TECHNICAL NOTE

DATASENSING
easing automation challenges

SMART-VS SOFTWARE RELEASE 1.2.0

The Smart-VS have been upgraded to implement new communication protocol functionalities that will make your Smart-VS more flexible and easier about your machinery integration and interface. The new software includes a complete TCP/IP communication protocol which enables your device to get easily in communication with a PLC or Industrial PC client avoiding passing through the WEB Server GUI.
The TCP/IP communication requires that a connection be initiated between the two devices. The Smart-VS (server) is listening on port 1023 and the PLC or PC (clients), must send a connection request to establish communication. When the communication is established, commands can be sent to the device.

These new functionalities include the following:

- Create new jobs
- Change jobs
- Job modifies and handling
- Image banks modify and handling
- Device status monitoring
- Job training

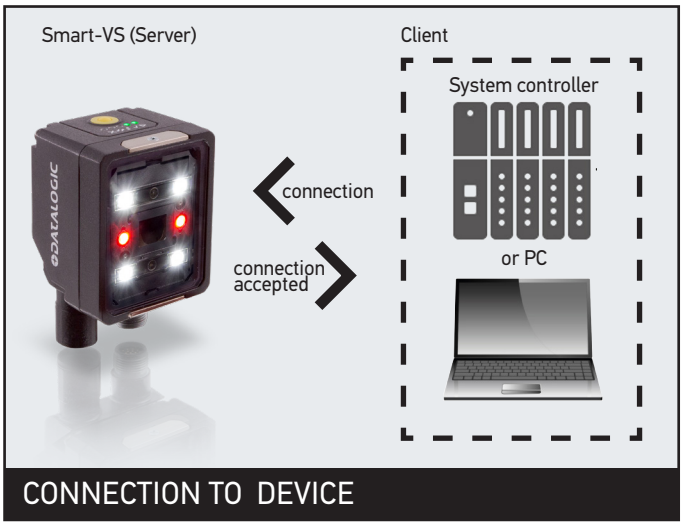
- AI enabled
- MLAS - Machine Learning Assisted Setting

NOTE: For a more detailed information on command and implementation example please download the reference manual at this link

The new firmware release R1.2.0 will be introduced on Smart-VS starting by March 2022 a further notice will be sent later reporting the first S/N with R1.2.0 loaded. The firmware will be also downloadable from Datasensing WEB site.

CONNECTION

The TCP/IP communication requires that a connection be initiated between the two devices. The Smart-VS (server) is listening on port 1023 and the PLC or PC (clients), must send a connection request to establish communication. When the communication is established, commands can be sent to the device.



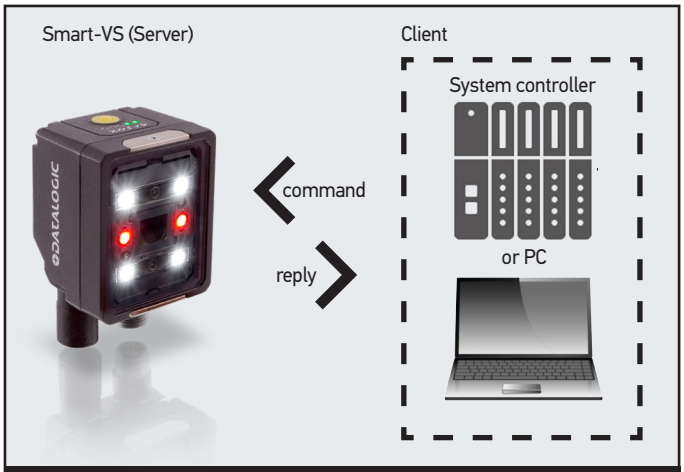
CONNECTION TO DEVICE

Product Marketing Manager: Alberto Fabbri
alberto.fabbri@datasensing.com
Sensor & Safety Application Engineer Manager: Luca Mazzetti
luca.mazzetti@datasensing.com
Customer Administration Manager: Raffaella Casini
raffaella.casini@datasensing.com

PROTOCOL OVERVIEW

The client sends a command to the Smart-VS and receives a reply with the result. The Smart-VS (server) is able to process only one command at a time. There are two types of commands:

1. Device control commands
2. Commands for Upload and Download of binary files



EXAMPLE OF COMMUNICATION

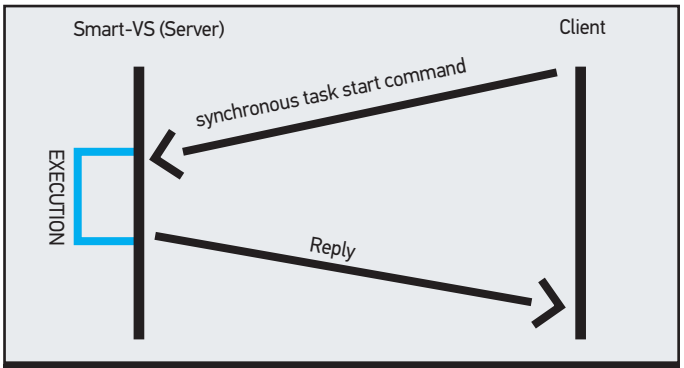
DESCRIPTION OF AVAILABLE FEATURES

There are two macro-groups of features:

1. Synchronous features: begins features that require short execution times. The result will be available directly in the reply.
2. Asynchronous features: begins features that take long execution times. The result must be requested once the completion of the task has been verified. In the event that the task is immediately finalized without previously verifying that the status has ended, the call will be blocked and the reply will be sent at the end of the task execution.

USE OF SYNCHRONOUS FEATURES

When a synchronous feature is performed, the reply is sent at the end of the execution. See example in the figure.



EXAMPLE OF USING A SYNCHRONOUS FEATURE

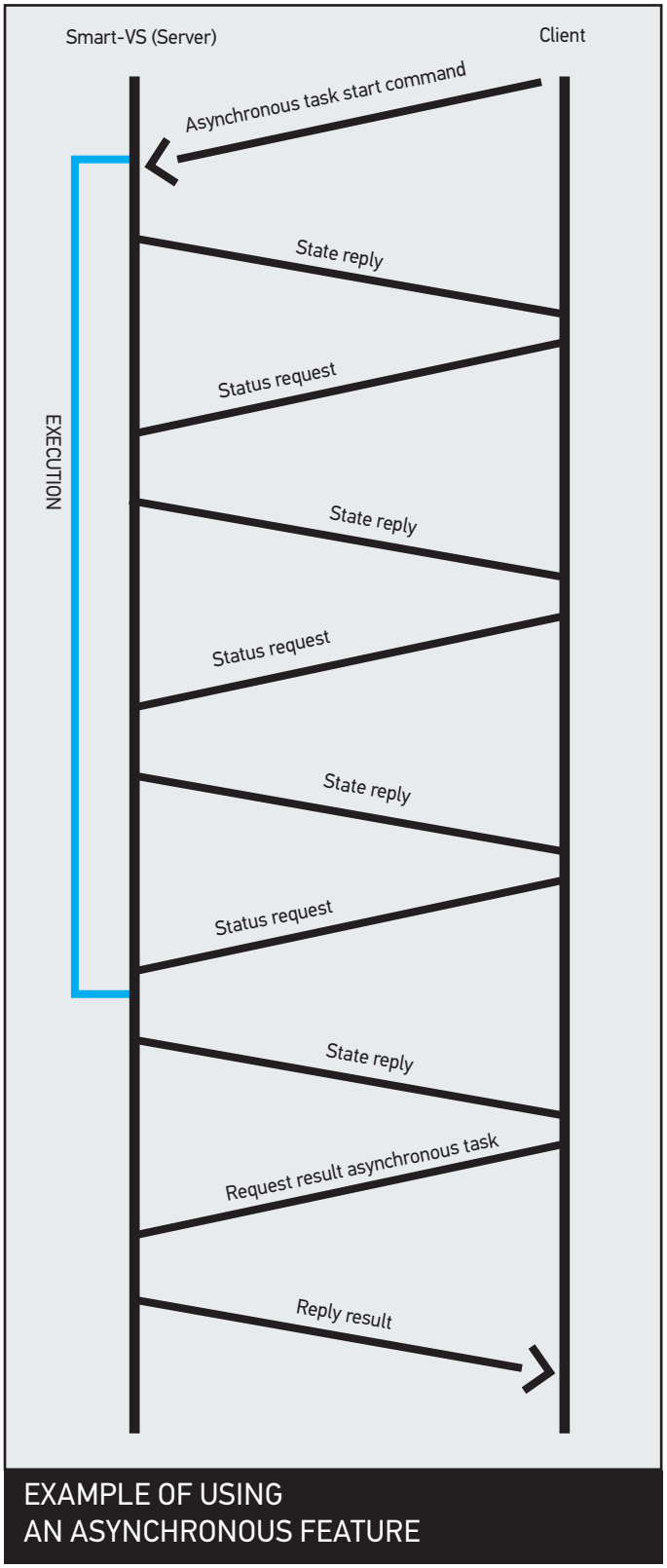
If the feature requires results to be sent, they will be contained in the reply

USE OF ASYNCHRONOUS FEATURES

An asynchronous feature must be started with the appropriate command. It will not be possible to run multiple asynchronous features at the same time. During the execution of the asynchronous feature, the device can be queried about the status of the activity. Once the activity is terminated, it will be possible for the client to request the finalization of the task, if provided, the finalization command will also contain the reply. It will not be possible to start another asynchronous task if the previously executed task is not terminated first.

SCENARIO 1

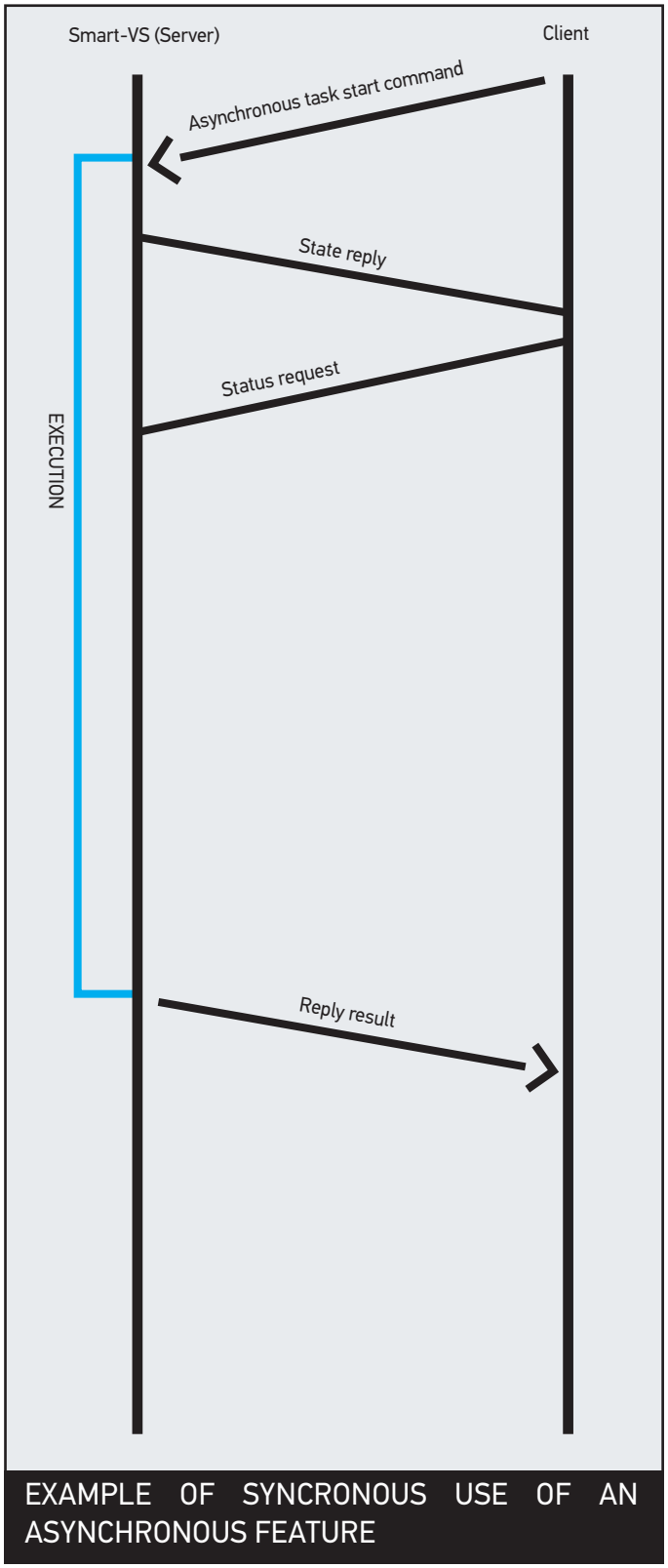
The client queries the device on the status of the task and, once the completion is veri¬fied, finalizes the task.



EXAMPLE OF USING
AN ASYNCHRONOUS FEATURE

SCENARIO 2

The client starts the task and immediately asks for its finalization. The reply will be sent at the end of the task.



EXAMPLE OF SYNCRONOUS USE OF AN ASYNCHRONOUS FEATURE

DELIMITERS AND SEPARATOR

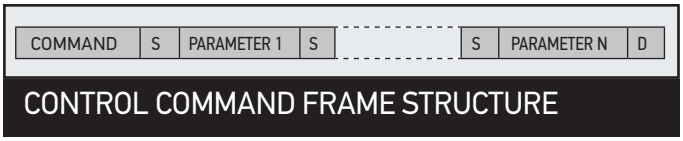
The grammar of the protocol provides a character used as a separator (S) of the fields within the message and a two-character sequence used as a delimiter (D) of the message.

	CHARACTER\S	DECIMAL VALUE	HEX VALUE
SEPARATOR	;	59	0x3B
DELIMITER	<CR><LF>	1 3 10	0X0D 0X0A

DEVICE CONTROL COMMANDS

These commands allow to remotely control some features of the device. They are encoded in ASCII with a defined grammar.

COMMAND FORMAT



CONTROL COMMAND FRAME STRUCTURE

- Command: Encoding containing the command
- S: Separator
- Parameter 1 N: Command parameter list. The number of parameters is variable.
- D: Delimeter

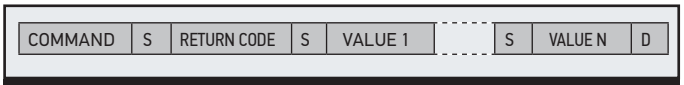
Example of command with one parameter.

EXAMPLE USING COMMAND WITH ONE PARAMETER.

Example of command without parameter

EXAMPLE OF USING COMMANDS WITHOUT PARAMETERS.

REPLY FORMAT



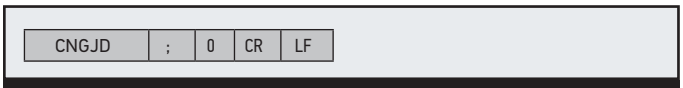
CONTROL COMMAND REPLY FRAME STRUCTURE

- Command: command to which the reply is related
- Return code: contains the result of the command, a code that identifies whether the command was accepted or not and identifies the type of error
- S: Separator
- Value 1-N: Return values of the command. It depends on the type of command
- D: Delimeter

Positive reply example with return values:

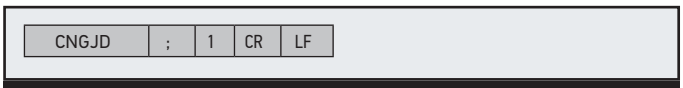
EXAMPLE OF REPLY TO A SUCCESSFUL COMMAND WITH RETURN VALUES.

Positive reply example with no return values



POSITIVE REPLY EXAMPLE WITH NO RETURN VALUES

Example of negative reply:



EXAMPLE OF A REPLY TO A FAILED COMMAND

LIST OF AVAILABLE DEVICE CONTROL FUNCTIONS

COMMAND	FUNCTION	PARAMETER	TYPE
CRTJB	Create a new job	Bank number to be configured, name to be assigned to the job	asynchronous
MDFJB	Modify a job	Bank number to be changed	synchronous
EXTJB	Exit the configuration without saving		synchronous
TRNJB	Train job		aynchronous
CNGJB	Change job	Bank number to load	synchronous
BKNST	Bank status		synchronous
GTRJB	Current Bank	See here	synchronous
GTDVCS	Device Status		synchronous
ACQIMG	Reference image acquisition	Class of the image	synchronous
CLRBNK	Clear Bank	Bank number	synchronous
CLRJBS	Clear Jobs		synchronous
GTATS	Get Async Task Status		synchronous
FNZJB	Finalize job creation		synchronous
FNZTRN	Finalize train		synchronous

Device control command list

RETURN CODES

The return code is contained in all replies as a parameter following the name of the command to which the reply is related. The return code allows to understand if the command was successful or rejected.

CODE	NAME	DESCRIPTION
0	Success	The command was successful
1	NotInSession	The device is running and the command cannot be executed
2	Failed	Generic command failure
3	-	(Reserved)
4	NotInJobEditing	The creation or a modification of a job has not be started
5	-	(Reserved)
6	OtherInProgress	Another asynchronous task is in progress or an asynchronous task has not been finalized, therefore is not possible to start a new one
7	-	(Reserved)
8	Invalidinput	Invalid input
9	-	(Reserved)
10	AlreadyInConfiguration	A configuration has already been started, it is not possible to start a new one before closing the previous one
11	MaxNumberOfImage	Maximum number of images reached
12	NotInProgress	No Asynchronous task running
13	Protocol error	Syntax error in protocol
14	UnknownMethod	Call unaivalable (call name error)
99	Not relevant	Parameter not relevant

Possible return code in reply.