



The Gent Driver

The Gent driver connects to the range of Gent by Honeywell's Vigilon fire detection panels supporting the Open Protocol Specification. Available for Commander and ObSys.

This document relates to Gent driver version 2.0

Please read the *Commander Manual* or *ObSys Manual* alongside this document, available from www.northbt.com

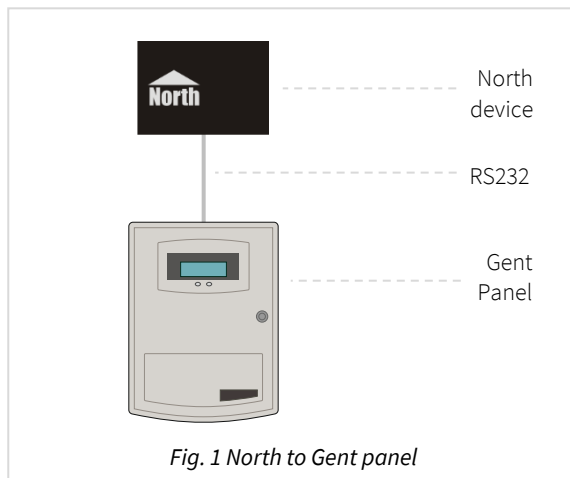
Contents

Compatibility with the Gent System	3
Equipment	3
Values	3
Prerequisites	4
Driver Operation	5
Events from the panel.....	5
Alarms	5
Event Storage and Latching Options	5
Using the Driver	6
Making the Cable	6
Starting the Interface.....	6
Setting up the Driver.....	6
Checking Communications	6
Alarms	7
Format.....	7
Examples	7
Point Field	7
Condition and Priority Field	8
Object Specifications.....	9
Example Object Reference	9
Device Top-Level Objects	9
Gent Driver Setup.....	10
Gent Driver Filter Events.....	11
Event Storage Options.....	12
Gent System	13
System Summary.....	14
Gent Panel.....	15
Commands.....	15
Loop	16
Device.....	17
Sub Address	17
Sector	18
Zone	19
Supervisory Inputs.....	20
Driver Versions	21

Compatibility with the Gent System

The Gent driver allows North to interface with Gent by Honeywell's Vigilon fire detection system.

The driver connects to a Gent fire control panel (Fig. 1), and can communicate with a network of up to 32 panels.



Equipment

Gent fire control panels supporting the Open Protocol Specification, compatible with the driver include:

- Vigilon Compact Panel (EN)
- Vigilon Compact VA panel (EN)
- Vigilon 4/6 loop panels (EN/BS)
- Vigilon network nodes
- Vigilon 34000 panels
- 3404 Control panel (1-4 loops)
- 2408 Control panel (1-8 loops)
- 3500 Network terminal node

Values

The driver can typically access the following values from a fire control panel:

- | | | |
|----------------|---------------------|---------------------|
| • Reset panel | • Supervisory state | • Sub-address state |
| • Sounders | • Loop state | • Zone state |
| • System state | • Sector state | |
| • Panel state | • Loop device state | |

States for fire, fault, and isolation are available.

Fire control panels can send alarms to the Gent driver. Long labels are not supported – the maximum label length available from a panel is 32 characters.

On starting, the driver will be unaware of the current fire system status.

Prerequisites

Vigilon 4/6 loop panels require an IO card (VIG-IOC-DOM) fitting in socket P2 of the backplane. On the card, set RS232 port mode to 'universal full-duplex'. The RS232 serial cable should be attached to connector P4 on the terminal card.

Vigilon Compact panels do not require an additional IO card. The RS232 serial cable should be attached to connector PB6.

The Vigilon panel's port must be configured with a baud rate and, on a Vigilon Compact panel, also the mode.

Configure the Vigilon port as follows (requires a level 3 user PIN):

Press **Menu On/Off** button and then select **[Test/Eng]**, **[Usercode]**, select **<etc>** and then select **[Config]**. Momentarily press **<etc>** to select **[Comms]**. Then select **[Card]** and proceed depending on the panel model.

On a Vigilon 4/6 loop panel, enter card '15'. Select **[Baud]**, scroll to '19200' and press **[Enter]**.

On a Vigilon Compact panel, enter card '0', then port '1'. Select **[Mode]** and scroll to 'Universal' and press **[Enter]**. Next, set select **[Baud]** and scroll to '19200' and press **[Enter]**.

Refer to the Gent Vigilon commissioning manual for further information.

To enable supervisory events, the command build label should be prefixed with an '&' (ampersand) character.

Driver Operation

Events from the panel

The driver connects to a Gent Vigilon fire alarm panel and listens for change-of-state events. It processes these events in real time to maintain a local database of active alarm states.

On reading an object from the Gent system, the driver responds immediately with the current state stored in this database.

Important operational notes

- **Initial state:** Upon startup, the driver is unaware of the current fire system status. The panel may need to be reset to regenerate events and populate the driver database.
- **Communication loss:** The driver monitors communication with the connected panel and reports a fault if connectivity is lost. Once communication is restored, the driver cannot backfill or request events that occurred during the outage.
- **Disablement Identification:** Disablement events received from the panel do not include specific zone or device identifiers. Consequently, the driver can only report the panel and loop numbers associated with active disablements.

Alarms

When an event is received from the panel, the driver converts it into a standard North-format alarm and forwards it to the North device's alarm processing system.

Event Storage and Latching Options

Because the event data sent from the panel can vary, the driver provides specific configuration objects to manage how events are stored and synchronized:

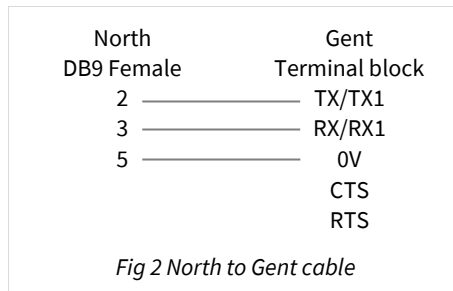
- **Database Resynchronization:** Specific event types can be manually cleared from the driver database to aid in resynchronization.
- **Latching Behaviour:** Configuration objects allow engineers to set faults, disablements, and supervisory events to operate in latching or non-latching mode.
 - **Fire events** always operate in latching mode. They remain active in the driver database until cleared by a manual panel reset event.
 - **Latching mode** (faults, disablements, supervisory) events remain active in the driver database until cleared by a panel reset.
 - **Non-latching mode** (faults, disablements, supervisory) events dynamically follow the state of the panel. They are automatically cleared from the driver database upon receipt of a matching clear event from the panel.
- **Persistence:** When supervisory events are configured for non-latching mode, their states are stored in the North device's persistent memory. This ensures that the supervisory input state remains available following a device restart.

Using the Driver

On ObSys, the Gent driver is pre-installed. On Commander, the Gent driver is available to download in the file 'Bank7 Gent.cdm'. On both these North devices, you can use the driver to create an interface to Gent. Once started, you will need to set up the driver before it can communicate with the Gent system.

Making the Cable

Using the RS232 cable specification (Fig. 2), connect the North device COM port to the Gent Vigilon panel – port P4 on a Vigilon 4/6 loop panel, or PB6 on a Vigilon Compact panel. Connector types at each end of the cable are shown:



The maximum RS232 cable length is 15m and should be as short as possible.

Cables are available from North, order code CABLE/GENT/TERM.

Starting the Interface

- 🖥 To start an interface using the Gent driver, follow these steps:
 - **Start Engineering** your North device using ObSys
 - Navigate to **Configuration, Interfaces**, and set an unused **Interface** to 'Gent' to start the particular interface
 - Navigate to the top-level of your North device and re-scan it

The driver setup object (Mc), labelled **Gent Setup**, should now be available. If this object is not available, check an interface licence is available and the driver is installed.

Setting up the Driver

- 🖥 To set up the driver, follow these steps:
 - Navigate to the **Gent Setup** object (Mc). For example, if you started interface 1 with the driver earlier, then the object reference will be 'M1'
 - Set **RS232 Com Port** (RS.COM) to select the serial port number on the North device the panel is connected to.

Checking Communications

Gent Setup contains a **Comms Online** (DS) object. A value of 'Yes' indicates the driver is receiving data from the Gent system.

Alarms

When the Gent system reports an event to the driver, the driver sends a North-format alarm to the device's alarm processing.

Format

North-format alarms contain six text fields. The Gent driver places the following information into these fields:

System – copied from System Label object (DL) within driver setup

Point – see Point Field section below

Condition – see Condition and Priority Field section below

Priority – see Condition and Priority Field section below

Date & Time – when the condition was detected by the fire panel

Examples

System	Point	Condition	Priority	Date	Time
Gent System	Panel 1 Loop 1 Dev 3.0 Zone 1	Fire	1	19/05/26	14:29:48
Gent System	Panel 1	Alarms Sounding	3	19/05/26	14:29:49
Gent System	Panel 2 Loop 8	Fault	3	19/05/26	14:35:12
Gent System	Panel 1	Reset	1	19/05/26	14:38:26
Gent System	Panel 2 Supv 5	On	4	19/05/26	14:42:34
Gent System	Panel 1 Loop 2	Disabled	2	19/05/26	14:51:03
Gent System	Panel 1 Loop 2	Enabled	2	19/05/26	15:17:41

Point Field

Selected by the **Alarm Point Field** object (AT) within driver setup.

If 'PLD reference' option is selected, Point field can be:

System

Panel *p*

Panel *p* Loop *l*

Panel *p* Loop *l* Dev *d.s*

Panel *w* Loop *l* Dev *d.s* Zone *z*

Panel *w* Supv *u*

If the device label starts 'MCP', then 'MCP' is appended to the field.

If 'Sector' option is selected, Point field can be:

System

Panel *p*

Panel *p* Loop *l*

Panel *p* Loop *l* Sect *t,t,t*

Panel *w* Supv *u*

If 'Detector label' option is selected, Point Field contains:

Device location from panel (up to 32 chars)

Condition and Priority Field

The following alarm conditions can be sent by the driver:

Condition	Priority
Fire	1
Reset	1
Fault	3
System Fault	3
Fault Cleared	3
Disabled	2
Enabled	2

Condition	Priority
System Fault	3
Alarms Silenced	3
Alarms Sounding	3
Communications Lost	3
Communications Regained	3
On	4
Off	4

Notes

The priority field is typically used as follows:

- 1 – fire and evacuate events
- 2 – isolation events
- 3 – fault events
- 4 – status events

Object Specifications

Once an interface is started, one or more extra objects become available within the top-level object of the device. As with all North objects, each of these extra objects may contain sub-objects, (and each of these may contain sub-objects, and so on) - the whole object structure being a multi-layer hierarchy. It is possible to navigate around the objects using the ObSys Engineering Software.

Each object is specified below, along with its sub-objects.

Example Object Reference

An example of a reference to an object in the same device: the Gent System (S1) contains Panel 1 (P1), which contains Loop 3 (L3), which has Device 35 (D35), which contains a condition (C). Therefore, the complete object reference will be 'S1.P1.L3.D35.C'.

An example of a reference to an object in a different device: the IP network object (IP) contains Default Commander object (CDIP), which contains the object above (S1.P1.L3.D35.C) – therefore the complete object reference is 'IP.CDIP.S1.P1.L3.D35.C'.

Device Top-Level Objects

When an interface is started using the Gent driver, the objects below become available within the top-level object of the device. For example, if Interface 1 is started, then the object with references 'M1' and 'S1' become available.

Description	Reference	Type
Gent Setup Set up the Gent driver, started on interface c (c is the interface number)	Mc	Fixed Container: On the Commander platform this will be <i>[CDM v20\Gent v20]</i> On the ObSys platforms this will be <i>[OSM v20\Gent v20]</i>
Gent System Access Gent system connected to interface c (c is the interface number)	Sc	Variable Container: <i>[Gent v20]</i>

Gent Driver Setup

Object Type: *[OSM v20\Gent v20]*

Object Type: *[CDM v20\Gent v20]*

The Gent driver contains the following objects:

Description	Reference	Type
System Label Label displayed when scanning the system and within alarms	DL	Obj\Text: 20 Chars; Adjustable
RS232 COM Port	RS.COM	Obj\Num: 1...8; Adjustable
Baud Rate	RS.BR	Obj\Num; Adjustable Values: 1200, 4800, 9600, 19200
Domain Domain number of fire system	DN	Obj\Num; Adjustable 0: auto-detect
Comms Online Indicates whether communications has been established with the panel	DS	Obj\NoYes
Alarm Point Field Selects source of the alarm message point field	AT	Obj\Enum: 0...3; Adjustable Values: 0=PLD reference, 1=Detector label (if available), 2=Sector reference
Event storage available Each event from the system must be remembered by the driver. If no storage is available for a new event, the driver will not be able to remember it. Event storage excludes supervisory states, which are stored separately.	SC	Obj\Num: 0...500
Reset Driver Clears the internal database	RST	Obj\NoYes; Adjustable
Filter Events Ignore listening for particular event types or panel addresses. This can provide more event storage for other event types.	FE	Fixed Container: On the Commander platform this will be [CDM v20\Gent v20\Filter] On the ObSys platforms this will be [OSM v20\Gent v20\Filter]
Event Storage Options Manage how events are stored and synchronized	A	Fixed Container: On the Commander platform this will be [CDM v20\Gent v20\Advanced] On the ObSys platforms this will be [OSM v20\Gent v20\Advanced]

Gent Driver Filter Events

Object Type: [OSM v20\Gent v20\Filter]

Object Type: [CDM v20\Gent v20\Filter]

Each event from the Gent system must be remembered by the driver. On a large fire system with a significant number of active events at any one time, use this object to select which event types are ignored by the driver – disablement, fault, supervisory events, etc.

Description	Reference	Type
Ignore Disablement events Enable to ignore isolation events from the system	I.C1	Obj\NoYes; Adjustable
Ignore Fault events Enable to ignore fault events from the system	I.C2	Obj\NoYes; Adjustable
Ignore Fire events Enable to ignore fire events from the system. This provides more event storage for other event conditions	I.C4	Obj\NoYes; Adjustable
Ignore Supervisory events Enable to ignore supervisory events from the system	I.C5	Obj\NoYes; Adjustable
Store events from panel (start) Lowest address of network interface panel to store events from	PS	Obj\Num: 0...31; Adjustable
Store events from panel (end) Highest address of network interface panel to store events from	PE	Obj\Num: 0...31; Adjustable

Event Storage Options

Object Type: [OSM v20\Gent v20\Advanced]

Object Type: [CDM v20\Gent v20\Advanced]

Because the event data sent from the panel can vary, the driver provides these objects to manage how events are stored and synchronized.

Clear Events of Type: Specific event types can be manually cleared from the driver database to aid in resynchronization.

Latching Behaviour: Set faults, disablements, and supervisory events to operate in either a latching or non-latching mode.

- **Fire events** always operate in latching mode. They remain active in the driver database until cleared by a manual panel reset event.
- **Latching mode** (faults, disablements, supervisory) events remain active in the driver database until cleared by a panel reset.
- **Non-latching mode** (faults, disablements, supervisory) events dynamically follow the state of the panel. They are automatically cleared from the driver database upon receipt of a matching clear event from the panel.

Persistence: When supervisory events are configured for non-latching mode, their states are stored in the North device's persistent memory. This ensures that the supervisory input state remains available following a device restart.

Description	Reference	Type
Clear Events of Type Remove specific event types from the driver's event storage	RST	Obj\Enum; Adjustable Values: 0=None, 1= Disablement, 2=Fault 4=Fire, 5=Supervisory, 8=All events
Disablement Events Latching	EL1	Obj\NoYes; Adjustable
Fault Events Latching	EL2	Obj\NoYes; Adjustable
Supervisory Events Latching	EL5	Obj\NoYes; Adjustable

Gent System

Object Type: *[Gent v20]*

The Gent system is a network of Gent fire detection panels. It contains objects to view the status of the whole system (P) and objects to access information from each connected panel (Px).

The Gent system will contain one or more panels.

Description	Reference	Type
System Summary	P	Fixed container: <i>[Gent v20\System]</i>
Panel x The panel number, x, is a number in the range 1...31	Px	Fixed container: <i>[Gent v20\Panel]</i>

System Summary

Object Type: *[Gent v20\System]*

The Gent System Summary object contains network-wide status for the Gent system.

Note that system counts may lag the true status of the Gent system by up to two minutes (or more on complex networks).

Description	Reference	Type
System Alarm State	C	Obj\Enum: 0...4; Where: 0=OK, 1=Disablement, 2=Fault, 4=Fire
System OK	C0	Obj\NoYes
Disablement	C1	Obj\NoYes
Fault	C2	Obj\NoYes
Fire	C4	Obj\NoYes
Disablement Count	N1	Obj\Num
Fault Count	N2	Obj\Num
Fire Count	N4	Obj\Num

Deprecated objects

For compatibility with previous driver versions, the following objects are also available. Their use is not recommended for new configurations.

Description	Reference	Type
Disablement Count	DC	Obj\Num
Fault Count	LC	Obj\Num
Fire Count	FC	Obj\Num

Gent Panel

Object Type: *[Gent v20\Panel]*

A Gent panel contains the following objects:

Description	Reference	Type
Commands Contains objects for resetting latched events and silencing sounders	A	Fixed container: <i>[Gent v20\Actions]</i>
Panel Alarm State	C	Obj\Enum: 0...4; Where: 0=OK, 1=Disablement, 2=Fault, 4=Fire
Panel OK	C0	Obj\NoYes
Disablement	C1	Obj\NoYes
Fault	C2	Obj\NoYes
Fire	C4	Obj\NoYes
Loop x The loop number, x, is in the range 0...8	Lx	Fixed container: <i>[Gent v20\Loop]</i>
Zone y The Zone number, y, is in the range 1..128	Zy	Fixed container: <i>[Gent v20\Zone]</i>
Supervisory Inputs	V	Fixed container: <i>[Gent v20\Supv]</i>

Deprecated objects

For compatibility with previous driver versions, the following objects are also available. Their use is not recommended for new configurations.

Description	Reference	Type
Reset	R	Obj\NoYes; Adjustable only
Sounder	S	Obj\OffOn; Adjustable
Value	V1	Obj\Enum Value: 0=OK, 15=Fault, 17=Isolation, 19=Alarms Sounding, 22=Fire
Fault	F15	Obj\NoYes
Disablement	F17	Obj\NoYes
Alarm Sounding	F19	Obj\NoYes
Fire	F22	Obj\NoYes
Supervisory z	Sz.S	Obj\OffOn

Commands

Object Type: *[Gent v20\Actions]*

The Gent panel Commands object contains the following:

Description	Reference	Type
Reset Performs a reset on the panel	R	Obj\NoYes; Adjustable only
Sounders Silences or re-sounds sounders	S	Obj\OffOn; Adjustable

Loop

Object Type: *[Gent v20\Loop]*

A Gent Loop contains the following objects:

Description	Reference	Type
Loop Alarm State	C	Obj\Enum: 0...4; Where: 0=OK, 1=Disablement, 2=Fault, 4=Fire
Loop OK	C0	Obj\NoYes
Disablement	C1	Obj\NoYes
Fault	C2	Obj\NoYes
Fire	C4	Obj\NoYes
Device x The device address, x, is in the range 1...207.	Dx	Fixed container: <i>[Gent v20\Device]</i>
Sector t The sector number, t, is in the range 1..32	Tt	Fixed container: <i>[Gent v20\Sector]</i>

Deprecated objects

For compatibility with previous driver versions, the following objects are also available. Their use is not recommended for new configurations.

Description	Reference	Type
Value	V1	Obj\Enum Value: 0=OK, 15=Fault, 17=Isolation, 22=Fire
Fault	F15	Obj\NoYes
Disablement	F17	Obj\NoYes
Fire	F22	Obj\NoYes
Sector t The sector number, t, is in the range 1..32	St	Fixed container: <i>[Gent v20\Sector]</i>

Device

Object Type: *[Gent v20\Device]*

A Gent device contains the following objects.

Isolation events from the Gent system do not include a device number, so these are only reported at loop or panel level.

Description	Reference	Type
Device Alarm State	C	Obj\Enum: 0...4 Where: 0=OK, 2=Fault, 4=Fire
Device OK	C0	Obj\NoYes
Device in Fault	C2	Obj\NoYes
Device in Fire	C4	Obj\NoYes
Sub address x The sub address or channel number, x, is in the range 0...7	Sx	Fixed Container: <i>[Gent v20\SubDev]</i>

Deprecated objects

For compatibility with previous driver versions, the following objects are also available. Their use is not recommended for new configurations.

Description	Reference	Type
Value	V1	Obj\Enum Value: 0=OK, 15=Fault, 22=Fire
Fault	F15	Obj\NoYes
Fire	F22	Obj\NoYes

Sub Address

Object Type: *[Gent v20\SubDev]*

A Gent sub-address contains the following objects:

Description	Reference	Type
Alarm State	C	Obj\Enum: 0...4 Where: 0=Ok, 2=Fault, 4=Fire
OK	C0	Obj\NoYes
Fault	C2	Obj\NoYes
Fire	C4	Obj\NoYes

Sector

Object Type: *[Gent v20\Sector]*

A Gent loop sector contains the following objects:

Description	Reference	Type
Sector Alarm State	C	Obj\Enum: 0...4; Where: 0=OK, 1=Disablement, 2=Fault, 4=Fire
Sector OK	C0	Obj\NoYes
Disablement	C1	Obj\NoYes
Fault	C2	Obj\NoYes
Fire	C4	Obj\NoYes

Deprecated objects

For compatibility with previous driver versions, the following objects are also available. Their use is not recommended for new configurations.

Description	Reference	Type
Value	V1	Obj\Enum Value: 0=OK, 15=Fault, 17=Isolation, 22=Fire
Fault	F15	Obj\NoYes
Disablement	F17	Obj\NoYes
Fire	F22	Obj\NoYes

Zone

Object Type: *[Gent v20\Zone]*

A Gent zone contains the following objects.

Isolation events from the Gent system do not include a zone number, so these are only reported at loop or panel level.

Description	Reference	Type
Zone Alarm State	C	Obj\Enum: 0...4; Where: 0=OK, 2=Fault, 4=Fire
Zone OK	C0	Obj\NoYes
Fault	C2	Obj\NoYes
Fire	C4	Obj\NoYes

Deprecated objects

For compatibility with previous driver versions, the following objects are also available. Their use is not recommended for new configurations.

Description	Reference	Type
Value	V1	Obj\Enum Value: 0=OK, 15=Fault, 22=Fire
Fault	F15	Obj\NoYes
Fire	F22	Obj\NoYes

Supervisory Inputs

Object Type: *[Gent v20\Supv]*

A Gent Supervisory is a general purpose, non-fire event typically used for monitoring other equipment, or performing non-fire actions on the fire system.

To enable supervisory events from the Gent system, the command build label should be prefixed with an ‘&’ (ampersand) character.

When supervisory events are configured for non-latching mode in the driver, their states are stored in the North device's persistent memory. This ensures that the supervisory input state remains available following a device restart.

Description	Reference	Type
Supervisory <i>i</i> The input number, <i>i</i> , is in the range 1...255	<i>li</i>	Obj\OffOn

Driver Versions

Version	Build Date	Details
1.2	07/03/2000	Implement sector bit array, Channel fields, re-worked supervisory elements
1.2	18/04/2000	Added driver alarm buffer
1.3	18/06/2003	Protocol change - warning state changed to isolation
1.4	18/06/2003	Driver updated with protocol changes, added write commands
1.4	27/02/2009	Fixed problem clearing an event. Increased database size.
1.4	04/06/2013	Fixed problem with .V objects not returning system fault
1.4	04/10/2023	Modified isolation objects (I) to return '' (blank) rather than '0' when reading. Improving operation in Essential Data. Default baud rate 19200
2.0	05/06/2026	Redesigned to bring up to the latest North fire interface design. Driver settings will wipe on update. Fix: Reset event cleared all events. This is not always required. Added Latching event options. Mod: Added filter objects. Added Domain object. Mod: Increased event storage to 500 (was 130). Added .SC driver object. Mod: Supervisory event states are now stored separately in persistent memory. Mod: New .C objects added. Mod [breaking change]: .V1 and .Fx objects retained for backward compatibility, but sub-device and V objects are no longer supported. Mod [breaking change]: Removed write-only objects for Time and Detector Label Mod [breaking change]: Alarm point field changed.

Next Steps...

If you require help, contact support on 01273 694422 or visit www.northbt.com/support



North Building Technologies Ltd
+44 (0) 1273 694422
support@northbt.com
www.northbt.com

This document is subject to change without notice and does not represent any commitment by North Building Technologies Ltd.

ObSys and Commander are trademarks of North Building Technologies Ltd. All other trademarks are property of their respective owners.

© Copyright 2026 North Building Technologies Limited.

Author: JF
Checked by: JM

Document issued 11/06/2026.