Floor1_smk': On 'Floor1_stb';

[Alarm2]
Alarm 'Floor2_smk': On 'Floor2_stb';

[Alarm3]
Alarm 'Floor3_smk': On 'Floor3_stb';
```

Where each Floor's smoke turns on it's corresponding Strobe.

Specific.

By using an N-Variable these rules can easily be written as:

```
[Floor of Incident Alarm]
Alarm 'Floor<n:1-3>_smk': On 'Floor<n>_stb';
```

An N-Variable specifying numbers in the object label can only be used in the input statement side of a rule.

Optionally, an N-Calculator specifying numbers in the object label can be used in the output statement of a rule.

N-Variable '<N:>'

Let's look at another example where we want to replace a range of numbers with leading 0's.

In this example we have 25 floors where some smoke labels contains a leading 0.

'Floor01_smk' — through → 'Floor25_smk'

Using the N-variable as presented so far and due to the fact that an N-Variable will not recognize a leading 0, we would have to write two rules.

```
[Floor of Incident Alarm]
Alarm 'Floor0<n:1-9>_smk': On 'Floor<n>_stb';

[Floor of Incident Alarm]
Alarm 'Floor<n:10-25>_smk': On 'Floor<n>_stb';
```

As you can see, this makes the process of writing more complex and labor intensive.

N-Variable '<N:>'

The N-Variable is a programming function which replaces the numbers contained in an object label, within a rule, with a programmable variable N.

The proper syntax (input statement):

Use '<N:#:W>' in an input object label to replace a single number.

Use '<N:# - #:W>' in an input object label to replace a range of numbers.

Use '<N:#,# - #,#,# - #:W>' in an input object label to replace combinations.

The 3-SDU eliminates this problem by providing a width (W) syntax for the N-Variable function, Specifying the number of integers in the number range.

N-Variable '<N:>'

Let's look at another example where we want to replace a range of numbers with leading 0's.

In this example we have 25 floors where each smokes label contains a leading 0.

'Floor01_smk' ⟶ through ⟶ 'Floor25_smk'

Now our task is simplified and we only need to write one rule to accomplish this task.

[Floor of Incident Alarm]
Alarm 'Floor<n:1-25:2>_smk': On 'Floor<n:2>_stb';

If the output object labels also include leading 0's, We can use a width for the N-Calculator also.

Incrementing and decrementing

[Floor of Incident]
Alarm 'Floor<n:1-10>_smk': On 'FL<n>_stb',
On 'FL<n>_horn',
AmpOn 'L*_AMP' to 'ALERT',
AmpOn 'L<n>_AMP' to 'EVAC',
On 'FL<n+1>_stb',
On 'FL<n+1>_horn',
AmpOn 'L<n+1>_AMP' to 'EVAC',
On 'FL<n-1>_stb',
On 'FL<n-1>_horn',
AmpOn 'L<n-1>_AMP' to 'EVAC';

Every Floor's Amps ON to the ALERT channel.

Fire Floor's Amps ON to the EVAC channel.

Floor Above

Floor Below

Rule Execution & Restoration

[Floor of Incident]
Alarm 'Floor<N:1-10>_smk' : On 'FL<n>_stb',
On 'FL<n>_exhfan',
AmpOn 'L<n>_AMP' to 'EVAC',

On 'FL<n+1>_stb',
AmpOn 'L<n+1>_AMP' to 'EVAC',
On 'FL<n+1>_presfan',

On 'FL<n-1>_stb',
On 'FL<n-1>_presfan',
AmpOn 'L<n-1>_AMP' to 'EVAC',

On 'FL<n>_doorholders',
On 'Elev_deluge',

AmpOn 'L*_AMP' to 'ALERT';

Initially an input device goes off normal starting the rules execution or activation sequence.

Execution

Rule Execution & Restoration

[Floor of Incident]
Alarm 'Floor<N:1-10>_smk' : On 'FL<n>_stb',
On 'FL<n>_exhfan',
AmpOn 'L<n>_AMP' to 'EVAC',

On 'FL<n+1>_stb',
AmpOn 'L<n+1>_AMP' to 'EVAC',
On 'FL<n+1>_presfan',

On 'FL<n-1>_stb',
On 'FL<n-1>_presfan',
AmpOn 'L<n-1>_AMP' to 'EVAC',

On 'FL<n>_doorholders',
On 'Elev_deluge',

AmpOn 'L*_AMP' to 'ALERT';

After this off normal condition has been cleared, and you have acknowledged & reset alarm at the front panel the rules restoration begins.

Restoration

Rule Execution & Restoration

Rules execute during the rules activate sequence top-down and execute during the restore sequence bottom-up.

This means that if you turned ON a device during the rule's activation sequence, it would then turn OFF during the restoration sequence.

Dialer applications create some unique circumstances during a rule's activation and restoration sequences.

Let's send a common trouble event message to the central station. The following rule looks like it might work.

[Common Trouble - Contact ID]
CMSFT : SEND 'Central_Station_Acct' MSG "130000000";

Rule Execution & Restoration

The rule sends the common trouble message to the central station on the rule's activation sequence but then resends the same message on the rule's restoration sequence.

To make this even more problematic, we need to send a common trouble event restoral message to the Central Station when the trouble has been resolved.

If you add the send command to send the restoral message to this rule we now send it twice also.

```
[Common Trouble - Contact ID]
CMSFT    :    SEND 'Central_Station_Acct' MSG "130000000";
SEND 'Central_Station_Acct' MSG "330000000";
```

Command Qualifiers

To resolve this behavior for rules, Command Qualifiers are available in the syntax of rules.

A Command Qualifier is simply a + or - added to the front of the Command in the output statement of a rule.

A + causes the command to only execute on rule activation.

A - causes the command to only execute on rule restoration.

Now the common trouble event message is sent only during the rule's activation sequence

and the common trouble event restoral message is sent only during the rule's restoration sequence.

```
[Common Trouble - Contact ID]
 CMSFT    : +SEND 'Central_Station_Acct' MSG "130000000",
            -SEND 'Central_Station_Acct' MSG "330000000";
```

Command Qualifier

A Command Qualifier is used to latch system output events without the use of priorities.

For example: In our project if we wanted to turn on the air conditioning when we disarmed the system when opening the facility, we could use:

[Opening Sequence]
SecurityDisarmed 'Security_Partition' : +ON 'AC_UNIT';

Write a rule to turn the air conditioning off when we armed away the system when closing the facility.

[Closing Sequence]
SecurityAway 'Security_Partition' : -OFF 'AC_UNIT';

In both cases, the Command Qualifier effectively latches the relay controlling the AC unit ON or OFF.

Let's add Delays to our rule

[Floor of Incident]
Alarm 'Floor<N:1-10>_smk' : On 'FL<n>_stb',
On 'FL<n>_exhfan',
AmpOn 'L<n>_AMP' to 'EVAC',
DELAY 0005, 5 second Delay
On 'FL<n+1>_stb',
AmpOn 'L<n+1>_AMP' to 'EVAC',
On 'FL<n+1>_presfan',
DELAY 0010, 10 second Delay
On 'FL<n-1>_stb',
On 'FL<n-1>_presfan',
AmpOn 'L<n-1>_AMP' to 'EVAC',
DELAYACTIVATE 0010, 10 second Delay
On 'FL<n>_doorholders',
On 'Elev_deluge',
DELAYRESTORE 0010, No Delay
AmpOn 'L*_AMP' to 'ALERT';

First DELAY - Which delays during execution and restoration.

Then DELAYACTIVATE – Which delays during the execution sequence only.

Finally DELAYRESTORE – Which delays during the restoration sequence only

Execution

Delays

[Floor of Incident]
Alarm 'Floor<N:1-10>_smk' : On 'FL<n>_stb',
On 'FL<n>_exhfan',
AmpOn 'L<n>_AMP' to 'EVAC',
DELAY 0005, 5 second Delay
On 'FL<n+1>_stb',
AmpOn 'L<n+1>_AMP' to 'EVAC',
On 'FL<n+1>_presfan',
DELAY 0010, 10 second Delay
On 'FL<n-1>_stb',
On 'FL<n-1>_presfan',
AmpOn 'L<n-1>_AMP' to 'EVAC',
DELAYACTIVATE 0010, No Delay
On 'FL<n>_doorholders',
On 'Elev_deluge',
DELAYRESTORE 0010, 10 second Delay
AmpOn 'L*_AMP' to 'ALERT';

First DELAY - Which delays during execution and restoration.

Then DELAYACTIVATE – Which delays during the execution sequence only.

Finally DELAYRESTORE – Which delays during the restoration sequence only

Restoration

Delays

[Floor of Incident]
Alarm 'Floor<N:1-10>_smk' : On 'FL<n>_stb',

A DELAYACTIVATE delays everything below (after) it in a rule.

A DELAYRESTORE delays everything above (before) it in a rule.

AmpOn 'L<n+1>_AMP' to 'EVAC',
On 'FL<n-1>_presfan',
AmpOn 'L<n-1>_AMP' to 'EVAC',
DELAYACTIVATE 0010,
On 'FL<n>_doorholders',
On 'Elev_deluge',
DELAYRESTORE 0010,
AmpOn 'L*_AMP' to 'ALERT';

First DELA delays dur execution restoration.

Then DE – Which the exec sequenc

Finally DELAYRESTORE – Which delays during the restoration sequence only

Delays

You can also use a Delay to control the duration of an event.

Let's take the example of the Service Groups in our project which turn on the corresponding Floor's strobe for a duration of 10 seconds in response to testing a device within the group.

Let's write this rule:

What's the input event type?

First we need a rule label.

```
[Floor3 through 5 Service Test]
SG
```

Delays

You can also use a Delay to control the duration of an event.

Let's take the example of the Service Groups in our project which turn on the corresponding Floor's strobe for a duration of 10 seconds in response to testing a device within the group.

Let's write this rule:

Next I'll use the labels that I created for my three service groups, taking advantage of the N-Variable function.

```
[Floor3 through 5 Service Test]
SG 'Floor<N:3-5>_Service' :
```

Delays

You can also use a Delay to control the duration of an event.

Let's take the example of the Service Groups in our project which turn on the corresponding Floor's strobe for a duration of 10 seconds in response to testing a device within the group.

Let's write this rule:

Turn on the corresponding strobe using the N-Calculator.

```
[Floor3 through 5 Service Test]
SG  'Floor<N:3-5>_Service'  :     On 'L<N+2>_STB',
```

Why N+2?

Delays

You can also use a Delay to control the duration of an event.

Let's take the example of the Service Groups in our project which turn on the corresponding Floor's strobe for a duration of 10 seconds in response to testing a device within the group.

Let's write this rule:

Add a 10 second delay.

```
[Floor3 through 5 Service Test]
SG     'Floor<N:3-5>_Service'  :     On 'L<N+2>_STB',
                                     DLY 10,
```

Delays

You can also use a Delay to control the duration of an event.

Let's take the example of the Service Groups in our project which turn on the corresponding Floor's strobe for a duration of 10 seconds in response to testing a device within the group.

Let's write this rule:

Turn off the corresponding strobe.

```
[Floor3 through 5 Service Test]
SG 'Floor<N:3-5>_Service' :      On 'L<N+2>_STB',
                                 DLY 10,
                                 Off 'L<N+2>_STB';
```

Delays

You can also use a Delay to control the duration of an event.

Let's take the example of the Service Groups in our project which turn on the corresponding Floor's strobe for a duration of 10 seconds in response to testing a device within the group.

Let's write this rule:

Will this rule work?

The answer is NO! A Service Group activation is a momentary event, where this rule automatically restores. In this case, we get a 10 second delay on activation and an additional 10 second delay on restoration, for a total of 20 seconds.

```
[Floor3 through 5 Service Test]
SG 'Floor<N:3-5>_Service' :      On 'L<N+2>_STB',
                                 DLY 10,
                                 Off 'L<N+2>_STB';
```

20 second delay total

Delays

You can also use a Delay to control the duration of an event.

Let's take the example of the Service Groups in our project which turn on the corresponding Floor's strobe for a duration of 10 seconds in response to testing a device within the group.

Let's write this rule:

Fix this problem by using a delay activate.

This works, but is inefficient. How about if we:

```
[Floor3 through 5 Service Test]
SG  'Floor<N:3-5>_Service'  :     On 'L<N+2>_STB',
                                  DLYA 10;
```

Rules

Let's do a few example rules from your project

Practical exercise #19 wants you to turn on the supply fan at startup. It also requires that you steady light the auto segment LEDs of the 3x3x6 hand-off-auto panel, indicating that the system is in the AUTO Mode.

First I need a rule Label:

```
[STARTUP SUPPLY FAN]
```

Rules examples

What's the Input Event Type?

Is there an input device type or object label?

Finally end the input statement with a :

```
[STARTUP SUPPLY FAN]
STARTUP :
```

NO – because this is a system event

Rules examples

In this example, this fan is controlled by a relay which was configured as supervised fancontrol device.
What is the output command to turn on the supply fan?

We could have used just ON.

Next let's add the output device object label.

Finish this output statement with a ,.

```
[STARTUP SUPPLY FAN]
STARTUP : FANON 'L8_SUP_FAN',
```

Remember to use your 'Labels'!

Rules examples

Next, let's steady light the auto portion LEDs of the three switch combinations, which manually control your fans.
What is the output command to steady light a LED?

Then add the output device object label for the 2nd LED.

Finish this output statement with a ,

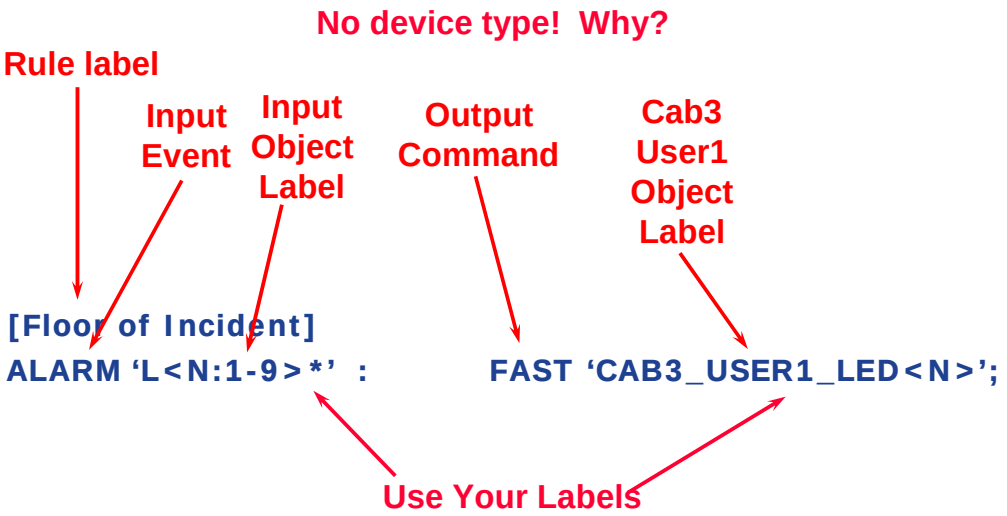
Then add the output statements for the 5th and 8th LEDs.

Then finish rule with a ;.

```
[STARTUP SUPPLY FAN]
STARTUP : FANON 'L8_SUP_FAN',
          STEADY 'CAB1_USER3_LED2',
          STEADY 'CAB1_USER3_LED5',
          STEADY 'CAB1_USER3_LED8';
```

Rules examples

Practical exercise #21 wants you to write one rule to fast blink the appropriate LEDs on the Cab3 User1 24 LED display panels to indicate the floor-of-incident when any main building alarm is reported to the panel.

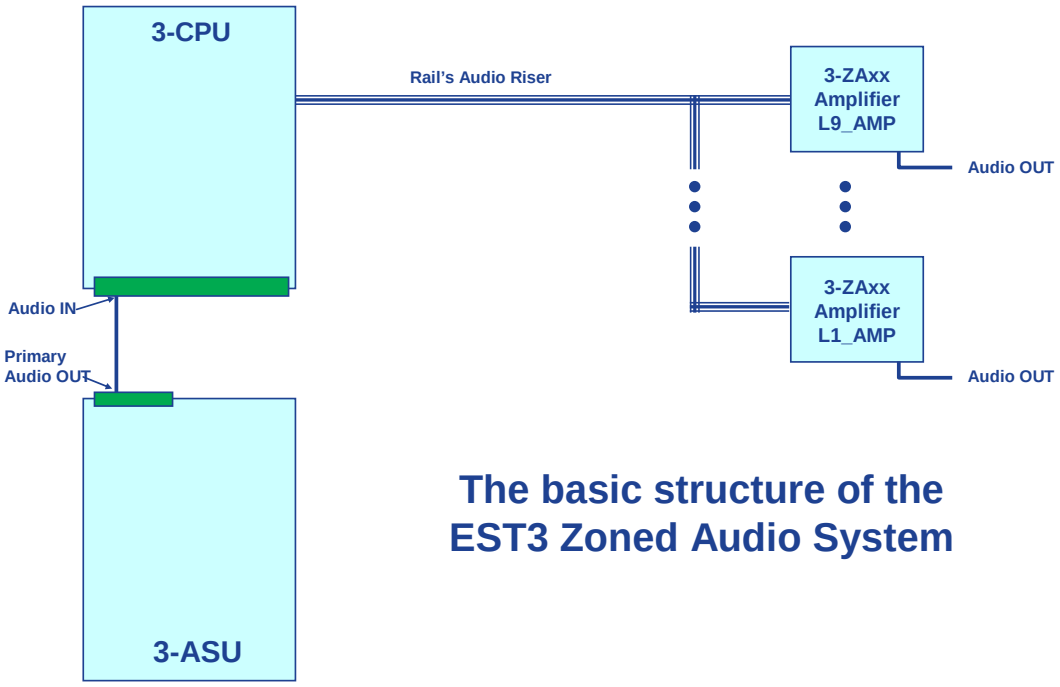


Rules examples

Practical exercise #25 wants you to write one rule to initiate audible and visible zone responses for the main building, per the specification, when a smoke detector goes active.

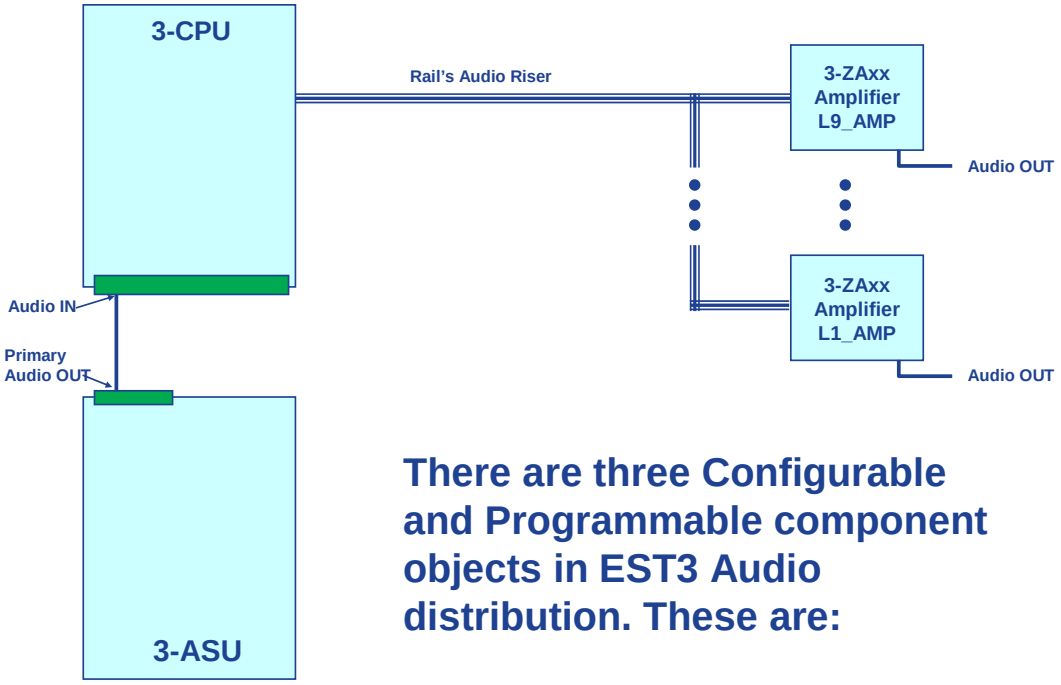
This requires writing a rule that sends EVAC and ALERT messages to the appropriate floors in our 9 level main building.

Rules examples



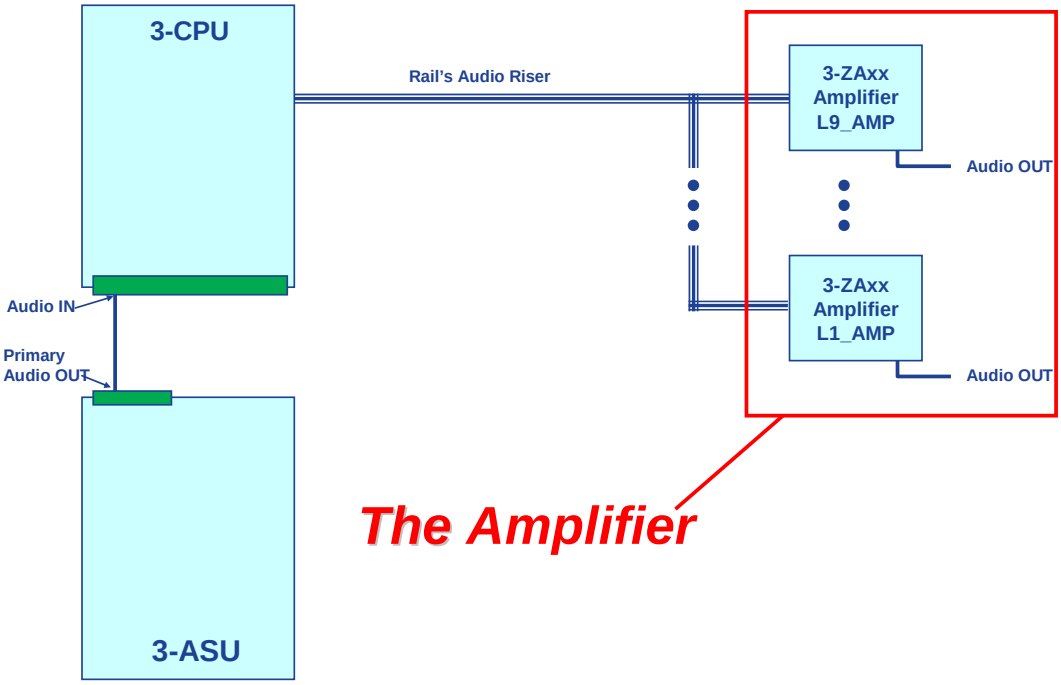
The basic structure of the EST3 Zoned Audio System

Rules examples

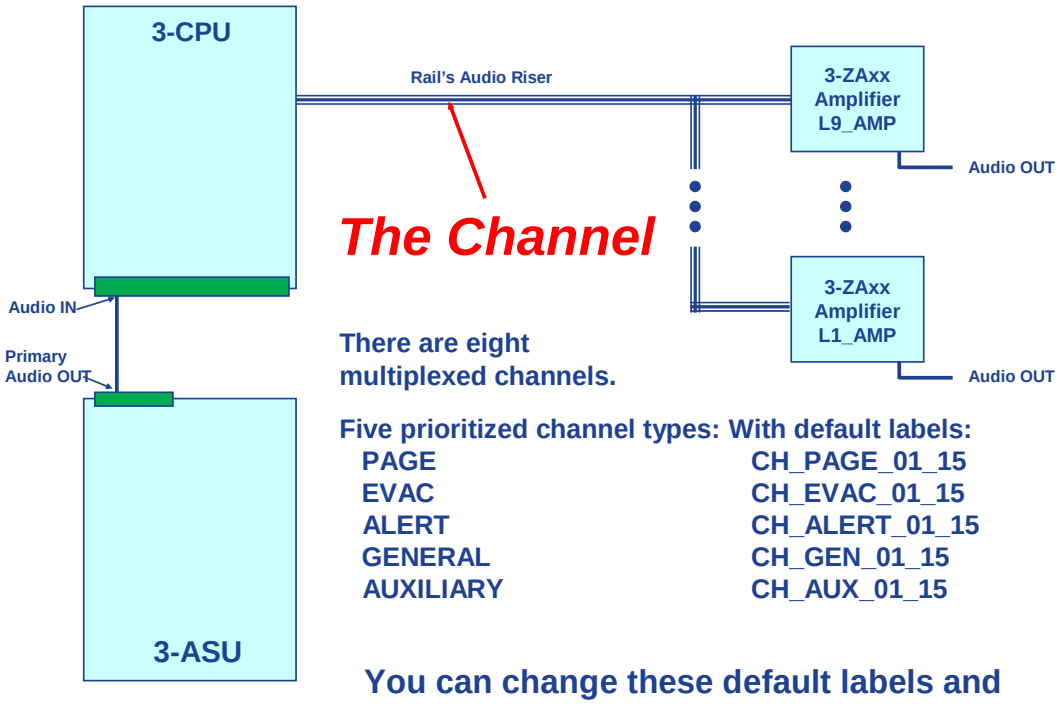


There are three Configurable and Programmable component objects in EST3 Audio distribution. These are:

Rules examples



Rules examples

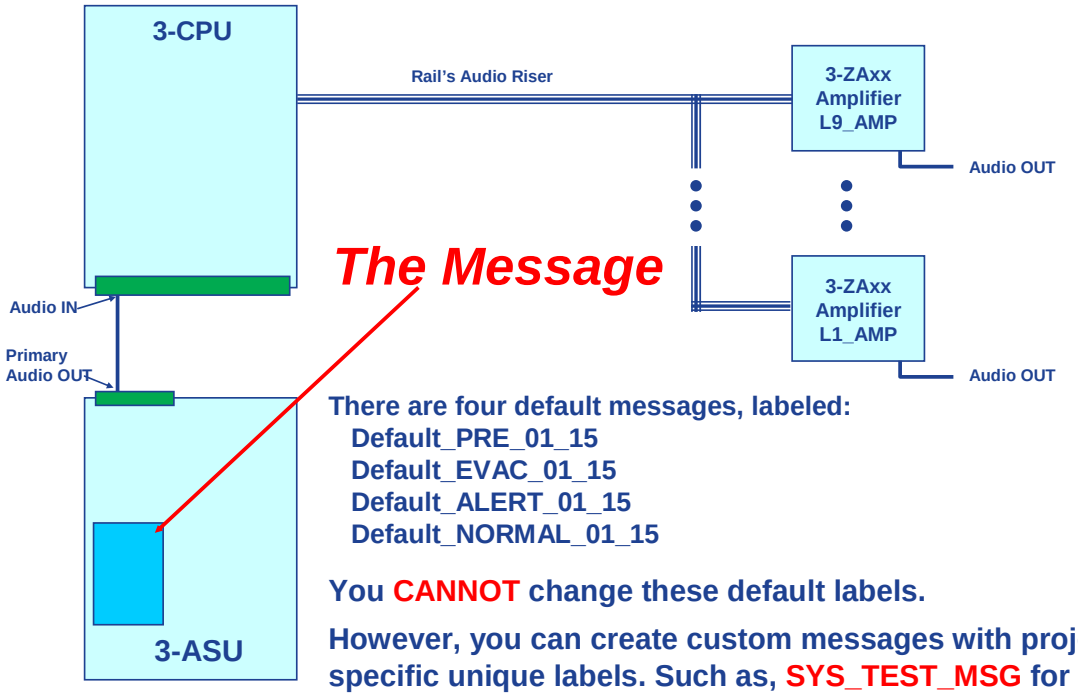


There are eight multiplexed channels.

Five prioritized channel types: With default labels:

PAGE	CH_PAGE_01_15
EVAC	CH_EVAC_01_15
ALERT	CH_ALERT_01_15
GENERAL	CH_GEN_01_15
AUXILIARY	CH_AUX_01_15

You can change these default labels and use duplicate channel types with different labels.



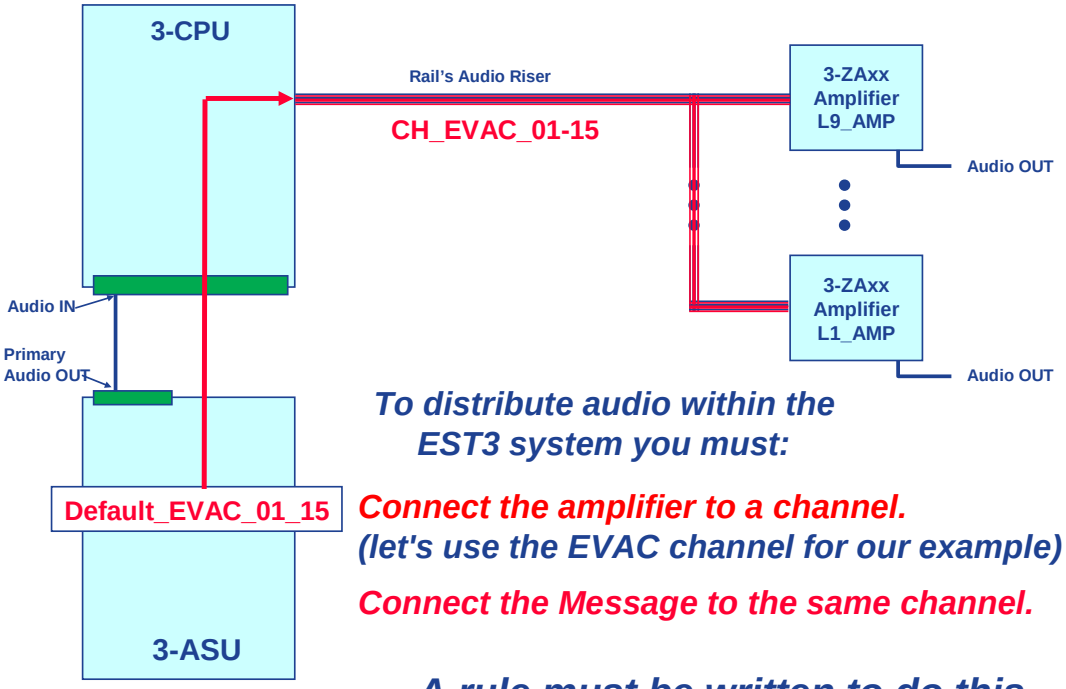
There are four default messages, labeled:

- Default_PRE_01_15
- Default_EVAC_01_15
- Default_ALERT_01_15
- Default_NORMAL_01_15

You **CANNOT** change these default labels.

However, you can create custom messages with project specific unique labels. Such as, **SYS_TEST_MSG** for our project's audio test message.

Thus, there are two types on messages, **DEFAULT** and **CUSTOM**. This is important when writing rules!

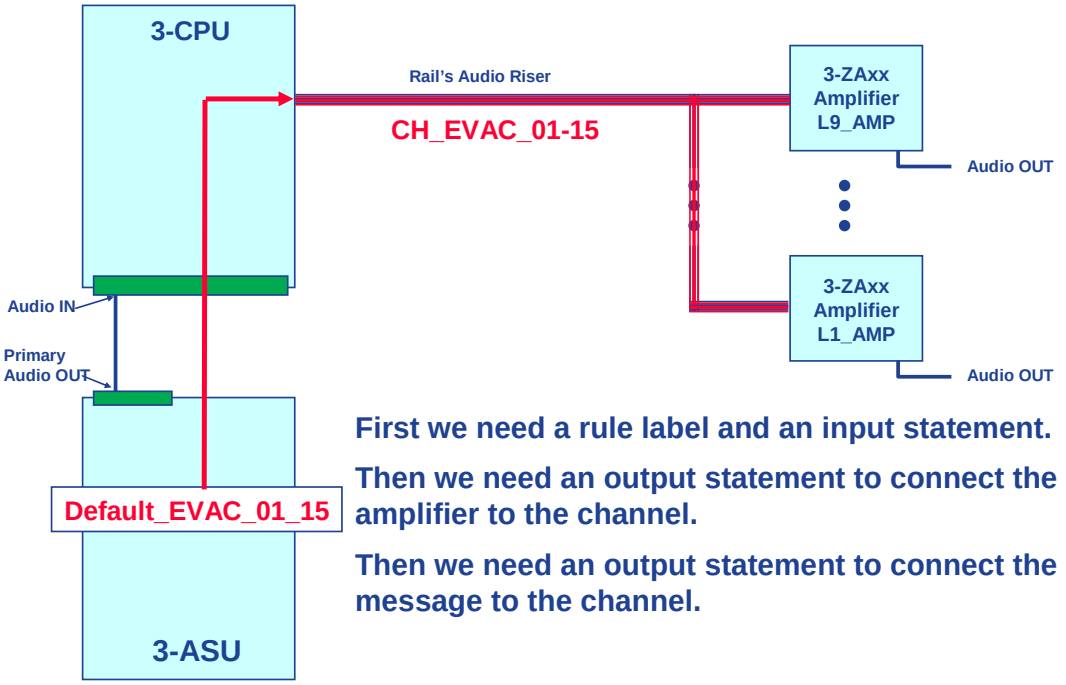


To distribute audio within the EST3 system you must:

Connect the amplifier to a channel. (let's use the EVAC channel for our example)

Connect the Message to the same channel.

A rule must be written to do this.

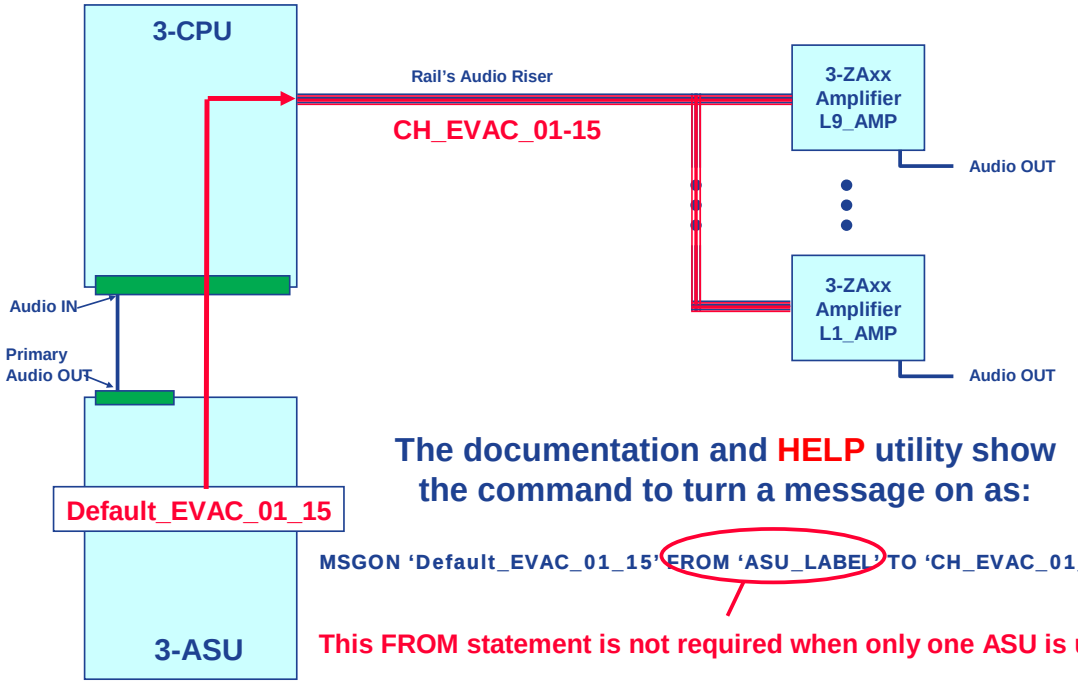


First we need a rule label and an input statement.

Then we need an output statement to connect the amplifier to the channel.

Then we need an output statement to connect the message to the channel.

```
[EVAC AUDIO RESPONSE]
ALARM SMOKE '*' :  AMPON 'Floor4_AMP' TO 'CH_EVAC_01_15',
                   MSGON 'Default_EVAC_01_15' TO 'CH_EVAC_01_15';
```

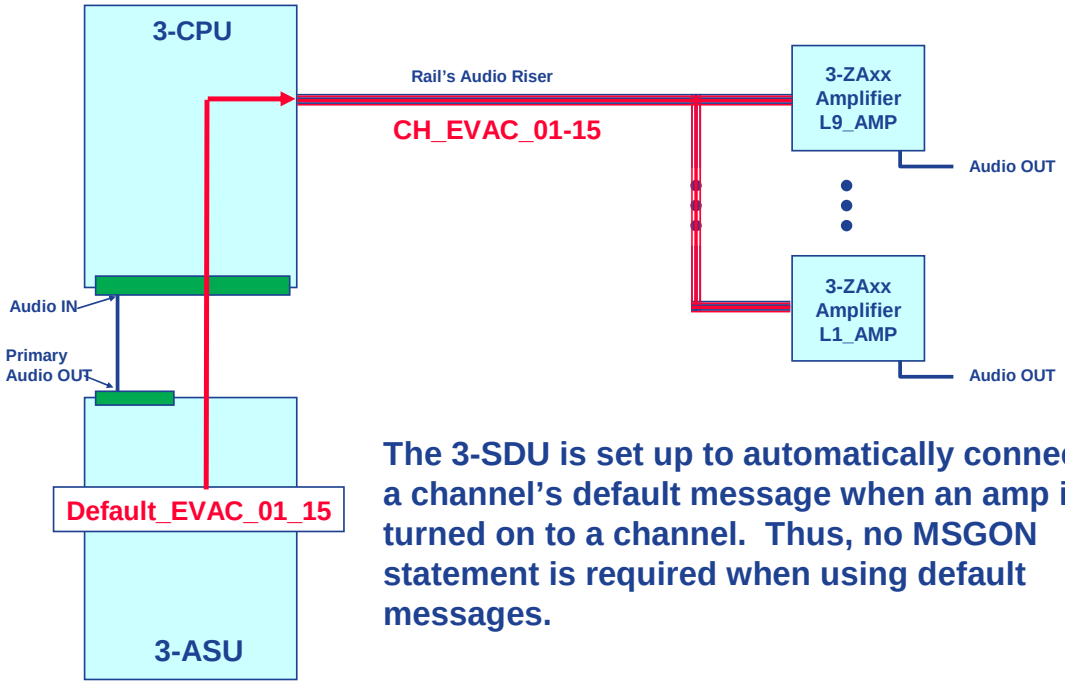


The documentation and HELP utility show the command to turn a message on as:

MSGON 'Default_EVAC_01_15' FROM 'ASU_LABEL' TO 'CH_EVAC_01_15';

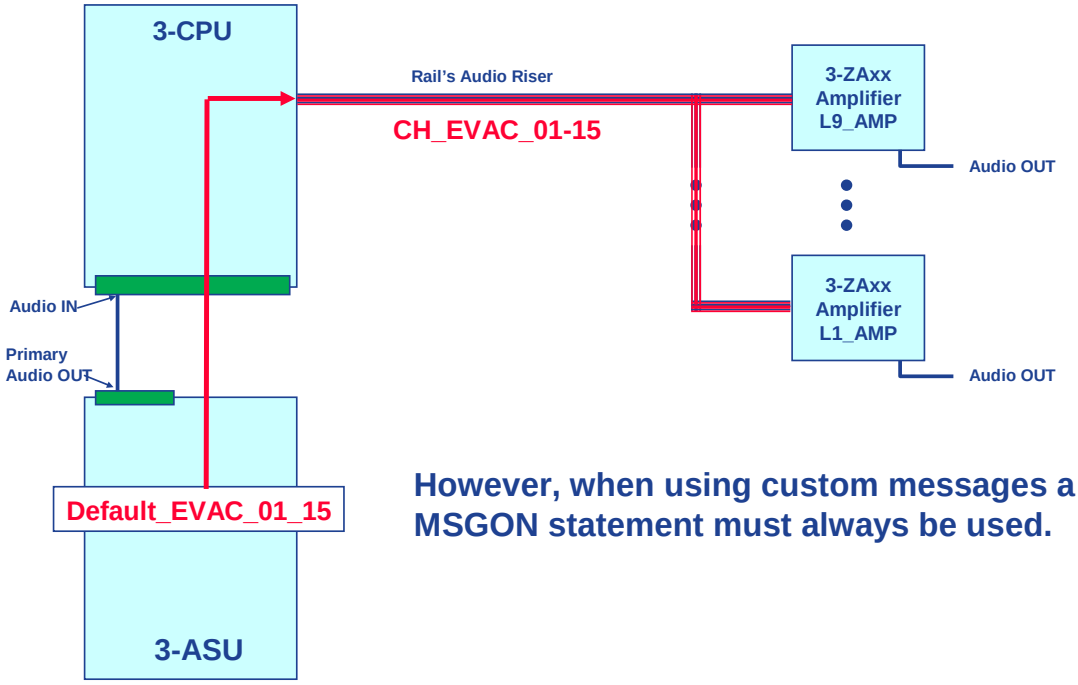
This FROM statement is not required when only one ASU is used.

[EVAC AUDIO RESPONSE]
ALARM SMOKE '*' : AMPON 'Floor4_AMP' TO 'CH_EVAC_01_15',
MSGON 'Default_EVAC_01_08' TO 'CH_EVAC_01_15';



The 3-SDU is set up to automatically connect a channel's default message when an amp is turned on to a channel. Thus, no MSGON statement is required when using default messages.

[EVAC AUDIO RESPONSE]
ALARM SMOKE '*' : AMPON 'Floor4_AMP' TO 'CH_EVAC_01_15';



However, when using custom messages a MSGON statement must always be used.

```
[EVAC AUDIO RESPONSE]
ALARM SMOKE '*' :  AMPON 'Floor4_AMP' TO 'CH_EVAC_01_15';
                   MSGON 'Custom_EVAC_MSG' TO 'CH_EVAC_01_15';
```

Rules examples

Practical exercise #25 wants you to write one rule to initiate audible and visible zone responses for the main building, per the specification, when a smoke detector goes active.

[Audible/Visible on Smoke]

```
ALARM SMOKE 'L<N:1-8>_*SMK' :
                AMPON 'L<N>_AMP' TO 'EVAC',
                AMPON 'L<N+1>_AMP' TO 'EVAC',
                AMPON 'L<N-1>_AMP' TO 'EVAC',
                AMPON 'L*_AMP' TO 'ALERT',
                ON 'L<N>_STB',
                ON 'L<N+1>_STB',
                ON 'L<N-1>_STI ,
                ON 'L1_AUDIBLE';
```

Is this rule finished?

No!

What about IDC/NAC5.

Appendix A
Labeling Conventions

Introduction

This section contains a description of the basic methods used to establish labeling conventions for EST3 field applications.

UTC Fire & Security

A United Technologies Company

Table of Contents

Topic **Page**

Basic Label Planning

Introduction

All fire alarm systems are divided into three fundamental parts:

- Inputs
- Control and operating functions
- Outputs

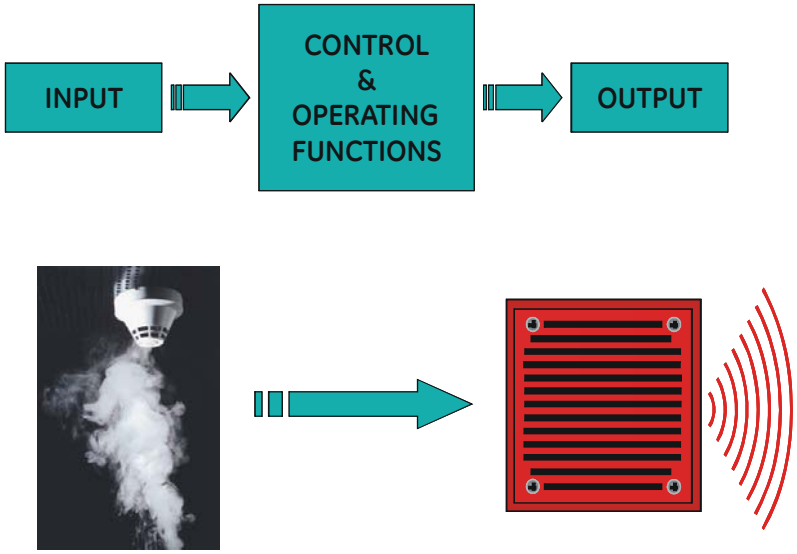


Figure 2-1. Three Fire Alarm System Parts

- Fire alarm system inputs include: smoke detectors, manual pull stations and waterflow switches

- Fire alarm system outputs include: bells, strobes, control messages and audio messages

The system programmer develops the fire alarm system's control and operating functions by configuring objects and creating rules. For example, in a basic fire alarm system, a function is stated:

When any smoke detector goes into alarm, turn on all audible devices.

[Example Rule]

ALARM SMOKE : ON AUDIBLE;

Input Event
Input Device Type
Output Device Type
Output Command

This statement is the programming instruction the fire alarm system follows to accomplish its purpose. An example rule is shown constructed in the syntax required to accomplish this statement. As fire alarm systems become more extensive, they require a more sophisticated set of rules to initiate more complex operations.

EST3 System Definition Utility (3-SDU) programming is accomplished by creating a series of rules specifying output action or actions and the execution order when a specific input or set of input events occurs. The effective EST3 programmer configures and labels objects within the fire alarm system's environment. Before developing, take time to:

- Identify objects, device types and labels
- Identify input and output objects and naming conventions (labels)
- Determine the relationship between input and output objects

Examples of objects include: panels, annunciator panels, hardware modules, switches, LEDs, detectors, manual pull stations, audio messages and pseudo points.

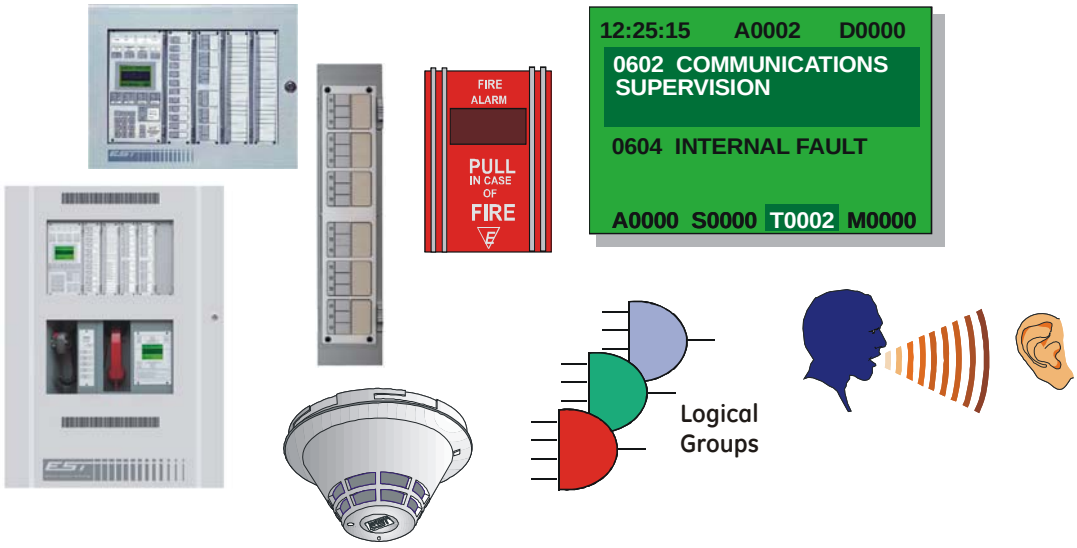


Figure 2-2. Example EST3 Objects

Rules

Rules are program statements specifying which operation(s) or action(s) is(are) executed when a specific event(s) take place. When the conditions are met to make the input statement of the rule true (active), the system executes the rule's output statement(s) and activates the output device(s.)

When an input device or object goes active, the system controller compares the corresponding object label to those in the database. If a match is found between an input device's object label and the input statement's object label within the rule, the rule is executed. Executing the rule requires a database search for all output object labels matching those in the rule's output statement. The matching output devices are then turned on or off, as specified within the executed rule.

When a rule contains multiple output statements, each output command is executed in the order listed. A rule may contain only one input statement and up to 32 output statements. When the input event activating a rule restores, the operations performed by the rule automatically restore in reverse order. Anything previously turned on is turned off and anything previously turned off turns on.

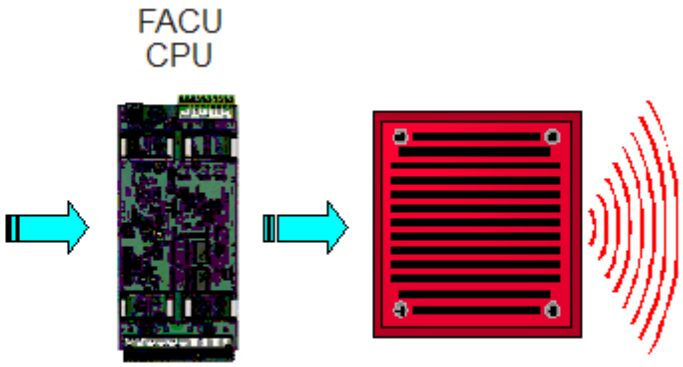


Figure 2-3. Automated System Environment

Objects

An object is any addressable device, circuit, audio or text message, pseudo point or other entity used to initiate an input event within a rule or is the subject of a rule's output command. For example:

- Initiating devices - Pull station and smoke detector alarm devices, Tamper switch supervisory devices
- Annunciator devices - Switch input devices and light emitting diode (LED) output devices
- Audio messages - EVAC and ALERT voice messages
- Logical groups - Combinations of physical system components or logical database objects including: AND, Matrix, Zone, Instruction Text, Check in, or Guard Patrol groups
- Pseudo points - Artificial and/or internal events designed to monitor system operation or integrity including trouble and monitor points

Objects are configured in the order added. During the configuration process, each object is assigned a device type and a label. An input device (e.g. pull station, smoke detector, switch, pseudo point, etc.) triggers the execution of a rule. An output device (e.g. NAC, LED, etc.) is the subject of an output command.

Device Types

A device type is assigned to an object to define the operating characteristics of the physical device or logical function the object represents. For example:

- A manual pull station has a device type of PULL
- A waterflow switch has a device type of WATERFLOW
- A tamper switch has a device type of TAMPER
- A smoke detector has a device type of SMOKE
- A heat detector has a device type of HEAT
- A LRM failure pseudo point has a device type of LOCALTROUBLE

For a rule to be valid, the device type used must be applicable for the specific event or command. The 3-SDU will not compile a rule when the wrong device type is used. For example, you can not have a rule which states:

When any LED goes into alarm, turn on all audible devices.

The system's LED is an output device and is turned on because of an active input event. LEDs do not initiate a rule's execution and cannot be used in a rule's input statement.

Labels

The 3-SDU permits identifying objects within the database by using labels instead of numeric addresses. The alphanumeric label can be custom created to describe an object's location, function, and device type. For example:

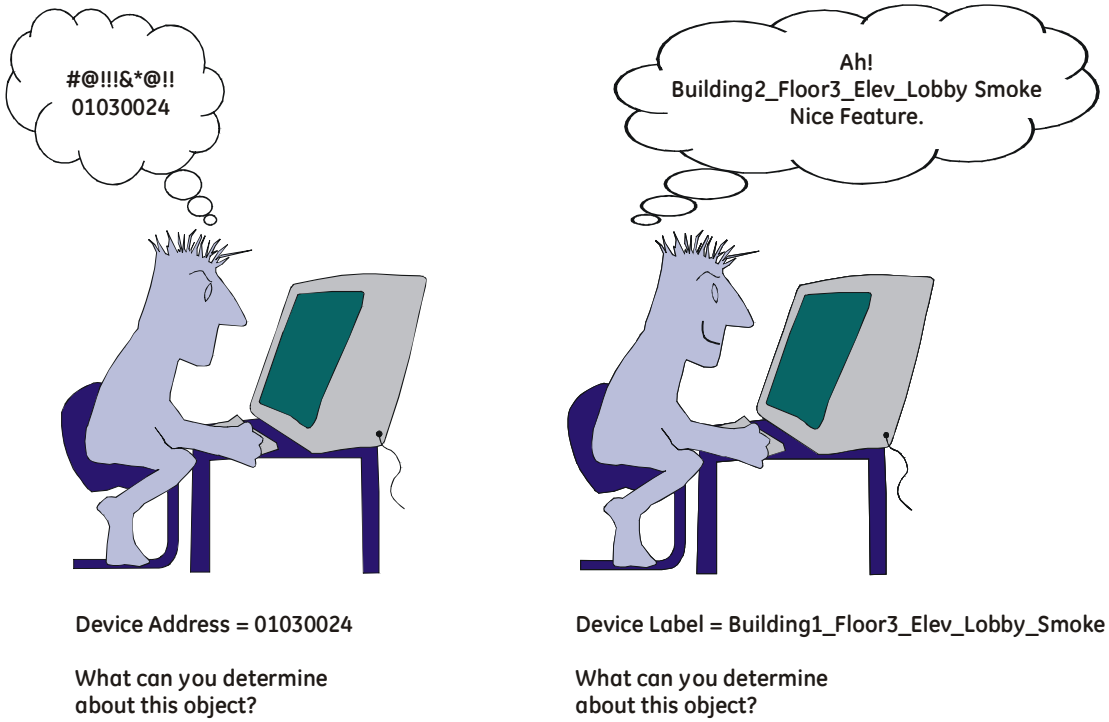


Figure 2-4. Device Address vs. Device Label

In the example above, both the device address and the device label identify the same smoke detector in the elevator lobby, located on the third floor of a multi-floor building, in the second building of a multi-building complex.

Label Parameters

A label is a descriptive alphanumeric combination of text and numbers assigned to an object. A label's syntax identifies a specific object within the database and the fire alarm panel's rules program. An object label should conform to the following parameters:

- **Each label must be unique** - Duplicate labels generate errors during the compiling process
- **Each object within the database should have a label** - Objects without a label generate warning messages and cannot be used within a rule
- **Labels are arbitrary** - Develop labeling conventions to simplify programming
- **Labels contain up to 40 alphanumeric characters and are not case sensitive** -A label is not unique simply by changing the case (e.g. **SMOKE** is the same as **smoke**)
- **Labels contain any alphanumeric ASCII text character syntax,** except for:

Braces	**{** or **}**
Brackets	**[** or **]**
Percent sign	**%**
Number sign	#
Less than symbol	**<**
Greater than symbol	**>**
Asterisk	*****
Comma	**,**
Semicolon	**;**
Colon	**:**
Single quotes	**‘**
Double quotes	**“**
Blank spaces (use underscore when space is desired)_	

A rule label is included between the **[** and **]** within rules. Rule labels differ from object labels since spaces are allowed.

In some cases, labels are automatically generated and assigned. For example, the 3-SDU automatically creates and assigns labels for pseudo points including: Internal Fault, Configuration Fault and Map Fault.

Labeling Plan

Labels must be created and assigned to all objects. Labels are broken into alpha and numeric modifiers.

Another important planning tool is the I/O matrix. An I/O matrix is a hardcopy or computer-based spreadsheet enabling the programmer to list input and output labels and to correlate input object activations to an output object.

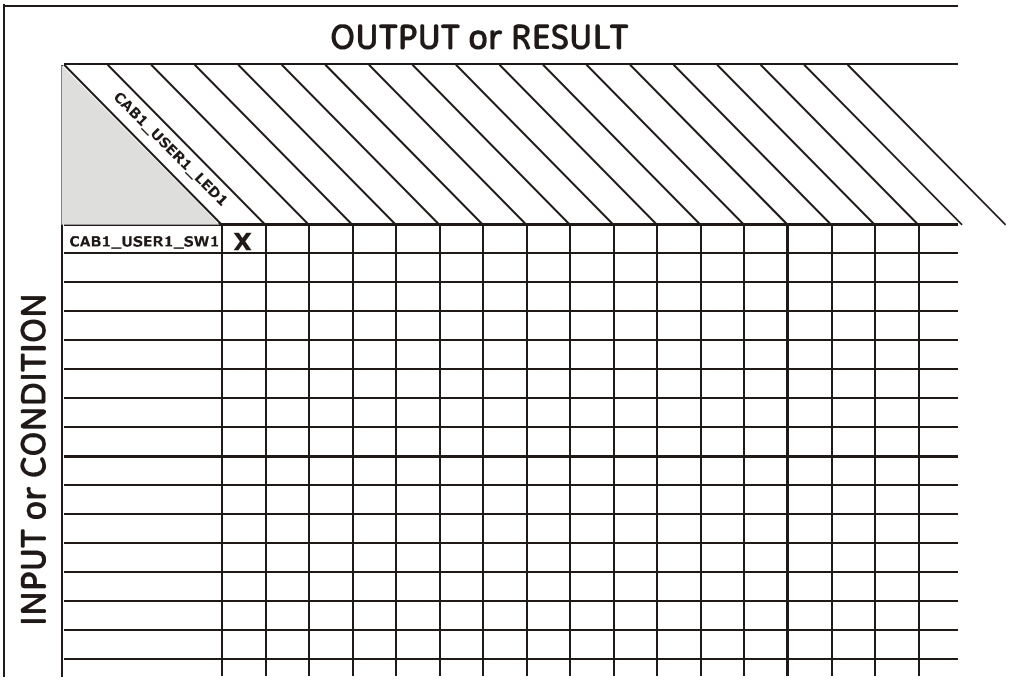


Figure 2-5. Example I/O Matrix Labeling Plan Worksheet

Modifiers

The descriptive component text and numbers used to identify specific locations, components, devices or functions within a label are called modifiers. For example:

Building2_Floor3_Elev_Lobby_Smoke of the previous example.

Building	Identifies the device's high level location
2	Identifies the device's building location
Floor	Identifies the device's location on a floor
3	Identifies the floor's location
Elev_Lobby	Identifies the device's elevator lobby location
Smoke	Identifies the device type

When developing a labeling plan, consideration the fact that labels may be viewed online, on printed reports and on the system display panels. Consistency is of the utmost importance when developing a labeling plan.

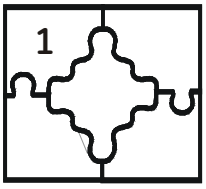
Each object within the database has:

- A **device address** – machine language used by system components for status and control communications
- An **object label** - program language used by the programmer to communicate with the system's CPU
- A **location message** – display language used to post displayed messages to communicate with system operators

Note: Optionally, the object label and location message text can be the same. You should only make the message equal to the label when it makes good business sense.

Five things to consider when developing a labeling plan:

- Label Format
- Label Content
- Common Label Modifiers
- Numbers in Labels
- Labels as Location Messages



Label Format

When formatting labels consider:

- Abbreviating label modifiers. Standardized and documented abbreviations aid those who follow you.
- Using all uppercase characters, all lower case characters, or a combination of both in the labels. Even though labels are not case sensitive, select conventions meaningful to your office.
- Separating label modifiers within labels. Spacing (underscore) in labels can make them easier to read.
- Ordering modifiers within the labels. Consider which modifier would contain the highest location identifier (e.g. Building1, Level2, Floor3, etc.)

An example would be creating labels for several cabinets in an industrial park containing several buildings. The following shows four label formats using different methods for describing the **1st floor** in **building 1** using the same modifiers with different cases:

BLDG1FLOOR1 **BLDG1_FLOOR1** **Bldg_1_Floor_1** **Bldg1_Floor1**

Any of these is correct. Its up to you, the programmer.

The above example abbreviates **Building** in the same way. In this case, avoid using both **BLDG1** and **Bldg1** as label modifiers to reference the same building. The 3-SDU is not case sensitive, but this makes it hard to see duplicate labels when debugging your project's rule program.

Avoid using differing label constructions such as **BLDG1FLOOR1**, **BLDG_1_FLOOR_1**, and **BLDG1_FLOOR1**. Each method uses a different technique to separate modifiers using underscores. Remember, the 3-SDU does not see the **BLDG1FLOOR1** and **BLDG1_FLOOR1** as the same object.

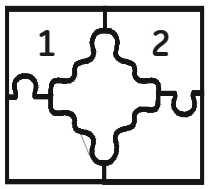
Avoid using differing label constructions such as **BLDG1_FLOOR1** and **FLOOR1_BLDG1**. In this case, the order of the label modifiers is reversed. Start with the biggest or highest location modifier and work down to the smallest modifier (device type for this example).

- First modifier **Building** which is bigger or higher than
- Second modifier **Floor** which is bigger or higher than
- Third modifier **Lobby** which is bigger or higher than
- Last modifier **Smoke** which is the device type (the smallest modifier)

When using this methodology, you do not need to remember which modifier is first, second, third or last. Simply remember the hierarchical order from biggest (first) to smallest (last).

Determine the following factors:

1. Abbreviations
2. Characters and case
3. Modifier separations
4. Modifier order (position)



Label Content

Rules which initiate fire alarm system operations are based on object location, function (alarm, supervisory trouble or monitor), and classification (device type).

A component on the 1st floor of a multi-story building may not perform the same function as a similar component located on the 6th floor. In a system performing audio notification on the floor of incident, the floor above the floor of incident and the floor below the floor of incident perform audio notification on the 1st floor and the 2nd floor. However, a 6th floor fire alarm should perform audio notification on the 6th floor, 7th floor and 5th floor. Thus, system component location modifiers are critical to effective and efficient programming.

If we compare this building's elevator lobby smoke detectors to the hallway smoke detectors, we find the elevator lobby smokes perform the elevator recall function in additional to the audio notification and the hallway smokes only perform audio notification. Thus, the system component function is critical to effective and efficient programming. In this case, we should incorporate an elevator recall modifier (**ELV_RECALL**) into the elevator lobby smoke labels to specify the unique function of these smokes.

Let's assume we are using both smoke and heat detectors in the hallway. The component classification is called device type. The hallway smoke detectors perform smoke detection and control and the hallway heat detectors do not. Thus, the device type for the object being labeled is also critical to effective and efficient programming. For this reason, we may want to incorporate the device type **SMOKE** and **HEAT** modifiers into the respective labels to distinguish between the different classifications.

After labels and modifiers are established for a project, the system uses the 3-SDU to create program statements known as rules to accomplish the required fire alarm system operation. A rule is a programming statement specifying the operation or action taking place when an input event occurs. A rule's input statement and output statement(s) are constructed using object labels and/or device types.

By using a standard set of conventions for modifiers and object label construction, the same rules can be used over and over again with minor editing. Remember, the content of these object labels should include modifiers specifying:

- Location
- Function
- Device type

Labels specify location

Object labels include a modifier specifying location. Label modifiers used to identify a fire alarm system component (device), should indicate the location of an alarm event. For example:

- **BUILDING1** or **BLDG1** or **B1**
- **FLOOR1** or **FLR1** or **F1**
- **LEVEL1** or **LEV1** or **L1**
- **SMOKE1** or **SMK1** or **S1**

A number is added to the modifier to indicate the appropriate location or component when there is more than one. For example, if a program is developed for a 15-story building with cabinets installed on the 1st, 5th , and 10th floors respectively, the programmer might need to know on which floor an alarm occurred. This location information would be necessary if the programmer was required to write a rule for audio notification (the floor above and the floor below) for the floor of the life safety incident. .

The programmer is required to program the system to sound the 3rd, 2nd, and 4th floor audio notification appliances if an alarm occurred on the 3rd floor. The 3rd floor smoke detector input object label in this high-rise application might look like:

FLOOR3_SMOKE1 or **FLR3_SMK1** or **F3_S1**

Inspection of these input labels, which initiate the floor alarm, reveals they are all located on the 3rd floor. This should signal to the programmer that a label modifier such as **FLOOR3**, **FLR3** or **F3** should be included as part of the object label for every 3rd floor device.

Subsequently, it should be easy to see that a **number** should be added to the text modifier to indicate the floor location of each device within the facility. In this example, the key modifier **FLOOR**, **FLR** or **F** should be incorporated into the object for all devices to be configured and programmed. In this way, the SDU checks any device containing **FLOOR3**, **FLR3** or **F3** as part of its object label and checks the number part (**3** for this example) of this modifier, selecting the appropriate input or output device by its floor and number designator.

A typical object label for a system smoke detector in a campus application with multiple buildings might look like:

BUILDING1_FLOOR1_SMOKE1 or **BLDSG1_FLR1_SMK1**

In an application where a system contains a number of LEDs and Switches on Annunciator panels, the installer places the Annunciator panels at the appropriate locations within the facility according to the blueprints. The system programmer needs

to create label modifiers to identify Annunciator locations of the many LEDs and Switches to be programmed. In this case, the floor location of the Annunciator panels may be of importance, but the floor location of the LEDs and Switches are not. The LEDs and Switches are typically located in a row on a panel installed into the Annunciator enclosure.

In this case, it would be important to be able to identify the Annunciator panel containing a LED or Switch. To accomplish this, the object label modifiers to identify LEDs and Switches might look like:

ANNUNCIATOT2_ROW1_LED1	or	**ANN2_ROW1_LED1**
ANNUNCIATOT2_ROW1_SWITCH1	or	**ANN2_ROW1_SW1**

In a system containing two Annunciators with five LED panels, each with 24 LEDs, the programmer identifies the LEDs on each panel. In this case, an object label of **LED24** means little to the programmer. However, an object label of **ANN2_PANNEL5_LED24** tells the programmer which annunciator (**ANN2**), which panel on the 2nd annunciator (**PANEL5**) and which LED on the 5th panel (**LED24**).

Labels specify function

When appropriate, an object label should describe the device of logical function. Some devices in a system by design provide a specific function. A system might include operator panel and amplifier devices performing specific functions. In this case, the object label should include modifiers identifying the device's function. Devices providing a specific system function might include:

1. A damper control switch's object label in a multi-floor building might be labeled:

   FLOOR1_DAMPER_CONTROL_SWITCH1
   FLR1_DMPR_SNTRL_SW1

2. An amplifier for the 7th floor designated as the 7th level in a multi-floor building might be labeled:

   AMPLIFIER_LEVEL7
   AMP_LEV7

3. The 1st LED on an Annunciator LED panel on the 1st row might be labeled:

   ANNUNCIATOR_ROW1_LED1
   ANN_ROW1_LED1

Let's add a label modifier to the previously constructed object label for the 3rd floor smoke detector to identify this detector as an elevator lobby smoke detector used in an elevator capture rule:

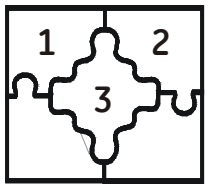
FLOOR3_SMOKE1_**ELEVATOR_LOBBY**
FLR3_SMK1_**ELV_LOB**

As you can see, the key modifier **ELEVATOR_LOBBY** or **ELV_LOB** is incorporated into the smoke detector's object label to be used in the elevator capture rule, excluding all non-elevator capture device labels. Therefore, any device having these modifiers as part of their object label will initiate the elevator capture rule. Remember to be consistent, use **ELEVATOR_LOBBY** or **ELV_LOB** not both.

Labels specify device type

Where appropriate, object labels should include a modifier to describe the component's device type. A system may contain any number of devices including: smoke detectors, heat detectors, manual pull stations, input modules and output modules. Consider using a device type to easily identify the device by reading the device's object label. For example:

- A smoke detector on the 1st floor might be labeled: **FLOOR1_SMOKE**
- A pull station on the 1st floor might be labeled**:** **FLOOR1_PULL**
- The 1st LED in row 1 of an annunciator panel might be labeled**:** **ANNUNCIATOR2_ROW1_LED1**
- The 1st Switch in row 1 of an annunciator panel might be labeled**:** **ANNUNCIATOR2_ROW1_SWITCH1**



Common Label Modifiers

Common label modifiers assign labels faster by using the 3-SDU's Prefabricated label editor and permit the usage of wildcards and N-variables when writing rules. The 3-SDU also contains a Prefabricated message utility.

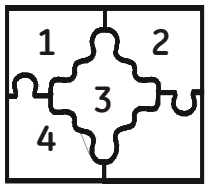
Basements, mezzanines and parking levels are examples of areas not typically referred to by floor numbers. Also the 13th floor is typically called the 14th floor, not a logical order.

On projects where non-numbered floors or floors with varying designators exist, you might want to use a more generic common label modifier convention such as level. This applies to all levels of the facility. By converting numbered floors, non-numbered areas and areas with different designators to levels, all areas can be labeled with common label modifiers. For example:

Each facility specific area might be labeled:

Maintenance Level	LEVEL16
Floor 14	LEVEL15
Floor 12	LEVEL14
Floor 11	LEVEL13
Floor 10	LEVEL12
Floor 9	LEVEL11
Floor 8	LEVEL10
Floor 7	LEVEL9
Floor 6	LEVEL8
Floor 5	LEVEL7
Floor 4	LEVEL6
Floor 3	LEVEL5
Floor 2	LEVEL4
Mezzanine	LEVEL3
Parking Level 1	LEVEL2
Parking Level 2	LEVEL1

Configure the location messages to use the original facility specific terminology (Parking Level, Mezzanine, Floor, etc).



Numbers in Labels

Using numbers in labels is particularly useful because of the programmer's need to uniquely identify many objects having common label modifiers. For example:

Example 1	Example 2
FLOOR3_DUCT1	**FLOOR1_DUCT1**
FLOOR3_DUCT2	**FLOOR2_DUCT1**
FLOOR3_DUCT3	**FLOOR3_DUCT1**
FLOOR3_DUCT4	**FLOOR4_DUCT1**

Using numbers has the added advantage of permitting the programmer to use N-variable syntax when writing rules. In Example 1 above, the numbers after **DUCT** are used to make each label unique by identifying four duct detectors on the 3rd floor. The common modifiers identify the 3rd floor location. **DUCT** identifies the device type and the numbers identify each individual duct detector. The labels in example 2 show the labels for the 1st duct detector on each floor.

Using the modifier LEVEL with numbers instead of the facility specific terminology makes it easier to write the floor of incident, floor above and floor below rule sequence. This is especially true when sub-levels, such as basements, mezzanines, and parking levels exist in the subject building.

Another example using numbers in levels is to use **LEVEL2_SMOKE1** as the label for a smoke detector on the 2nd level of a facility. The number 1 is added after smoke in the label to make it unique, assuming there is more than one smoke detector on the 2nd level performing a common function.

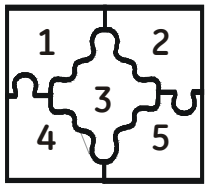
In reality, when rooms exist where smoke detectors are installed, the room number can replace the variable number to make the labels unique. For example:

LEVEL2_SMOKE_RM201, **LEVEL2_SMOKE_RM202**

From a programming prospective, a label with one or more common modifiers and unique numbers is advantageous. When it comes time to write rules, the **LEVEL2** can easily be replaced with **LEVEL3** for the next level's objects.

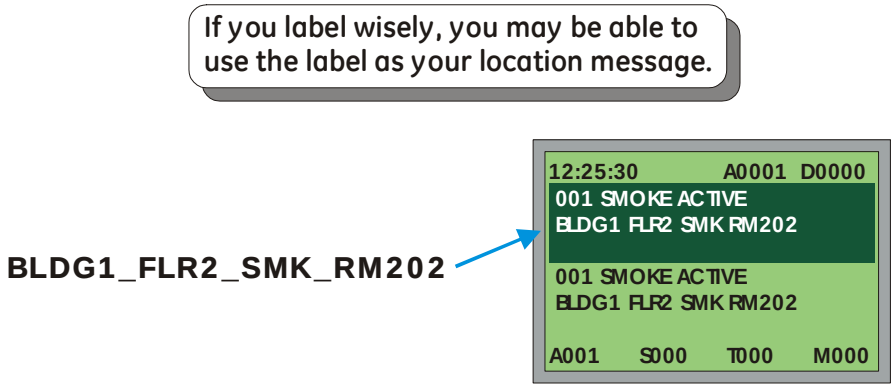
When using numbers in labels, use the advanced programming technique of N-variables to specify a range of specific objects (at the exclusion of others) using the numbers within their labels. If levels 1, 2, 4 through 11 elevator lobby smokes provide primary elevator recall and level 3 elevator lobby smoke provides alternate elevator recall, the input object label for the rule for primary recall is easily written as 'LEVEL**<N:1,2,4-11>**_ELV_LOB_SMK' using an N-variable.

LEVEL11_ELV LOB_SMK
LEVEL10_ELV LOB_SMK
LEVEL9_ELV LOB_SMK
LEVEL8_ELV LOB_SMK
LEVEL7_ELV LOB_SMK
LEVEL6_ELV LOB_SMK
LEVEL5_ELV LOB_SMK
LEVEL4_ELV LOB_SMK
LEVEL3_ELV LOB_SMK
LEVEL2_ELV LOB_SMK
LEVEL1_ELV LOB_SMK



Labels as Location Messages

The last piece of the puzzle is to use labels as location messages, when it is practical to do so. Simply program the system to display a different location message when each related object goes active. If you keep this in mind when designing your labels, you can use the labels as your location messages instead of creating a custom message for each object.



Standard Labeling Conventions

Standardized conventions significantly eliminate confusion. In order to facilitate the rules programming process, standardized labeling conventions should be created and used by the programmers, engineers and technicians. These should be added to your 3-SDU's Prefabricated label library.

To improve the readability and understanding of the project's rules for the installers, maintenance personnel and the end user's personnel, it is recommended to insert a text leader at the top of the rule file.

Consider the following parameters in your labeling plan:

1. Label format standard:

    - Abbreviations
    - Characters
    - Modifier separation
    - Modifier order

2. Label content includes:

    - Location
    - Function
    - Device type

3. Common label modifiers

4. Numbers in labels

5. Labels as location messages

Placing braces **{Header Text}** around the header containing the conventions used lets you include them in your rules file as text. This header is created in the 3-SDU's default rules editor or any ASCII text editor (e.g. WordPad or Notepad). Create a **Rules Conventions** directory to save your standardized header text file named **header.txt**. A common library of rules used from project to project should also be included. Remember, any example rules provided in the **HELP** utility may be copied into your rules editor.

{The labeling conventions and standard abbreviations shown are used for this project:

Smoke Detector	**SMK**
Manual Pull Station	**PULL**
Speaker	**SPK**
Amplifier	**AMP**
Duct Smoke Detector	**DUCT**
Sprinkler Waterflow Switch	**SPR**
Waterflow	**FLOW**
Tamper	**TAMP**
Door Monitor Contact	**DMC**
Monitor Contact	**MON**
Horn	**HRN**
Strobe	**STB**
Building	**BLDG**
Floor	**FLR**
Level	**LEV**
Annunciator	**ANN**
Cabinet	**CAB**
Fire Phone	**FPH**
Programmable display panel LED	**LED**
Programmable display panel switch	**SW**

Characters may be upper case, lower case or a combination.

Modifiers are separated with an underscore (spaces may not be used).

LEV1_SMK

Typical labels have higher location modifier listed first, followed by lower modifiers in descending order.

Typical labels end with a device type.}

Prefabricated Text Editor

The 3-SDU has been designed with a Prefabricated Label Editor. This is used during the object configuration process to automate the object labeling process.

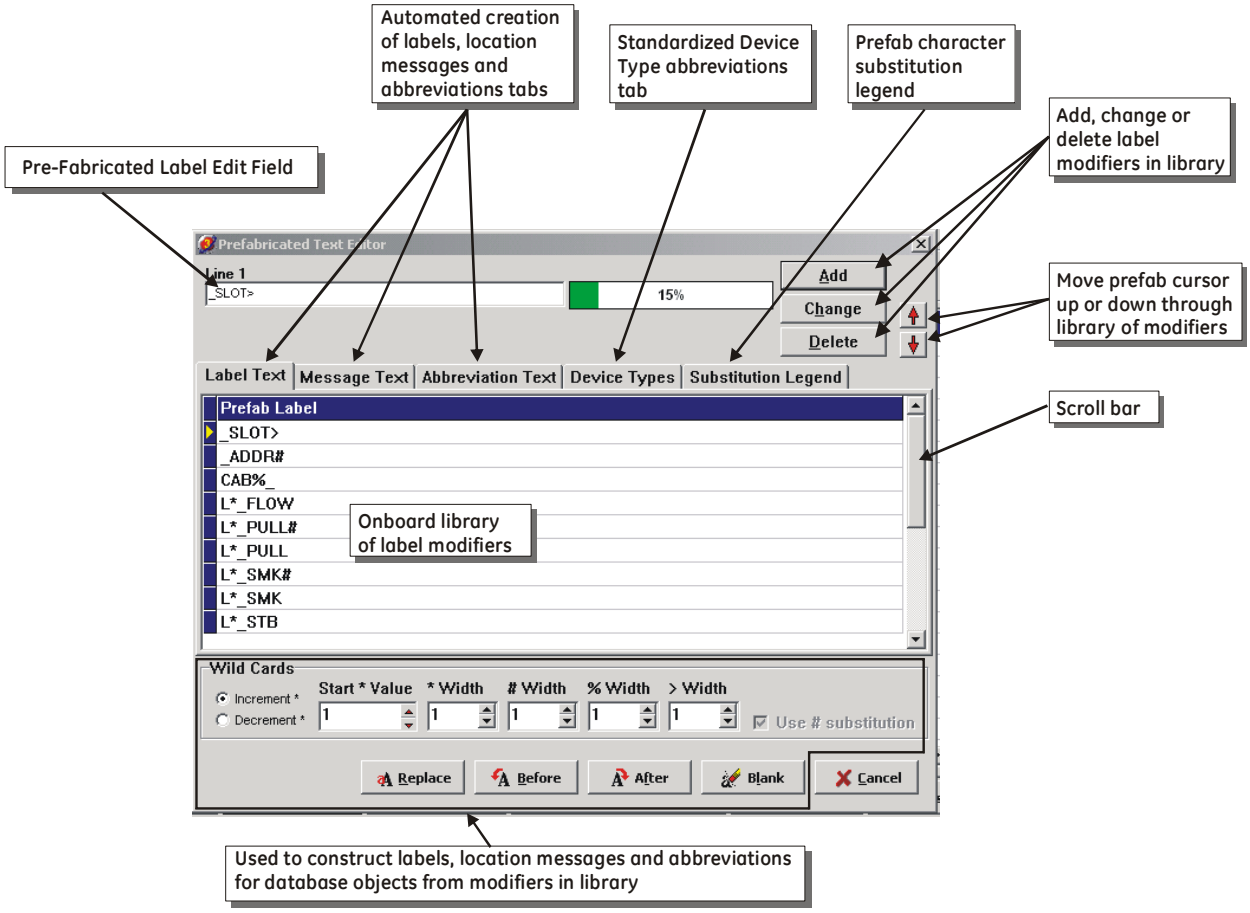


Figure 2-6. Prefabricated Text Editor

This Prefabricated Text Editor enables maintenance of an onboard library of label, location message and abbreviation modifier conventions used to create object labels, location messages and abbreviations. Special prefabricated substitution characters are available to automate the labeling process. These characters are defined under the **Substitution Legend** tab. These substitution characters are:

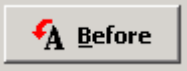
- ***** - Insert an increment or decrement sequence on numbers at a modifier position within multiple objects within the database labels
- # - Insert the device address as part of the object label for the object(s) labeled
- **%** - Insert the cabinet address as part of the object label for the object(s) labeled

- **>** - Insert the slot position as part of the object label for the object(s) labeled
- **<** - Insert the standardized device type abbreviation as part of the object label for the object(s) labeled

This editor enables you to Add, Change or Delete modifiers within the library. This editor and its library is part of the 3-SDU application and is not part of the project database. It is used from project to project to configure objects in the respective project's database.

The **Start * Value** and **Increment/Decrement** features enable you to sequentially add number modifiers to labels, location messages and abbreviations. This feature lets you number multiple quantities of common objects sequentially by simply placing a ***** at a modifier position in the edit field. The ***** uses an onboard increment or decrement counter by replacing the * with a sequential range of incrementing or decrementing numbers when labeling object in the database.

The **Replace** button enables you to replace object labels with a modifier selected in the Prefabricated library



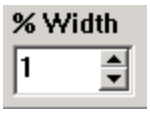
The **Before** button enables you to add a modifier from the library in front of the existing object labels in the object configuration screen

The **After** button enables you to add a modifier from the library behind or after the existing object labels in the object configuration screen

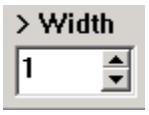
The **Blank** button enables you to erase existing object labels in the object configuration screen

The *** Width** field specifies the number of integers used for the sequence. A width of 1 inserts 1, 2, 3, etc. A width of 2 inserts 01, 02, 03 and a width of 3 inserts 001, 002, 003, etc. The maximum width is 6.

The **# Width** field specifies the number of integers used for the device address. This # width behaves like the * width. The maximum width is 4.



The **% Width** field specifies the number of integers used for the cabinet address. This % width behaves like the * width. The maximum width for a cabinet address is 2.



The **> Width** field specifies the number of integers used for the slot position. This > width behaves like the * width. The maximum width for a slot position is 2.

The above tools are used to create locations messages under the **Message Text** tab.

Labeling objects is accomplished in the Object Configuration screen. To label objects:

1. Filter the device type you wish to label
2. Pick the objects you wish to label
3. Select the Prefabricated Text Editor

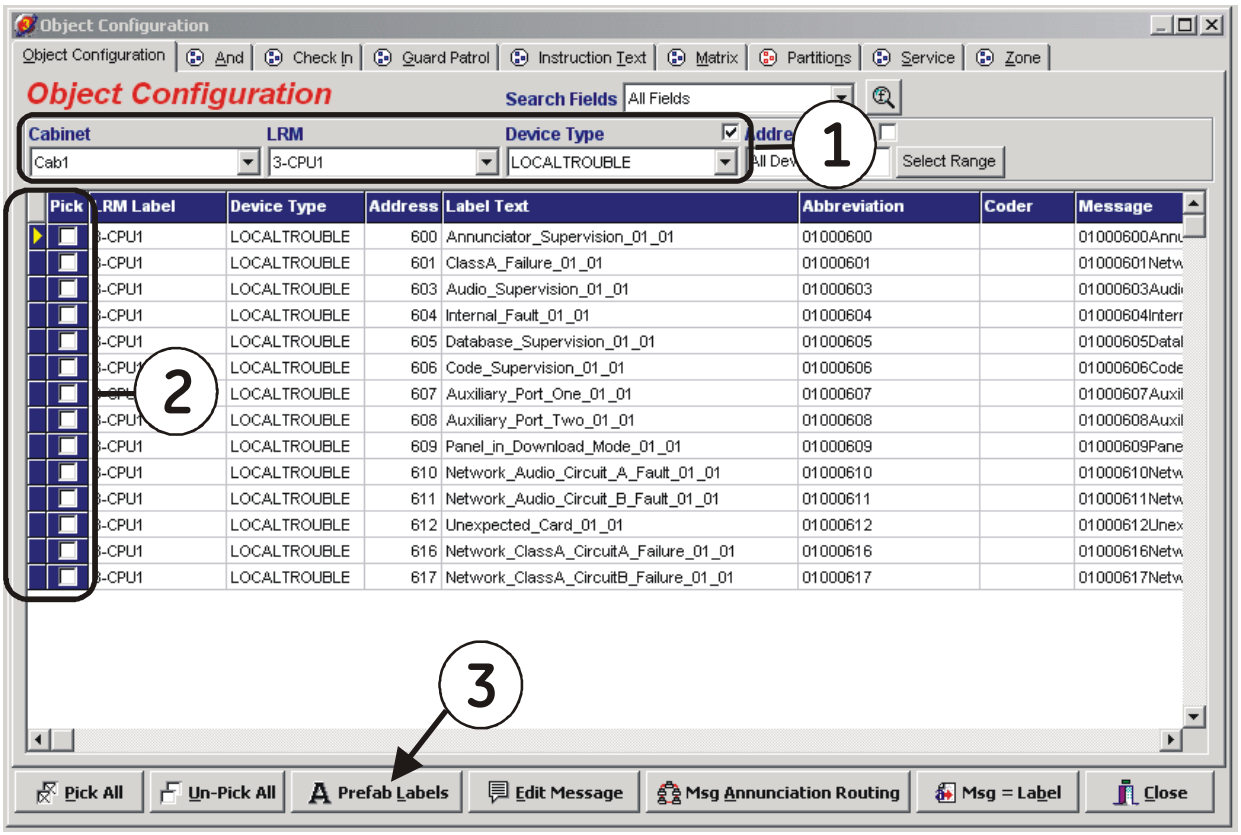


Figure 2-7. Label objects in the Objects Configuration Screen

Prefabricated Label Library

Establish a standardized modifier library for the Prefabricated Text Editor. The editor's substitution characters automatically assign numbers, address or slot locations and guarantees a unique labeling syntax.

- The **%** character automatically adds the cabinet or node address to the label
- The **>** character automatically adds the slot position to the label
- The # character automatically adds the device address to the label
- The **<** character automatically adds the device type abbreviation to the label
- The ***** character automatically sequentially increments or decrements the number within the label

A label modifier added to the Prefabricated Text Library might look like this:

ANNUNCIATOR%_ or **ANN%_**

In a system containing two annunciators (cabinet 2 and cabinet 3), the Prefabricated Text Editor automatically inserts **ANNUNCIATOR2**_ or **ANN2**_ for cabinet 2 objects and inserts **ANNUNCIATOR3**_ or **ANN3**_ for cabinet 3 objects.

If 3 rows of 12LED/12Switch panels are installed in the annunciator enclosures, each LED and Switch needs to be identified individually for programming purposes. To accomplish this, label modifiers would be added to the Prefabricated Text Editor library that might look like this:

For the LEDs:

ANNUNCIATOR%_ROW1_LED* or **ANN%_ ROW1_LED***
ANNUNCIATOR%_ROW2_LED* or **ANN%_ ROW2_LED***
ANNUNCIATOR%_ROW3_LED* or **ANN%_ ROW3_LED***

For the Switches:

ANNUNCIATOR%_ROW1_SW* or **ANN%_ ROW1_SW***
ANNUNCIATOR%_ROW2_SW* or **ANN%_ ROW2_SW***
ANNUNCIATOR%_ROW3_SW* or **ANN%_ ROW3_SW***

Another alternative might be to add modifiers to the library on a modifier by modifier basis such as:

ANNUNCIATOR%_ or **ANN%_**
ROW*_
LED*
SW*

In the first example, the programmer can pick the 12 LEDs of the 1st row on annunciator 2 in the object configuration screen and use the Prefabricated Text Editor, picking the **ANNUNCIATOR%_ROW1_LED*** or **ANN%_ROW1_LED*** to label these LEDs as:

ANNUNCIATOR2_ROW1_LED1	or	**ANN2_ ROW1_LED1**
ANNUNCIATOR2_ROW1_LED2	or	**ANN2_ ROW1_LED2**
ANNUNCIATOR2_ROW1_LED3	or	**ANN2_ ROW1_LED3**
ANNUNCIATOR2_ROW1_LED4	or	**ANN2_ ROW1_LED4**
ANNUNCIATOR2_ROW1_LED5	or	**ANN2_ ROW1_LED5**
ANNUNCIATOR2_ROW1_LED6	or	**ANN2_ ROW1_LED6**
ANNUNCIATOR2_ROW1_LED7	or	**ANN2_ ROW1_LED7**
ANNUNCIATOR2_ROW1_LED8	or	**ANN2_ ROW1_LED8**
ANNUNCIATOR2_ROW1_LED9	or	**ANN2_ ROW1_LED9**
ANNUNCIATOR2_ROW1_LED10	or	**ANN2_ ROW1_LED10**
ANNUNCIATOR2_ROW1_LED11	or	**ANN2_ ROW1_LED11**
ANNUNCIATOR2_ROW1_LED12	or	**ANN2_ ROW1_LED12**

In the second example, the programmer uses the Replace button to insert **ANNUNCIATOR2_** or **ANN2_** as the high order modifier, then uses the After button to insert **ROW*_** and the second modifier and finally uses the After button to insert **LED*** as the low order modifier.

The system programmer picks the 12 LEDs of the first row on annunciator 3 and uses either method to label these LEDs as:

ANNUNCIATOR3_ROW1_LED1	or	**ANN3_ ROW1_LED1**
ANNUNCIATOR3_ROW1_LED2	or	**ANN3_ ROW1_LED2**
ANNUNCIATOR3_ROW1_LED3	or	**ANN3_ ROW1_LED3**
ANNUNCIATOR3_ROW1_LED4	or	**ANN3_ ROW1_LED4**
ANNUNCIATOR3_ROW1_LED5	or	**ANN3_ ROW1_LED5**
ANNUNCIATOR3_ROW1_LED6	or	**ANN3_ ROW1_LED6**
ANNUNCIATOR3_ROW1_LED7	or	**ANN3_ ROW1_LED7**
ANNUNCIATOR3_ROW1_LED8	or	**ANN3_ ROW1_LED8**
ANNUNCIATOR3_ROW1_LED9	or	**ANN3_ ROW1_LED9**
ANNUNCIATOR3_ROW1_LED10	or	**ANN3_ ROW1_LED10**
ANNUNCIATOR3_ROW1_LED11	or	**ANN3_ ROW1_LED11**
ANNUNCIATOR3_ROW1_LED12	or	**ANN3_ ROW1_LED12**

By using the same modifier from the Prefabricated Text Editor library, with the appropriate characters, you can quickly label any number of objects.

Another example of using an appropriate character to automatically place numbers in a label is a 5-story building with 6 smoke detectors per floor. This location information is critical if the programmer is required to write a rule for audio notification for the floor of incident, the floor above and the floor below. The system programmer picks the six smokes for the 1st floor in the objects configuration screen. The programmer uses the Prefabricated Text Editor to insert modifiers to build labels for the smoke detectors.

FLOOR1_SMOKE*	or	**FLR1_SMK***
FLOOR*_	or	**FLR*_**
SMOKE*	or	**SMK***

The resulting smoke detector labels would be:

FLOOR1_SMOKE1	or	**FLR1_SMK1**
FLOOR1_SMOKE2	or	**FLR1_SMK2**
FLOOR1_SMOKE3	or	**FLR1_SMK3**
FLOOR1_SMOKE4	or	**FLR1_SMK4**
FLOOR1_SMOKE5	or	**FLR1_SMK5**
FLOOR1_SMOKE6	or	**FLR1_SMK6**

The system programmer picks the 6 smoke detectors for the 2nd floor and uses the Prefabricated Text Editor to label these:

FLOOR2_SMOKE1	or	**FLR2_SMK1**
FLOOR2_SMOKE2	or	**FLR2_SMK2**
FLOOR2_SMOKE3	or	**FLR2_SMK3**
FLOOR2_SMOKE4	or	**FLR2_SMK4**
FLOOR2_SMOKE5	or	**FLR1_SMK5**
FLOOR2_SMOKE6	or	**FLR2_SMK6**

The system programmer continues this sequence for the smoke detectors on the remaining floors.

Prefabricated Label Library Examples

Levels with numbers instead of facility specific terminology make it easier to write the floor of incident rules. Level is used in this document as an example of a common area modifier. Let's look at some examples of Prefabricated Text Editor modifiers using level:

Smoke detectors by level:

LEVEL1_SMOKE*	or	**LEV1_SMK***
LEVEL2_SMOKE*	or	**LEV2_SMK***
LEVEL3_SMOKE*	or	**LEV3_SMK***
or		or
LEVEL*_	or	**LEV*_**
SMOKE*	or	**SMK***

Manual pull stations by level:

LEVEL1_PULL*	or	**LEV1_PULL***
LEVEL2_PULL*	or	**LEV2_PULL***
LEVEL3_PULL*	or	**LEV3_PULL***
or		or
LEVEL*_	or	**LEV*_**
PULL*	or	**PULL***

Firefighters' telephone by level:

LEVEL1_PHONE*	or	**LEV1_PHONE***
LEVEL2_PHONE*	or	**LEV2_PHONE***
LEVEL3_PHONE*	or	**LEV3_PHONE***
or		or
LEVEL*_	or	**LEV*_**
PHONE*	or	**PHONE***

Heat detector by level:

LEVEL1_HEAT*	or	**LEV1_HEAT***
LEVEL2_HEAT*	or	**LEV2_HEAT***
LEVEL3_HEAT*	or	**LEV3_HEAT***
or		or
LEVEL*_	or	**LEV*_**
HEAT*	or	**HEAT***

Waterflow by level:

LEVEL1_FLOW*	or	**LEV1_FLOW***
LEVEL2_FLOW*	or	**LEV2_FLOW***
LEVEL3_FLOW*	or	**LEV3_FLOW***
or		or
LEVEL*_	or	**LEV*_**
FLOW*	or	**FLOW***

Visible circuit by level:

LEVEL1_VISIBLE*	or	**LEV1_VIS***
LEVEL2_VISIBLE*	or	**LEV2_VIS***
LEVEL3_VISIBLE*	or	**LEV3_VIS***
or		or
LEVEL*_	or	**LEV*_**
VISIBLE*	or	**VIS***

Amplifier output circuit by level:

LEVEL1_AMPLIFIER*	or	**LEV1_AMP***
LEVEL2_AMPLIFIER*	or	**LEV2_AMP***
LEVEL3_AMPLIFIER*	or	**LEV3_AMP***
or		or
LEVEL*_	or	**LEV*_**
AMPLIFIER*	or	**AMP***

Appendix B
Main building device board and schematics

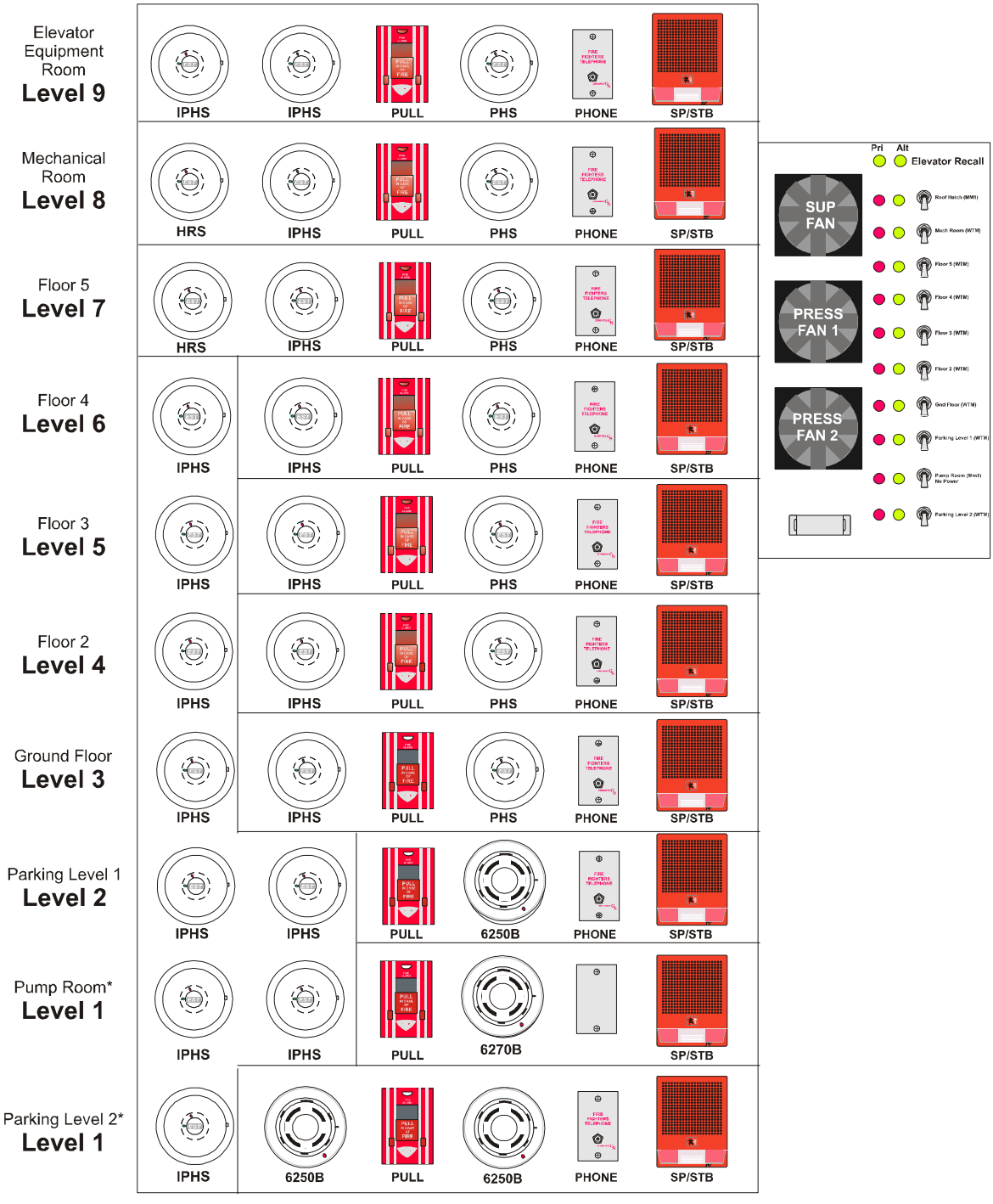
Introduction

This section contains the MacNeill Plaza Main Building drawings for the device board, simulation panel and schematics.

UTC Fire & Security

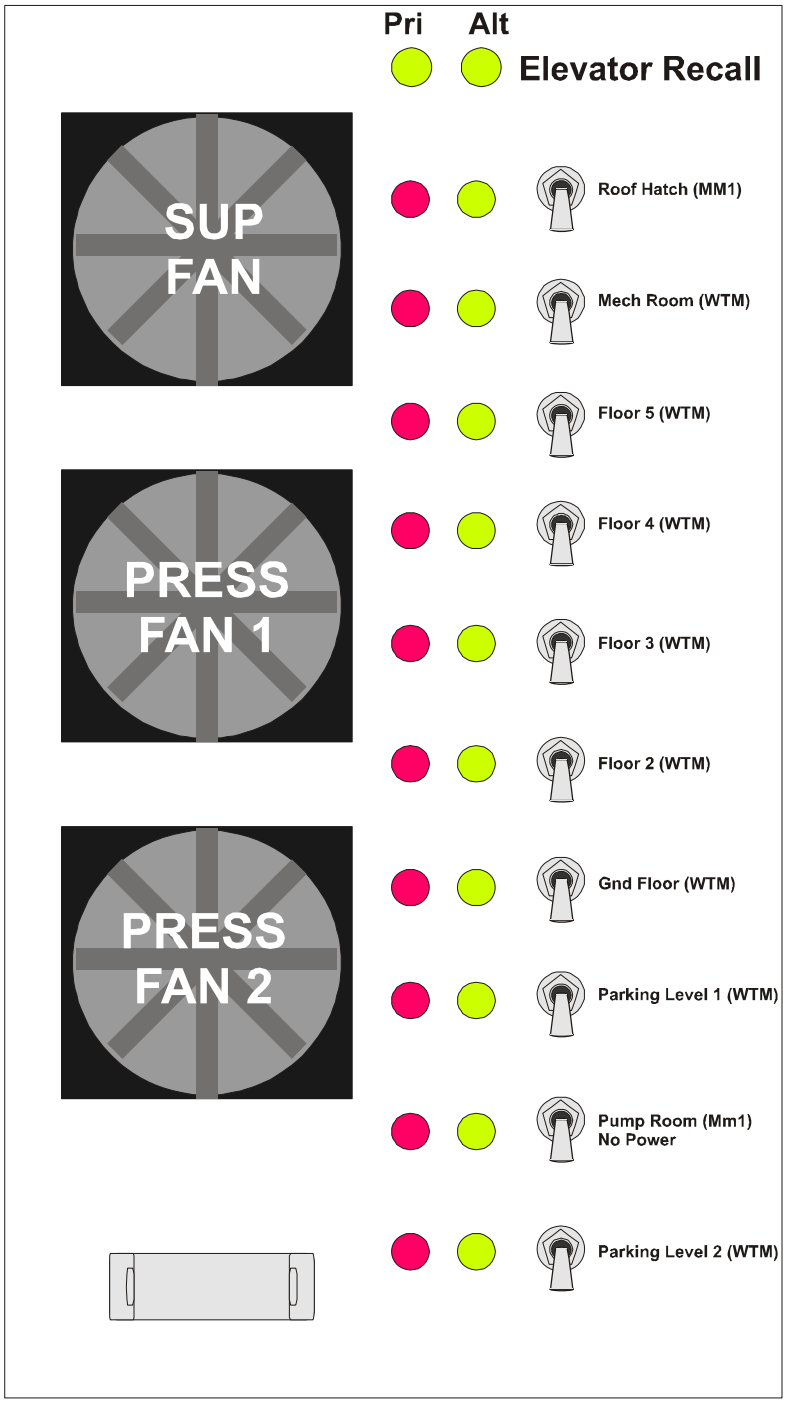
A United Technologies Company

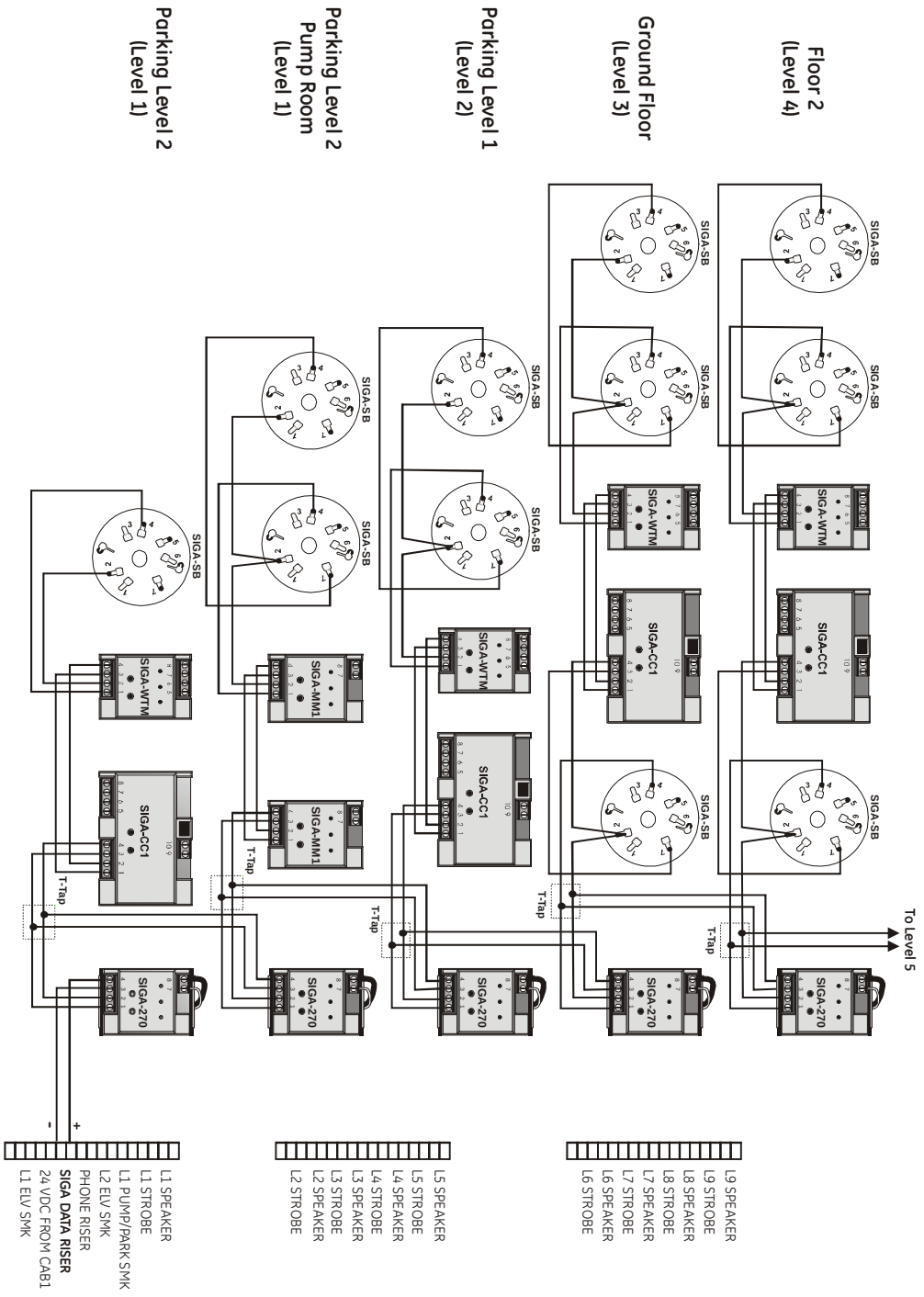
MacNeill Plaza Main Building Device Board Layout



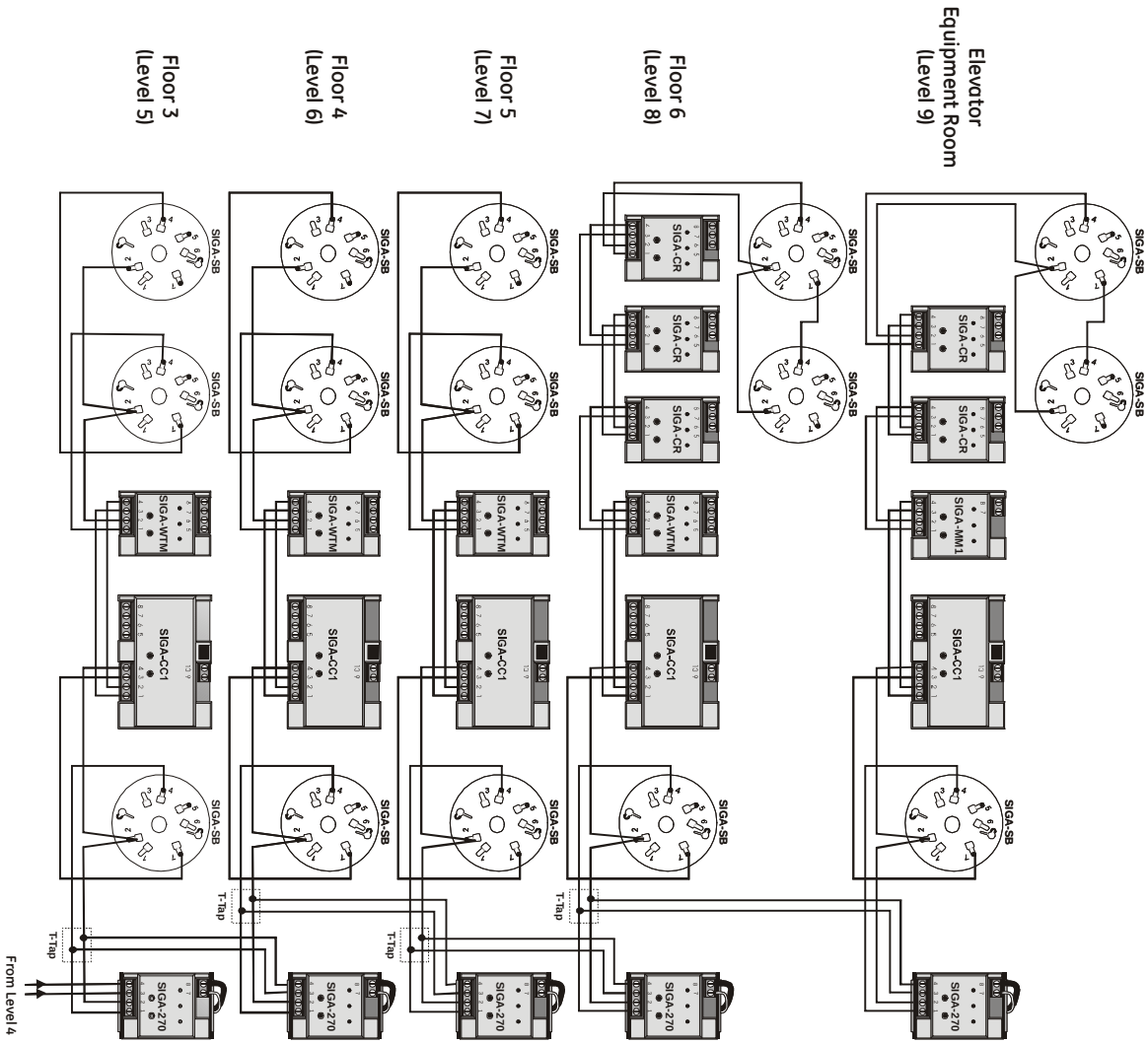
* Parking Level 2 & Pump Room are on the same Level.

MacNeill Plaza Main Building Simulation Panel

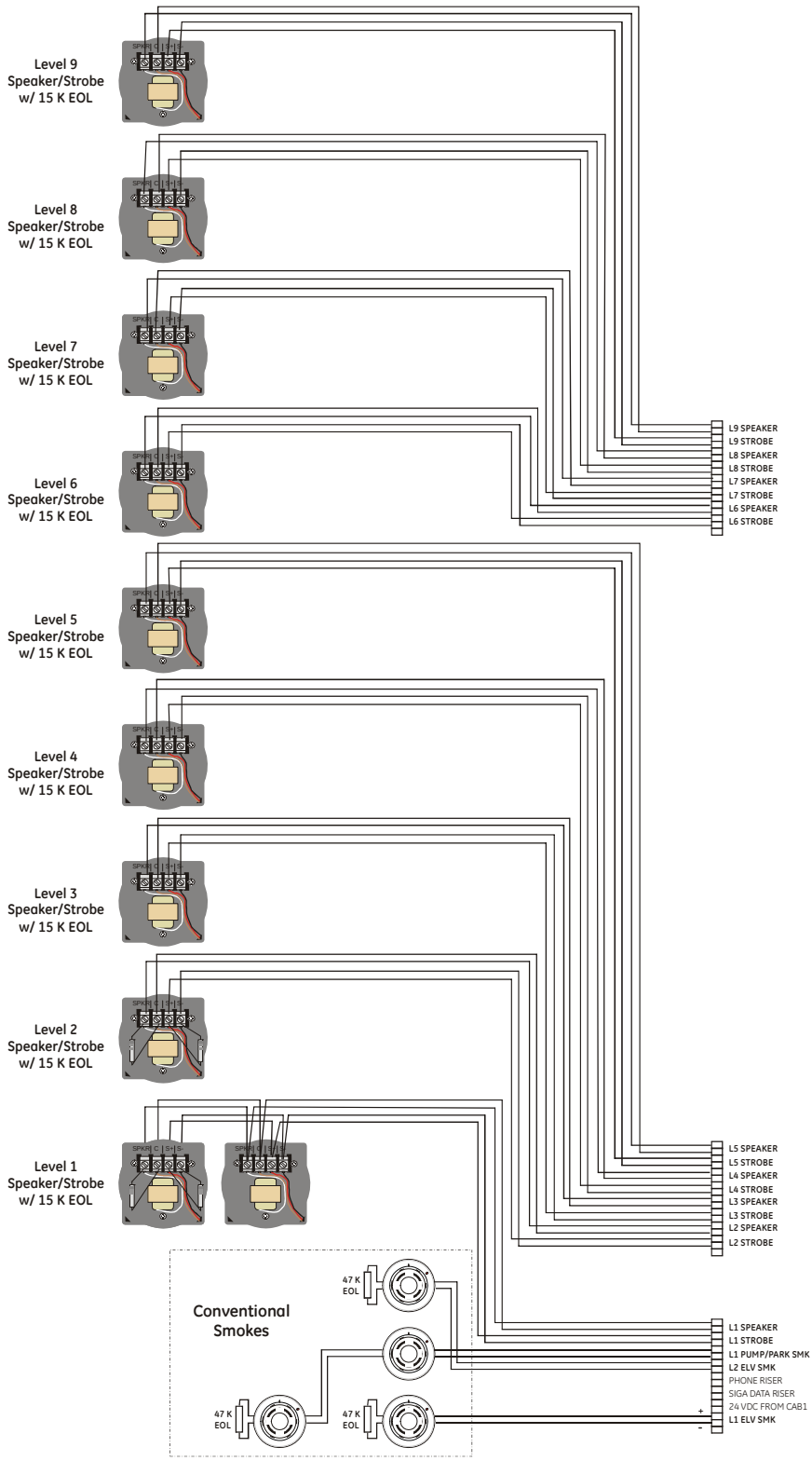




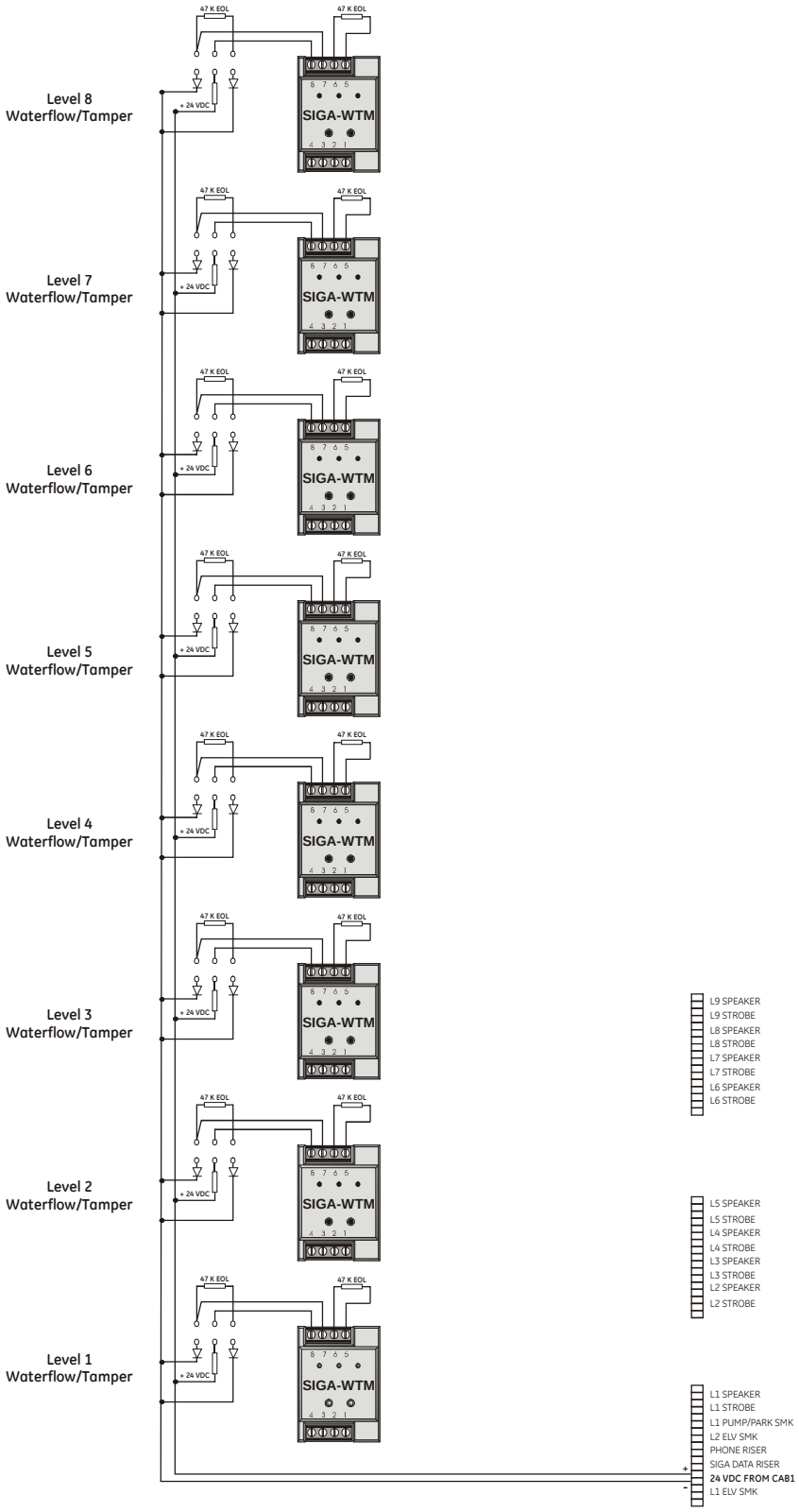
MacNeill Plaza Main Building SIGA Data Riser Levels 1 through 4



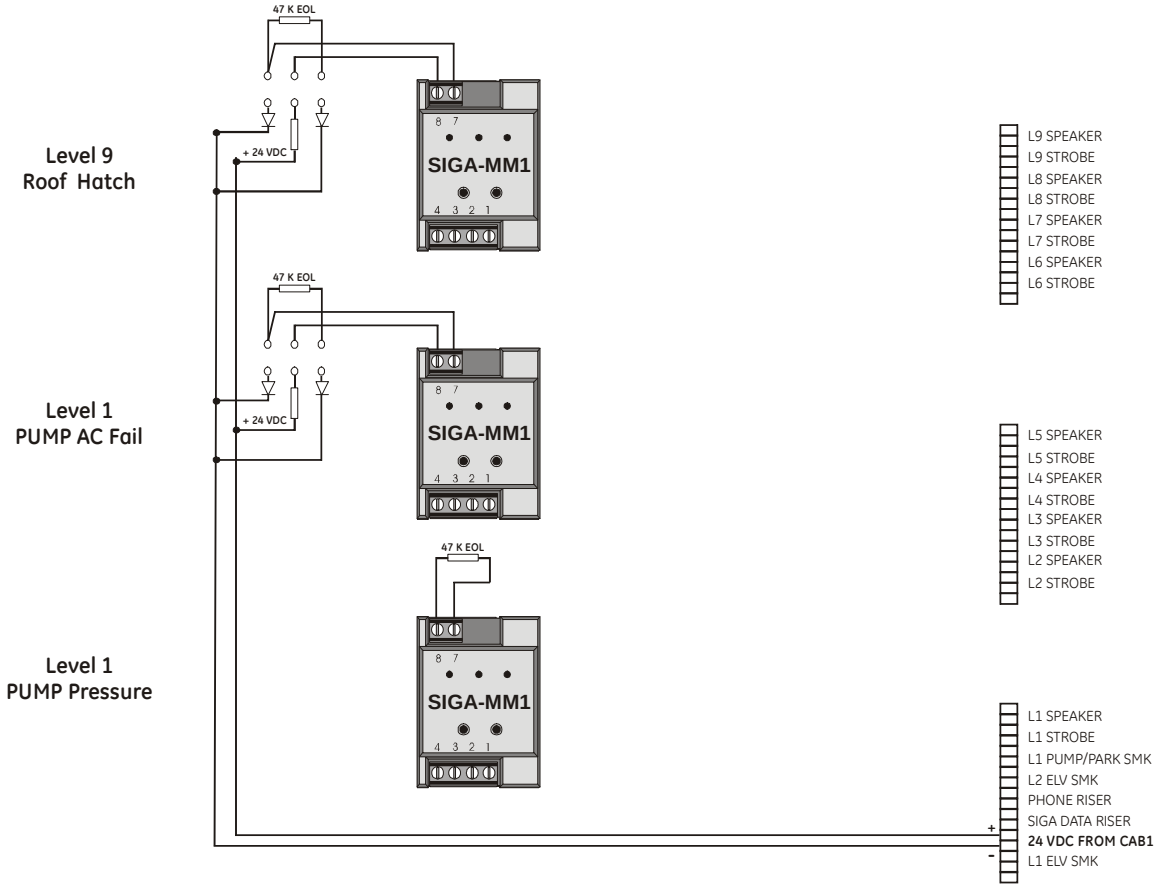
MacNeill Plaza Main Building SIGA Data Riser Levels 5 through 9



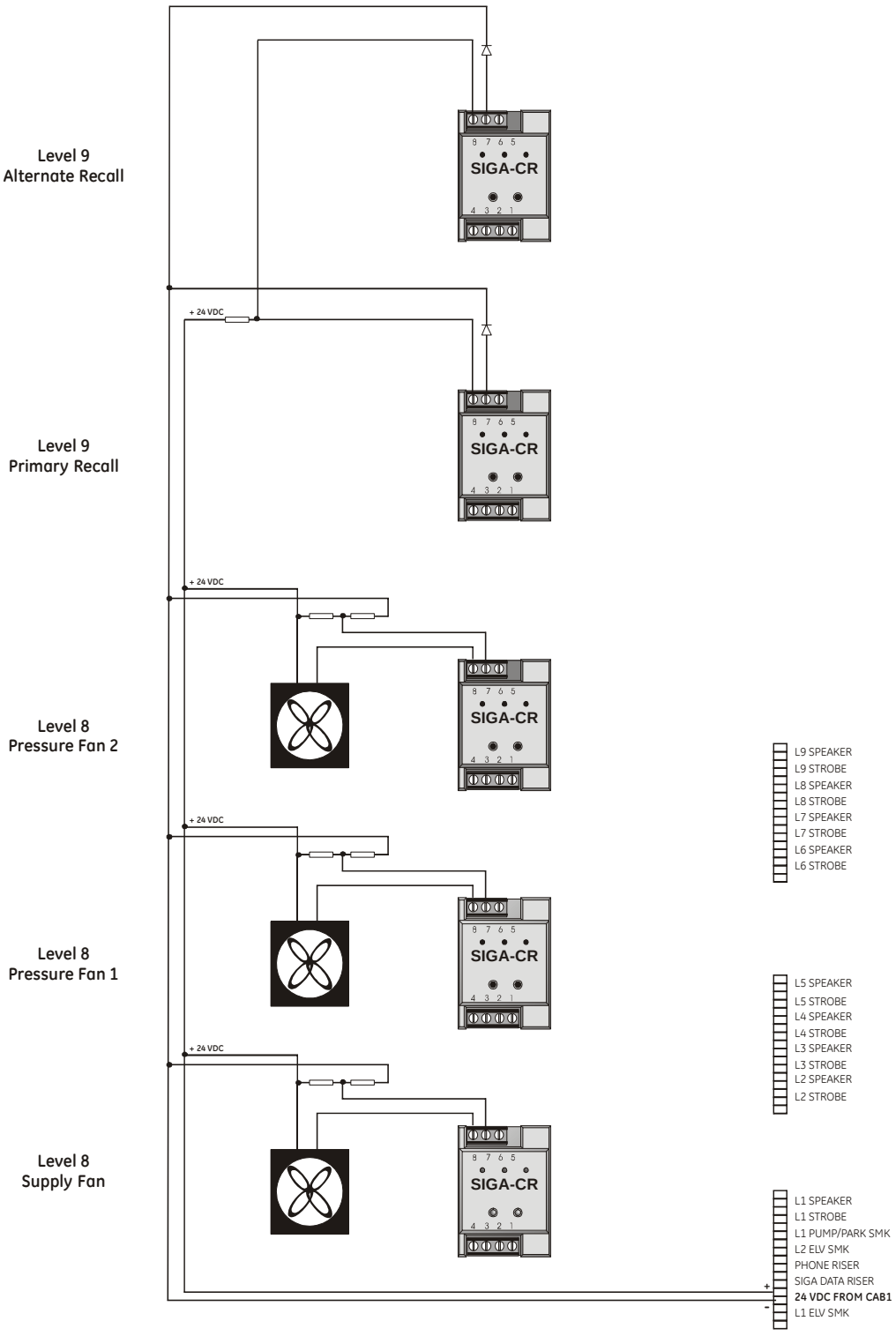
MacNeill Plaza Main Building Speaker, Strobe and Conventional Smoke Wiring



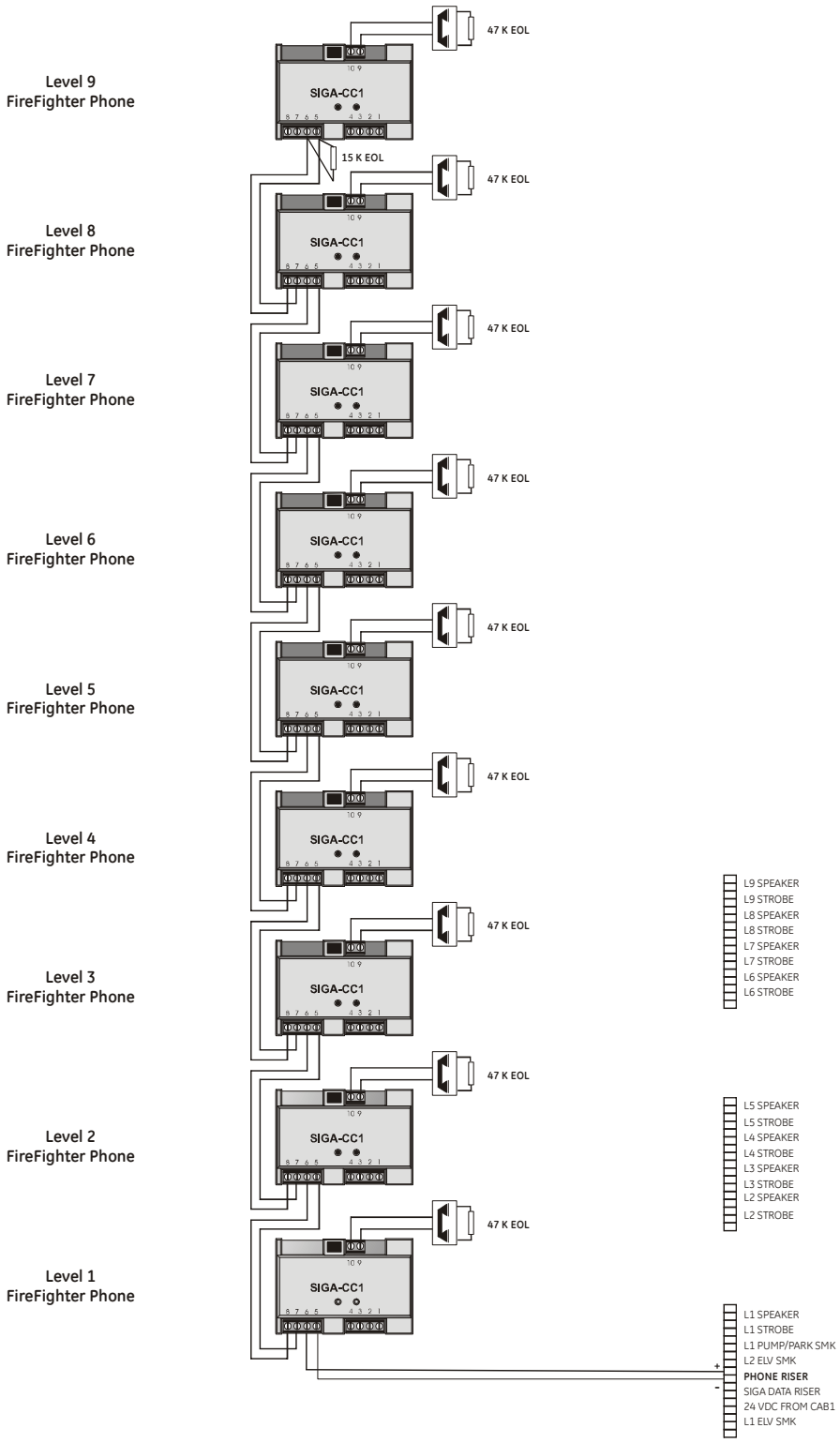
MacNeill Plaza Main Building Waterflow and Tamper Simulation Wiring



MacNeill Plaza Main Building Monitor Module Simulation Wiring



MacNeill Plaza Main Building Control Relay Simulation Wiring



MacNeill Plaza Main Building Phone Riser Wiring

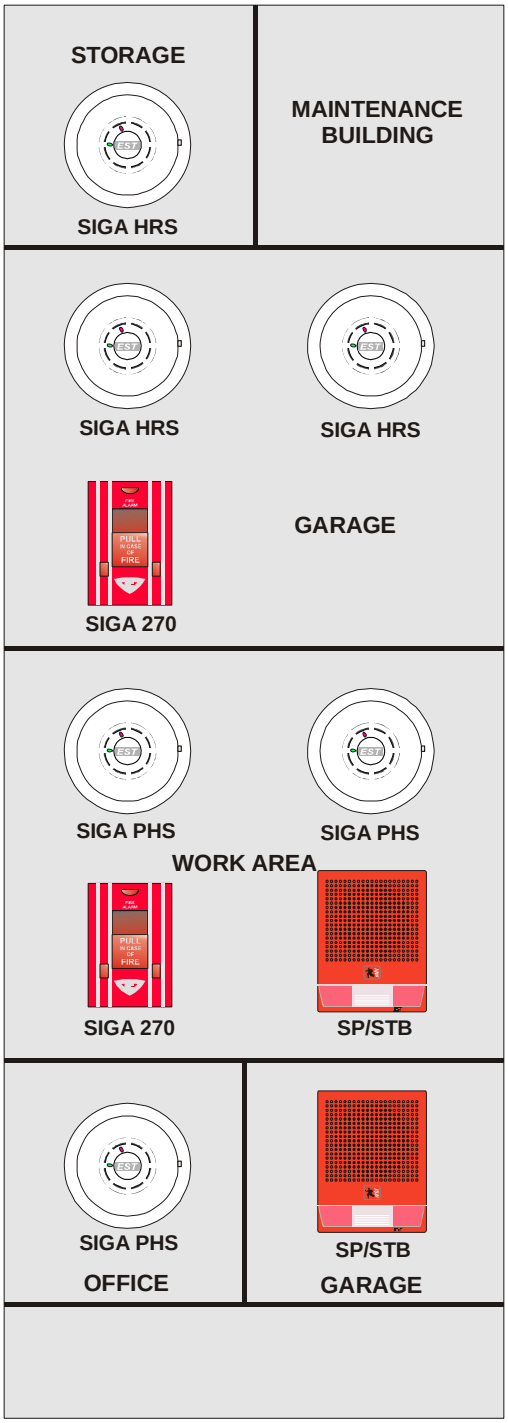
Appendix C
Maintenance building device board and schematics

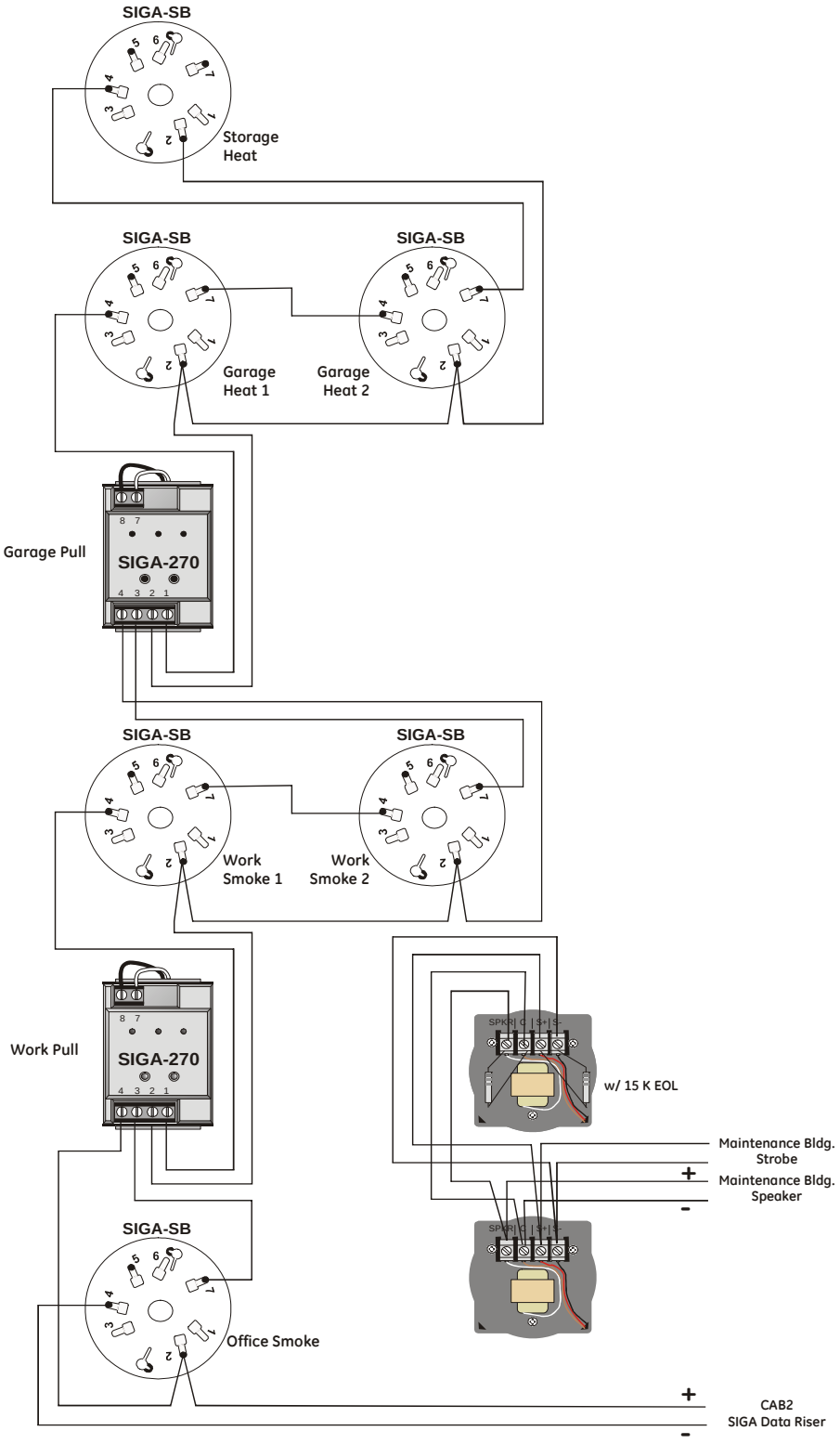
Introduction

This section contains the MacNeill Plaza Maintenance Building drawings for the device board and schematics.

UTC Fire & Security
A United Technologies Company

Maintenance Building Device Board Layout



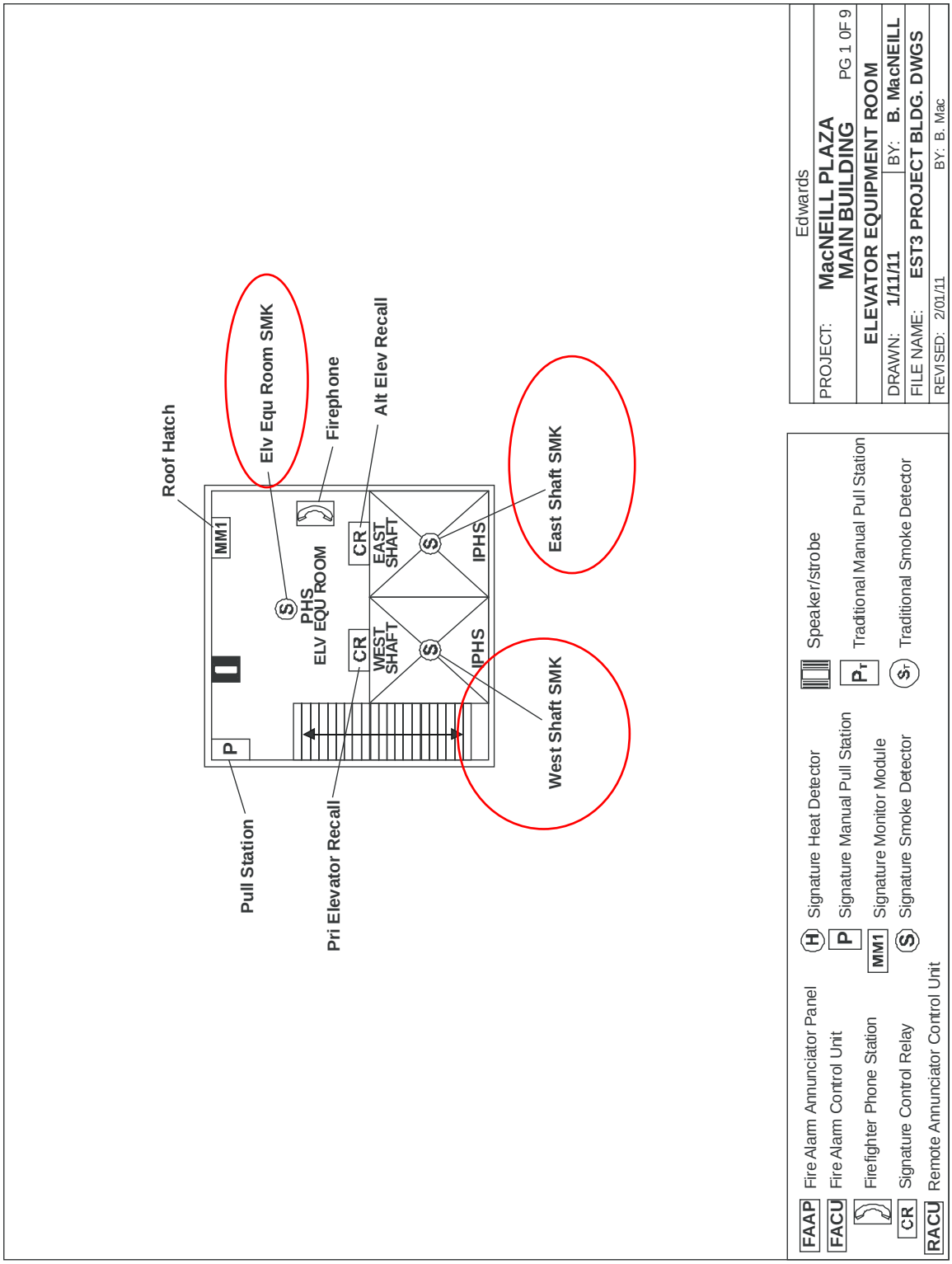


MacNeill Plaza Maintenance Building SIGA Data Riser, Speaker and Strobe Wiring.

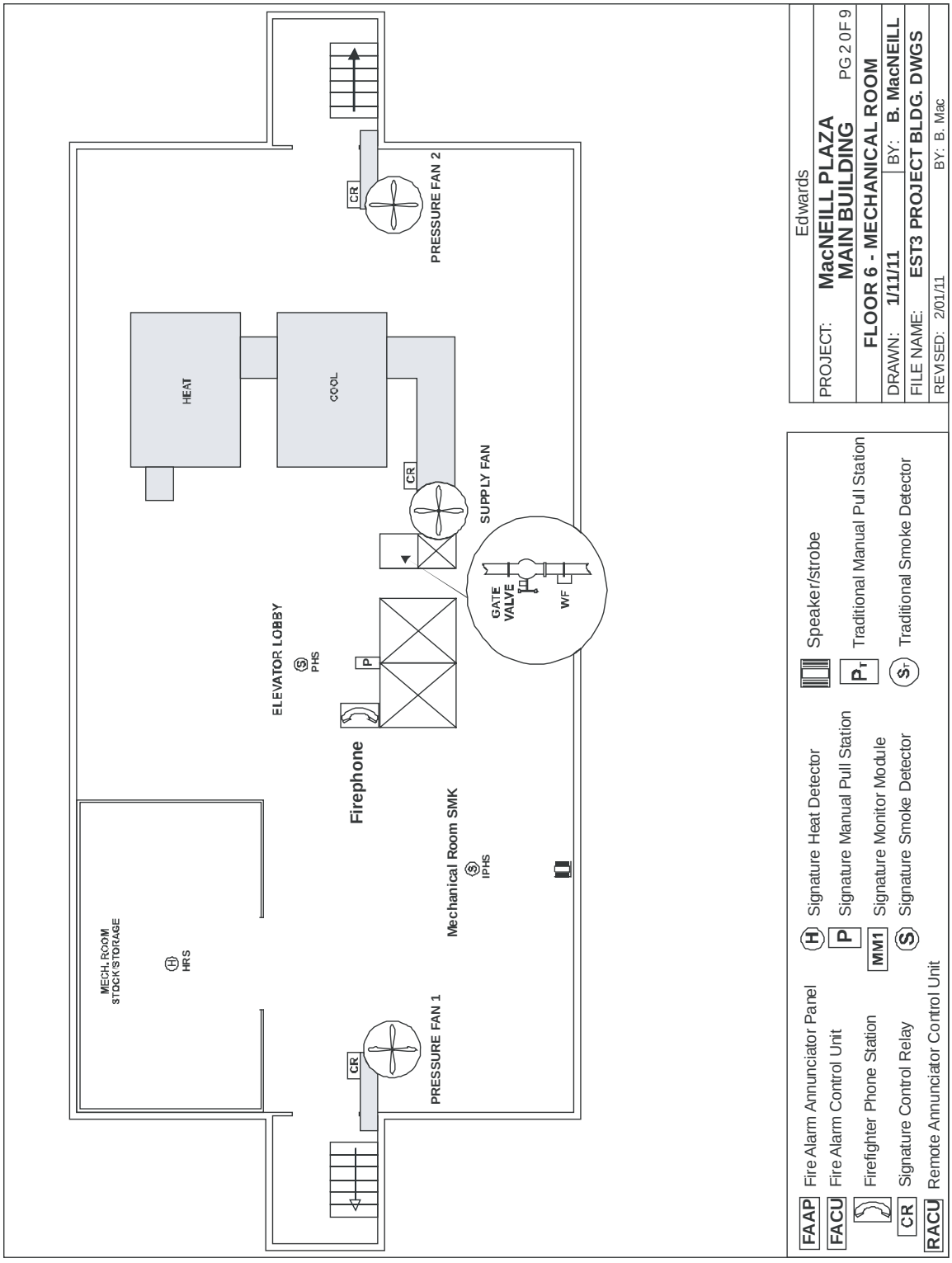
Appendix D
MacNeill Plaza building floor layouts

Introduction

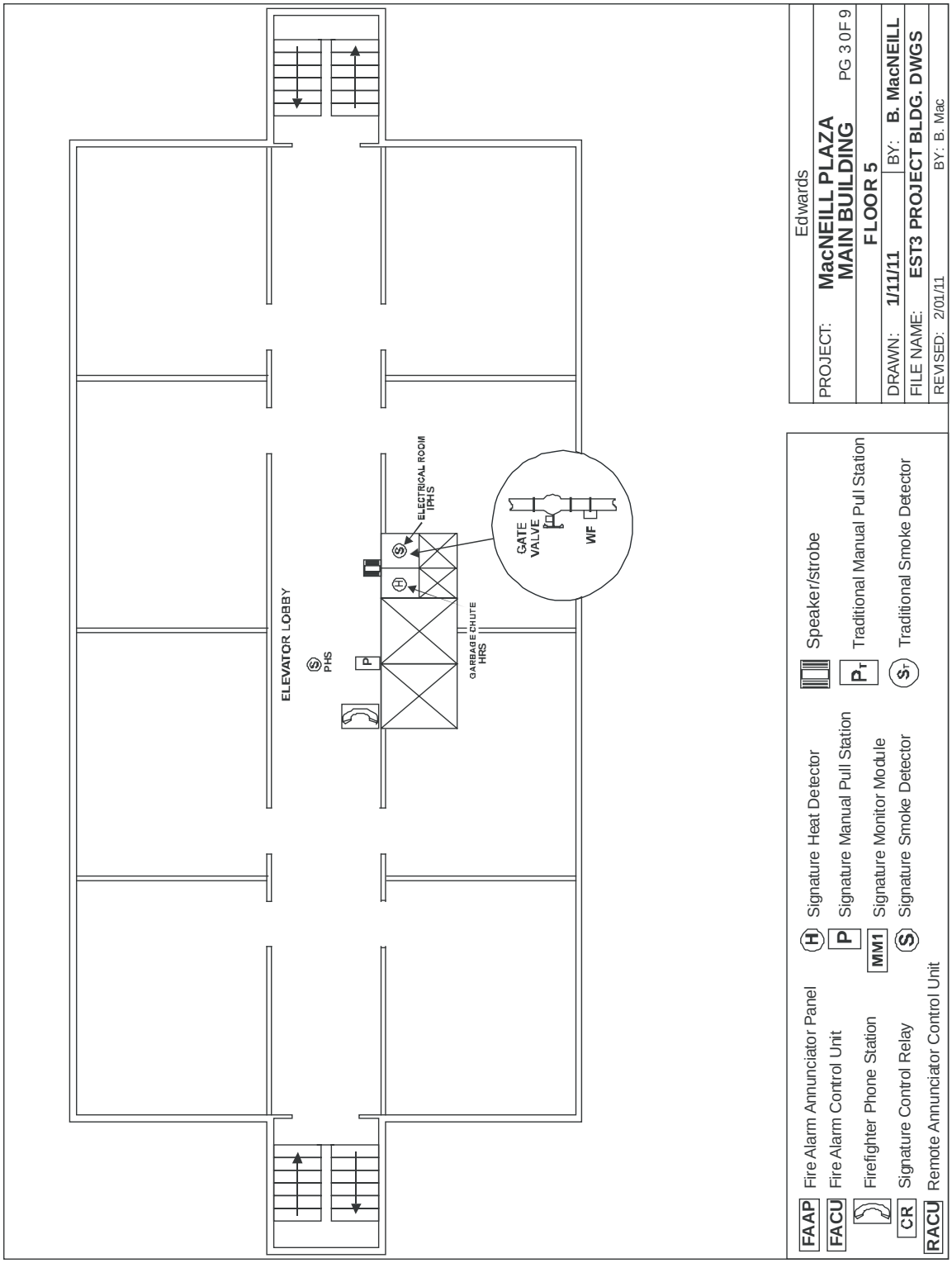
This section contains the MacNeill Plaza Main and Maintenance Building Floor layout drawings showing device locations.



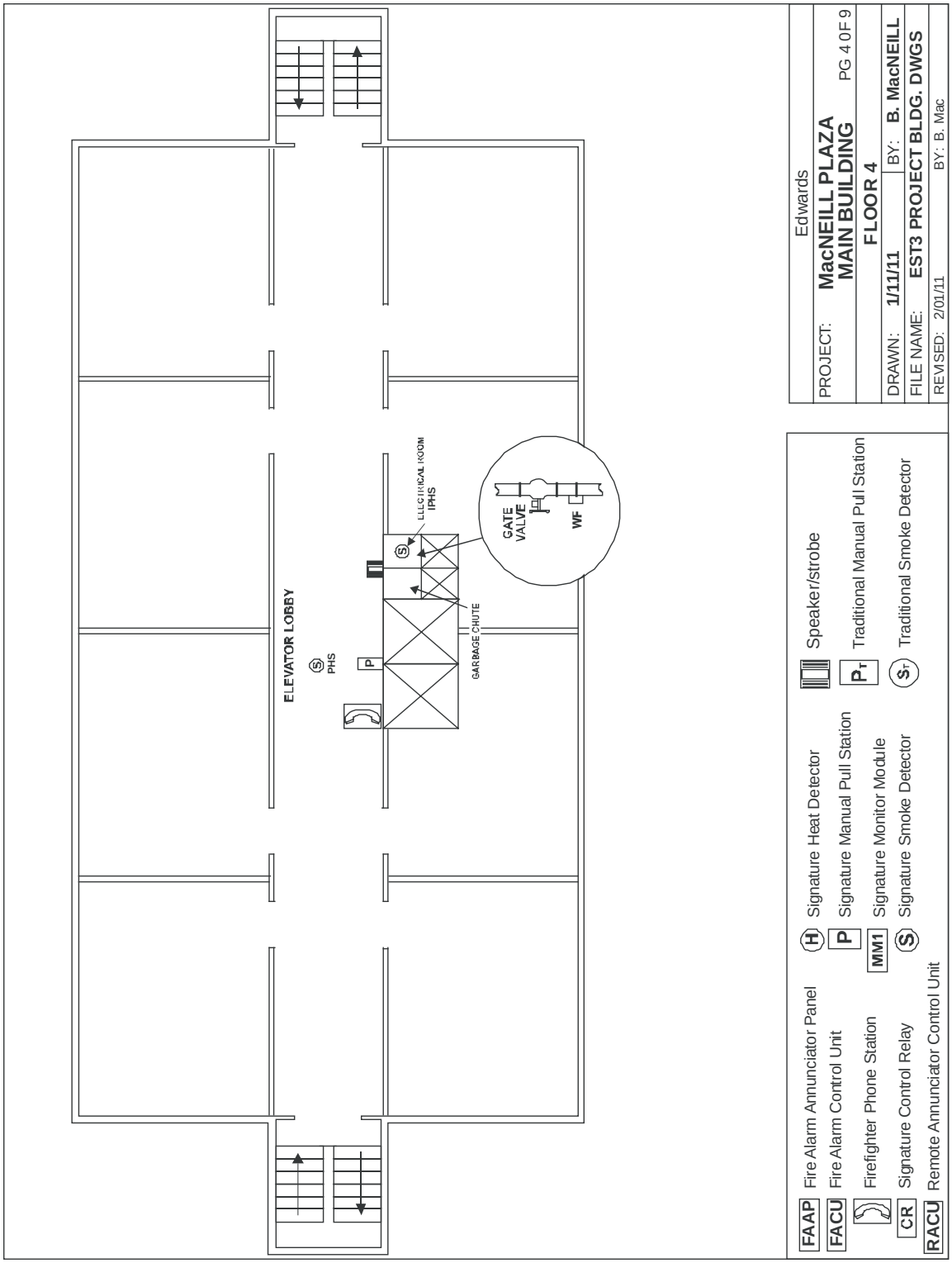
MacNeill Plaza Main Building Level 9 Elevator Equipment Room



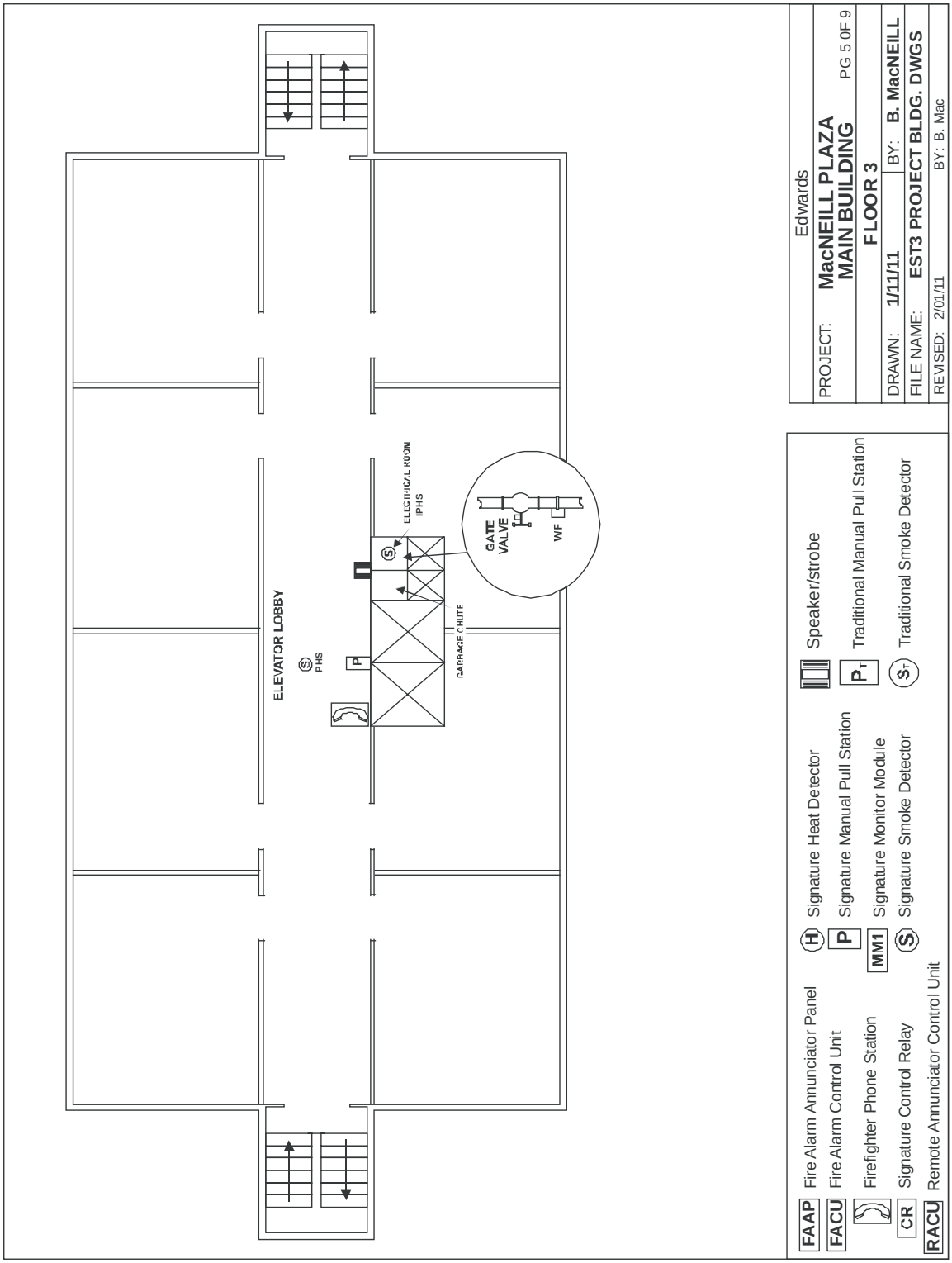
MacNeill Plaza Main Building Level 8 Mechanical Room



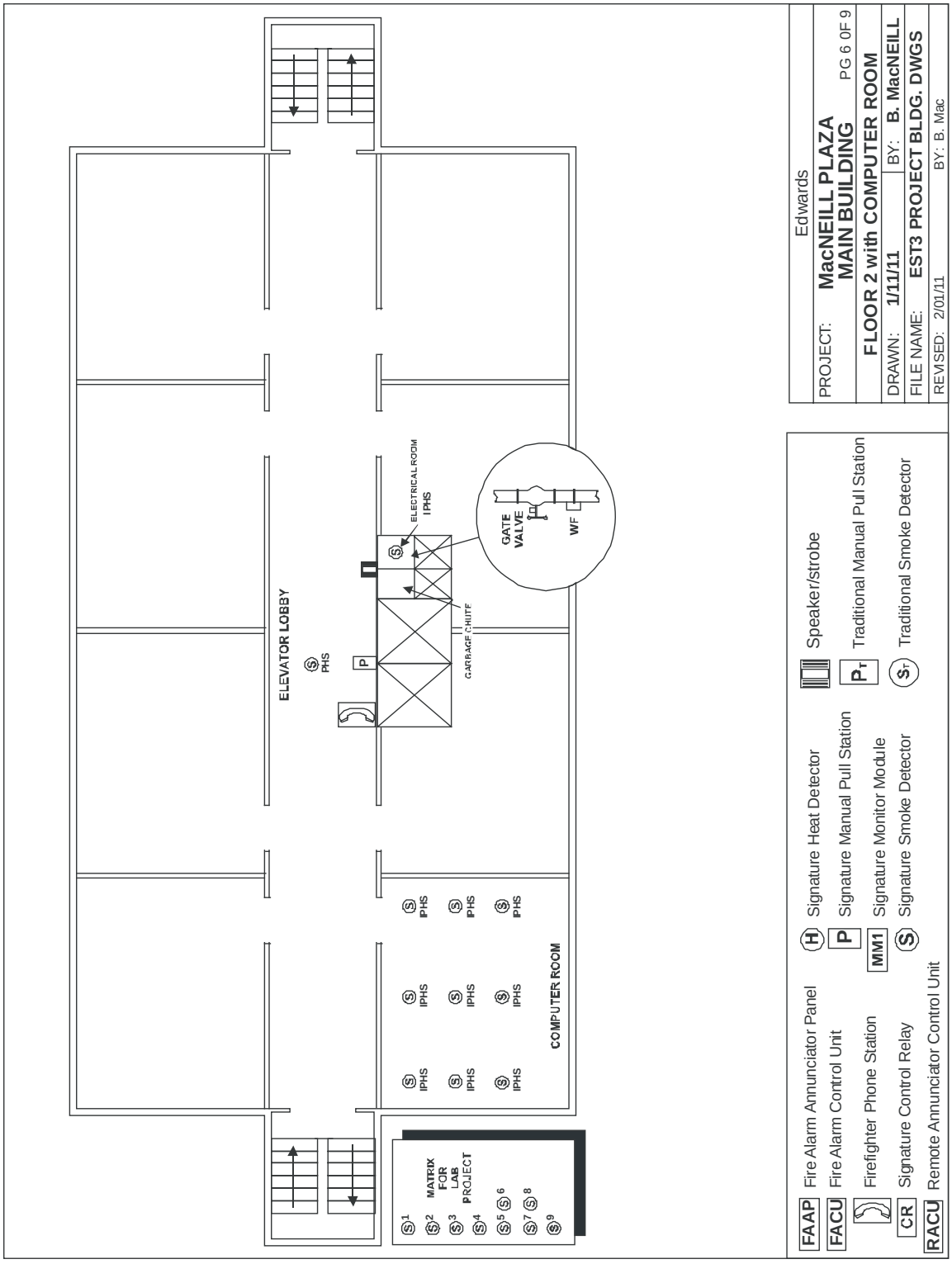
MacNeill Plaza Main Building Level 7 Floor 5



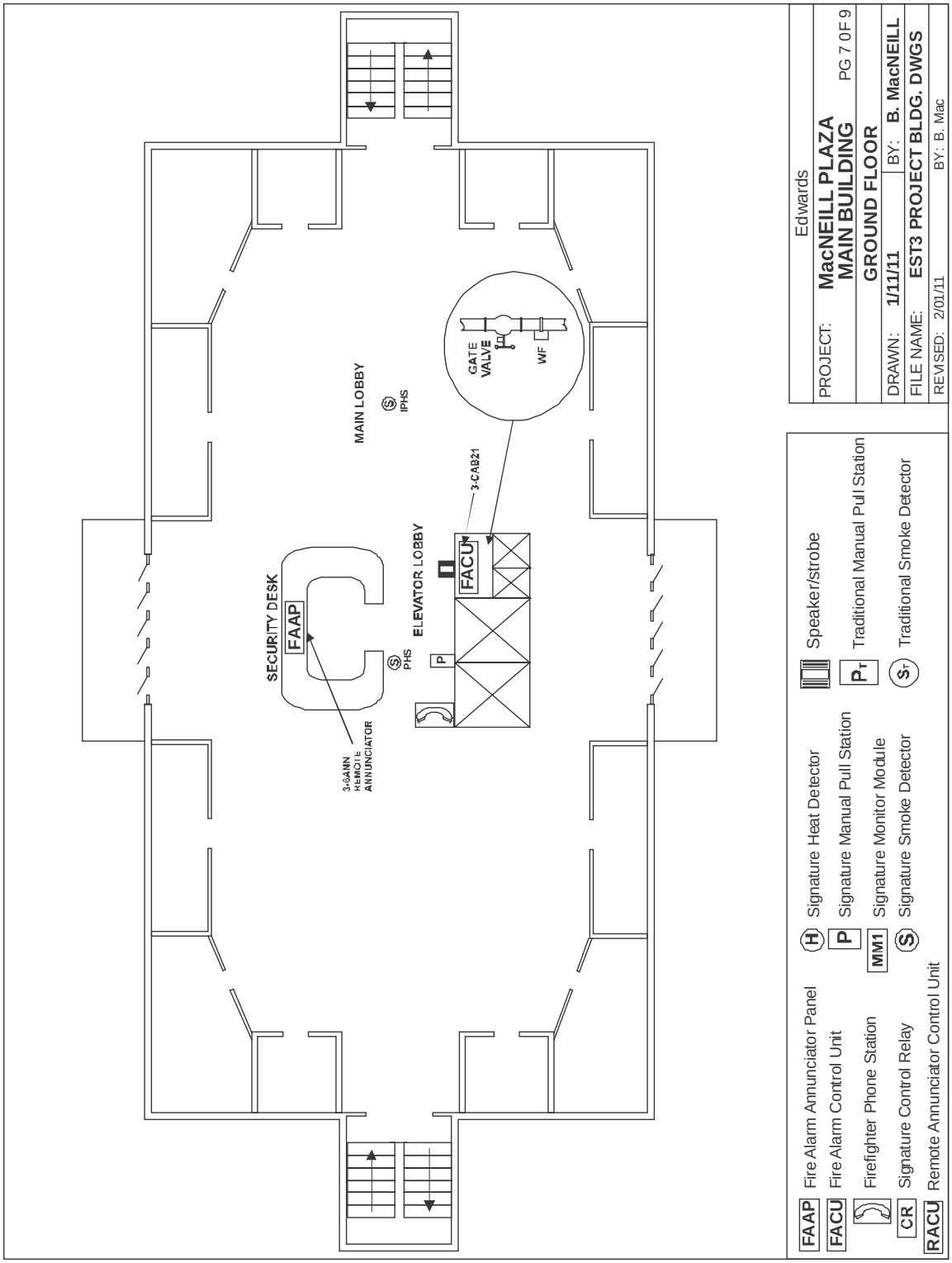
MacNeill Plaza Main Building Level 6 Floor 4



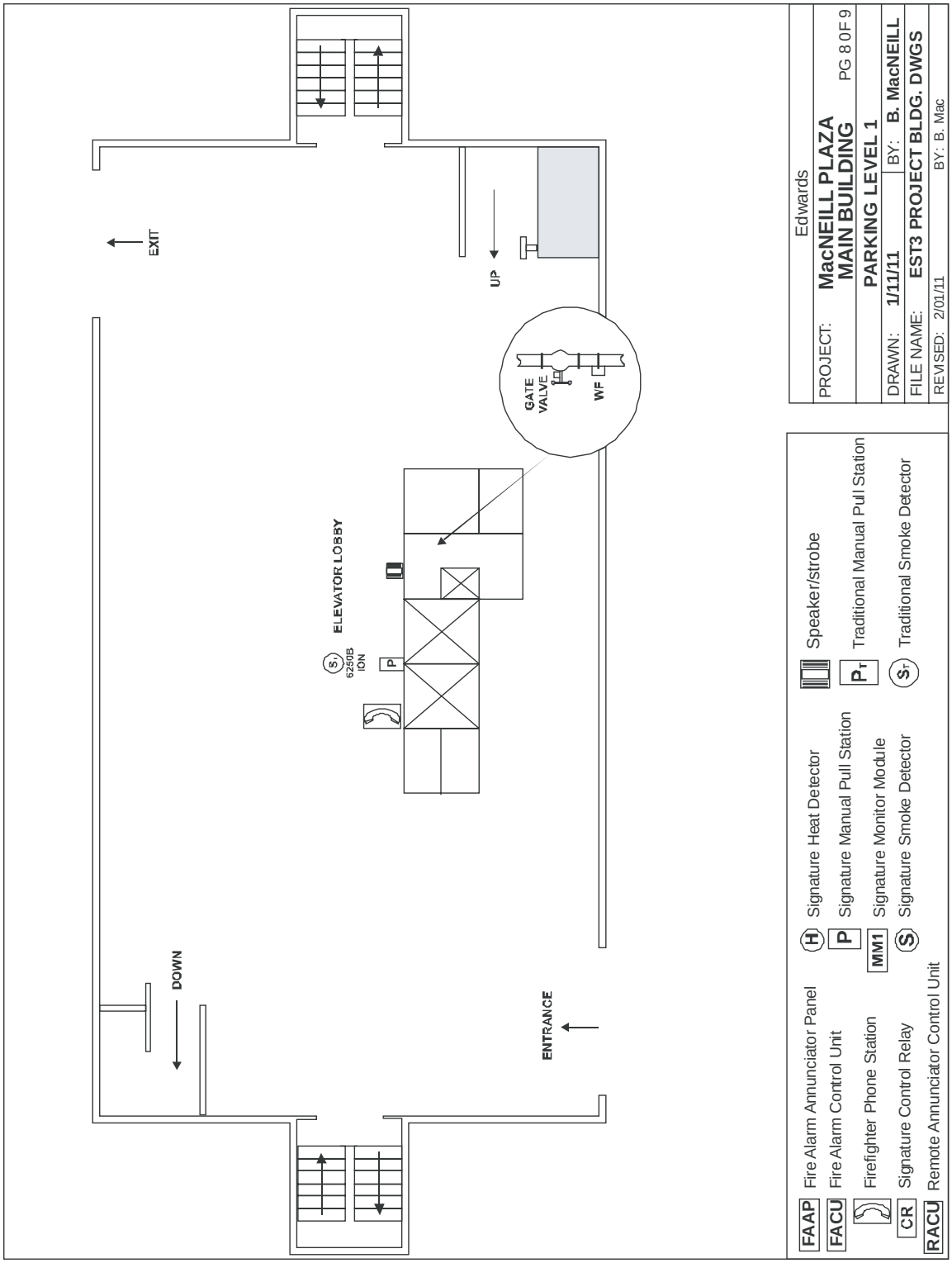
MacNeill Plaza Main Building Level 5 Floor 3



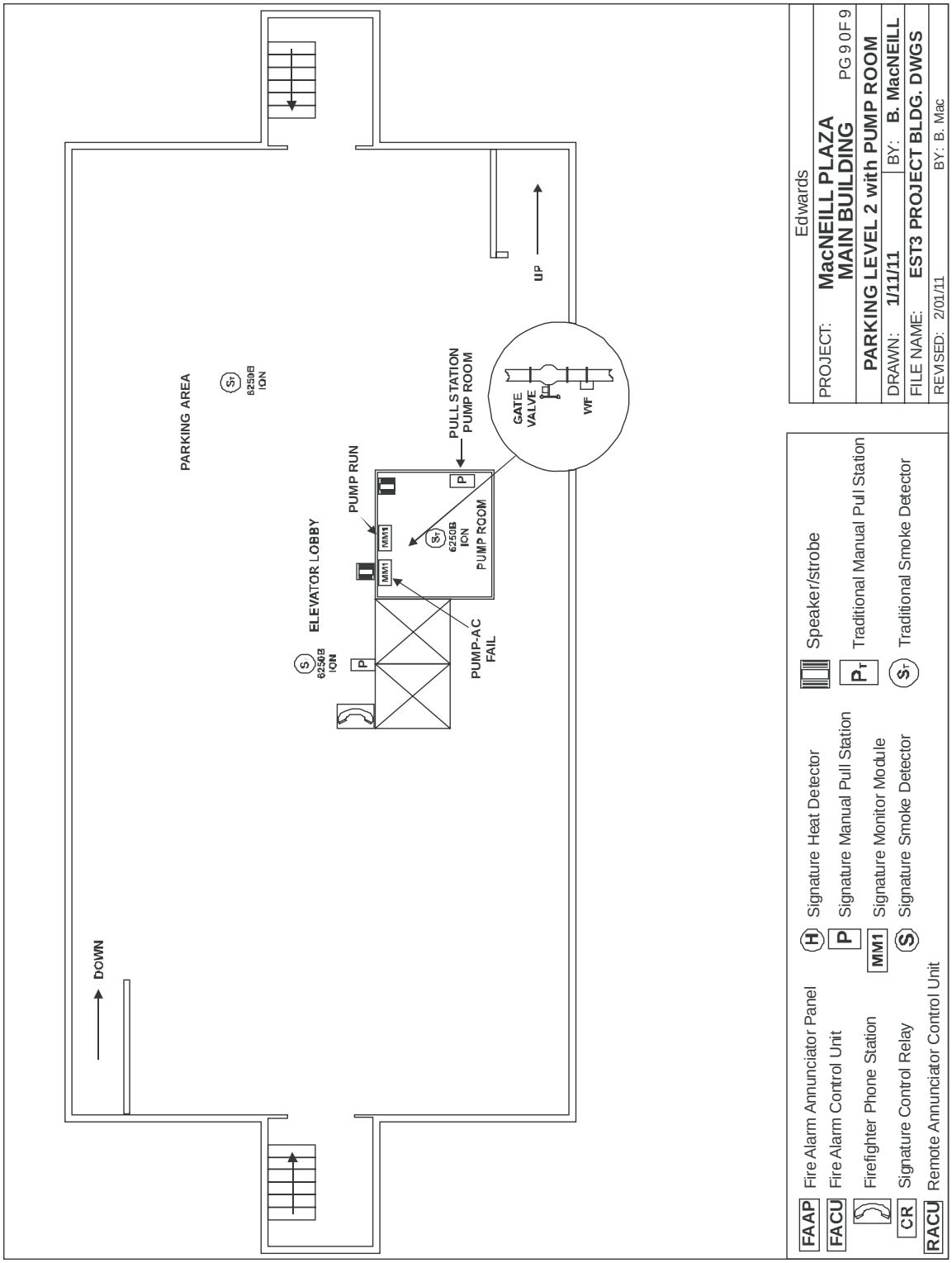
MacNeill Plaza Main Building Level 4 Floor 2 with Computer Room



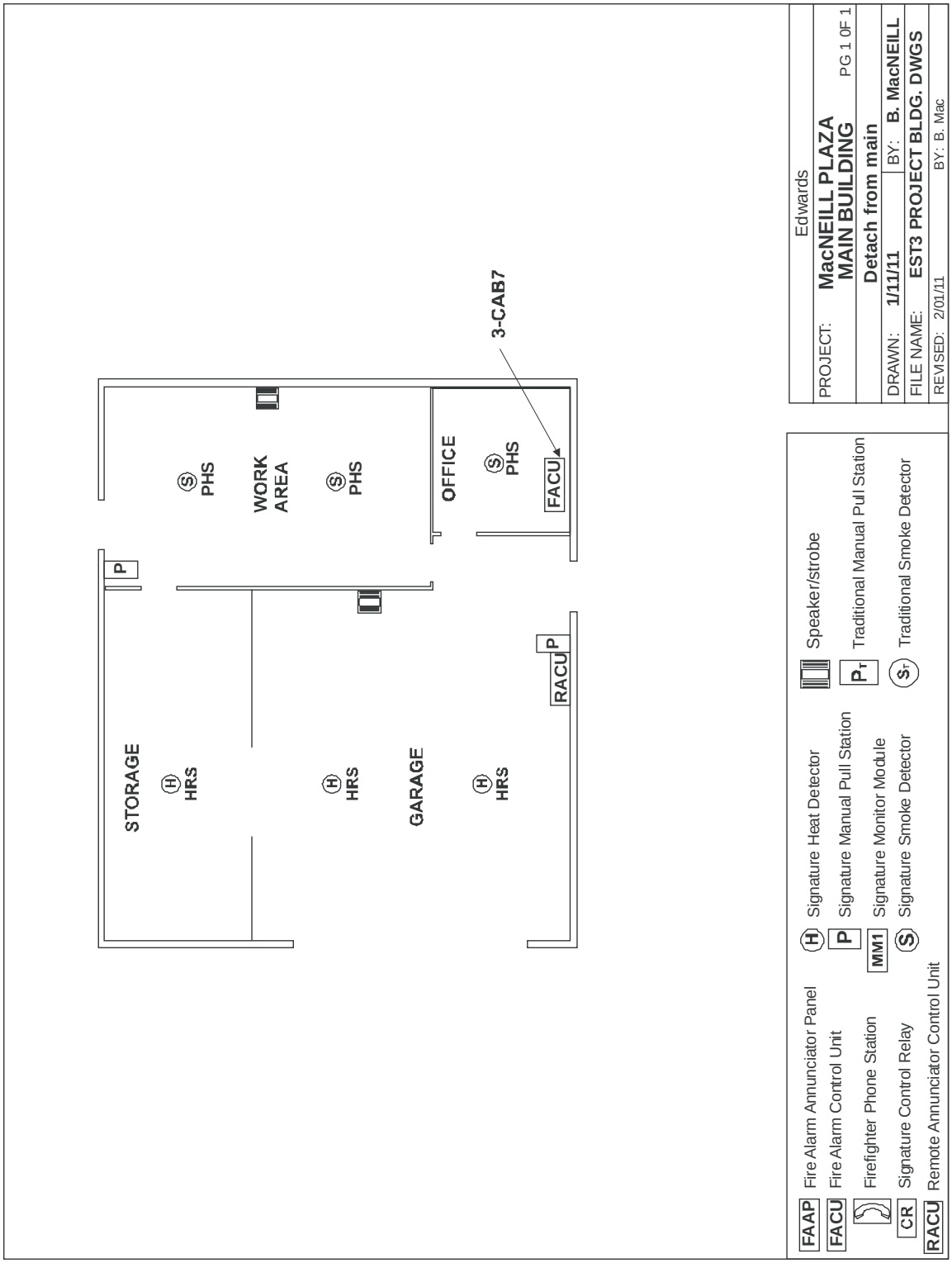
MacNeill Plaza Main Building Level 3 Ground Floor



MacNeill Plaza Main Building Level 2 Parking Level 1



MacNeill Plaza Main Building Level 1 Parking Level 2 with Pump Room



MacNeill Plaza Detached Maintenance Building

Appendix E
Lab panel operator layer display panel definitions

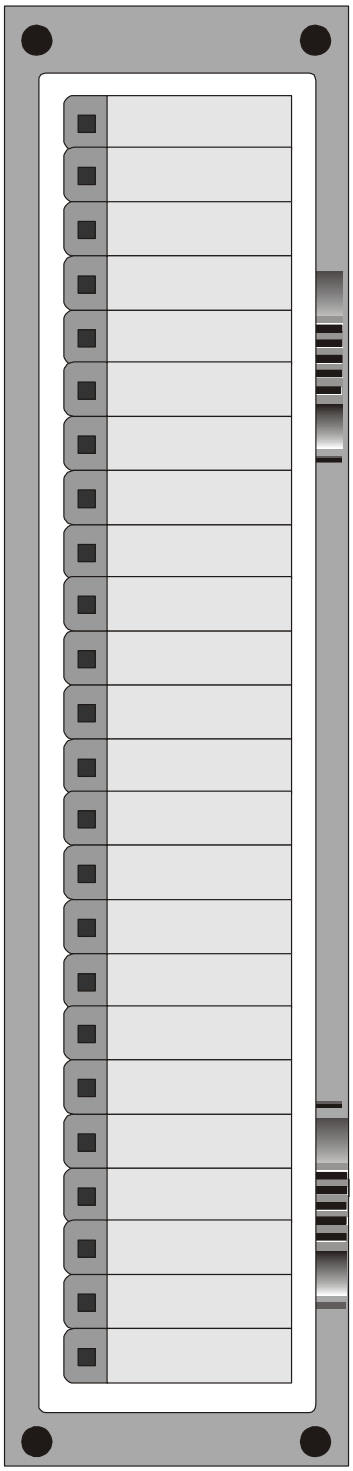
Introduction

This section contains definitions for the custom Control/LED panels.

UTC Fire & Security

A United Technologies Company

24 LED Panel
CAB #1, Slot 3



PHONE CALL IN **- Parking Level 2**

PHONE CALL IN **- Parking Level 1**

PHONE CALL IN **- Ground Floor**

PHONE CALL IN **- 2nd Floor**

PHONE CALL IN **- 3rd Floor**

PHONE CALL IN **- 4th Floor**

PHONE CALL IN **- 5th Floor**

PHONE CALL IN **- Mechanical Room**

PHONE CALL IN **- Elevator Equip. Room**

PHONE ACTIVATED **- Parking Level 2**

PHONE ACTIVATED **- Parking Level 1**

PHONE ACTIVATED **- Ground Floor**

PHONE ACTIVATED **- 2nd Floor**

PHONE ACTIVATED **- 3rd Floor**

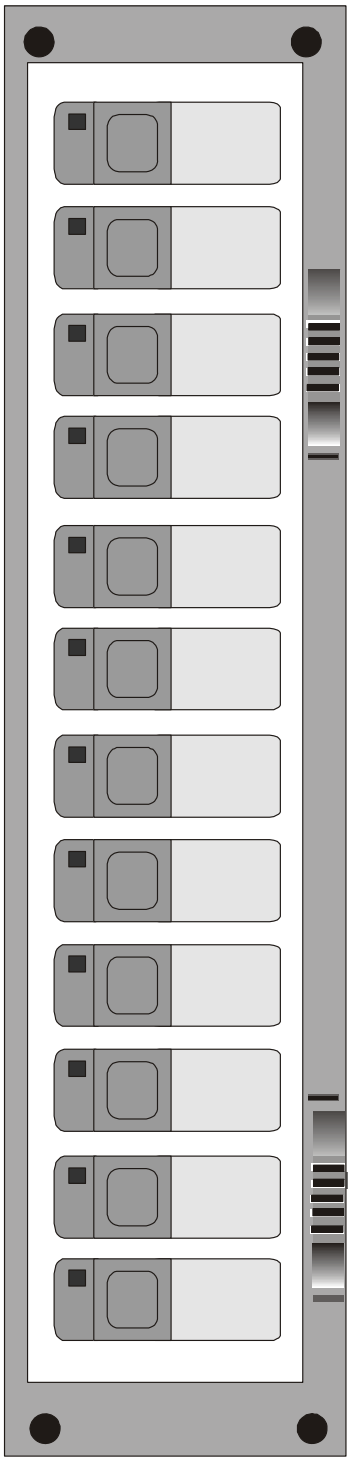
PHONE ACTIVATED **- 4th Floor**

PHONE ACTIVATED **- 5th Floor**

PHONE ACTIVATED **- Mechanical Room**

PHONE ACTIVATED **- Elevator Equip. Room**

12SW/12LED Panel
CAB 1, Slot 4



Parking Level2/Pump Page

Parking Level1 Page

Ground Floor Page

2nd Floor Page

3rd Floor Page

4th Floor Page

5th Floor Page

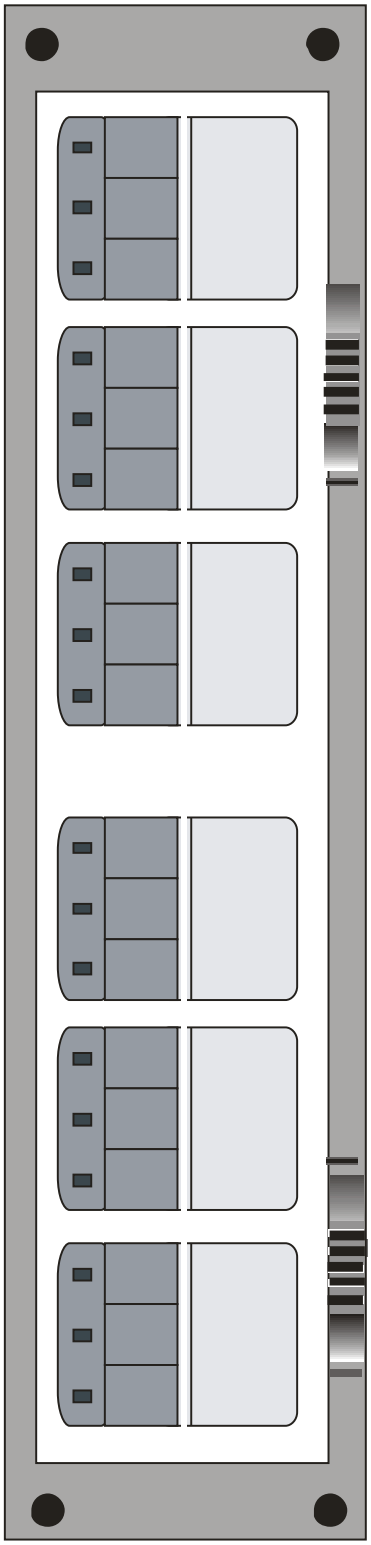
Mechanical Room Page

Elevator Equip Room Page

Maintenance Building Page

System Test Message

3SW x 3LED x 6 Panel
CAB 1, Slot 6



Supply Fan ON

Auto Mode (Program LED only)

Supply Fan OFF

Pressure Fan 1 ON

Auto Mode (Program LED only)

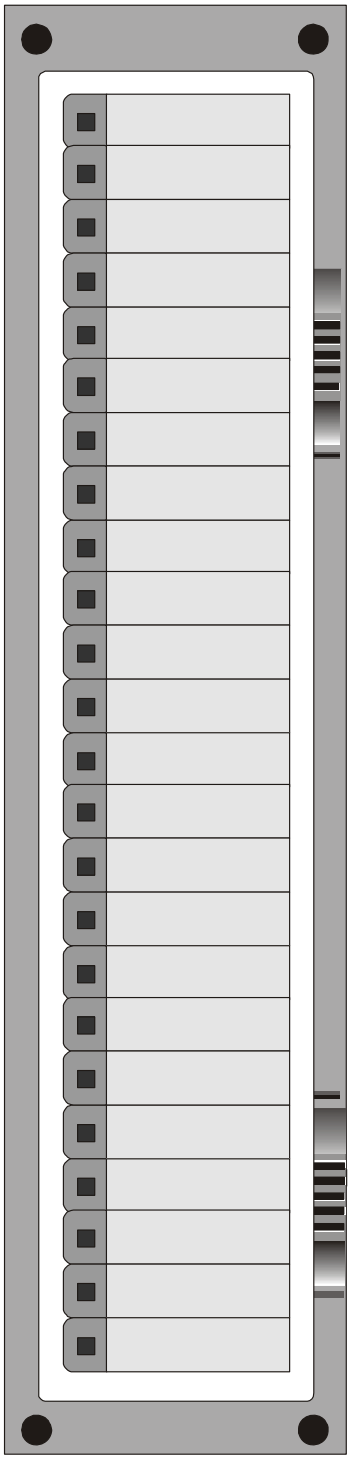
Pressure Fan 1 OFF

Pressure Fan 2 ON

Auto Mode (Program LED only)

Pressure Fan 2 OFF

24 LED Panel #1
CAB 3, Slot 4



Parking Level 2/Pump Alarm

Parking Level 1 Alarm

Ground Floor Alarm

2nd Floor Alarm

3rd Floor Alarm

4th Floor Alarm

5th Floor Alarm

Mechanical Room Alarm

Elevator Equipment Room Alarm

Elevator West Shaft Alarm

Elevator East Shaft Alarm

Maintenance Building Alarm

Supply Fan Activated

Pressure Fan #1 Activated

Pressure Fan #2 Activated

Primary Elev Recall Activated

Alternate Elev Recall Activated

Detector Maintenance Alert

Matrix Group Active

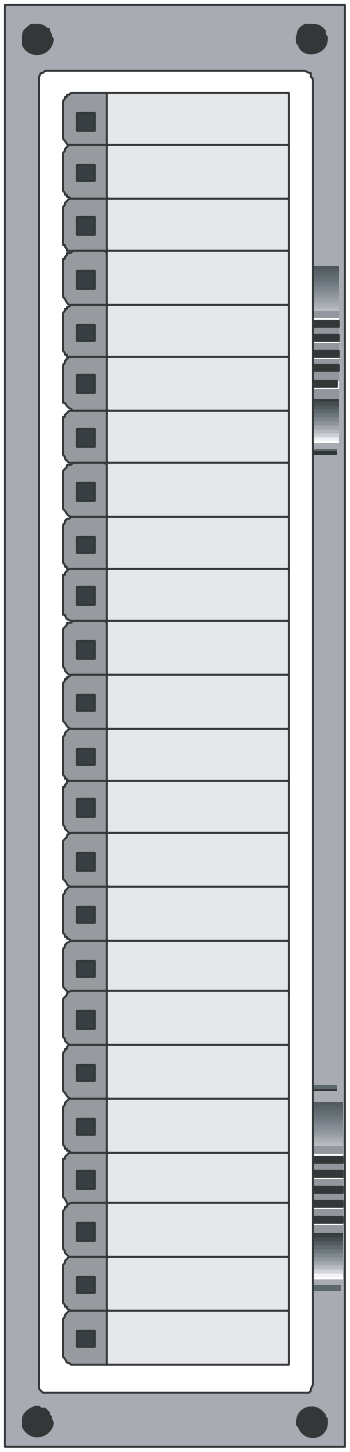
AND Group Active

Fire Pump - AC Fail

Roof Hatch Open

Device/Local Trouble

24 LED Panel #2
CAB 3, Slot 5



Cab #1 3-CPU3 Fail

Cab #1 3-PSMON Fail

Cab #1 3-SSDC Fail

Cab #1 Level9 3-ZA201 Amp Fail

Cab #1 3-LDSM Fail

Cab #1 Level8 3-ZA20 Amp Fail

Cab #1 Level7 3-ZA20 Amp Fail

Cab #1 Level6 3-ZA20 Amp Fail

Cab #1 Level5 3-ZA20 Amp Fail

Cab #1 Level4 3-ZA20 Amp Fail

Cab #1 Level3 3-ZA20 Amp Fail

Cab #1 Level2 3-ZA20 Amp Fail

Cab #1 Level1 3-ZA20 Amp Fail

Cab #1 3-ASU Fail

Cab #1 3-FTCU Fail

Cab #2 3-CPU3 Fail

Cab #2 3-PSMON Fail

Cab #2 3-SSDC1 Fail

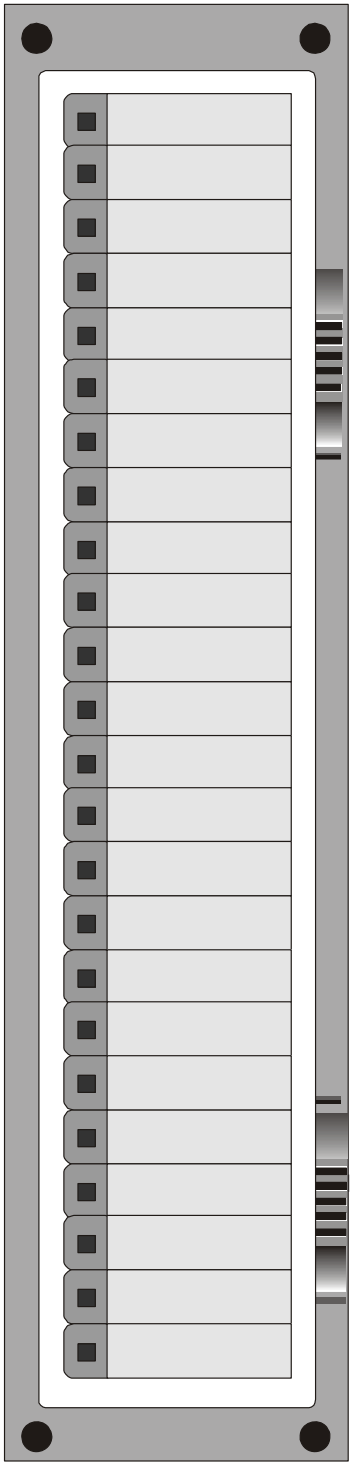
Cab #2 3-IDC8/4 Fail

Cab #2 Maint Bldg 3-ZA20 Amp Fail

Cab #2 Back-up 3-ZA20 Amp Fail

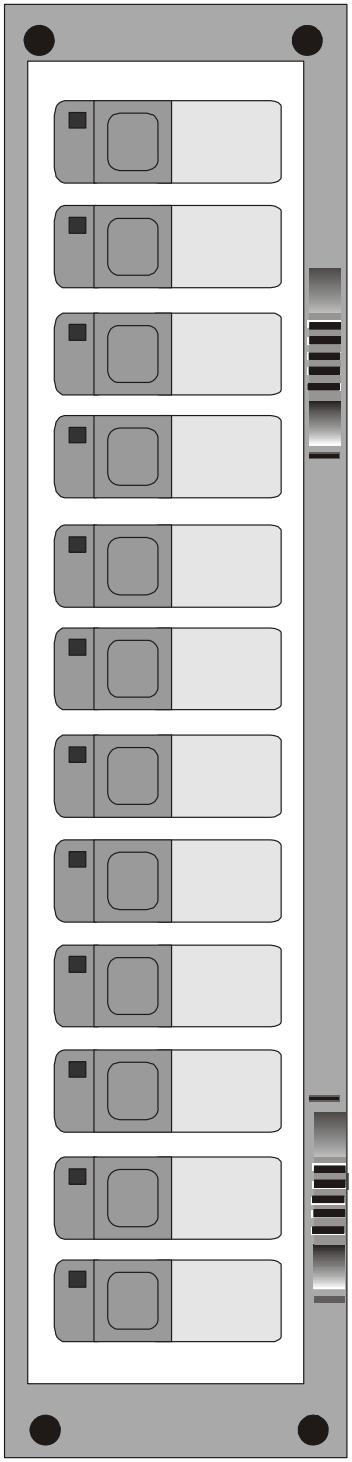
Cab #3 3-ANNCPU Fail

24 LED Panel #3
CAB 3, Slot 6



CURRENTLY UNUSED

12SW/12LED Panel
CAB 3, Slot7



CURRENTLY
UNUSED

Appendix F
Project Specifications

Introduction

This is the specification for your project. This project is worth 50% of your final grade. You must successfully complete 100% of the practical part of this course.

UTC Fire & Security
A United Technologies Company

MacNeill Plaza

Life Safety System Specification

Section A - Description

This specification includes information on the furnishing, installation and connection of the MacNeill Plaza networked, multiplex fire detection and control system. Its purpose is to form a complete, ready for operation, coordinated system. It shall include: alarm initiating devices, alarm indicating devices, central processing units, transponders, auxiliary control devices, pager and Firefighters' phone systems, status command center(s), power supplies and wiring.

Information provided in the following sections of this workbook and in the related manufacturer's documentation (e.g. installation sheets) is considered to be part of this project's requirements:

Section 4 – Main building device board and schematics
Section 5 – Maintenance building device board and schematics
Section 6 – Building drawings
Section 7 – Lab panel configurations
Section 8 – Display panel definitions

The networked system shall provide fire detection for the nine level Main Building, a six story building with two subterranean parking levels including an elevator equipment room and a single story Maintenance Building. The two buildings shall be connected together using Class A Network Data and Class B Audio Risers. Each local building system shall be non-coded and have electrically supervised Class B wired circuits. A standby evacuation tone must be supplied independently of the audio distribution. The audio amplification system must consist of one or more amplifiers for each evacuation-signaling zone, as necessary for the speaker load in each zone.

- Configured for US Market Local Mode operation per NFPA 72, where the system may be globally reset after alarm points have been cleared without individually acknowledging individual events

- Consist of three nodes and must respond to an alarm condition within 3 seconds of activation

- Network expandable and support up to 160K points including graphic annunciation

- Provide manually selectable alternate status and control, alternate sensitivity and bilingual operations

Node 1: Firefighters' Central Control Station

The Firefighters' Central Control Station node (CAB1 CPU) shall act as the master monitor and control panel for the entire system, including both buildings. This panel shall be installed in the Fire Equipment Room on the ground floor of the MacNeill Plaza main building and shall contain:

1. Central Processing Unit (CPU) with status/control display panel
2. Fire alarm and detection system status command panel
3. Status indicator and control of all alarm points, trouble conditions, service test and supervisory/monitor conditions on a system wide basis
4. Provisions for future alarm signal transmission to Firefighters' central alarm service via telephone lines
5. Manual page selections by zone for each level in the main building and the entire maintenance building
6. Manually activated system wide (both buildings) audio test message operation
7. Manual control of the HVAC system for smoke control
8. LED annunciation of main building Firefighters' phone call-in and connect status

Node 2: Maintenance Office Control Station

The Maintenance Office Control Station shall act as the master monitor panel for the status (state) of the entire system, including both buildings. This panel shall be capable of acknowledging any alarm, supervisory, trouble or monitor status message from the entire system. However, this panel shall not display service group or device test responses for ether building. This panel will have limited system control capability and shall be able to only command Reset, Alarm Silence, and Trouble/Panel Silence for the Maintenance building. This panel shall not be able to initiate a fire drill. This panel shall be installed in the Maintenance building office and shall contain:

1. Central Processing Unit (CPU) with status/control display panel
2. Fire alarm and detection system status command panel, as specified above
3. Status indicator and acknowledge of all alarm points, trouble conditions and supervisory/monitor conditions on a system wide basis
4. Remote Annunciator located in maintenance garage will display only messages (alarm, supervisory, trouble and monitor) displayed on Node 2 and mimic all command and control functions of Node 2
5. Maintenance panel shall be downloadable through EST3 network, serial connection or TCP/IP. Remote diagnostics may also be performed using ICP/IP.

Node 3: Security Desk Annunciator/Control Station

The Security Desk Annunciator/Control Station shall act as the master monitor panel for the status (state) of the entire system, including both buildings. This panel shall be capable of acknowledging any alarm, supervisory, trouble or monitor status message from the entire system. However, this panel shall not display service group or device test responses for either building. This panel will have limited control capability and shall be able to command Reset, Alarm Silence, and Trouble/Panel Silence for the MacNeill Plaza Main building only. . This panel shall not be able to initiate a fire drill. This panel shall be installed in the Security Desk in the lobby on the ground floor of the main building and shall contain:

1. Central Processing Unit (CPU) with status/control display panel
2. Fire alarm and detection system status command panel, as specified above

3. Status indicator and acknowledge of all alarm points, trouble conditions and supervisory/monitor conditions on a system wide basis
4. LED annunciation for Alarms by zones (each level and maintenance building)
5. LED annunciation to monitor fan and recall relay status, sensor maintenance, logic group activation, monitor points and local/system troubles
6. LED annunciation to display a local failure of each LRM module within the three system panels
7. LED annunciation for EVAC and ALERT notification by zone (each level and maintenance building)
8. Spare 24-LED and 12-LED/12-Switch panel for future applications

Section B - Fire Detection and Control System Functional Description

This section includes the functional and operational description of the MacNeill Plaza fire detection and control system.

MacNeill Plaza Main building:

1. Building type:
   - Six-story office building with 2 subterranean parking levels and an elevator equipment room (nine levels)

2. System configuration:
   - NFPA 72 with emergency generator and US Local Mode operation

3. Detectors:
   - Ground Floor through Elevator Equipment Room use intelligent Ion, Photo, and/or ROR Heat detectors
   - Smoke detectors shall not be configured for Alarm Verification or Alternate Sensitivity
   - Primary Sensitivity for smoke detectors shall be set to normal, 2.5% Obscuration per foot
   - Parking areas use traditional or conventional Ion and/or Photo detectors

4. LED Annunciation:
    - Zone of fire alarm, Firefighters' telephone status per zone, manual page per zone, HVAC auto/manual mode operation, system wide LRM module local trouble status per panel, trouble/monitor condition status, logic group status, detector maintenance status, fan activation confirmation and elevator recall activation confirmation

5. Data and Audio communications:
    - Network Data Riser - Class A
    - Audio Data Riser - Class B
    - Field Wiring - Class B

MacNeill Plaza, Maintenance Building:

1. Building type:
    - Detached 1-story maintenance shop and storage with integrated garage

2. System configuration:
    - NFPA 72 with emergency generator and US Local Mode operation

3. Detectors:
    - Use intelligent Photo with CO detection and/or ROR Heat detectors
    - Smoke detectors shall not be configured for Alarm Verification or Alternate Sensitivity
    - Primary Sensitivity for smoke detectors shall be set to normal, 2.5% Obscuration per foot

4. LED Annunciation:
    - None
    - All annunciation at Maintenance building is located at its CPU status/control display panel

5. Data and Audio communications:
    - Network Data Riser - Class A
    - Audio Data Riser - Class B
    - Field Wiring - Class B

Alarm Detection System:

1. Shall operate the audible and visible notification appliance circuits (NACs) per this specification
2. Shall operate the supply and pressure fans per paragraph **HVAC Supply and Pressure Fans**
3. Elevator lobby smoke detectors shall also recall the elevators:
   - A set of **Primary Recall Contacts** shall be provided by the Elevator Lobby Smokes on the Mechanical Room Floor, Floor 5, Floor 4, Floor 3, Floor 2, Parking Level 1 and Parking Level 2
   - The Elevator Lobby Smoke on the Ground Floor shall provide a set of **Alternate Recall Contacts**

Matrix Group:

1. The nine 2nd Floor (Level 4) Computer Room Smoke Detectors shall be configured and programmed as a Matrix Group with an Activation Number of 4 and a Radius of 1. When this Matrix Group is activated, audible and visible indications shall occur per this specification's main building requirements.
2. The Matrix Group detectors shall not individually initiate an audible and visible NAC alarm response. Individual detector activations shall still report to the control panel and shall activate the alarm response of the main building fans.
3. Matrix group activation shall be annunciated (flash fast) on the appropriate Node 3, Security Panel LED

Service Groups:

1. A Service Group shall be configured and programmed for each of the 3rd (Level 5), 4th (Level 6) and 5th (Level 7) Floors of the main building. Each floor's Service Group shall contain that floor's two Smoke Detectors and Manual Pull Station for Testing. The corresponding floor's Strobe shall be programmed to activate for a period of 10 seconds and then reset, when a smoke detector or pull is tested during the service group TEST mode.
2. When in the TEST mode, the service group's devices shall not activate the normal NAC alarm sequence. If the TEST mode is canceled when a tested device is still active, the system shall not false alarm but shall display a service device trouble message.

3. If the TEST mode is left unattended without testing activity, it shall automatically reset after a period of 30 minutes

AND Group:

1. The Elevator Equipment Room Smoke Detectors shall be configured and programmed as an AND Group. When the AND Group is activated (2 detectors in Alarm), audible and visible indications shall occur per this specification's Main building requirements.
2. The AND Group detectors will not individually initiate an audible and visible NAC alarm response. Individual detector activations shall still report to the control panel and shall activate the alarm response of the Main building fans.
3. AND group activation shall be annunciated (flash fast) on the appropriate Node 3, Security Panel LED

HVAC Supply and Pressurization Fans:

1. The Supply Fan shall be ON at system startup and Pressurization Fans shall be OFF
2. The MacNeill Plaza fire detection and control system shall shut down the supply fan and start the pressurization fans when:
    - Any Main building Smoke detector detects smoke*
    - Any Main building Sprinkler system waterflow device senses waterflow
    - Any Main building Heat detector is activated
    - Any Main building Manual Pull Station is operated

    ***NOTE:** If one detector of the AND or Matrix Group reports an Alarm, turn the fans ON or OFF as this specification directs

    NOTE: A five second delay is required between Pressurization Fan 1 and Fan 2 when activated to minimize the effect of current surge

3. The fire detection and control system shall also shut down the supply fan and start the pressurization fans when the time control function is activated per paragraph **Time Control**
4. The fire detection and control system shall provide manual control of individual fans per paragraph **Manual Control**

Main Building Audible Notification:

1. A recorded default EVACUATION voice message shall sound on the floor of incident, floor above, and floor below basis when any:
    - Main building smoke detector or group goes into alarm (activates)
    - Main building sprinkler system waterflow device senses waterflow
    - Main building heat detector is activated
    - Main building manual Pull Station is operated
2. A recorded default ALERT voice message shall sound to all non-EVAC areas in the Main building when an alarm condition occurs
3. The audio system shall sound an All-call page, All-call minus page, page to EVAC or page to Alert through manual control at the ASU/FT pager control panel. A pre-announcement message shall be broadcast prior to any page and page **will not** be inhibited.
4. Each speaker circuit shall be capable of individual paging from the Node 1, Firefighters' Central Station without affecting the zone paging for other signal areas per paragraph **Manual Control**
5. A custom audio system-wide test message shall be broadcast to all Main building zones through manual control per paragraph **Manual Control**

Maintenance Building Audible Notification:

1. A recorded custom EVACUATION voice message shall sound within the Maintenance building only when any:
    - Maintenance building smoke detector detects smoke
    - Maintenance building heat detector is activated
    - Maintenance building manual Pull Station is operated
2. A recorded ALERT voice message shall **NOT BE** broadcast to either building
3. The audio system shall sound an All-call page, All-call minus page, or page to EVAC through manual control at the ASU/FT pager control panel. A pre-announcement message will be broadcast prior to any page and page **will not** be inhibited.
4. This speaker circuit shall be capable of individual paging from the Firefighters' Central Station without affecting the

zone paging for other signal areas per paragraph **Manual Control**

5. A custom audio system-wide test message shall be broadcast to the Maintenance building through manual control per paragraph **Manual Control**
6. The Node 2 Maintenance Control panel will provide backup amplifier operation for the maintenance building audio system

Visible Circuits:

1. Shall follow the operation of the audible EVACUATION circuits for both buildings
2. Shall be configured as silenceable

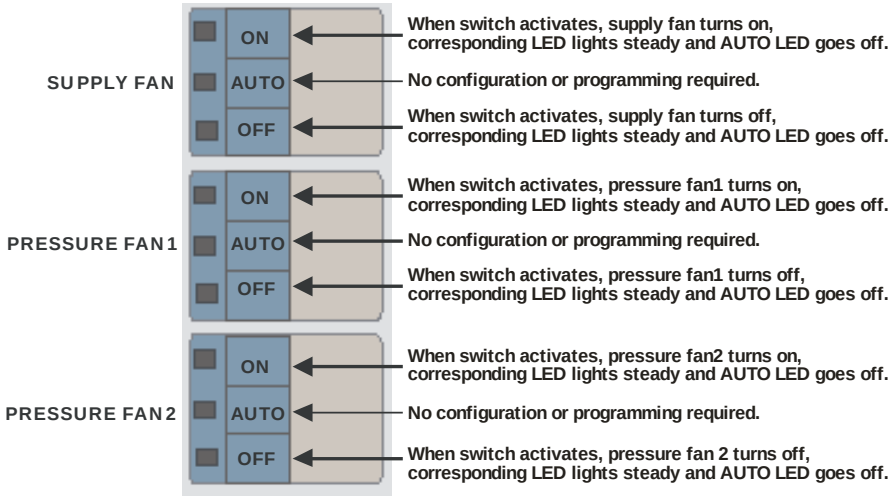
Alarm Silence:

1. Both audible and visible notification responses shall be silenced with the Alarm Silence control as follows:
    - Node 1, Firefighters' Central Station - both buildings
    - Node 3, Security Panel - Main building only
    - Node 2, Maintenance Panel - Maintenance building only

2. Alarm Silence and System Reset will not be inhibited (delayed for 1, 2 or 3 minutes)

Manual Control:

1. Shall provide page select for each zoned audible circuit. Each switch is configured for toggle operation and its corresponding LED lights steady when the switch is activated. Switch activation shall be annunciated as monitor events on the Node display panels.
2. Shall provide manual on-off control of fans:
    - The nine switch segments are configured for interlocked operation
    - The corresponding ON and OFF segment LEDs light steady and the corresponding auto segment LED turns off when the corresponding switch is activated



- The ON and OFF switch activation shall be annunciated as monitor events on the Node display panels
- The auto switch activation shall not annunciate on the Node display panels

3. Shall provide a system-wide (both buildings) broadcast custom audio test message. The switch is configured for toggle operation and its corresponding LED lights steady when the switch is activated. Switch activation shall be annunciated as monitor events on the Node display panels.

Time Control:

1. A Time Control shall be configured and programmed to turn OFF the supply fan and turn ON the two pressurization fans on Monday through Friday at 9:00 PM (2100 hours) for 1 minute. A 5 second delay is required between Pressurization Fan 1 and Fan 2 when activated to minimize the effect of current surge.

   NOTE: See HVAC Supply and Pressurization Fans

Annunciation:

1. Shall provide visible indication (flash slow) of a system wide and/or local trouble event
2. Shall provide visible indication (flash slow) of Roof Hatch and Pump AC Fail Monitor point events
3. Shall provide visible indication (flash fast) of the Matrix and the AND Group activation events
4. Shall provide visible indication (flash fast) of individual Main building floor alarm, Maintenance building alarm and each elevator shaft alarm
   Note: Even if one of the AND or Matrix detectors reports a Alarm, light (flash fast) the appropriate floor's LED
5. Shall provide visible indication (flash slow) of confirmation of Fan activation and Primary and Alternate Elevator Recall activation. Visible indication will occur by confirming relay activation.
6. Shall provide visible indication (flash slow) if any smoke detector reports a dirty head maintenance alert condition
7. The auto segment LEDs of the 3SW/3LEDx6 Panel will light at system startup, indicating automatic operation, where system controls fans. These LEDs will go OFF

when Firefighters' panel's ON/OFF/AUTO switch segments are used to manually control fan operation.

8. Provide visible indication (flash fast) of Firefighters' phone call-in and confirmed activated (connected) status
9. Provide visible indication (flash fast)of each panel's LRM module fail status (a local trouble event)
10. Each switch's corresponding LED should light (Steady) on switch activation

Notes:

1. Elevator Equipment room wall mounted speaker / strobe – Qty. 1, @ 1 watt, 105 mA @ 24VDC
2. Mechanical room (floor 6) wall mounted speaker / strobe – Qty. 1, @ 1 watt, 105 mA @ 24VDC
3. Floors 2 through 5 wall mounted speaker/strobes –

   Qty. 4, @ .5 watt, 105 mA @ 24VDC
4. Ground Floor wall mounted speaker/strobe –

   Qty. 1, @ .5 watt, 105 mA @ 24VDC
5. Parking Level 1 wall mounted speaker/strobe –

   Qty. 1, @ 2 watt, 105 mA @ 24VDC
6. Parking Level 2 wall mounted speaker/strobes –

   Qty. 2, @ 2 watt, 105 mA @ 24VDC
7. Detached Maintenance building wall mounted speaker/strobes –

   Qty. 2, @ .5 watt, 105 mA @ 24VDC
8. Floors 2 through 6 Main building speaker/strobe runs are approximately 200 feet per floor
9. Elevator Equipment room Main Building speaker/strobe run is approximately 25 feet
10. Floors 2 through 6 Main building detection loop is approximately 150 feet per floor
11. Elevator Equipment room Main building detection loop is approximately 30 feet
12. Ground Floor and Parking Level Main building speaker/strobe runs are approximately 500 feet per floor
13. Ground Floor and Parking Level Main building detection loop are approximately 300 feet per floor
14. Maintenance building speaker/strobe runs is approximately 75 feet

15. Maintenance building detection loop is approximately 200 feet

Main Building devices:
(See project floor drawings for facility locations)

Elevator Equipment room (Level 9):

- 3 Smoke detectors
- 2 Elevator control relays
- 1 Manual pull station
- 1 Wall mounted phone station with communications device
- 1 Roof Hatch monitor point
- 1 Speaker/Strobe

Floor 6 Mechanical room (Level 8):

- 2 Smoke detectors
- 1 Heat detector
- 1 Manual pull station
- 1 Waterflow device
- 1 Spare Gatevalve device
- 1 Supply fan control relay
- 2 Pressurization fan control relays
- 1 Wall mounted telephone station with communications device
- 1 Speaker/Strobe

Floor 5: (Level 7)

- 2 Smoke detectors
- 1 Heat detector
- 1 Manual pull station
- 1 Waterflow device
- 1 Spare Gatevalve device
- 1 Wall mounted telephone station with communications device
- 1 Speaker/Strobe

Floors 2 - 4 (Levels 4 - 6):

- 2 Smoke detectors per floor
- 1 Manual pull station per floor
- 1 Waterflow device per floor
- 1 Spare Gatevalve device per floor
- 1 Wall mounted telephone station/floor with communications devices
- 1 Speaker/Strobe per floor

Floor 2 Computer room (Level 4):

- 9 smoke detectors

Ground Floor (Level 3):

- 2 Smoke detectors
- 1 Manual pull station
- 1 Waterflow device
- 1 Spare Gatevalve device
- 1 Wall mounted telephone station with communications device
- 1 Speaker/ Strobe

Parking Level 1 (Level 2):

- 1 Traditional Smoke detector
- 1 Manual pull station
- 1 Waterflow device
- 1 Spare Gatevalve device
- 1 Wall mounted telephone station with communications device
- 1 Speaker/Strobe

Parking Level 2, Parking area (Level 1):

- 2 Traditional Smoke detectors
- 1 Manual pull station
- 1 Waterflow device
- 1 Spare Gatevalve device
- 1 Wall mounted telephone station with communications device
- 1 Speaker/Strobe

Parking Level 2, Pump room (Level 1):

- 1 Fire pump-run monitor point
- 1 Fire pump-no power monitor point
- 1 Traditional Smoke detector
- 1 Manual pull station
- 1 Speaker/Strobe

Maintenance Building Devices:
(See project floor drawings for facility locations)

- 3 Smoke/CO detectors
- 3 Heat detectors
- 2 Manual pull stations
- 2 Speaker/Strobes

Appendix G
Answer Sheets

Introduction

This section contains copies of Practical Checklist and Answer Sheets to be used during this course.

UTC Fire & Security

A United Technologies Company

Front Panel Operations/ Logical Addressing Exercise

1. Using the 3-CAB 21 3-LCD front panel Main Menu functions:
   - Light the 10th LED **STEADY** on the 3-6/3S1 LED/Switch Panel
   - Turn this LED off from the front panel

2. Using the 3-CAB21 3-LCD Main Menu, list the firmware version resident in the 3-LDSM module _____

3. Perform the following practice sequence on the EST3 panel device board:

   Note: Do NOT use the detectors in the Elevator Equipment Room or the Computer room for this exercise

   a. Remove a smoke detector. What happened? _____
   b. List this detector's logical address _____
   c. Reinstall this detector and allow the panel to return to normal
   d. Use the front panel to disable this detector. What happened?

      _____
   e. Smoke this disabled detector. What happened? _____
   f. While this detector is still in alarm, enable it. What happened?

      _____
   g. Acknowledge this alarm and reset at the front panel. Allow the panel to return to normal.

4. Perform the following practice sequence on the EST3X panel device board:

    a. Remove a smoke detector. What happened? _____
    b. List this detector's logical address _____
    c. Reinstall this detector and allow the panel to return to normal
    d. Use the front panel to disable this detector. What happened?

        _____
    e. Smoke this disabled detector. What happened? _____
    f. While this detector is still in alarm, enable it. What happened?

        _____
    g. Acknowledge this alarm and reset at the front panel. Allow the panel to return to normal.

5. Perform the following practice sequence:

    a. Activate any one of the pull stations. What happened?

        _____
    b. While the pull station is still active, silence the NACs. What happened? _____
    c. While the pull station is still active, acknowledge the alarm and reset the panel. What happened? _____
    d. Reset the pull station, acknowledge this alarm and reset using the front panel. Allow the panel to return to normal.

6. Perform the following practice sequence:

    a. Place the panel in test mode, select the available service group and start the test. What happened? _____
    b. Test the 3rd floor smoke detectors. Did any points go into alarm? Yes ☐ No ☐ What happened? _____
    c. Cancel this test. What happened? _____

d. Place the panel in test mode again and test one of the 3rd floor smokes
e. Activate the 2nd floor pull station while the panel is in test mode. What happened? _____
f. Restore the pull station, acknowledge the alarm and reset the panel. What happened? _____

7. Will the panel let you cancel the test if a tested device remains off normal? Yes ☐ No ☐

8. List the function of 01 02 0142? _____

9. List the active points:

_____ _____ _____

_____ _____ _____

_____ _____ _____

_____ _____ _____

Grade_____

EST3 Series Practical Checklist

Name: _____Date Completed_____

Task		**pts**.
1. Customize the 3-SDU behavior and configuration aspects	☐	**1**
2. Create a new project (1st letter of project name is password)	☐	**1**
3. Configure Cabinets, Cab #1, Cab #2 & Cab #3	☐	**1**
• Add R Series Annunciator		
4. Configure Network Routing, Cab #1, Cab #2 & Cab #3	☐	**1**
5. Configure Cab #1 LRMs and LED Switch Panels	☐	**1**
6. Configure 3-ASU channels and messages	☐	**1**
7. Configure Cab #2 LRMs and LED Switch Panels	☐	**1**
8. Configure Cab #3 LRMs and LED Switch Panels	☐	**1**
9. Configure Time Controls (fan control)	☐	**1**
10. Configure non-SIGA Objects (LEDs, Switches, Strobes, 3-IDC8/4 Zones for all cabinets)	☐	**2**
11. Enter and Configure SIGA devices for both SIGA loops	☐	**1**
12. Label SIGA devices for both SIGA loops	☐	**1**
13. Scan the serial numbers for the 3-CAB21 SIGA Loop devices	☐	**1**
14. Reconcile the 3-CAB21's SIGA Loop (scanning method)	☐	**2**

15. SIGA convert and download reconciled database to 3-CAB21 SIGA loop monitor using Status and re-map to get a white map ☐ **2**

16. Reconcile the 3-CAB7's SIGA Loop (mapping method) ☐ **2**

17. SIGA convert and download reconciled database to 3-CAB7 SIGA loop monitor using Status and re-map to get a white map ☐ **2**

18. Downloading to all three 3-CPUs ☐ **1**

19. Configure Project's Logical Groups (AND, Matrix and Service) ☐ **1**

20. Write a rule that turns on the Supply Fan and steady lights the AUTO Segment LEDs (2, 5 & 8) of Cab 1's 3SW/3LED x 6 panel at start up (per the specification) ☐ **1**

21. Configure the three AUTO segment switches (2, 5 & 8) of Cab 1's 3SW/3LED x 6 panel to **NOT** report a monitor event to the LCD panels when each switch is actuated ☐ **1**

22. Write one rule which lights the appropriate **floor of incident** LED to flash fast on the 24 LED panel #1 display of CAB #3 when **ANY** Alarm (including just one AND or Matrix Detector) is reported by the panel for the Main building ☐ **1**

23. Write rules which light the appropriate LED to flash fast on the 24 LED panel #1 display of CAB #3 when **ANY** Alarm is reported by the panel for the Maintenance Building and the two shaft smokes in the Main building ☐ **1**

24. Write two rules that light the appropriate LED **SLOW** on the appropriate 24 LED panel #1 display of CAB #3 when any System Trouble and Local Trouble is reported by the panel. Use TROUBLE for input and output device system troubles (e.g. NACs and smokes) and LOCALTROUBLE for any of the LRM failure pseudo point events ☐ **1**

25. Write one rule that lights the appropriate LED to flash slow on the appropriate 24 LED panel 1 in the CAB3 (3-6ANN) when any SIGA smoke detector, in either building, reports a dirty head maintenance alert condition ☐ **1**

26. Write one rule for **audible** and **visible** zone responses for the Main building, per the specification, when a smoke detector goes into alarm. Exclude: Matrix & And Group. ☐ **1**

NOTE: The Level 1 Speaker/Strobes are connected to the IDC/NAC circuit 1 and 5 outputs of the 3-IDC8/4 module and the Level 1 Amp is the Audio source for this module's audible zone on circuit 5

27. Write one rule for **audible** and **visible** zone responses for the Maintenance building, per the specification, when a smoke detector goes into alarm ☐ **2**

28. Write one rule for **audible** and **visible** zone responses for the Main building, per the specification, when a manual pull station is activated ☐ **2**

29. Write one rule for **audible** and **visible** zone responses for the Maintenance building, per the specification, when a manual pull station is activated ☐ **2**

30. Write one rule for **audible** and **visible** zone responses for the Main building, per the specification, when a waterflow device is activated ☐ **2**

31. Write one rule for **audible** and **visible** zone responses for the Main building, per the specification, when a heat detector is activated ☐ **2**

32. Write one rule for **audible** and **visible** zone responses for the Maintenance building, per the specification, when a heat detector is activated ☐ **2**

33. Write one rule to manually select individual zone paging from the 3-CAB21, 12SW/12LED panel, as specified for the Main building switch configured for toggle operation and the corresponding LED lights steady when each switch segment is activated ☐ **1**

34. Write one rule to manually select paging from the 3-CAB21, 12SW/12LED panel, as specified for the Maintenance building. switch configured for toggle operation and the corresponding LED lights steady when switch segment is activated ☐ **1**

35. Write two rules to display (on the 24 LED user panel 1 display of the CAB3 (3-6ANN)), the state of the Roof Hatch (**SLOW**) and the state of Fire Pump AC Fail (**SLOW**), per the specification ☐ **1**

36. Write six rules to manually **ON-OFF-AUTO** select all three fans from the 3-CAB21 3SW x 3LED x 6 panel per the specification:
    - The ON switch segment (1, 4, and 7) for each turns the corresponding fan ON, lights the corresponding LED segment STEADY, and turns the corresponding auto-LED segment OFF
    - The OFF switch segment (3, 6, and 9) for each turns the corresponding fan OFF, lights the corresponding LED segment STEADY, and turns the corresponding auto-LED segment OFF ☐ **2**

37. Write one rule, using N-Variables, to recall the elevators when a Primary Elevator Lobby Smoke detectors (Level 1, 2 and Level 4 through 8) goes into Alarm ☐ **1**

38. Write one rule to recall the elevators when the Alternate Elevator Lobby Smoke detector (Level 3) goes into Alarm. ☐ **1**

39. Write two rules to slow blink the designated LEDs on the appropriate Cab #3 (3-6ANN) 24 LED panel when the elevator recall control relays are confirmed to be activated ☐ **1**

40. Write three rules to slow blink the designated LEDs on the appropriate Cab #3 (3-6ANN) 24 LED panel when the three fan control relays are confirmed to be activated ☐ **1**

41. Write one rule which turns off the Supply Fan and turns on the two Pressure Fans (with a 5 second delay), per the specification, when any Alarm is reported in the main building only ☐ **2**

42. Write one rule, per the specification, for the 3rd, , 4th and 5th Floor Service Groups ☐ **3**

43. Write one rule, per the specification, for the Time Control for the fans ☐ **1**

44. Write one rule, per the specifications, for **audible** and **visible** zone responses and to light the appropriate LED **FAST** for the 2nd floor Computer Room Matrix Group ☐ **2**

45. Write a rule, per the specifications, for **audible** and **visible** zone responses and to light the appropriate LED **FAST** for the Elevator Equipment Room AND Group ☐ **2**

46. Write one rule to light the appropriate CALL IN LED to blink **FAST** on the Cab #1 24 LED display to indicate Call In Status ☐ **1**

47. Write one rule, per the specifications, to light the appropriate ACTIVATE LED to blink **FAST** on the Cab #1 24 LED display to indicate Phone Activated Status ☐ **1**

48. Write five rules, using N-variables with a width (**<N:n-n:W>**), per the specification, to fast blink the appropriate LED on the 24 LED panel #2 in Cab # 3 to indicate an LRM failure (local trouble) for the LRMs in each of the three system cabinets ☐ **2**

49. Write one rule to manually initiate the system wide audio test message from the appropriate switch on the 3-CAB21, 12SW/12LED panel per the specifications (the corresponding LED lights steady when the switch is activated) ☐ **1**

50. Create two new audio sound clip libraries and record Audio Sound Clips in these two libraries. ☐ **1**

51. Record the Default_Pre, Default_EVAC (Main Building), Default_ALERT (Main building), Default_Normal (header file only), custom Maint_EVAC (Maintenance building) and System Test (both buildings). Create at least one message using the pre and post clip features to construct it from the clip library. ☐ **2**

52. DB convert, download all three CPUs, and ASU databases only ☐ **1**

Total 70

Turn in your practical exercise including a copy of your rules. Remember to test your panel.

Grade_____

FINAL EXAM
ANSWER SHEET

TEST VERSION _____

NAME: _____DATE: _____

CLASS NUMBER: _____

1 _____	11 _____	21 _____
2 _____	12 _____	22 _____
3 _____	13 _____	23 _____
4 _____	14 _____	24 _____
5 _____	15 _____	25 _____
6 _____	16 _____	26 _____
7 _____	17 _____	27 _____
8 _____	18 _____	28 _____
9 _____	19 _____	29 _____
10 _____	20 _____	30 _____