

Driver Manual
(Supplement to the FieldServer Instruction Manual)

FS-8705-37

Honeywell HUS CCTV System

Ethernet Driver

TABLE OF CONTENTS

TABLE OF CONTENTS	3
Honeywell HUS Driver Description.....	5
Driver Scope of Supply	6
1.1 Supplied with this driver.....	6
ABBREVIATIONS AND GLOSSARY	7
Hardware Connections.....	12
1.2 Block Diagram.....	12
1.1 Cable Connections.....	13
Configuring the FieldServer as a Honeywell HUS Client	14
1.2 Data Arrays	15
1.2.1 Data Arrays – Example.....	15
1.3 Client Side Connections.....	16
1.3.1 Client Side Connection Descriptions - Example	16
1.4 Client Side Nodes.....	17
1.4.1 Client Side Nodes – Example	17
1.5 Client Side Map Descriptors	18
1.5.1 FieldServer Related Map Descriptor Parameters	18
1.5.2 Driver Related Map Descriptor Parameters	19
1.5.3 Actions (actions.csv).....	20
1.5.4 Priorities (priority.csv)	22
1.6 Examples.....	26
1.6.1 Map Descriptor - Required	26

Configuring the FieldServer as a Honeywell HUS Server27

Appendices.....28

1.7 Sample Configuration 28

1.1 HUS Subscriptions..... 32

1.2 HUS Camera – PointID , AlarmCode, AlarmType Code. 32

1.3 HUS TCP/IP 32

1.4 Direct control of each Camera 33

1.5 Revision History 34

Honeywell HUS Driver Description

This Ethernet driver supports an Ethernet TCP/IP connection between the HUS and the gateway. Once connected, the driver can send alarm notifications to the HUS system. Within the HUS system these alarm notifications can be processed with HUS rules to trigger actions or further notifications (e.g. triggering cameras to designated preset positions).

The driver can be linked to any of the library of over 120 protocols supported by the FieldServer products and thus the HUS system can be driven with data from Modbus, BACnet, Rockwell, Lonworks, GE PLC, Omron, JCI and many other systems.

The driver is fully compatible with other FieldServer drivers and meets FieldServer's quality assurance standards. The driver was developed by Chipkin Automation Systems, an Approved FieldServer Integrator.

This driver comes in 2 flavors. 1) Provides a simplified interface which triggers configured presets 2) provides full real time control over preset parameters – They do not need to be preconfigured in the gateway.

Max Nodes Supported

FieldServer Mode	Nodes	Comments
Client	1	One gateway supports a connection to a single Honeywell HUS system.
Server	0	Not supported or documented.

Driver Scope of Supply

1.1 Supplied with this driver

FieldServer Technologies PART #	Description
Cables	No specific cables are shipped with this driver.
FS-8705-37	Driver Manual.

ABBREVIATIONS AND GLOSSARY

PointID, AlarmCode, AlarmType

Keywords used in the configuration of this Driver. Their definition and meaning is in the Honeywell HUS system – these are HUS keywords.

The HUS performs actions based on alarm notifications. An alarm notification is defined by these 3 numeric fields.

Camera

A physical CCTV Camera is one meaning. Another meaning is specific to the operation/configuration of this driver. In the context of the gateway, a camera is collection of up to 64 triggers.

Its important to understand that there is no correlation between the gateway's camera's and the physical camera. The gateway sends alarm notification to the HUS in the form of a triplet of information – PointID, AlarmCode, AlarmTypeCode. Using the HUS configuration software an engineer configures the actions that will be taken when an alarm notification is received. Such an action can be to drive a physical camera. The HUS has the relationship between (PointID, AlarmCode, AlarmTypeCode) and (Physical Camera). The Gateway simply has 500 channels that can have up to 64 triggers only one of which will be acted on at a time.

It is sensible to configure the gateway and the HUS to use the same camera numbers.

Subscription

The HUS system will only take actions based on Alarm Notifications that it expects. Thus the HUS and driver must be configured with the same set of Alarm Notification parameters (PointID, AlarmCode, AlarmTypeCode). When the HUS connects with the Gateway it sends the gateway 'Subscription' messages to tell the Gateway which Alarm Notifications it knows about.

Action

When the Gateway sends an alarm notification message to the HUS system we say the gateway is performing an action.

When the HUS drives a CCTV camera to a given preset in response to an given alarm notification we call that an action too.

Event / Trigger

An automated plant has various modes of operating and can be in various operational modes and states. An event is an operational state that needs to be monitored and which may trigger some actions by humans and other parts of the automation system.

In the case of the automated physical plant, some system like a PLC system might monitor for Fire, Gas and other alarm conditions. When notice of one of these events is sent to the gateway it triggers an action by the driver – being the sending of an alarm notification to the HUS

Protocol

A protocol is a standardized set of rules that specify how a conversation is to take place, including the format, timing, sequencing and/or error checking

Driver

Refers to the component of an computer / microprocessor application which handles communications.

In relation to a gateway – A driver is the code that handles one communication protocol. It is linked together with another protocol driver and a kernel of services to make gateway firmware.

Firmware

The application loaded on the gateway. The program that does the work of making the gateway operate like a gateway. It is non-human readable digital file. Firmware has a version number that increases each time a change is made to the firmware.

Version Numbers

A number used to identify a particular version of something. A number that increases each time a change is made to the firmware. In FieldServer gateways the following items have unique version numbers – the hardware, the kernel, each driver, the configuration.

Configuration

Gives personality to the firmware. In a gateway the firmware has general capability and the configuration is used to direct that general capability to perform specific (required) operations. Firmware can send a TCP/IP message. Configuration can tell it the IP address to send the message to.

Each driver is configured in a different way because each protocol has its own characteristics such as how data is addressed.

This driver is configured as part of the gateway file called “config.csv”. In addition the driver expects to find these 2 files – “actions.csv” and “priority.csv” These 3 files form the full configuration of the driver.

Honeywell HUS

A product line made and sold by Honeywell. HUS stands for Honeywell Universal Surveillance.

FieldServer

A FieldServer connects two networks of the same access method, for example, Ethernet to Ethernet or Token Ring to Token Ring. A FieldServer works at the OSI's Media Access Layer and is transparent to upper-layer devices and protocols. FieldServers operate by filtering packets according to their destination addresses. Most FieldServers automatically learn where these addresses are located, and thus are called learning FieldServers.

Gateway

In OSI terminology, a gateway is a hardware and software device that connects two dissimilar systems such as a LAN and a mainframe. It operates at the fourth through seventh layers of the OSI model. In Internet terminology, a gateway is another name for a router.

Bandwidth

Bandwidth is the amount of data that can be transmitted over a channel, measured in bits per second. For example, Ethernet has a 10Mbps bandwidth and FDDI has a 100 Mbps bandwidth. Actual throughput may be different than the theoretical bandwidth.

TCP/IP, Transmission Control Protocol/ Internet Protocol

TCP/IP is the protocol suite developed by the Advanced Research Projects Agency (ARPA), and is almost exclusively used on the Internet. It is also widely used in corporate internetworks, because of its superior design for WANs. TCP governs how packets are sequenced for transmission. IP provides a connectionless datagram service. "TCP/IP" is often used to generically refer to the entire suite of related protocols.

Local Area Network (LAN)

A LAN is a group of computers, each equipped with the appropriate network adapter card and software and connected by a cable, that share applications, data and peripherals. All connections are made by cable or wireless media, but a LAN does not use telephone services. It typically spans a single building or campus.

Wide Area Network (WAN)

A WAN consists of multiple LANs that are tied together via telephone services and/or fiber optic cabling. WANs may span a city, state, a country or even the world

Client

In Data communications terminology a client is the consumer of information. An active client, requests the information. A passive client, listens for it.

Server

In Data communications terminology a server is the source of information. An active server transmits data without being requested. A passive server, wait for a request and then transmits the data in a response.

Client / Server Side of a Gateway

A gateway always has at least two protocol drivers because it exchanges data between 2 protocols.

A gateway can be configured to be any of the following combos

1. An active client, passive server – Reads data using protocol #1 and waits passively to be asked for data on protocol #2.
2. A passive client, passive server – Waits, listening for unsolicited data messages on protocol #1 and also waits passively to be asked for data on protocol #2
3. A passive client, active server – Wait,s listening for unsolicited data messages on protocol #1 and transmits the data on protocol#2 without being asked – spits it out.
4. A active client, active server – Reads data using protocol #1 and transmits the data on protocol#2 without being asked – spits it out.

Data Array

A place in which data is stored internally in the gateway. Data Arrays have length and Type. Eg. 100 Unsigned 16 bit numbers.

They are defined in the Gateway's configuration. Both protocols have access to the Data Arrays and that is how they share data.

Adapter

The keyword using in the gateway configuration to refer to an Ethernet

N1, N2

Some FieldServer gateways have more than one Ethernet Port. They are labelled N1 and N2. Each port can be configured to be on a different subnet.

Node

Keyword used in the gateway configuration. Refers to a device. In the case of a client side node it defines an external device that is the source of the data.

Map Descriptor

In the gateway, this is a keyword used to define a task (like a client side task to read particular data from a particular device.) or capability (how the data is identified on the server side).

UINT16

16 Bit Unsigned Integer.

ALLSUB

The identifier the driver uses to accept a connection from the HUS. This parameter is configured in the HUS system and hardcoded in the driver.

Preloads

A section in the gateway configuration where some/all data arrays are prepopulated with constants.

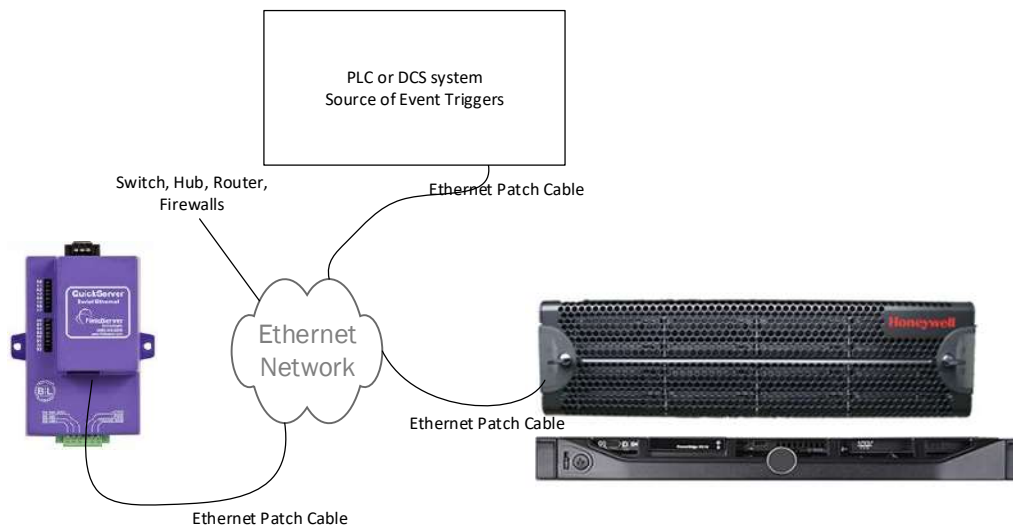
Hardware Connections

1.2 Block Diagram

Multiple WorkStation protocols and connection supported. See list of FieldServer Drivers.



1.1 Cable Connections



Configuring the FieldServer as a Honeywell HUS Client

For a detailed discussion on FieldServer configuration, please refer to the FieldServer Configuration Manual. The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer (See “.csv” sample files provided with the FS).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with an Honeywell HUS system.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for Honeywell HUS interaction, the driver independent FieldServer buffers need to be declared in the “Data Arrays” section, the destination device addresses need to be declared in the “Client Side Nodes” section, and the data required from the servers needs to be mapped in the “Client Side Map Descriptors” section. Details on how to do this can be found below.

Note that in the tables, * indicates an optional parameter, with the bold legal value being the default.

1.2 Data Arrays

Section Title		
Data_Arrays		
Column Title	Function	Legal Values
Data_Array_Name	Provide name for Data Array	Up to 15 alphanumeric characters
Data_Array_Format	Provide data format. Each Data Array can only take on one format.	Recommended: Bit, UInt16, Also Supported: Float, UInt32, SInt16, Packed_Bit, Byte, Packed_Byte, Swapped_Byte
Data_Array_Length	Number of Data Objects. Must be larger than the data storage area required by the Map Descriptors for the data being placed in this array.	1-10,000

1.2.1 Data Arrays – Example

Data_Arrays

Data_Array_Name , Data_Format , Data_Array_Length

HUS-Stats , UInt16 , 1000

HUS-Interface , UInt16 , 1000

DA_HUS_MODBUS , UInt16 , 3000

HUS-RESETCtrl , UInt16 , 100

1.3 Client Side Connections

Create one connection for each Honeywell HUS Ethernet port. Each connection can only be used to connect to a single Honeywell HUS interface/port.

Section Title		
Connections		
Column Title	Function	Legal Values
Adapter	Specify which network port the device is connected to the FieldServer	N1, N2
Protocol	Specify protocol used	HoneywellHus

1.3.1 Client Side Connection Descriptions - Example

(Configure your gateway as shown in this example. Do not introduce your own variation.)

Connections

Adapter ,protocol

N1 ,HoneywellHUS

1.4 Client Side Nodes

Create one Node per FACP in the network only.

Section Title		
Nodes		
Column Title	Function	Legal Values
Node_Name	Provide name for node	The name is used to form a relationship with Map_Descriptors. Other than that it is simply a description.
Node_ID	Not used directly by the driver	0-255 Commonly omitted. If you are using Node_Status bits then allocate a Node_ID. Give each meter a unique number.
Protocol	Specify protocol used	HoneywellHus

1.4.1 Client Side Nodes – Example

(Configure your gateway as shown in this example. Do not introduce your own variation.)

Nodes

Node_Name ,Node_ID ,Protocol

Gateway ,1 ,HoneywellHUS

1.5 Client Side Map Descriptors

1.5.1 FieldServer Related Map Descriptor Parameters

Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer	One of the Data Array names from “Data Array” section above
Data_Array_Offset	Starting location in Data Array	0 to maximum specified in “Data Array” section above
Function	Function of Client Map Descriptor..	Server

1.5.2 Driver Related Map Descriptor Parameters

Column Title	Function	Legal Values
Node_Name	Name of Node to fetch data from	One of the node names specified in "Client Node Descriptor" above
Length	Length of Map Descriptor	1

1.5.3 Actions (actions.csv)

A file called actions.csv MUST be loaded onto the gateway. (Take care not to overwrite config.csv). It defines actions associated with each trigger for each camera. A Sample is provided below. Only the 1st 5 columns are used by the driver. The remainder of the line is ignored. Commented lines are ignored as is the header line if it begins with the word 'Camera'. The keyword 'end' terminates processing of the file. The file is loaded once when the gateway starts up.

```
Camera No.,Bit No.,PointID,AlarmTypeCode,AlarmCode,Fire Detector Tag,Modbus
Register,Register Bit,PCCTV Camera & Preset,Process Unit

// Lines that begins with double slash are ignored,,,,,,,,,
// Heading line is ignored if 1st word is Camera,,,,,,,,,
// Only 5 columns are processed - rest are reference data,,,,,,,,,
1,0,10001,111,11,01-AFT-00012,40001,0,01-TCAP-000001 Pos 1,Unit 1
1,1,10001,111,11,01-AFT-00013,40001,1,01-TCAP-000001 Pos 1,Unit 1
1,2,10002,111,11,01-AFT-00021,40001,2,01-TCAP-000001 Pos 2,Unit 1
1,3,10003,111,11,01-AFT-00022,40001,3,01-TCAP-000001 Pos 3,Unit 1
1,4,10003,111,11,01-AFT-00023,40001,4,01-TCAP-000001 Pos 3,Unit 1
1,5,10004,111,11,01-AFT-00046,40001,5,01-TCAP-000001 Pos 4,Unit 1
2,0,10006,111,11,01-AFT-00004,40005,0,01-TCAP-000002 Pos 2,Unit 1
2,1,10013,111,11,01-AFT-00014,40005,1,01-TCAP-000002 Pos 9,Unit 1
2,2,10014,111,11,01-AFT-00015,40005,2,01-TCAP-000002 Pos 10,Unit 1
2,3,10007,111,11,01-AFT-00016,40005,3,01-TCAP-000002 Pos 3,Unit 1
2,4,10005,111,11,01-AFT-00033,40005,4,01-TCAP-000002 Pos 1,Unit 1

end,,,,,,,,,

// End statement ends processing of file - all lines following are
ignored,,,,,,,,,
```

Column 1 – Camera Number

Column 2 – Bit Number (0-63)

Column 3 – Preset PointID

Column 4 – Preset AlarmCode

Column 5 – Preset AlarmTypeCode

Example

1,1,10001,111,11,01-AFT-00013,40001,1,01-TCAP-000001 Pos 1,Unit 1

If Camera 1 – bit 1 is activated then send camera 1 to preset 10001 (111,11)

Notes:

1. The action will have no effect on the HUS until the triplet (PointID,AlarmCode,AlarmTypeCode) have been specified as an alarm/action in the HUS CCTV system.
2. You do not need entries in this file for bits that will not be activated. The sort order of the file is not important.

1.5.4 Priorities (priority.csv)

Its possible that multiple triggers for a single camera could be active at the same time. How does the driver handle this. Normally it does it this way – when a trigger for a camera is active, all other triggers for that camera are ignored. When the active trigger-preset is cleared then the next trigger is immediately considered. The file is loaded once on startup.

Normal Operation – A camera that does not have a priority action defined.

Eg.

Camera 1 Bit 0 is allocated to 10001,111,1

Camera 1 Bit 1 is allocated to 10002,112,2

Camera 1 Bit 2 is allocated to 10003,113,4

Eg 1:

If 40001 = 7. All three bits are activated. Driver selects lowest number bit (bit 0 in this case) and drives camera 1 to preset 10001.

Now external system sets 40001 to 6. Bits 1 and 2 are active,. Bit 0 has been cleared which allows the driver to activate another preset. In this case 10002 because it looks at the active bits starting at the lowest bit number. Bit 2 will be ignored until bit 1 has been cleared.

Eg 2:

If 40001 = 2 then bit 1 is set so camera 1 is driven to 10002.

Now 40001 = 6. Bit 1 and 2 active. A camera preset is already active so the new active bit is ignored.

Now 40001 = 0. All bit are cleared. No action taken.

Now 40001 = 4. Bit 2 is set. Camera 1 driven to preset 10003.

A file called priority.csv MUST be loaded onto the gateway. (Take care not to overwrite config.csv). It can be an empty file.

Only entries for camera's with priority actions are specified here.

It defines actions associated with each trigger for each camera. A Sample is provided below. Only the 1st 5 columns are used by the driver. The remainder of the line is ignored. Commented lines are ignored as is the header line if it begins with the word 'Camera'. The keyword 'end' terminates processing of the file.

```
Camera No.,Active Bits,PointID for Priority Position,AlarmTypeCode,AlarmCode
// Lines that begins with double slash are ignored.,,,,
// Heading line is ignored if 1st word is Camera.,,,,
// Only 5 columns are processed - rest are reference data.,,,,
// Only include cameras with priority rules.,,,,
25,21,10216,333,33
26,21,10224,333,33
27,6,10228,333,33
28,6,10232,333,33
29,6,10236,333,33
30,6,10240,333,33
end,,,
// End statement ends processing of file - all lines following are ignored.,,,,
```

Priorities have been defined for camera 25,26,27,28,39,30. All other camera's will operate without priority actions.

This is how Priority Actions work

If A camera is define to have a priority action, then if that camera has more than one trigger bit active at a time, then that camera shall take the priority action.

Eg.

Camera 25 – Corresponds to Modbus 40097,98,99,100

Lets say Modbus 40097 has a value of 31. Ie bits 0-5 are on.

That means Camera 25 has multiple triggers active.

It is present in the priority.csv file so the driver applies the priority rule and in this example sends camera 25 to (10216,333,33)

Column1 – Camera

Column2 – Max Bits, How many of the 64 bits to evaluate as being simultaneously on.

Column 3 – Preset PointID

Column 4 – Preset AlarmCode

Column 5 – Preset AlarmTypeCode

1.6 Examples

1.6.1 Map Descriptor - Required

This Map Descriptor is required. Configure the gateway as shown below. Do not introduce your own variation. No other Map Descriptors connected to the HoneywellHus Node are permitted.

Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name ,Data_Array_Offset ,Node_Name ,Function ,Length

Rqd but not used ,DA_dummy ,0 ,Gateway ,Server ,1

Descriptive only. Not used.

Data Array and offset where data will be stored. Set as Shown. The Data Array specified here is not used.

Use the name of the node you created for the HUS.

Since the HUS initiates the connection, we have to wait patiently as a

Set the Length to 1

Configuring the FieldServer as a Honeywell HUS Server

This driver cannot be used to emulate a Honeywell HUS system. It always expects the HUS to be the HUS to establish the connection, to subscribe to points and to accept alarm notifications which drive cameras to presets.

Appendices

1.7 Sample Configuration

```

Bridge

Title

FSe7996-Phase2 Rev801

//=====
//
//  Data Arrays
//

Data_Arrays

Data_Array_Name ,Data_Format ,Data_Array_Length

HUS-Stats      ,UINT16      ,1000

HUS-Interface   ,UINT16      ,1000

DA_HUS_MODBUS   ,UINT16      ,3000

HUS-RESETCtrl   ,UINT16      ,100

Preloads,

Data_Array_Name ,Preload_Data_Index ,Preload_Data_Value

// This param is read once on startup. Its the duration of the the timeout after we send an alarm message that we
wait until retry if we dont get a response.

HUS-Stats      ,09          ,20

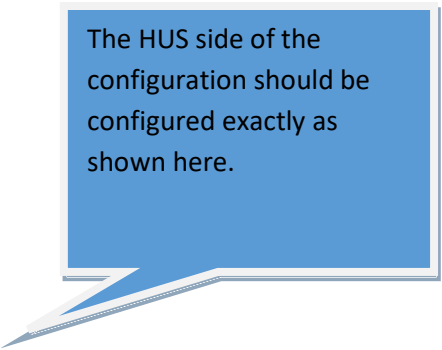
```

Notes on this section are provided at the end of the sample.

```
//=====
//
//   Honeywell HUS
//
Connections
Adapter ,protocol
N1      ,HoneywellHUS

Nodes
Node_Name ,Node_ID ,Protocol
Gateway   ,1      ,HoneywellHUS

Map_Descriptors
Map_Descriptor_Name ,Data_Array_Name ,Data_Array_Offset ,Node_Name ,Function ,Length
Rqd but not used   ,DA_dummy        ,0                ,Gateway   ,Server   ,1
```



The HUS side of the configuration should be configured exactly as shown here.

```
//=====
//
//  Modbus Side
//

Connections

Adapter , Protocol

N1      , Modbus/TCP

Nodes

Node_Name , Node_ID , Protocol

MbServer , 1      , Modbus/TCP

Map_Descriptors

Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Address , Length

Serve Data          , DA_HUS_MODBUS , 4 , Server , MbServer , 40001 , 2000

Listen for heartbeat, HUS-RESETCtrl , 0 , Server , MbServer , 45001 , 100
```

This is the other protocol. In this example – ModbusTCP. It could be BACnet or Rockwell's EthernetIP or any other protocol in our library of over 120 protocols.

The important part is that the other protocol must map data onto the array called DA_HUS_Modbus starting at offset 4.

AND map data onto offset 0 of HUS-RESETCtrl

See notes and end of example.

Dara Array = HUS-Stats ,UINT16 ,1000

Used to control some diagnostic and debugguing functions. Do not use this array unless directed to by a support engineer.

HUS-Interface ,UINT16 ,1000

Operational status of the 1st 100 camera's is displayed here. 10 registers per camera. Displays info like the last sequence number used, the point number, the number of retries, the state of the retry time, the success code and more.

DA_HUS_MODBUS ,UINT16 ,3000

4x 16 Bit registers are allocated per camera. Offset 0,1,2,3 are for camera zero, 4,5,6,7 are for camera 1 etc. This array is mapped onto the other protocol (eg Modbus) to allow a remote system to trigger presets for each camera.

In this example configuration camera zero is not mapped onto Modbus. Modbus is mapped onto offset 4,5,6 etc. with 40001 corresponding to offset 4, 40002 to offset 5 etc. (See the Modbus side of the configuration.)

HUS-RESETCtrl ,UINT16 ,100

This special data array is also mapped onto the other protocol (eg Modbus). The driver expects that the remote system will write a heartbeat to this point. The value must change all the time. Two presets should be configured in the HUS system to report the loss of the connection from the remote system. The gateway must be configured using camera zero for these two presets. If the heartbeat stops then one preset is triggered. If the heartbeat is re-established the other preset is triggered. NOW the HUS can notify the operator using these presets.

In this example the Data Array is mapped onto Modbus 450001

Preloads

The HUS-stats array is used to control some operation and diagnostic functions. In the fragment below offset 9 has been set to a value of 20. When the config loads, the value will be loaded into the data array.

```
Preloads,

Data_Array_Name ,Preload_Data_Index ,Preload_Data_Value

// This param is read once on startup. Its the duration of the the timeout after we send an alarm message that we wait
until retry if we dont get a response.

HUS-Stats      ,09      ,20
```

If you were using BACnet – Map up to 64 binary BO or BV objects into each of the 64 bits allocated to each camera if you want to control the presets with Binary objects.

1.1 HUS Subscriptions

When the HUS connects to the gateway it begins by subscribing to all the points that have been defined. The driver accepts these subscription requests, makes a note of them and respond with an ACK to the HUS. In normal operation the HUS should only subscribe once to each point. In actual operation, during testing, it was noted that multiple subscriptions to a point indicated some other configuration problem in the HUS.

The HUS does not accept alarm notifications from the Driver unless those points already exist in the HUS. In other words you cannot send a trigger for a preset that does not exist. In other words, the triplet of (PointID, AlarmCode, AlarmTypeCode) must already be defined in the HUS for this driver to activate the triplet. On connection to the driver the HUS will report all these triplets by means of sending subscription requests to the driver. (Alarm Notifications - this is how the driver tells the HUS to drive a camera to a preset)

It is normal to ignore the subscription messages during normal operation of a well configured system.

1.2 HUS Camera – PointID , AlarmCode, AlarmType Code.

The 3 parameters have no specific meaning until they have been configured in the HUS. Through the HUS configuration software the triplet is mapped onto an action for a specific camera.

From the driver's point of view a camera is a virtual concept. It's a collection of presets that may be viewed from a particular physical camera.

You cannot tell from the surface what camera a triplet (PointID, AlarmCode, AlarmTypeCode) will actually activate unless you go into the HUS configuration of actions.

1.3 HUS TCP/IP

Driver uses port 6001.

Driver uses "ALLSUB" as the key identifier.

New firmware is required if this changes.

1.4 Direct control of each Camera

A version of this driver is available where the remote system can driver any camera to any preset without requiring them to be configured in csv files in advance. 10x 16bit registers are provided for control and monitoring of every camera. If you are interested in this option please request the additional manual provided.

Here they are shown as Modbus Registers but they could be mapped onto any protocol such as BACnet and Rockwells EthernetIP There are one Chunk of 10 Registers per point.

Offset	Name	Notes	Read/Write on Modbus
0	Trigger	Modbus Writes '1' to this point to trigger alarm. Retriggers ignored while timeout is running.	40001,11,21,31... Read / Write (to trigger)
1	Point Number	Preload with a value to define a point. Eg. Value =9 then this chunk of 10 registers is for point 9	40002,12,22,32... Read
2	AlarmTypeCode	Preload or write to using Modbus.	40003,13,23,33... Read
3	AlarmTypeCode	Preload or write to using Modbus.	40003,13,23,34... Read
4	Subscription Count	Increase each time we receive a subscription for the point	40005,15,25,35... Read
5	Not Used		40006,16,26,36... Read
6	Timer/Counter	Will wait this long for response from HUS before retrying or accepting new triggers	40007,17,27,37... Read
7	Xmite Counter	Number of times the AR message has been sent	40008,18,28,38... Read
8	Transaction Success Code	Set to 3 when message is sent. Result extracted from response if we get one and then overwrites the 3 with 0 (success),1,2	40009,19,29,39... Read
9	Sequence Number.	The number used in the Alarm Notification message to report the alarm. We track it because we might need to re-use it	40010,20,30,40... Read

1.5 Revision History

Date	Resp	Format	Driver Ver.	Doc. Rev.	Comment
2019Jan21	PMC		8.0	1.0	Created
2019Feb07	PMC		8.0	2.0	Released