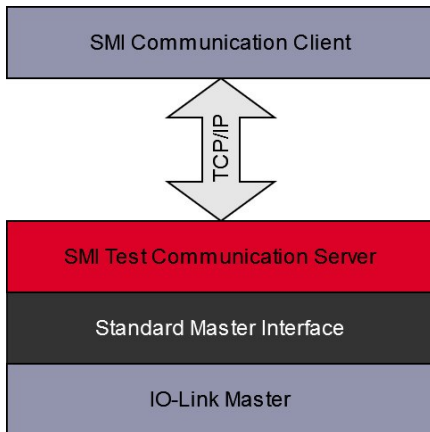




TEConcept SMI Communication Client



```
C:\Windows\System32\cmd.exe - python command_line_tool.py
Welcome to TEConcept SMI Command Line Interface!
(SMI-client)# connect 127.0.0.1 49852 1
Connection established with 127.0.0.1:49852
(SMI-client)# portconfiguration 1 IOL_AUTOSTART
(SMI-client)#
[+] 1 0xFFFF0
(SMI-client)#
[+] 1 0xA001 0xFF26 Event single shot. Notification. Master/Port. Port status changed.
(SMI-client)#
[+] 1 0xA001 0x1FF0 Event disappears. Error. Master/Port. Vendor specific.
(SMI-client)#
[+] 1 0xA001 0x1800 Event appears. Error. Master/Port. No Device (communication).
(SMI-client)#
[+] 1 0xA001 0xFF26 Event single shot. Notification. Master/Port. Port status changed.
(SMI-client)#
```

Overview

The TEConcept SMI Communication Client “SCC” – written in Python – provides access and control to any V1.1.3 compatible IO-Link Master which can be controlled via the SMI Communication Test Server (STCS) as specified in the IO-Link V1.1.3 test specification. The SMI communication client runs on any desktop or embedded Linux or Windows systems with Python 3 support.

Description

The SCC can be used for easy interaction with IO-Link Masters. The SMI services are executed serially in a request – response style. It is useful for debugging purposes and for automated test systems. The SCC provides a command line interpreter for interactive operation and for scripted operation based on customer specific Python scripts.

Features

- Interactive Command Line Interface
- Simple, text-based commands
- Command history
- Detailed SMI Service Arguments interpretation
- TCP/IP socket connection
- Automatic reconnection
- Support of “raw” SMI Service Octetstrings (for vendor specific ArgBlocks)
- Simultaneous interaction with multiple Masters

Advantages

- Useable from remote location (over TCP/IP socket)
- Hides communication complexities
- Suitable for analysing the SMI Service Arguments and ArgBlocks.
- An STCS server that connects to TEConcept IO-Link Masters via USB, SPI, UART etc. is provided together with the SCC.

Optional

- Safety extension
- Vendor specific extensions

Delivery

- Manual with detailed examples
- STCS binary (Windows or Linux)
- Python SCC package