INDUSTRIAL CONTROL COMMUNICATIONS, INC.

Baumer VeriSens Client Driver Manual

October 30, 2014

TABLE OF CONTENTS

1 Baumer VeriSens Client

The VeriSens client driver can be used to access information on Baumer vision sensors that support the VeriSens process interface.

Some other notes of interest are:

- Supports up to 4 simultaneous connections (defined by the connection objects).
- Supports the following commands: CS (Clear Statistics), GD (Get Data), SJ (Switch Job), TE (Teach Image), TR (Trigger Image), and VB (VeriSens Reboot).
- The server device must not use starting or terminating characters.
- The server device must use a separator/delimiter to separate values in the RD response data.
- Only "polling mode" is supported.

1.1 Client Settings

Scan Rate

This is the time in milliseconds the driver will wait between sending requests. This is a useful feature to reduce overall network utilization, or for certain devices or infrastructure components (such as radio modems) that may not be capable of sustaining the maximum packet rates that the driver is capable of producing. The start time for this delay is taken with respect to the moment at which the driver is capable of sending the next packet (due to either reception or timeout of the previous request). If no additional time is required, setting this field to 0 instructs the driver to send its next request packet as soon as possible.

1.2 Connection Object Settings

The VeriSens client driver uses a construct called a connection object in order to target VeriSens server devices. A connection object can be thought of as a communication channel or "pipe" which is created between the driver and the server device, independent of the service objects that may later make use of that communication channel to transfer service object requests. A connection object defines a connection to a specific endpoint (IP address and port).

IP Address

Defines the IP address of the server device to be targeted by the connection object. All connection object endpoints (IP address and port settings) must be unique.

Port

Defines the TCP port number (1…65535) of the server device to be targeted by the connection object. All connection object endpoints (IP address and port settings) must be unique.

Separator

Enter the delimiter character used to separate value fields in the RD response data. This must match the separator that is configured on the vision sensor. To avoid parsing errors, ensure that this character is unique (i.e. not included as an element within the value fields). Space (" ") characters are not allowed, as it is possible for Result data type values to contain a trailing space character. Similarly, if floating point values are expected, the period (".") character should not be used.

1.3 Command Service Object Settings

A variety of different command service object types are available. Each type initiates a specific action on the server device, and can only be added once per connection object.

1.3.1 Clear Statistics Service Object Settings

This service object sends the "CS" Clear Statistics command and includes a *Job Number* (located at a specified internal database address) as its only parameter. If this service object is not configured for *Manual Trigger*, then a change of value of the job number itself triggers the service object.

Job Number Database Address

When the Clear Statistics command is sent, it includes the job number located at this database address in the payload.

Job Number Data Type

Fixed at 16-Bit Unsigned. The job number is a 16-bit unsigned value and therefore always occupies 16 bits (2 bytes) in the internal database.

1.3.2 Get Data Service Object

This service object sends the "GD" Get Data command, to which the vision sensor replies with an "RD" Response Data packet. Therefore, at least one *Response Data Object* must be included with this connection object to parse the response data. If this service object is not configured for *Manual Trigger*, then it is automatically triggered every scan cycle.

1.3.3 Switch Job Service Object

This service object sends the "SJ" Switch Job command and includes a *Job Number* (located at a specified internal database address) as its only parameter. The vision sensor replies with an "RS" Response State packet containing the current status, which is then stored at a specified internal database address. If this service object is not configured for *Manual Trigger*, then a change of value of the job number itself triggers the service object.

Job Number Database Address

When the Switch Job command is sent, it includes the job number located at this database address in the payload.

Job Number Data Type

Fixed at 16-Bit Unsigned. The job number is a 16-bit unsigned value and therefore always occupies 16 bits (2 bytes) in the internal database.

Status Database Address

When the “RS” Response State packet is received, the current status in the payload is stored at this database address.

Status Data Type

Fixed at 16-Bit Unsigned. The current status is a 16-bit unsigned value and therefore always occupies 16 bits (2 bytes) in the internal database.

1.3.4 Teach Image Service Object

This service object sends the “TE” Teach Image command. This service object cannot be automatically triggered: it must be manually triggered, and the *Manual Trigger* element is therefore automatically included (refer to section 1.5.)

1.3.5 Trigger Image Service Object

This service object sends the “TR” Trigger Image command. This service object cannot be automatically triggered: it must be manually triggered, and the *Manual Trigger* element is therefore automatically included (refer to section 1.5.)

Response Expected

If the “RD” Response Data is expected as a result of the Trigger Image command, then enable this checkbox and include at least one *Response Data Object* to parse the response data. Otherwise, disable this checkbox.

1.3.6 VeriSens Reboot Service Object

This service object sends the “VB” VeriSens Reboot command and includes a reboot mode (fixed at a value of 0) as its only parameter. If this service object is not configured for *Manual Trigger*, then a change in the value located at the Reboot Mode Database Address triggers the service object.

Reboot Mode Database Address

Specifies a database location to be used as a trigger mechanism: a trigger occurs whenever the value changes. The value located at this database address is not used as the actual reboot mode in the VB command packet: the reboot mode value is fixed at 0 when the VB command is sent to the server.

Reboot Mode Data Type

Fixed at 16-Bit Unsigned. The reboot mode trigger is a 16-bit unsigned value and therefore always occupies 16 bits (2 bytes) in the internal database.

1.4 Response Data Object Settings

Separator-delimited “RD” response data packets can be parsed by sequentially-defined Response Data Objects. A Response Data Object must be added for each value returned in the RD response packet. The Response Data Objects must be added to the connection object according to the order in which the values are contained in the response packet (i.e. the position of a Response Data Object in the configuration must match the position of the corresponding value in the RD response packet). Up to 50 Response Data Objects can be added per connection object.

Description

This 32-character (max) field is strictly for user reference: it is not used at any time by the driver.

Database Address

Defines the database address where the value of this service object will be mapped. The configuration studio will not allow entry of a starting database address that will cause the value to run past the end of the database. The highest valid database address, therefore, will depend on the *Database Data Type*, as well as the *String Length* for String elements.

Element Data Type

Select the appropriate data type of the element. Note that binary data types are not supported. “Result” data type codes are converted to a hexadecimal value prior to being stored in the database: the result-to-value mapping is detailed in Table 1.

Table 1: Result-to-Value Mapping

Result Code	Value (Hexadecimal)
F	0x00
P	0x01
I	0x02
FA	0x80
PA	0x81
IA	0x82

Database Data Type

Specifies how many bytes are used to store values in the database, after they are converted from the *Element Data Type*. Select the desired data type from this dropdown menu.

String Length

Enabled only when Element Data Type is set to “String”. Defines the length of the string-encoded value (1…100 bytes).

Multiplier

Disabled when Element Data Type is set to “String”. The amount that associated values are scaled by prior to being stored into the database. Response values are divided by this multiplier before being stored into the database.

1.5 Manual Trigger

1.5.1 Overview

The processing action of every command service object is triggered to execute periodically by some form of stimulus. Once triggered, this processing action entails sending the command and processing incoming received data (where applicable). By default, the driver automatically provides the trigger stimulus for most command service objects based on expiration of the *Scan Rate* timer (note that there are several command service objects that do not support auto trigger). However, there may be situations where external (manual) control of this trigger stimulus is desirable in order to achieve a certain degree of control over a service object's behavior. This can be accomplished by adding an optional Manual Trigger element to the service object. Once added, a manual trigger can act as the service object's execution stimulus by manipulating a single bit in the internal database. Trigger bits can be manipulated either by actively injecting data into the database from a remote client via any supported server protocol, or by new data values being actively read into the database via service objects associated with a client protocol.

1.5.2 Manual Trigger Settings

Trigger Address

Specifies the database address that contains the byte-size *Trigger Bit* structure.

Trigger Bit

Specifies which bit in the byte designated by the *Trigger Address* is to be used as the trigger bit. Only one bit may be selected in the *Trigger Bit* structure, and a given *Trigger Address* may only be associated with one manual trigger.

1.5.3 Behavior

- When auto trigger is used, cyclic expiration of the *Scan Rate* timer causes the service object to unconditionally trigger.
- When manual trigger is used, cyclic expiration of the *Scan Rate* timer causes the service object to only evaluate whether or not it should trigger. This evaluation is performed by inspecting the state of the trigger bit and reacting to it as a "one-shot", meaning that when an external source sets the trigger bit to "1", then the corresponding service object is unconditionally triggered. Once the processing action has been completed (either successfully or unsuccessfully), the driver will update the database with the received data (where applicable), update the diagnostics object data structure (if implemented) to indicate the success or failure of the transaction, and then automatically clear the trigger bit. In this way, a remote device can both trigger an action, as well as be notified of the completion and resulting status by appropriately mapping the trigger bit, service object data, and diagnostics object structure into a block of data that is accessible via another network (such as a read/write service object controlled by a client driver on another network). If the service object evaluates its trigger bit to be "0", then it takes no further action: no command is transmitted to the server device.

Note that because the internal database is initialized to "0" values after every boot cycle, all defined manual trigger bits will cause their respective service objects to be disabled until explicitly enabled from an external source.

1.6 Diagnostics Object

Does not apply to Response Data Objects. Each command service object can optionally include a diagnostics object for debugging and diagnostics.

Diagnostics Database Address

Enter the database address at which to store the diagnostics information.