

# **CyFlex® Device Communication User Guide**

Version 6

January 22, 2024

**Developed by Transportation Laboratories** 



### **Version History**

| Version Date Revision Description |           | Revision Description   |
|-----------------------------------|-----------|--|
| 1                                 | 5/10/2016 | Initial publication  |
| 2                                 | 8/23/2018 | <ul> <li>Formatted with SGS brand</li> <li>Revised the following sections: <ul> <li>1.3. The DevCom Subsystem – added device driver (dc_GenStream)</li> <li>2.2. Procedure Steps – removed notice about device readme files, which were not implemented</li> <li>2.3.4.2. Limit Report – revised table headings</li> <li>2.3.5 Error Log – revised table headings</li> <li>2.3.6 Trace Log – revised table headings</li> <li>2.3.7. Linux Diagnostics Command – revised command to display communication read from and written to the test device</li> <li>2.3.8. Linux minicom – revised description and added basic steps for configuring minicom and testing communication over a serial line.</li> <li>3.3. Collector (devcom_coll) – revised statement about gp_test using the keyword @DEVCOM to start the task.</li> <li>3.3. Collector (devcom_coll), and 3.4 Monitor (devcom_mon) – revised statement about gp_test using the keyword @DEVCOM to start the task.</li> <li>3.4. Monitor (devcom_mon) – revised statement to say that gp_test must start the task using the keyword @BACKGROUND. The program gp_test does not start it automatically because of keywords @DEVCOM or @DEVCOM_ACTIONS.</li> <li>4.1. AK Protocol (dc_AKg) – added basic info about TCP/IP, Ethternet and serial connections (RS232, RS485).</li> <li>4.2. AK Protocol (dc_AKgm) – added type of emissions analyzer at Fridley that dc_AKgm communicates with.</li> <li>5. Example Files – <ul> <li>Changed font to New Courier for column alignment</li> <li>Corrected flow diagram to show device driver name and to fit landscape page orientation.</li> <li>Added note that debug in the command line turns on the same debug flag that is turned on if debug is true in the specification file. The spec file does not override the command.</li> </ul> </li> </ul></li></ul> |



| Version | Date      | Revision Description  |  |
|---------|-----------|---|--|
|         |           | <ul> <li>start event, stop event or ms", with "# event or ms, device, command string, start event, stop event"</li> <li>Replaced the Monitor spec file for GasMuxMon with one for an AVL415</li> </ul>      |  |
| 3       | 4/7/2020  | Retrofit to new template  |  |
|         |           | Revised Section 6.1.6 Query Commands on page 61 to add additional warnings about the format of optional returns to query commands.  |  |
| 4       | 12/3/2021 | Revised Section 3 DevCom Applications on page 14 and<br>Section 4 DevCom Device Drivers on page 18 to remove inline<br>usage content and add hypertext linked cross-references to<br>cyflex.com usage help. |  |
| 5       | 6/7/2022  | Updated all hypertext linked cross-references to cyflex.com usage help descriptions   |  |
| 6       | 1/22/2024 | Rebrand to TRP Laboratories<br>Added Section 2.3.7.1 Fixing a Device Communication Error<br>on page 9   |  |

#### **Document Conventions**

This document uses the following typographic and syntax conventions.

 Commands, command options, file names or any user-entered input appear in Courier type. Variables appear in Courier italic type.

Example: Select the cmdapp-relVersion-buildVersion.zip file....

• User interface elements, such as field names, button names, menus, menu commands, and items in clickable dropdown lists, appear in Arial bold type.

Example: **Type**: Click **Select Type** to display drop-down menu options.

- Cross-references are designated in Arial italics.
   Example: Refer to *Figure 1...*
- Click intra-document cross-references and page references to display the stated destination.

Example: Refer to Section 1 Overview on page 1.

The clickable cross-references in the preceding example are 1, Overview, and on page 1.

#### **CyFlex Documentation**

CyFlex documentation is available at <u>https://cyflex.com/</u>. View **Help & Docs** topics or use the **Search** facility to find topics of interest.



## **Table of Contents**

| 1 | ٥V  | /ERVIEW1 |   |    |  |
|---|-----|----------|---|----|--|
|   | 1.1 | Dev      | COM SUBSYSTEM                                 | 1  |  |
| 2 | US  | ING I    | DEVCOM  | 2  |  |
|   | 2.1 | TES      |   | 2  |  |
|   | 2.2 | Dev      | Com Procedure                                 | 2  |  |
|   | 2.3 | TRC      | DUBLESHOOTING A DEVCOM SETUP                  | 5  |  |
|   | 2.3 | .1       | Equipment Connections                         | 5  |  |
|   | 2.3 | .2       | Test Device Errors                            | 5  |  |
|   | 2.3 | .3       | Test Device Specification File                | 5  |  |
|   | 2.3 | .4       | Failed gp_test                                | 5  |  |
|   | 2.3 | .5       | Error Log                                     | 7  |  |
|   | 2.3 | .6       | Trace Log                                     | 7  |  |
|   | 2.3 | .7       | Linux Diagnostics Command                     | 8  |  |
|   | 2.3 | .8       | Linux Minicom                                 | 10 |  |
| 3 | DE  | VCO      | M APPLICATIONS                                | 14 |  |
|   | 3.1 | Gen      | NERAL PURPOSE TEST MANAGER                    | 14 |  |
|   | 3.1 | .1       | Procedure Files and Test Modes                | 14 |  |
|   | 3.1 | .2       | Keywords                                      | 14 |  |
|   | 3.1 | .3       | Variables                                     | 15 |  |
|   | 3.2 | TES      | ۲   | 16 |  |
|   | 3.3 | COL      | LECTOR  | 16 |  |
|   | 3.4 | Mon      | NITOR   | 17 |  |
| 4 | DE  | VCO      | M DEVICE DRIVERS                              | 18 |  |
|   | 4.1 | AK       | PROTOCOL DC_AKG                               | 18 |  |
|   | 4.2 | AK       | PROTOCOL DC_AKGM                              | 18 |  |
|   | 4.3 | Gen      | NERAL SYNCHRONOUS COMMUNICATION DC_GENSYNC    | 19 |  |
|   | 4.4 | Gen      | NERIC ASYNCHRONOUS COMMUNICATION DC_GENSTREAM | 19 |  |
| 5 | EX  |          | LE FILES                                      | 20 |  |
|   | 5.1 | Sta      | RTUP FILE                                     | 21 |  |
|   | 5.2 | TES      | T MANAGER PROCEDURE FILE                      | 22 |  |
|   | 5.3 | Ινιτ     | IALIZATION FILE                               | 25 |  |
|   | 5.4 | COL      | LECTOR FILE                                   | 32 |  |
|   | 5.5 | Mor      | NITOR SPECIFICATION FILE                      | 43 |  |
|   | 5.6 | AKe      | G SPECIFICATION FILE                          | 44 |  |



|   | 5.7  | AKG  | M SPECIFICATION FILE               |
|---|------|------|------------------------------------|
|   | 5.8  | Gen  | SYNC SPECIFICATION FILE            |
| 6 | SUF  | PPOF | RTED PROTOCOLS                     |
|   | 6.1  | Gen  | IERIC COMMUNICATION AK PROTOCOL55  |
|   | 6.1. | 1    | Command Format55                   |
|   | 6.1. | 2    | Protocol Framework                 |
|   | 6.1. | 3    | Command Types                      |
|   | 6.1. | 4    | Control Commands                   |
|   | 6.1. | 5    | Setting Commands                   |
|   | 6.1. | 6    | Query Commands61                   |
|   | 6.1. | 7    | Example70                          |
|   | 6.2  | Gen  | IERAL ASCII SYNCHRONOUS PROTOCOL70 |



## List of Figures

| FIGURE 1: DEVCOM COMMUNICATION FILES           | 2  |
|--|----|
| FIGURE 2: SAMPLE GP_TEST STATUS OUTPUT         | 3  |
| FIGURE 3: SAMPLE TEST VARIABLES OUTPUT         | 4  |
| FIGURE 4: EXAMPLE TRACE OUTPUT FILE            | 8  |
| FIGURE 5: PID AND TEST DEVICE OUTPUT           | 9  |
| FIGURE 6: ADDITIONAL STRACE OUTPUT             | 9  |
| FIGURE 7: MINICOM CONFIGURATION MENU           | 10 |
| FIGURE 8: SERIAL PORT SETUP OPTIONS            | 11 |
| FIGURE 9: MINICOM WELCOME SCREEN               | 12 |
| FIGURE 10: MINICOM COMMAND SUMMARY             | 12 |
| FIGURE 11: CYFLEX SYSTEM OVERVIEW USING DEVCOM | 20 |



## LIST OF TABLES

| TABLE 1: ON-SCREEN TERMINATION INDICATORS            | 6  |
|--|----|
| TABLE 2: ERROR LOG METHODS                           | 7  |
| TABLE 3: DEVICE COMMUNICATION COMMANDS AND RESPONSES | 13 |
| TABLE 4: TEST DEVICES, PROTOCOLS, AND REQUIRED FILES | 54 |
| TABLE 5: COMMAND MESSAGE PROTOCOL                    | 55 |
| TABLE 6: ACKNOWLEDGMENT MESSAGE PROTOCOL             | 56 |
| TABLE 7: COMMAND GROUPS                              | 57 |
| TABLE 8: CONTROL COMMANDS                            | 57 |
| TABLE 9: SETTING COMMANDS                            | 59 |
| TABLE 10: QUERY COMMANDS                             | 62 |
| TABLE 11: EXAMPLE MEASUREMENTS                       | 70 |



## 1 Overview

Device Communication (DevCom) is a subsystem of CyFlex, and is a collection of applications, device drivers and user-configurable specification files. DevCom communicates with and controls Intelligent Electronic Devices (IEDs) that support a serial communications protocol. Smoke meters are an example.

This document describes DevCom and how to use it.

## 1.1 DevCom Subsystem

Communication can be customized with a certain device by modifying the specification file to work with the device's attributes without having to develop a unique software application for the device.

Before editing a specification file, understand the device's communication protocol. Refer to *Section 6 Supported Protocols* on page 54. The required editing may be simple and to only a small subset of the device's capabilities or the file may require a complicated configuration to manage a complex device.

The DevCom applications are:

- Collector (devcom\_coll)
- Monitor (devcom\_mon)
- Test Manager (gp\_test)
- Test (devcom\_smsg\_tst)

Refer to Section 3 DevCom Applications on page 14.

DevCom device drivers include:

- AK protocol (dc\_AKg)
- Generic Synchronous Communication (dc\_GenSync)
- Generic Asynchronus Communication (dc\_GenStream)

Refer to Section 4 DevCom Device Drivers on page 18.

DevCom applications, device drivers and examples of specification files are separately in this document.

#### ØNote:

A DevCom device driver can be customized by TRP Labs per special agreement with the customer for a unique customer application.



## 2 Using DevCom

DevCom applications and device drivers are launched from CyFlex.

## 2.1 Test Device

The example test device used in this document is an AVL smoke meter which measures the soot content of exhaust from a diesel engine.

## 2.2 DevCom Procedure

- 1. Connect the smoke meter. The smoke meter connects to a power supply, the sampling hose/probe, any accessories, and the test cell computer by a serial port. See the smoke meter manufacturer's manual for connections, warnings, specifications, parameters, calibration, and operating instructions.
- 2. Switch on the smoke meter per the manufacturer's instructions.
- 3. If the test cell computer is not already on, switch it on. This automatically launches the Goscript. The startup file can be run from the Go-script or from the command line.
- 4. DevCom uses the files shown in *Figure 1* to communicate with the example AVL 415 smoke meter test device. This is the first screen to appear.

Figure 1: DevCom Communication Files

```
File Edit View Search Terminal Help
[tc91@cmx-091 specs]$ cd cmds
[tc91@cmx-091 cmds]$ ls start_415
start_415
[tc91@cmx-091 cmds]$ cd ../
[tc91@cmx-091 specs]$ ls *4156
ls: cannot access *4156: No such file or directory
[tc91@cmx-091 specs]$ ls *4156*
AVL4156_mon.spec AVL4156.spec
[tc91@cmx-091 gp]$ ls *4156*
gp_4156 gp_4156_collect gp_4156_error gp_4156_init
[tc91@cmx-091 gp]$
```

5. Enter less *filename* at the command line to view the contents of a file. For example, to view the contents of the smoke meter initialization file listed in the preceding screen, enter:

\$ less gp\_415G\_init

Refer to Section 5 Example Files on page 20 for file examples including gp\_415G\_init.

6. A test cell typically runs separate tests, taking samples over a period of time. Enter the following command to view the status of the currently running test procedures:

\$ dump\_it TEST\_DESC

Figure 2 on page 3 shows a sample status.



#### Figure 2: Sample gp\_test Status Output

| 📮 tc5@cnsnode5:/specs/gp - Shell - Konsole                                  |
|---|
| Session Edit View Bookmarks Settings Help                                   |
| DISPLAY STRING VARIABLES  |
| 0 TEST DESC "gp idle,95   lower limits are enabled - Engine Running         |
| 1 TEST DESC 2 "   |
| 2 TEST DESC 3 "gp 415G,4   Ready - Waiting for events SM init or SM collect |
| 3 TEST DESC 4 "   |
| 4 TEST DESC 5 "   |
| 5 TEST DESC 6 "   |
| 6 TEST DESC 7 "   |
| 7 TEST DESC 8 "gp test failed in reading the furball file.                  |
| (END)   |
| Shell   |

7. Enter the following command to view the test variables:

## \$ dump\_it SM

Figure 3 on page 4 shows a sample variables list.



#### Figure 3: Sample Test Variables Output

| 💭 tc5@  | cnsnode5:/spec    | s/gp - Shell - Kons | ole      |        |                   |
|---------|-------------------|---------------------|----------|--------|-------------------|
| Session | Edit View Bo      | okmarks Settings    | Help     |        |                   |
| GENERAL | USE REAL VARIA    | ABLES               |          |        |                   |
| index   | label             | value               | units h  | istory | /tol owner status |
| 230     | SMPaperLeft       | 1391211340          | sec      | OFF    | 1.00 GL_SM415     |
| 232     | SMSootMn          | 8.2500              | mg/m3    | OFF    | 1.00 GL_SM415     |
| 233     | SMSoot1           | -99000000.0000      | mg/m3    | OFF    | 1.00 GL SM415     |
| 234     | SMSoot2           | -99000000.0000      | mg/m3    | OFF    | 1.00 GL_SM415     |
| 235     | SMSoot3           | -99000000.0000      | mg/m3    | OFF    | 1.00 GL_SM415     |
| 236     | SMSoot4           | -99000000.0000      | mg/m3    | OFF    | 1.00 GL_SM415     |
| 237     | SMSoot5           | -99000000.0000      | mg/m3    | OFF    | 1.00 GL_SM415     |
| 238     | SMSmKMN<br>SMSmk1 | 0.5330              | fen      | OFF    | 1.00 GL_SM415     |
| 239     | SMSmk2            | -99.0000            | fsn      | OFF    | 1.00 GL_SH415     |
| 241     | SMSmk3            | -99.0000            | fsn      | OFF    | 1.00 GL SM415     |
| 242     | SMSmk4            | -99.0000            | fsn      | OFF    | 1.00 GL_SM415     |
| 243     | SMSmk5            | -99.0000            | fsn      | OFF    | 1.00 GL_SM415     |
| 244     | SMBlkPcnt         | -99                 | 26       | OFF    | 1.00 GL_SM415     |
| 245     | SMWhtVal          | -99.0000            | none     | OFF    | 1.00 GL_SM415     |
| 246     | SMGreyVal         | -99.0000            | none     | OFF    | 1.00 GL_SM415     |
| 247     | SMBlkVal          | -99.0000            | none     | OFF    | 1.00 GL_SM415     |
| 248     | SMEffVol          | -99000000.0000      | cm3      | OFF    | 1.00 GL_SM415     |
| 249     | SMEffLen          | -99000.0000         | mm       | OFF    | 1.00 GL_SM415     |
| 250     | SMDelPdig         | -99.0000            | none     | OFF    | 1.00 GL_SM415     |
| 251     | SMDelPmbar        | -1                  | mbar     | OFF    | 1.00 GL_SM415     |
| 252     | SMRelPdig         | -99                 | none     | OFF    | 1.00 GL_SM415     |
| 253     | SMRelPmbar        | -1                  | mbar     | OFF    | 1.00 GL_SM415     |
| 254     | SMTemp            | 1                   | deg_C    | OFF    | 1.00 GL_SM415     |
| 255     | SMSupVolt         | - 99                | volt     | OFF    | 1.00 GL_SM415     |
| 256     | SMDeadVol         | Θ                   | ml       | OFF    | 1.00 GL_SM415     |
| 257     | SMAmbPress        | 0                   | mbar     | OFF    | 1.00 GL_SM415     |
| 258     | SMTempCor         | 0                   | none     | OFF    | 1.00 GL_SM415     |
| 259     | SMVolCor          | 0                   | none     | OFF    | 1.00 GL_SM415     |
| 260     | SMMinSampVol      | 0                   | ml       | OFF    | 1.00 GL_SM415     |
| 261     | SMMaxSampVol      | Θ                   | ml       | OFF    | 1.00 GL_SM415     |
| 262     | SMPurgeTime       | Θ                   | sec      | OFF    | 1.00 GL_SM415     |
| 263     | SMTrigDelay       | Θ                   | sec      | OFF    | 1.00 GL_SM415     |
| 264     | SMTolerance       | Θ                   | fsn      | OFF    | 1.00 GL_SM415     |
| 265     | SMPrintMode       | Θ                   | none     | OFF    | 1.00 GL_SM415     |
| 266     | SMPrintLang       | Θ                   | none     | OFF    | 1.00 GL_SM415     |
| 267     | SMIPapBlk         | Θ                   | none     | OFF    | 1.00 GL_SM415     |
| 268     | SMIWhtVal         | Θ                   | none     | OFF    | 1.00 GL_SM415     |
| 269     | SMIGreyVal        | 0                   | none     | OFF    | 1.00 GL_SM415     |
| 270     | SMIBLKVal         | 0                   | none     | OFF    | 1.00 GL_SM415     |
| 2/1     | SMIVOLStd         | 0                   | mu       | OFF    | 1.00 GL_SM415     |
| 272     | SMIVOLPAR         | 0                   | ml       | OFF    | 1.00 GL_5M415     |
| 273     | SMILETTLEN        | 0                   | mm<br>ml | OFF    | 1.00 GL_5H415     |
| 274     | SMILeakage        | 70.10               | mt       | 055    | 1.00 GL_5H415     |
| 275     | SMULT             | /0.10               | C C      | OFF    | 1.00 GL_5H415     |
| 270     | SMALET            | 63 10               | C        | OFF    | 1.00 GL_5H415     |
|         | SUMTLI            | 05.10               | C        | OFF    | 1.00 GL_30413     |
|         |                   |                     |          |        |                   |
|         | Shell             |                     |          |        |                   |
|         |                   |                     |          |        |                   |



## 2.3 Troubleshooting a DevCom Setup

Problems with a test setup may result from any of the following:

- Equipment connections or cables/hoses
- Test device error or malfunction
- Application, device driver or specification file(s)
- Parameter(s) limits exceeded

Refer to the following sub-sections if the setup does not properly work.

#### 2.3.1 Equipment Connections

Check the following:

- All hardware connections are seated.
- Cables and hoses are not kinked, etc.

#### 2.3.2 Test Device Errors

If the test device displays an error message or appears to be malfunctioning, consult the manufacturer's manual for a description of error messages and recommended actions.

#### 2.3.3 Test Device Specification File

Confirm the serial port of the test device is correctly identified in the device's specification file.

View the test device's specification file to confirm the correct serial port is identified.

In Section 5 Example Files on page 20, the test device specification file for the smoke meter is used by the initialization file and Collector file. The example spec file name is AVL415G.spec. The smoke meter is connected using serial port 2:

\$Device
/dev/ser2:9600,8,1,N
# 192.168.222.200:5304

### 2.3.4 Failed gp\_test

A gp\_test can abnormally terminate for various reasons. The following are possible indicators:

- Indicators and messages on-screen
- Limit Report
- Error Log
- Trace Log

5



## 2.3.4.1 *On-Screen Information*

Displayed messages and indicators may help explain why a test was terminated. Refer to *Table 1*.

| Indicator                | Description   |  |  |
|--------------------------|---|--|--|
| Blinking variable        | A blinking value means that a limit associated with a variable has been tripped.                                    |  |  |
| Active test mode         | The active mode will show the latest mode executed by the <code>gp_test</code> scheduler.                           |  |  |
| Previous test modes      | The previous modes will show which <i>gp_test</i> modes were executed before the active mode.                       |  |  |
| Notification<br>messages | The NOTIFY message indicates the most recent message spawned from the Event_Response test or from the command line. |  |  |

#### Table 1: On-Screen Termination Indicators

## 2.3.4.2 *Limit Report Utility*

The limit report utility tracks exceeded (tripped) variable limits and records them in a memory resident utility. The report remains active until the operator enters a reset or Go command.

Enter the following command to produce an active limit report that shows variable limit violations:

```
$ limit_rpt
```

If the limit report utility does not show a limit violation that may have caused the gp\_test failure, check the limit\_rpt file. The file may contain additional tripped variable limits.

Perform the following:

1. Change directory:

```
$ cd data/errors
```

2. Enter:

```
$ edit limit_rpt
```

The edit limit\_rpt command produces an output file with the following naming convention:

limit\_rpt.YYMMDD where YYMMDD is the date in year, month and day format.

The file lists all variable limit violations.

#### ØNote:

Recent limit violations are more likely than older ones to have contributed to a test failure

## 2.3.4.3 Error Log and Trace List

Events leading to termination of a test normally generate entries in the error log. Trace logs may also provide insight as to why a test failed. Refer to the following:

- Section 2.3.5 Error Log on page 7
- Section 2.3.6 Trace Log on page 7

6



## 2.3.5 Error Log

Errors are logged to error files during initialization. Errors are normally logged to both the error\_log file and to an application-specific file. However, runtime errors are typically logged only to the application-specific error log file. *Table 2* lists the methods to produce error logs, associated purpose, and outputs.

| Command  | Purpose   | Output File                    |  |
|--|---|--------------------------------|--|
| \$ errs  | Lists errors with dates and times, except application-specific runtime errors | /data/errors/error.log         |  |
| \$ errc  | Displays general error code information                                       | /cyflex/inc/errors.h           |  |
| <pre>\$ less dc_xxx.MMDD   dc_xxx is the name of the   application, such as   gp_test.</pre> | Displays errors for the application with dates and times                      | Application-specific error log |  |

#### Table 2: Error Log Methods

## 2.3.6 Trace Log

The trace log contains entries generated by the  $gp\_test$  trace feature. These entries contain useful information, such as when a test was started and stopped, and events that may have spawned a new test.

Perform the following:

- 1. Change directory:
  - \$ cd specs/gp
- 2. Enter:
  - \$ less /specs/gp/TRACE\_smk

The command produces the TRACE\_smk file that contains the dates and times entries occurred.

*Figure 4* on page 8 is an example of the trace feature showing the gp\_415\_init initialization procedure file commands.



Figure 4: Example Trace Output File

```
T/D-16:23:44 02/03/14
  PROCEDURE called - gp 415G init
gp_415G_init,2 | Set the SM_init_strt event to indicate initialization started
T/D-16:23:49 02/03/14
  timeout - 5.0[sec]
gp 415G init,4 | Clear the buffer and reset error parameters
T/D-16:23:49 02/03/14
  immediate -
gp 415G init,6 | wait for reset to complete
T/D-16:23:54 02/03/14
  timeout - 5.0[sec]
gp_415G_init,8 | set 415 into remote
T/D-16:23:54 02/03/14
  DEVCOM command -
gp_415G_init,10 | set 415 into remote mode
T/D-16:23:54 02/03/14
  DEVCOM command -
gp 415G init,12 | SRDY to clear error
T/D-16:23:54 02/03/14
   - RETURN to /specs/gp/gp_415G
/gp 415G init
T/D-16:23:54 02/03/14

    RETURN to /specs/gp/gp_415G

  DEVCOM command -
gp_415G,4 | Ready - Waiting for events SM_init or SM_collect
T/D-16:24:51 02/03/14
  TERMINATION EVENT - SM collect
gp_415G_collect,2 | Set SM_coll_strt event to indicate meas seq started.
T/D-16:24:51 02/03/14
  immediate -
gp_415G_collect,4 | Send "SPUL" and purge for 10 sec.
(END)
```

## 2.3.7 Linux Diagnostics Command

The Linux strace command displays information that can help diagnose a problem. It reveals activity such as calls within a procedure or communication with devices that are useful, for example, if the application calls a file that does not exist.

In the example below, strace is used to show communication between a DevCom device driver and a test device. This requires knowing the process identification number (PID) of the driver. To determine this, at the command line enter:

\$ sin -P dc\_xxx

where dc\_xxx is the DevCom device driver name, such as dc\_AKg

The executed command produces the example shown in *Figure 5* on page 9.

8



#### Figure 5: PID and Test Device Output

| 营 tc91@cmx-091:/specs/gp - Shell - Konso  | ple         |
|---|-------------|
| Session Edit View Bookmarks Settings  | Help        |
| <pre>[tc91@cmx-091 gp]\$ sin -P dc_AKg 11866 -16 dc_AKg /specs/AVL415G.spec 13809 20 /bin/bash /cvflex/cmds/sin</pre> | -P. dc. AKa |

The preceding output shows the device driver PID is 11866 and test device is an AVL 415G smokemeter.

Add an argument(s) to the strace command to specify which information to view. Include the PID. For example:

\$ strace -p 11866 -eread,write

Figure 6: Additional strace Output

| 🚽 tc91@cmx-091:/specs/gp - Shell - Konsole |                   |  |  |
|--|-------------------|--|--|
| Session Edit View Bookmarks                | Settings Help     |  |  |
| [tc91@cmx-091 gp]\$ strace -p 11           | .866 -eread,write |  |  |
| Process 11866 attached - interr            | upt to quit       |  |  |
| write(5, "\2 ASTZ\3", 7)                   | = 7               |  |  |
| read(5, "\2", 1)                           | = 1               |  |  |
| read(5, " ", 1)                            | = 1               |  |  |
| read(5, "A", 1)                            | = 1               |  |  |
| read(5, "S", 1)                            | = 1               |  |  |

Depending on the system, use the following methods to view all available strace arguments:

- \$ use strace
- \$ man strace

#### 2.3.7.1 *Fixing a Device Communication Error*

A response from strace that contains the characters ???? indicates that communications are not working properly.

Take the following actions to correct a device communication error response that contains the characters ????:

- Add stty -F <path to the file that corresponds with the device> -echo to the script used to start dc\_AKg as in the following example: stty -F /dev/rckt1 -echo
- 2. Add the user to the dialout group to avoid sudo in the start script for dc\_AKg as in the following example:

sudo usermod -a -G dialout <user>

Note: You can also put sudo chown in the rc.local file to avoid the use of sudo tty in the dc\_AKg start script.



## 2.3.8 Linux Minicom

The minicom utility communicates directly with a test device via a serial line. This program is useful for determining whether basic communication is working before having to troubleshoot the associated device specification file used by the DevCom driver.

Use the following procedure to communicate with a serial device using minicom:

- 1. Determine which serial port to use for the connection.
  - a. View serial ports on the machine:
    - \$ dmesg | grep tty
  - b. Note which serial port to use (for example, ttyS0).
- 2. Set up minicom:
  - \$ sudo minicom -s

The minicom configuration menu should appear as in Figure 7.

Figure 7: Minicom Configuration Menu



- a. If minicom does not start, locate the program using the which command:
  - \$ which minicom
- b. If the program is not found, install it:
  - \$ sudo
  - \$ yum install minicom

#### ØNote:

If entering sudo does not work, try switching to the root account using su and entering the root password.



- 3. Configure settings for the connection:
  - a. In the minicom configuration menu shown in *Figure* 7 on page 10, use the arrow keys to select the **Serial port setup** option and press Enter to display the following screen.

Figure 8: Serial Port Setup Options

| A - Serial Device<br>B - Lockfile Location<br>C - Callin Program<br>D - Callout Program<br>E - Bps/Par/Bits<br>F - Hardware Flow Control<br>G - Software Flow Control<br>Change which setting? | : /dev/ttyS0<br>: /var/lock<br>:<br>: 9600 8N1<br>: Yes<br>: No |
|--|---|
| +<br>  Screen and keyboar<br>  Save setup as dfl<br>  Save setup as<br>  Exit<br>  Exit from Minicom   | rd  <br> <br> <br> <br>   |

#### ØNote:

Pressing the Esc key returns the cursor to the previous selection.

- b. Edit the Serial Device if using a port other than the one displayed.
- c. Adjust communication parameters as necessary for the test device, referring to the manufacturer's documentation. Other parameters are available through the minicom configuration menu.
- 4. To save the settings as the default for future connections, choose the option **Save setup** as dfl. This saves the text file /etc/minirc.dfl.

An alternate method is to edit the .dfl file (or copy it from another test cell). For example:

a. In the text file, include the following lines:

```
# Machine-generated file - use <CNTL> + <A> + <Z> to
change parameters
menu 0 port /dev/ttyS0
```

ØNote:

The serial port named in the line above ( $tt_{YS0}$ ) is an example. Make sure the correct serial port is identified in the file.

- b. Save the file as: /etc/minirc.dfl
- c. Exit minicom:
  - \$ sudo exit minicom

#### ØNote:

In order for minicom to communicate with the test device, make sure there are no processes running on the serial port, including all



instances of minicom.

- 5. Start minicom:
  - \$ sudo minicom

The minicom welcome screen is displayed as in Figure 9:

Figure 9: Minicom Welcome Screen



- 6. Connect the serial port identified above to the serial cable/test device.
- 7. Confirm the test device is powered on.
- 8. Open the Minicom Command Summary menu to see options (CTRL-A-Z).

Figure 10: Minicom Command Summary

```
Minicom Command SummaryCommands can be called by CTRL-A <key>Main FunctionsOther FunctionsDialing directory..Drun script (Go)....G | Clear Screen.....CSend files.....SReceive files.....R | cOnfigure Minicom..0comm Parameters...PAdd linefeed.....A | Suspend minicom...JCapture on/off...LHangup.....H | eXit and reset....Xsend break.....Finitialize Modem...M | Quit with no reset.QTerminal settings.Trun Kermit......K | Cursor key mode...IlineWrap on/off...Wlocal Echo on/off..E | Help screen.....ZPaste file.....Yscroll Back....B
```



9. Send basic commands to the device to determine whether communication is working.

| Table 3: Device | Communication  | Commands a | nd Responses |
|-----------------|----------------|------------|--------------|
| 10010 01 201100 | •••••••••••••• | eennanae a | na neopeneee |

| Function Keys and<br>Commands            | Device Response  |
|--|--|
| <ctrl-b> ASTF <ctrl-c></ctrl-c></ctrl-b> | Current error code, or if no error: 0                  |
| <ctrl-b> ASTZ <ctrl-c></ctrl-c></ctrl-b> | Current status of device                               |
| <ctrl-b> APAP <ctrl-c></ctrl-c></ctrl-b> | Amount of paper left                                   |
| <ctrl-b> AEVL <ctrl-c></ctrl-c></ctrl-b> | Last effective volume and length measured              |
| <ctrl-b> SMES <ctrl-c></ctrl-c></ctrl-b> | Starts or continues with the measurement set with EMZY |

### ØNote:

For the commands in the preceding table, press and release CTRL-B, type the command (ASTF, for example), then press CTRL-C.



## **3 DevCom Applications**

This section describes the DevCom applications.

### 3.1 General Purpose Test Manager

Use gp\_test to enable the Test Manager application to manage parts of the general test process:

- Communicate with external test devices, such as smoke meters
- Control specialized measurement devices
- Communicate with a support application that can off-load parallel computations, for example, to determine whether testing variables are within limits
- Set operating points for closed loop controllers
- Start and stop data acquisition processes

#### 3.1.1 Procedure Files and Test Modes

Use Test Manager to design tests using a language written in text files called procedure files. This language is unique to CyFlex and Test Manager.

A procedure file contains the instructions for a test or part of a test. These instructions are organized into blocks of text called test modes.

Test modes describe:

- How the test object (usually an engine) is to be operated
- Testing data to be captured
- Optional features

Issue a command to start a certain procedure by using its file name. This initial procedure ("main" or "root" procedure) may call or jump to other procedures which can be thought of as sub-procedures.

Refer to Section 5 Example Files on page 20 for an example of a Test Manager procedure file.

#### 3.1.2 Keywords

Keywords within a procedure file determine how the test functions. When the Test Manager application reads a procedure file, it recognizes the keyword at the beginning of each test mode.

The keyword is a special character string calling for a specific operation, and all data following the keyword goes with that operation.

Each keyword:

- Begins with the @ character
- Is followed by a string, which must not contain any blanks
- Is the only information on that line
- Consists of all upper-case characters
- Must be spelled correctly to be recognized by Test Manager

The first keyword for each test mode is always @MODE. This signals the start of a new mode and the end of the previous mode specification.



Specification lines following the @MODE E keyword define:

- Mode number (between 1 and 999)
- Maximum time to spend in the mode
- Which mode to execute next
- Purpose of the test mode

#### Example:

```
@MODE
```

```
# mode_number max_time default_next_mode
```

```
110[sec]2
```

```
# description
```

This is mode number one.

The mode number) is a tag only and does not imply order or sequence. The modes may be numbered in any order, with control passing from one mode to another in any sequence. It is recommended, however, that user edits keep orderly numbering and sequencing so the file stays easy to understand.

#### ØNote:

Test Manager ignores any comment line. Each begins with the # symbol and helps explain for the user the specification line below the comment.

Refer to the <u>Test Manager User Guide</u> and cyflex.com usage help for <u>gp\_test</u> for additional information.

### 3.1.3 Variables

Variables are the basic means of identifying, characterizing, and using the measurements and parameters of the test object and its environment. Variables hold information needed throughout the test process, i.e., values for displaying, logging, computing, and controlling.

Variables describe values and how they are used. This includes descriptive labels, display format, and how the values are measured (unit of measure). Values can be integers, floating point numbers, logical (on/off) values, or text strings.

Fields in a procedure file (or in a sub-procedure file called by the procedure) specify values for use by the Test Manager. The instructions (test mode) for a particular keyword may indicate how a value is to be expressed – as a constant, variable label, or computed expression.

The example in *Section 5.4 Collector File* on page 32 specifies variables for the test device (AVL415G smoke meter) to measure and return.

```
@DEVCOM
# instrument name
AVL415G /specs/AVL415G.spec
# command key strings
# ASTZ,%s %s %s #%s #%s
"ASTZ SMsmode SMstate SMpapeco"
```

ASTZ is a query command, while the rest of the lines above describe the variables.

The comment line in the file # ASTZ, %s %s %s #%s, means the variables are text strings.

The next line tells the Collector task to return values to variables: SMsmode, SMstate, and SMpapeco.

## 3.2 Test

Test is a utility program that interfaces directly with and controls device drivers. This application is not to be confused with the Test Manager, which controls high level tasks such as calling procedures.

Communication: devcom\_smsg\_tst

**Command syntax:** Refer to cyflex.com usage help for devcom\_smsg\_tst.

Refer to Section 4 DevCom Device Drivers on page 18 and Section 6 Supported Protocols on page 54 for related information.

Example Test File:

```
Get device id, firmware rev to variable at string variable, and
serial number
> devcom_smsg_tst AVL483 "AKEN Deviceid @SringVar AKSerialN"
```

```
Send command to get device ID, but do not do anything with returned data > devcom_smsg_tst AVL483 "AKEN - - -"
```

```
Tell device driver task to exit, closing access to the AK device. > devcom_smsg_tst AVL483 exit
```

### 3.3 Collector

The Collector application provides synchronous communication to supported devices. The Collector interface controls the top-level task, while device drivers interface with the devices.

Communication: devcom\_coll

**Command syntax:** Refer to cyflex.com usage help for devcom\_coll.

Example Collector File:

devcom\_coll /specs/MON2.spec mydone\_event myconfig\_event &

ØNote:

This task is automatically started by gp\_test when the @DEVCOM keyword is used.



### 3.4 Monitor

The Monitor program provides event-driven or time-based polling of Collector (devcom\_coll) tasks.

Communication: devcom\_mon

**Command syntax:** Refer to cyflex.com usage help for devcom\_mon.

Example Monitor File:

devcom\_mon /specs/GasMux.mon &

#### ØNotes:

- This application is commonly started by gp\_test using the @BACKGROUND keyword. The gp\_test program will not start devcom\_mon using a @DEVCOM or @DEVCOM\_ACTIONS keyword.
- Use devcom\_smsg\_tst to test commands first that were added to the specification file.
- If any commands do not work using devcom\_smsg\_tst, they will not work with devcom\_mon.



## 4 DevCom Device Drivers

This section describes the DevCom device drivers.

## 4.1 AK Protocol dc\_AKg

Device driver dc\_AKg is for the Collector (devcom\_coll) tasks. The AK protocol is generic and AVL-compatible, and works with TCP/IP, Ethernet, or serial (RS-232, RS-485) based devices.

#### Communication: dc\_AKg

**Command syntax:** Refer to cyflex.com usage help for dc\_AKg.

#### ØNote:

This device driver is usually spawned by the Collector application and is similar to but differs from AK\_sync. Refer to cyflex.com usage help for AK\_sync.

TCP/IP (Transmission Control Protocol/Internet Protocol) can be used as a communications protocol in a private network (either an intranet or extranet).

Ethernet supports physical network architectures (such as hub and spoke topology) that use a central connecting point. A network hub is hardware that functions as a central hub to all nodes. Ethernet allows for transfer rates between 100Mb and 1Gb at distances up to approximately 300 ft.

RS-232 is point-to-point only and used for short distances, typically 65 ft or less. RS-485 supports daisy chain/multi-drop topology and is used for distances up to approximately 3,200 ft. Rates are affected by cable length, ranging from 1.5Mb at close distances to only hundreds of Kb at longer distances.

## 4.2 AK Protocol dc\_AKgm

The dc\_AKgm device driver task follows the AK protocol, but was modified to work with the unique response format of an emissions analyzer (Gas Mux) at a customer facility in Fridley, MN. This device driver is for Collector (devcom\_coll) tasks and works with either serial or TCP/IP based devices.

#### ØNote:

The emissions analyzer mentioned above is a HORIBA MEXA-6000FT, using the Fourier transform infrared (FTIR) method of measurement. Another example is the HORIBA MEXA-1170HFID.

#### Communication: dc\_AKgm

**Command syntax:** Refer to cyflex.com usage help for dc\_AKgm.

#### ØNote:

This task is usually spawned by the Collector (devcom\_coll) application



## 4.3 General Synchronous Communication dc\_GenSync

The dc\_GenSync device driver is for Collector (devcom\_coll) tasks and works only with synchronous communication (master-slave, command-response). This driver functions with either serial or socket-based devices.

**Communication:** dc\_GenSync

**Command syntax:** Refer to cyflex.com usage help for dc\_GenSync.

#### ØNotes:

- A default dc\_GenSync specification file named GS610.specs is available in the directory on the test cell / specs.def (specifications defaults).
- This device driver task (dc\_GenSync) is usually spawned by the Collector task.
- This driver does not support unsolicited or streamed data. dc\_GenSync does not support Cyclical Redundancy Code (CRC) calculations, which can be added as needed.

#### 4.4 Generic Asynchronous Communication dc\_GenStream

The dc\_GenStream driver is for Collector (devcom\_coll) tasks and functions with either serial or socket-based devices.

This driver works with either synchronous or asynchronous communication. However, if communication is synchronous only, it is recommended to use dc\_GenSync instead. The driver dc\_GenStream can support synchronous to start the stream if needed.

**Communication:** dc\_GenStream

**Command syntax:** Refer to cyflex.com usage help for dc\_GenStream.

#### ONOTES:

- A default dc\_GenStream specification file named hydrometer.spec is available in the test cell directory /specs.def.
- The device driver task (dc\_GenStream) is usually spawned by the Collector task.



## 5 Example Files

Example files in this section are for an AVL 415G smoke meter, and include the file types below:

- Startup; refer to Section 5.1 Startup File on page 21
- Test Manager Procedure; refer to Section 5.2 Test Manager Procedure File on page 22
- Initialization; refer to Section 5.3 Initialization File on page 25
- Collector; refer to Section 5.4 Collector File on page 32
- Monitor Specification; refer to Section 5.5 Monitor Specification File on page 43
- AKg Specification; refer to Section 5.6 AKg Specification File on page 44
- AKgm Specification; refer to Section 5.7 AKgm Specification File on page 47
- GenSync Specification; refer to Section 5.8 GenSync Specification File on page 51

Figure 11 shows an overview of a CyFlex system using DevCom, with files for a smoke meter.

#### Figure 11: CyFlex System Overview using DevCom



#### ØNote:

Entering debug on the command line turns on the same debug flag that is turned on if debug is true in the specification file. The specification file setting does not override the command



## 5.1 Startup File

The startup file is normally launched by a Go-script, which runs automatically when the computer is powered on. The example startup file shown below is for a test cell controlling an AVL 415G smoke meter.

#!/bin/sh release\_gptest smoke release\_task AVL415G\_mon sleep 2 /cyflex/bin/gp\_test /specs/gp/avl\_smoke\_header & sleep 2 svar gp\_trace\_smk on sleep 1 /cyflex/bin/nt /specs/gp/gp\_415G sleep 10 devcom\_mon 15 /specs/AVL415G\_mon.spec & dt=`date "+%A %D %r"` echo "AVL415 restart/started on \$dt" >>/tmp/415\_start\_log



### 5.2 Test Manager Procedure File

The example Test Manager procedure file below calls two sub-procedures, gp\_415G\_init and gp\_415G\_collect, to initialize and control an AVL 415G smoke meter.

ØNote:

```
The Test Manager file may be used to call sub-procedure qp 415G error, or the Monitor application, but in the example below
does not.
*****
# gp_415G
******
        # PURPOSE:
# This is the main procedure for operating the AVL 415 smoke meter using a serial communications link and
# the AK protocol.
# This procedure is designed to run as a parallel qp test driven by events sent from a second qp test that
 uses a test table (commonly referred to in CyFlex as, "furball") to do engine mapping.
#
# The standard mapping procedure will have a name in the form gp map Xha to indicate that is can take HSDA
#
 and AVL smoke meter data along with the fuel readings.
# This procedure waits for the SM_init_start and SM_coll_start events to be received so that it can call
# the appropriate sub-procedure.
# PROCEDURES CALLED:
# /specs/qp/qp 415G init
# /specs/gp/gp_415G_collect
# Notes:
# 1. The events used to communicate with the mapping test procedure need to be created in gen_labels.
# start mode (mode to begin the test)
2
@INSTANCE
smoke
```



```
# @GLOBAL EVENTS and @REGISTERED EVENTS specify events that are handled the same regardless of which
# test mode is active.
# If neither the next_mode or the test_procedure are specified, then the mode is terminated and the
# normal path is taken.
# If the next mode is specified, but the test procedure is not, then the test will jump to that mode in
# this procedure.
# If the test procedure is specified, but the next mode is not, then the test will jump to that sub-
# procedure.
# If the sub-procedure exits via a RETURN, the current procedure will begin executing at the
# default_next_mode specified for the test mode that was interrupted.
# If both next mode and test procedure are specified, it will jump to the next mode upon returning from
# the procedure.
# If the test procedure file does not exist or cannot be opened, then this mode will be executed next.
@GLOBAL_EVENTS
# event_name
                       next_mode
                                        test_procedure
@REGISTERED EVENTS
# event name
                       next mode
                                       test procedure
# Initialize AVL 415 when test starts
@MODE
# mode number
             max_timeout
                              default_next_mode
 2
               0.0[sec]
                              4
# description
Initialize AVL 415 when test starts
@PROCEDURE
# pathname
/specs/qp/qp 415G init
```



@MODE

#### @TERMINATION\_EVENTS

| # | event_name | next_mode/procedure       |
|---|------------|---------------------------|
|   | SM_init    | /specs/gp/gp_415G_init    |
|   | SM_collect | /specs/gp/gp_415G_collect |
|   | FR_RQST    | /specs/gp/gp_415G_collect |
|   |            |                           |



## 5.3 Initialization File

The gp\_415G\_init file initializes the smoke meter in the following example.

```
#
                            qp 415G init
                    * * * * * * * * * * * * * * * * * *
                                    *****
#
# PURPOSE:
# This sub-procedure is called by qp 415G to initialize the AVL 415 smoke meter. The SM init done event
# is emitted upon successful completion of this sub-procedure.
#
 PROCEDURES CALLED:
 Notes:
#
# 1. The events used to communicate with the mapping test procedure need to be created in gen labels.
# 2. The maximum sample volume is set to 20 liters (the maximum allowed) and the sample mode is set to
# auto-range. This means that the instrument will take a small sample and then decide on the basis of
# that measurement what sample volume (up to 20 liters) will be required to give a reading in the
# sensor's ideal range. At high fuel/air ratios, this has the potential to wet the paper and cause a
# tear or an inaccurate reading.
# If either of these problems is encountered, consider limiting the maximum sample volume.
# start mode (test begins)
 2
# @GLOBAL_EVENTS and @REGISTERED_EVENTS specify events that are handled the same regardless of which
# test mode is active.
# If neither the next mode or the test procedure are specified, then the mode is terminated and the
# normal path is taken.
# If the next mode is specified, but the test procedure is not, the test will jump to that mode in this
# procedure.
# If the test_procedure is specified, but the next_mode is not, the test will jump to that sub-
# procedure.
```



```
# If the sub-procedure exits via a RETURN, the the current procedure will begin executing at the
# default_next_mode specified for the test mode that was interrupted.
# If both next_mode and test_procedure are specified, it will jump to the next_mode upon returning from
# the procedure.
# If the test procedure file does not exist or cannot be opened, this mode will be executed next.
@GLOBAL EVENTS
# event name
                     next mode
                                     test procedure
@REGISTERED_EVENTS
# event_name
                     next mode
                                     test_procedure
# Set the SM init strt event to indicate initialization started.
@MODE
# mode number
              max_timeout
                            default next mode
   2
               5.0[sec]
                             4
# description
 Set the SM init strt event to indicate initialization started
@SET EVENTS
# start type
             event name
 AT START
             SM init strt
# Clear the buffer and reset error parameters.
@MODE
# mode number
              max timeout
                            default next mode
              1.0[sec]
  4
                            6
# description
 clear the buffer and reset error parameters
@DEVCOM_ACTIONS
                            failure_action
# start_type
              stop_code
```



|              | AT_START  | MODE_TERMINATE                                    | 30                                      |   |
|--------------|---|---|---|---|
| @I<br>#<br># | DEVCOM<br>instrument name<br>AVL415G /specs/AV<br>command key strir<br>"SRES" | /L415G.spec<br>ngs                                |   |   |
| @۲           | ARAMETERS   |   |   |   |
| #            | start_type<br>AT_START<br>AT_START<br>AT_START                                | label<br>SMErrorCnt<br>SMErrorNum<br>SM_repeat_md | value<br>0[none]<br>-1[none]<br>4[none] | restore(YES/NO)<br>NO<br>NO             |
| #*           | * * * * * * * * * * * * * * * * * *   | *           | * * * * * * * * * * * * * * *           | * |
| #            | START SESSION   |   |   |   |
| #*           | * * * * * * * * * * * * * * * * * * *   | *           | * * * * * * * * * * * * * * * *         | * |
| @N           | IODE  |   |   |   |
| #            | mode_number<br>6  | <pre>max_timeout 5.0[sec]</pre>                   | default_next_n<br>8                     | node                                    |
| #            | description<br>wait for reset to  | complete  |   |   |
| ൭൛           | ARAMETERS   |   |   |   |
| #            | start type  | label   | value                                   | restore (YES/NO)                        |
|              | AT_START<br>AT_START  | SMErrorNum<br>SM_repeat_md                        | -2[none]<br>6[none]                     | NO                                      |
| #*           | * * * * * * * * * * * * * * * * * *   | *           | * * * * * * * * * * * * * * * *         | * |
| #<br>#       | SREM to set 415 i   | into remote mode                                  |   |   |
| #*           | ·*************************************  | *           | * * * * * * * * * * * * * * * * *       | * |
| @№<br>#      | mode number   | timeout   | next mode                               |   |
|              | 8   | 10.0[sec]   | 10                                      |   |
| #            | mode_description<br>set 415 into remo   | (60 characters max<br>ote                         | .)                                      |   |
| ொ            | FUCOM ACTIONS   |   |   |   |
| ≝∟<br>#      | start type  | stop code   | failure acti                            | on                                      |
|              | AT_START  | MODE_TERMINATE                                    | 30                                      |   |



| @DEVCOM                           |  |   |
|-----------------------------------|--|---|
| # instrument name                 |  |   |
| AVL415G /spec                     | s/AVL415G.spec                               |   |
| <pre># command key s "SREM"</pre> | trings                                       |   |
| @PARAMETERS                       |  |   |
| <pre># start_type</pre>           | label  | value                                   |
| AT_START                          | SM_repeat_md                                 | 8[none]                                 |
| #****                             | *      | * |
| # send SRDY to                    | clear any errors                             |   |
|                                   | *      | * |
| # mode number                     | timeout                                      | next mode                               |
| 10                                | 10.0[sec]                                    | 12                                      |
| # mode descript                   | ion (60 characters ma                        | <br>ax)                                 |
| set 415 into                      | remote mode                                  |   |
|                                   |  |   |
| @DEVCOM_ACTIONS                   |  |   |
| <pre># start_type</pre>           | stop_code                                    | failure_action                          |
| # AT_START                        | MODE_TERMINATE                               |   |
| # Ignore error                    | SLALUS, SINCE LHIS CO<br>F TERMINATE MODE TE | DMMMANA SHOULA CLEAF IL                 |
| AI_DIARI MODI                     |  |   |
| @DEVCOM                           |  |   |
| # instrument nam                  | me   |   |
| AVL415G /spec                     | s/AVL415G.spec                               |   |
| # command key s                   | trings                                       |   |
| "SRDY"                            |  |   |
| @PARAMETERS                       |  |   |
| <pre># start_type</pre>           | label  | value                                   |
| AT_START                          | SM_repeat_md                                 | 10[none]                                |
| #****                             | *      | * |
| # SRDY to clear                   | any pre-existing er                          | cor, again                              |
| #*****                            | *****  | ****                                    |
| @MODE                             |  |   |


| <pre># mode_number 12</pre>  | timeout<br>10.0[sec]                              | next_mode<br>14   |
|--|---|---|
| <pre># mode_description SRDY to clear e:</pre>   | n (60 characters max)<br>rror                     |   |
| @DEVCOM_ACTIONS  |   |   |
| <pre># start_type AT_START</pre>   | stop_code<br>MODE_TERMINATE                       | failure_action<br>30  |
| <pre>@DEVCOM # instrument name    AVL415G /specs/i # command key str:    "SRDY"</pre>        | AVL415G.spec<br>ings                              |   |
| @PARAMETERS  |   |   |
| <pre># start_type</pre>  | label   | value   |
| AT_START   | SM_repeat_md                                      | 12[none]  |
| <pre>#************************************</pre>   | **************************************            | **************************************                          |
| # mode number  | timeout   | next mode   |
| 14   | 5.0[sec]  | 16  |
| <pre># mode_description Send "EGPA" to a</pre>   | n (60 characters max)<br>set purge time to 10 sec |   |
| @DEVCOM_ACTIONS  |   |   |
| <pre># start_type</pre>  | stop_code   | failure_action  |
| AT_START   | MODE_TERMINATE                                    | 30  |
| <pre>@DEVCOM # instrument name    AVL415G /specs/# # Command key str: # Note: this has</pre> | AVL415G.spec<br>ings<br>different units than in   | 415 "compatible" interface and a different number of arguments. |
| "EGPA 500 8000 .   | 10 0 0.2"   |   |



| @PARAMETERS              |   |  |
|--------------------------|---|--|
| <pre># start_type</pre>  | label                                   | value  |
| AT_START                 | SM_repeat_md                            | 14[none]   |
| #*****                   | * | *                          |
| "<br># Send "EMZY" to    | set sampling parameters.                | . "O" as a third argument tells the instrument to auto-range the |
| sample volume to         | achieve a reading in the                | e sensor's ideal range.  |
|                          | * | *                          |
| @MODE                    |   |  |
| <pre># mode_number</pre> | timeout                                 | next_mode  |
| 16                       | 5.0[sec]                                | RETURN   |
| # mode_descriptio        | n (60 characters max)                   |  |
| Send "EMZY" to           | set sampling parameters                 |  |
| @DEVCOM ACTIONS          |   |  |
| # start type             | stop code                               | failure action   |
| AT_START                 | MODE_TERMINATE                          | 30   |
|                          |   |  |
| @DEVCOM                  |   |  |
| # instrument name        | 44                                      |  |
| AVL415G /specs/          | AVL415G.spec                            |  |
| # command key str        | ings                                    |  |
| EMZI V U I               |   |  |
| @SET EVENTS              |   |  |
| # start_type             | event_name                              |  |
| AT_END                   | SM_init_done                            |  |
|                          |   |  |
| @PARAMETERS              |   |  |
| <pre># start_type</pre>  | label                                   | value  |
| AT_START                 | SM_repeat_md                            | 16[none]   |
| #*****                   | * | *                          |
| # Do nothing.            |   |  |
| #*****                   | * | *                          |
| @MODE                    |   |  |
| <pre># mode_number</pre> | timeout                                 | next_mode  |
| 18                       | 0.0[sec]                                | 18   |
| # mode descriptio        | n (60 characters max)                   |  |



Do nothing

| #*************************************                             | **************************************     | *************                           |
|--|--|---|
| <pre># mode_number     30 # mode_description</pre>                 | timeout<br>0.0[sec]<br>(60 characters max) | next_mode<br>32                         |
| Call error handl   | ing sub-procedure                          |   |
| <pre>@PROCEDURE # pathname /specs/gp/gp_4150</pre>                 | G_error                                    |   |
| #*****   | *    | * |
| <pre># Retry mode where #***********************************</pre> | failure occurred                           | ****                                    |
|  |  |   |
| # mode number  | timeout                                    | next mode                               |
| 32   |  | 2                                       |
| # mode description   | (60 characters max)                        | -                                       |
| Retry mode where   | failure occurred                           |   |
| @SWITCH  |  |   |
| # switch variable  |  |   |
| SM_repeat_md   |  |   |
| # case   | path                                       |   |
| 4  | 4  |   |
| 6  | б  |   |
| 8  | 8  |   |
| 10   | 10   |   |
| 12   | 12   |   |
| 14   | 14   |   |
| 16   | 16   |   |
| #*****   | *    | ******                                  |



# 5.4 Collector File

This file controls the top-level task, while the device driver interfaces with the smoke meter in the following example.

```
*****
                         qp 415G collect
±**
                                   #
# PURPOSE:
# This sub-procedure is called by gp_415G to initiate a collection sequence for an AVL 415 smoke meter
# equipped with two sample probes for two exhaust stacks. The SM_coll_done event is emitted upon
# successful completion of this sub-procedure.
#
# PROCEDURES CALLED:
# /specs/qp/qp 415G error
#
# Notes:
# 1. The events used to communicate with the mapping test procedure need to be created in gen_labels.
# 2. You need to know the dead volumes for both probes and the temperature and volume correction
# factors to "hard-wire" both this sub-procedure and the AVL_415.nnn config file before using this
# qp test. This will change once we make qp test and the config file more flexible.
# start mode ( mode where the test begins )
 2
# @GLOBAL_EVENTS and @REGISTERED_EVENTS specify events that are handled the same regardless of which
# test mode is active.
# If neither the next_mode nor the test_procedure are specified, then the mode is terminated and the
# normal path is taken.
#
# If the next mode is specified, but the test procedure is not, then the test will jump to that mode in
# this procedure.
# If the test_procedure is specified, but the next_mode is not, then the test will jump to that sub-
# procedure.
#
```



# If the sub-procedure exits via a RETURN, the the current procedure will begin executing at the # default\_next\_mode specified for the test mode that was interrupted. # If both next\_mode and test\_procedure are specified, it will jump to the next\_mode upon returning # from the procedure. If the test\_procedure file does not exist or cannot be opened, then this mode # will be executed next. @GLOBAL EVENTS #event name next mode test procedure @REGISTERED\_EVENTS next\_mode test\_procedure #event\_name @CREATE\_VAR (up to 128 variables per procedure ) # label initial value type units display format SMsmode STRING \_ . 1 1 SMstate STRING \_ . SMpapeco STRING @INSTANCE smoke # Set the SM coll strt event to indicate measurement sequence started. @MODE #mode number max timeout default next mode 2 -1.0[sec] 4 # description Set SM\_coll\_strt event to indicate meas seg started. @SET EVENTS # start type event name SM coll strt AT START @PARAMETERS # start type label value restore(YES/NO) AT\_START avl\_415\_smk -99[fsn] NO AT\_START SMErrorNum -99[none] NO



| AT_START                 | SMSmkN                          | -99[none]                   | NO                                      |
|--------------------------|---------------------------------|-----------------------------|---|
| AT_START                 | SMSmkMn                         | -99[none]                   | NO                                      |
| AT_START                 | SMSmk1                          | -99[none]                   | NO                                      |
| AT_START                 | SMSmk2                          | -99[none]                   | NO                                      |
| AT_START                 | SMSmk3                          | -99[none]                   | NO                                      |
| AT_START                 | SMSmk4                          | -99[none]                   | NO                                      |
| AT_START                 | SMSmk5                          | -99[none]                   | NO                                      |
| AT_START                 | SMSootMn                        | -99[kg/m3]                  | NO                                      |
| AT_START                 | SMSoot1                         | -99[kg/m3]                  | NO                                      |
| AT_START                 | SMSoot2                         | -99[kg/m3]                  | NO                                      |
| AT_START                 | SMSoot3                         | -99[kg/m3]                  | NO                                      |
| AT_START                 | SMSoot4                         | -99[kg/m3]                  | NO                                      |
| AT_START                 | SMSoot5                         | -99[kg/m3]                  | NO                                      |
| AT_START                 | SMBlkPcnt                       | -99[none]                   | NO                                      |
| AT_START                 | SMWhtVal                        | -99[none]                   | NO                                      |
| AT_START                 | SMGreyVal                       | -99[none]                   | NO                                      |
| AT_START                 | SMBlkVal                        | -99[none]                   | NO                                      |
| AT_START                 | SMEffVol                        | -99[none]                   | NO                                      |
| AT_START                 | SMEffLen                        | -99[none]                   | NO                                      |
| AT_START                 | SMDelPdig                       | -99[none]                   | NO                                      |
| AT_START                 | SMDelPmbar                      | -99[none]                   | NO                                      |
| AT_START                 | SMRelPdig                       | -99[none]                   | NO                                      |
| AT_START                 | SMRelPmbar                      | -99[none]                   | NO                                      |
| AT_START                 | SMTemp                          | -99[none]                   | NO                                      |
| AT_START                 | SMSupVolt                       | -99[none]                   | NO                                      |
| AT_START                 | SM_repeat_md                    | 2[none]                     |   |
|                          |                                 |                             |   |
| #********                | * * * * * * * * * * * * * * * * | * * * * * * * * * * * * * * | * |
| # Send "SPUL"            | and purge for                   | 40 sec.                     |   |
| #********                | * * * * * * * * * * * * * * * * | * * * * * * * * * * * * * * | * |
| @MODE                    |                                 |                             |   |
| <pre># mode_number</pre> | c timeout                       | t next_                     | _mode                                   |
| 4                        | 10.0[se                         | ec] 6                       |   |
| <pre># mode_descr:</pre> | iption (60 char                 | racters max)                |   |
| Send "SPUL'              | " and purge for                 | r 10 sec.                   |   |
|                          |                                 |                             |   |
| @ELSE_MODE               |                                 |                             |   |
| 30                       |                                 |                             |   |
|                          |                                 |                             |   |

@DEVCOM\_ACTIONS





# SMES to start measurement @MODE # mode\_number timeout next\_mode 27 10[sec] 28 # mode\_description (60 characters max) Wait! #\*\*\*\*\*\*\*\* # ASTZ to be sure measurement is done @MODE # mode number timeout next\_mode 28 1.83[sec] 29 # mode\_description (60 characters max) ASTZ to check on measurement @ELSE\_MODE 30 @DEVCOM\_ACTIONS # start\_type stop\_code failure\_action AT\_START MODE\_TERMINATE ELSE\_MODE @DEVCOM # instrument name AVL415G /specs/AVL415G.spec # command key strings # ASTZ,%s %s %s #%s #%s "ASTZ SMsmode SMstate SMpapeco" @PARAMETERS # start\_type label value AT START 28[none] SM\_repeat\_md #\*\*\*\*\*\* # Repeat ASTZ until ready mode @MODE # mode number timeout next\_mode 29 1.2[sec] 28



Check if done with reading. Switch on SMstate.

@SWITCH # switch variable SMstate # case path 'SRDY' 8 # AFSN to get smoke values @MODE # mode\_number timeout next\_mode 8 5[sec] 10 # mode\_description (60 characters max) AFSN to get smoke values @ELSE MODE 30 @DEVCOM\_ACTIONS # start\_type stop\_code failure\_action AT\_START MODE\_TERMINATE ELSE\_MODE @DEVCOM # instrument name AVL415G /specs/AVL415G.spec # command key strings # AFSN,%d #%f #%f #%f #%f #%f #%f "AFSN SMTemp SMSmkMn" @PARAMETERS # start\_type label value AT\_START 8[none] SM repeat md AT END avl 415 smk "SMSmkMn/100[none]" # APAP to see how much paper left @MODE



# mode number timeout next mode 2[sec] 10 11 # mode\_description (60 characters max) APAP to see how much paper left @ELSE MODE 30 @DEVCOM\_ACTIONS stop code failure action # start type AT START MODE\_TERMINATE ELSE MODE @DEVCOM # instrument name AVL415G /specs/AVL415G.spec # command key strings "APAP SMPaperLeft" @PARAMETERS # start\_type label value AT\_START SM\_repeat\_md 10[none] # AKON to get soot concentration @MODE # mode number timeout next mode 11 10[sec] 12 # mode\_description (60 characters max) AKON to get soot concentration @ELSE MODE 30 @DEVCOM ACTIONS failure action # start type stop code MODE TERMINATE ELSE MODE AT\_START @DEVCOM # instrument name



AVL415G /specs/AVL415G.spec # command key strings # AKON,%d #%f #%f #%f #%f #%f #%f "AKON SMTemp SMSootMn SMSoot1 SMSoot2 SMSoot3 SMSoot4 SMSoot5" @PARAMETERS # start type label value AT START 11[none] SM repeat md # RETURN unless SMDiag is ON @MODE # mode number next mode timeout 12 -1.0[sec] RETURN # mode description (60 characters max) RETURN unless smoke diag is ON @IF\_FALSE SMDiag @ELSE MODE 14 @SET EVENTS # start type event name AT END SM\_coll\_done # AADC to read instrument temperature @MODE # mode number timeout next mode 10.0[sec] 14 16 # mode\_description (60 characters max) AADC to read instrument temperature



@ELSE\_MODE 30

@DEVCOM\_ACTIONS # start\_type stop\_code failure\_action AT START MODE\_TERMINATE ELSE MODE @DEVCOM # instrument name AVL415G /specs/AVL415G.spec # command key strings # AADC,%f %f %f %f "AADC SMDelPmbar SMRelPmbar SMTemp SMSupVol" @PARAMETERS # start type label value restore(YES/NO) AT START SM repeat md 14[none] "SMcount + 1[none]" AT START SMcount NO # AOPT to get ADC values @MODE # mode number timeout next mode 16 5[sec] 18 # mode description (60 characters max) AOPT to get ADC values @ELSE MODE 30 @DEVCOM ACTIONS stop code failure action # start type AT START MODE TERMINATE ELSE MODE @DEVCOM

# instrument name
AVL415G /specs/AVL415G.spec



# command key strings # AOPT %f %f %f %f "AOPT SMBlkPcnt SMWhtVal SMGreyVal SMBlkVal" @PARAMETERS # start type label value AT START 16[none] SM repeat md # AEVL to get volume and effective length @MODE # mode\_number timeout next\_mode 18 5[sec] RETURN # mode description (60 characters max) AEVL to get volume and effective length @ELSE MODE 30 @DEVCOM\_ACTIONS # start\_type stop\_code failure action AT START MODE TERMINATE ELSE MODE @DEVCOM # instrument name AVL415G /specs/AVL415G.spec #command key strings # AVEVL %d %d "AEVL SMEffVol SMEffLen" @PARAMETERS # start type label value AT START SM repeat md 18[none] SMTime "time - SMTime" AT END @SET EVENTS # start\_type event\_name AT\_END SM\_coll\_done



#\*\*\*\*\*\*\*\* # Call error handling sub-procedure @MODE # mode number timeout next\_mode 30 1.0[sec] 32 # mode description (60 characters max) Call error handling sub-procedure @PROCEDURE # pathname /specs/gp/gp\_415G\_error # Retry mode where failure occurred @MODE # mode\_number timeout next\_mode 32 1.0[sec] 2 # mode\_description (60 characters max) Retry mode where failure occurred @SWITCH # switch variable SM repeat md # case path 2 2 4 4 б б 8 8 10 10 11 11 12 12 14 14 16 16 18 18 #\*\*\*\*\*



### 5.5 Monitor Specification File

```
@REG NAME
AVL415G mon
# This is a devcom monitor list spec file.
# The file is activated with devcom_mon priority /pathtothisspecfile.
# The command string (column 3) will be sent to the devcom device/instrument server (column 2),
# on each event (1st column).
# The action can be started and stopped with optional events specified in columns 4 and 5.
# The column 1 event can be a named event or a millisecond timer value.
# Start and stop events must be named events.
# All events will be created as temporary events if they do not exist.
#
# The user should test the command string with devcom_smsg_tst before adding it to this file.
# If an error is detected executing the command the event "REG_NAME"_err will be set.
# The list is sorted by event id (column 1), so like-events are grouped, but otherwise executed
# sequentially.
# There is NO threading across different devices, so the whole list will be paced by the slowest
# command/device.
# There is also a limit of 200 commands. If more are required, break them into multiple lists.
#
# Log to /data/errors/devcom mon.DATE
$Debuq
true
# false
# This MUST be located before the first command entry.
SCMDS
# event, device, command string, start event, stop event or ms
5000, AVL415G_TC, "ASTZ SMstate - - - "
#5000, AVL415G TC, "ASER - - - - - SMHtFanT SMAirT SMHtHosT - - - -"
#5000, AVL415G_TC, "APAP SMPaperLeft"
5000, AVL415G TC, "ASTF SMErrorNum"
#start int, AVL489, "SINT"
#stop int, AVL489, "SINA"
# The list MUST end with a dollar sign.
```



# 5.6 AKg Specification File

```
$Device
/dev/ttyUSB0:9600,8,1,N
# 192.168.222.200:5304
#
$Timeout
# default timeout in milliseconds
2000
#
# Name task will attach to. If there are multiple smoke meters, each name must be unique.
#
$Instrument
AVL415G
#
$Protocol
AKq
#
# Command formats
# Command format Response format [timeout]
#
$CmdDef
# Control Commands have no data response
# CommandString formats AK_Command [Timeout(msec)]
# comma or tab delmited
# command string,[response],[timeout]
SRES
SMAN
SREM
SRDY
SASB
SMES, -, -, 60000
SPUL, -, -, 65000
SPSE
SPSA
SFPF
SEX1
SEX2
SVOP
SKOR
```



```
SMKA
SMRE
SLEC
SBSZ
# Query Commands get data from device
#
 IMPORTANT - A # sign before % implies that the returned parameter is optional. If the # sign is not used
#
             for an optional return, an error will be flagged indicating that the device is not properly
#
             responding to a query
#
# comma or tab delimited
# command string,[response],[timeout]
AKEN,-,%s %s
AMZY,-,%d %d %d
ASYS,-,%f %d %d
APLP, -, %d %d %f
AMMP, -, %d %d %d
ACPA,-,%d %d %d %f
AGPA,-,%d %d %d %d %d
ASTF, -, %d
# ASTZ 4 or 5 returns depends on if single-flow or two-channel Measurement
ASTZ,-,%s %s %s #%s #%s
APAP, -, %d
# AKON, AFSN, APOL first parameter indicates number of other parameters returned
AKON,-,%d #%f #%f #%f #%f #%f #%f #%f
AFSN,-,%d #%f #%f #%f #%f #%f #%f #%f
APOL,-,%d #%f #%f #%f #%f #%f #%f #%f
AOPT,-,%f %f %f %f
AEVL,-,%d %d
AIZU,-,%s %d %d %s %d %d %d %d %d %d
AIDA,-,%f %f %f %f %d %d %d %f
AADC,-,%d %d %f %f
AVKF, -, %d %d %f
ABSZ,-,%d %d
#
# Setting commands set data
# comma or tab delimited
# Note: NO optional output parameters
# command string,[response],[timeout]
```



CyFlex® Device Communication User Guide

EMZY,%d %d %d ESYS,%f %d %d %d EPLP,%d %d %f EMMP,%d %d %d %d ECPA,%d %d %d %f # This section ends with a \$. \$



# 5.7 AKgm Specification File

```
#
$Device
/dev/ser1:38400,8,1,N
$Timeout
# default timeout in milliseconds
1000
$Debug
false
# true
# Name specific to the device
Note: If using multiple smoke meters, the name will need to be unique.
#
$ Instrument
GasMux
# AK Gas Mux specific
$Protocol
AKqm
# Note: change back to AKg if any problems (clf 9/28/11)
# Command formats
# Command format
                     Response format [timeout]
#
$CmdDef
# Control commands have no data response
# CommandString formats AK_Command [Timeout(msec)]
# comma or tab delmited
# command key,[command arguments],[response],[timeout]
# Command key,[command arguments],[response],[timeout]
# Auto range off
SARA,%s
# Auto range on
SARE,%s
# Auto zero/span cal
SATK,%s
```



# Trap drain SDRN,%s # Span analyzer # SEGA KO # Set analyzer range # SEMB KO # Enable analyzer # SENA # Select sample point # SENT K0,2 SENT,%s %d # HC Hangup check # SHUP # Leak check # SLEC # Sample point leak check # SLKS # Start sampling SMGA,%s # Zero analyzer # SNGA # Purge SSPL,%s # Standby STBY,%s # Mux Commands # Release control of bench SABT # Log results SLOG 1 -> tun on mux logging, SLOG 2 -> log zero/span results SLOG,%d # Open data file # SNDF,-,%s # Request control of bench

SREQ



```
# Priority request control of bench
# SRQP,-,%s
#
#
 Query Commands get data
#
 IMPORTANT - A # sign before % implies that the returned parameter is optional. If the # sign is not used
              for an optional return, an error will be flagged indicating that the device is not properly
#
#
              responding to a query
#
# comma or tab delmited
# command string,[response],[timeout]
# Request span calibration results
# range,measured,deviatoin in ppm,deviation in FS%
# AAEG,%f %f %f %f
# Request analyzer gas concentrations
# zero.mid.test
# AANG,%f %f %f
# Request analyzer range
# range of analyzer 1 - N
# TODO - is this an integer range number or a full scale value?
# AEMB,%f #%f #%f #%f #%f #%f #%f
# Request span and cal gas concentrations
# concentration of: zero,midspan,test
# AGAS,K1,-,%f %f %f
AGAS,%s,%f %f %f
# Request HC hangup
# HC Concentration per manifold
# AHUP,%f #%f #%f #%f #%f
# Request leak check results - one manifold at a time
# leak test results: 0=fail,1=pass
# ALEC,%d #%d #%d #%d #%d #%d #%d
# Request concentration
# concentration from analyzer 1-N
```



# AKON,%f #%f #%f #%f #%f #%f #%f AKON,%s,%f #%f #%f #%f #%f #%f #%f

# Request temperatures
# Temperatures from 1 to ?
# ATEM,%f

# Request position in queue
# position in queue
AQUE,-,%s
# This section should end with a \$
\$



# 5.8 GenSync Specification File

```
$Device
# serial device or socket
# IP:port
# serail port:baud,bits,stobits,parity,flow control(HW,XON,NONE-HW default)
/dev/ttyS0:9600,8,1,N,NONE
# maximum message rate is 1/sec
$MaxMsgRate
.5
$Debug
True
$Timeout
# default timeout in milliseconds
2000
#
# Name task will attach to, or key name so multiple device of same type can be supported.
#
$ Instrument
# Yokogawa GS610 Multi meter
GS610
# Protocol
$Protocol
# GenSync - general synchronous (command response) serial/socket protocol
GenSync
#message structure
$CmdStruct
# Structure
#
## This section shows the sequence of message elements
## H = header string
## S = station identifier
## M = message
## T = trailer string
## C = crc
##
```



MT

# response structure if any response expected \$RspStruct ΜT # \$Header -1 # trailer \$Trailer <CR><LF> # Note: NONE supported at this time 7/1/10, will be added as needed **\$CRC** -1 # Command formats # Command format Response format [timeout] # \$CmdDef # Control Commands have no data response # CommandString formats AK Command [Timeout(msec)] # comma or tab delimited #Command key, [command arguments],[response],[timeout] #\*IDN? ID and version of instrument \*IDN?,-, %s %s %s #\*RST Initializes settings Set up known conditions \*RST,-,-# enable/disable output ON/OFF :OUTPUT:STAT,%s,-#:SOUR:FUNC CURR Current sourcing mode Coil measurement mode :SOUR:FUNC, %s,-#:SOUR:SHAP DC Sourcing mode: DC DC mode for sourcing :SOUR:SHAP, %s,-#:SOUR:MODE FIX Sourcing pattern: Fixed :SOUR:MODE, %s,-#:SOUR:CURR:LEVEL <2uA,200uA,2mA,20mA,200mA,.5A,1A,2A,3A> :SOUR:CURR:LEV,%s,-



#:SOUR:CURR:RANG <2uA,200uA,2mA,20mA,200mA,.5A,1A,2A,3A> :SOUR:CURR:RANG, %s,-#:SENS:FUNC RES Resistance mode :SENS:FUNC,%s,-#:SENS:RANG:AUTO ON Sensing auto mode on :SENS:RANG:AUTO,%s,-#:SENS:AVER:STAT ON Averaging on :SENS:AVER:STAT, %s,-#:SENS:AVER:MODE MOV Moving average :SENS:AVER:MODE, %s, -#:SENS:AVER:COUN 10 100 averaging counts :SENS:AVER:COUN,%d,-#:SENS:AZER:STAT ON Auto zeroing on :SENS:AZER:STAT,%s,-#:SENS:RSEN ON Remote sense (4 wire mode) on :SENS:RSEN,%s,-#:TRIG:SOUR EXT External trigger source :TRIG:SOUR,%s,-**#:INIT** Initiate :INIT,-,-#\*TRG Trigger \*TRG,-,-# \*SAV save setup 1,2,3,4 \*SAV,%d,-# \*RCL recall setup 1,2,3,4 \*RCL,%d,-#:FETCH? Read results :FETC?,-,%f #:READ? internal trigger and read results :READ?,-,%f # check status of measurement :STAT:SENS:COND?,-,%d # check status of source :STAT:SOUR:COND?, -, %d # This section should with a \$ \$



# 6 Supported Protocols

DevCom applications and device drivers support the following industry standard protocols:

- AK
- GENSYNC

DevCom supports other protocols by customer request:

- GOAL
- MON2

Table 4 lists commonly used test devices, their protocols, and required files.

TRP Laboratories provides files (and example files) with each installation of CyFlex. The current versions of all executable and specification files are available to customers on a TRP Laboratories server in the directories shown below.

#### ONOTE:

In general, the /cell.def (cell defaults) directory contains hardware and site-specific spec files. The /specs.def (specs defaults) directory contains application specific spec files. The /cyflex/bin directory contains executables.

| Device Type  | Supported<br>Protocol | Files Required  |
|--|-----------------------|---|
| <ul> <li>Smoke meter (AVL 415G)</li> <li>Filter Sampler (AVL 472)</li> </ul> | AK                    | <ul> <li>/cyflex/bin/dc_AKg</li> <li>/specs.def/AVL415G.specs</li> <li>/specs.def/AVL472.specs</li> </ul>     |
| General ASCII synchronous protocol devices                                   | GENSYNC               | <ul> <li>/cyflex/bin/dc_GenSync</li> <li>/specs.def/CytecMux.specs</li> <li>/specs.def/GS610.specs</li> </ul> |
| Genset controller  | GOAL                  | <ul><li>/cyflex/bin/dc_GOAL</li><li>/specs.def/GOAL.specs</li></ul>   |
| Genset controller  | MON2                  | <ul><li>/cyflex/bin/dc_MON2</li><li>/specs.def/MON2.specs</li></ul>   |

#### Table 4: Test devices, Protocols, and Required Files

#### ØNote:

TRP Laboratories does not supply a startup file. However, most test cells using a smoke meter have a version of that script. Users typically name it start\_415 and place it in the /specs/cmds directory



# 6.1 Generic Communication AK Protocol

This section describes the generic communication AK protocol. Most test cell configurations using DevCom follow this protocol, so it is described here in detail.

This document uses an AVL smoke meter as an example test device. Smoke meters equipped with RS232 25-pin serial ports typically use the generic communication AK protocol. Commands will be the same for smoke meters connected with DB-9 serial ports.

For serial port pin assignments and baud rates, refer to the manufacturer's manual for your test device.

Communication is based on the master-slave principle.

### 6.1.1 Command Format

All commands and data are transferred in ASCII format.

The first characters of the command must be <STX> so that the command can be interpreted as an AK command. If the first characters are not < STX >, the device will acknowledge (respond) with <STX> ????<ETX>. The rest of the command follows the < STX > prefix, depending on the command type as described in *Section 6.1.3 Command Types* on page 57.

#### ØNote:

An AVL 415 smoke meter responds to a command with an acknowledgment. The following special conditions may apply – refer to the manufacturer's manual, which should state:

- The channel specification can be omitted from the command message.
- The don't care byte is not evaluated, but only copied into the acknowledgment.

### 6.1.2 Protocol Framework

#### 6.1.2.1 *Command Message*

#### Table 5: Command Message Protocol

| Byte | Function   |
|------|--|
| 1    | <stx></stx>                                      |
| 2    | Ignored  |
| 3 6  | Function code                                    |
|      | SXXX Control commands                            |
|      | EXXX Setting commands                            |
|      | AXXX Query commands                              |
| 7    | Blank  |
| 8    | К  |
| 9    | Channel number (always 0)                        |
| -    | Data (variable)                                  |
| -    | Can also be omitted (depending on function code) |



| Byte     | Function    |
|----------|-------------|
| -        |             |
| nth byte | <etx></etx> |

### 6.1.2.2 Acknowledgement Message

 Table 6: Acknowledgment Message Protocol

| Byte     | Function   |
|----------|--|
| 1        | <stx></stx>                                      |
| 2        | Ignored  |
| 3 6      | Function code (same as in command)               |
| 7        | Blank  |
| 8        | Error status                                     |
|          | 0 no error                                       |
|          | 1 9 error (cyclical exponent)                    |
| -        | Data (variable)                                  |
| -        | Can also be omitted (depending on function code) |
| -        |  |
| nth byte | <ett>&gt;</ett>                                  |

Acknowledgment message when function code not recognized:

- <STX> ???? n<ETX>
- n ... error status

General definitions and possible acknowledgment messages in the event of an error:

- The command message must start with <STX> and end with <ETX>.
- Blanks are used as separators.
- The variable-length floating decimal format is used to display numerical values. Integers have no decimal points. Prefixed signs are specified only for negative numbers. Physically meaningless digits are omitted.
- If a control or setting command is transmitted in SMAN mode (which only permits query commands), the acknowledgment is as follows:

```
<STX> XXXX n K0 OF<ETX>
XXXX ... function code
n ... error status
OF ... offline
```

• If a command is transmitted that cannot be executed in the current state or, because an error has occurred, the acknowledgment is as follows:

```
<STX> XXXX n K0 BS<ETX>
XXXX ... function code
n ... error status
```



BS ... busy

• If a command message contains a syntax error (e.g. missing or too many parameters), the acknowledgment is as follows:

<STX> XXXX n K0 SE<ETX> XXXX ... function code n ... error status

SE ... syntax error

• If the value of a parameter is outside the permissible range, the acknowledgment is as follows:

<STX> XXXX n K0 DF<ETX>

XXXX ... function code

- n ... error status
- DF ... data error

### 6.1.3 Command Types

Commands are defined as four-digit function code and can be divided into three groups:

| Command Type | Function Code                 |
|--------------|-------------------------------|
| Control      | 1st character: S (e.g., SREM) |
| Setting      | 1st character: E (e.g., EMZY) |
| Query        | 1st character: A (e.g., ASTZ) |

#### Table 7: Command Groups

## 6.1.4 Control Commands

*Table 8* lists control commands with their functions. These commands apply to an AVL 415 smoke meter which is used as the example test device in this procedure. Because there may be differences between smoke meter models, refer to the manufacturer's manual for a comprehensive list of commands.

#### ØNote:

The variable data block is not evaluated for these commands.

| Command | Function                               | Comments   | Permitted |  |
|---------|--|--|-----------|--|
| SRES    | Reset                                  | After an initialization phase (such as after power-up), the AVL 415 is in manual mode.                       | Any time  |  |
| SMAN    | Switches the AVL 415 to manual mode    | Procedures started in remote mode are aborted.   | Any time  |  |
| SREM    | Switches the AVL<br>415 to remote mode | The settings (operating mode, paper<br>economy mode and sampling point)<br>are transferred from manual mode. | Any time  |  |

#### Table 8: Control Commands



| Command | Function   | Comments  | Permitted   |
|---------|--|---|---|
| SRDY    | Change to ready mode   | Stops all actions and acknowledges errors   | Remote  |
| SASB    | Change to mode,<br>ready for sampling<br>for synchronized<br>measurement     | Starts the mode set with EMZY as a synchronized measurement. The sampling process is prepared and the state is then ready for sampling.   | Remote ready  |
| SMES    | Starts or continues<br>with the<br>measurement set<br>with EMZY              | If the device was in ready for sampling<br>state, a sampling process is started<br>immediately, and in multiple<br>measurements returns to ready for<br>sampling state after each sample is<br>taken. | Remote ready<br>or, remote ready<br>for sampling    |
| SPUL    | Starts purge for max.<br>60 s per sampling<br>line                           |   | Remote ready  |
| SPSE    | Activates paper<br>economy mode  |   | Remote ready  |
| SPSA    | Deactivates paper<br>economy mode  |   | Remote ready  |
| SFPF    | Starts a filter paper feed   |   | Remote ready  |
| SEX1    | Activates sampling<br>point 1 (for two-<br>channel<br>measurement only)      |   | Remote ready<br>with two-<br>channel<br>measurement |
| SEX2    | Activates sampling<br>point 2 (for multiple-<br>channel<br>measurement only) |   | Remote ready<br>with two-<br>channel<br>measurement |
| SVOP    | Starts volume testing mode   |   | Remote ready  |
| SKOR    | Sets volume correction factor  | Saves the volume correction factor  | Remote ready<br>when data<br>backup<br>deactivated  |
| SMKA    | Starts reflectometer<br>head calibration                                     |   | Remote ready  |
| SMRE    | Starts reflectance measurement   |   | Remote ready  |
| SLEC    | Starts the leak check  |   | Remote ready  |



| Command | Function                           | Comments | Permitted  |
|---------|------------------------------------|----------|--|
| SBSZ    | Resets the operating-hour counter+ |          | Remote ready<br>when data<br>backup<br>deactivated |

### 6.1.5 Setting Commands

*Table 9* lists setting commands, their functions, and parameters. These commands apply to an AVL 415 smoke meter which is used as the example test device in this procedure. Because there may be differences between smoke meter models, refer to the manufacturer's manual for a comprehensive list of commands.

| Command | Function                          | Parameters   | Permitted                                 |
|---------|-----------------------------------|--|---|
| EMZY    | Measurement cycle for remote mode | Variable data block for measurement cycle with preselected sampled volume:   | Remote ready                              |
|         |                                   | <ul> <li>1st datum: V</li> </ul>   |   |
|         |                                   | <ul> <li>2nd datum: Sampled volume [ml]<br/>(50 20000 or 0 = auto-range with<br/>pre-measurement)</li> </ul>   |   |
|         |                                   | <ul> <li>3rd datum: Number of samples         <ul> <li>(1 5 or 0 = special<br/>measurement, i.e. with tolerance<br/>monitoring)</li> </ul> </li> </ul> |   |
|         |                                   | Variable data block for measurement cycle with preselected sampling time:  |   |
|         |                                   | <ul> <li>1st datum: Z</li> </ul>   |   |
|         |                                   | <ul> <li>2nd datum: Sampling time [s]<br/>(1 120, i.e. 1 s 2 min or 0 =<br/>auto-range online)</li> </ul>  |   |
|         |                                   | <ul> <li>3rd datum: number of samples         <ul> <li>(1 5 or 0 = special<br/>measurement, i.e. with tolerance<br/>monitoring)</li> </ul> </li> </ul> |   |
| ESYS    | System parameters                 | <ul> <li>1st datum: Effective filter area</li> <li>[mm<sup>2</sup>]</li> <li>(1 3276.7)</li> </ul>   | Remote when<br>data backup<br>deactivated |
|         |                                   | 2nd datum: Temperature   |   |



| Command | Function                                     | Parameters   | Permitted                                 |
|---------|--|--|---|
|         |  | conditioning<br>0: Standard version without heating<br>or heating off<br>1: Heated version   |   |
|         |  | <ul> <li>3rd datum: Gray value error<br/>evaluation</li> <li>0: Standard monitoring</li> <li>1: Gray value &lt; white value is<br/>permitted (that also activates the<br/>negative measurement value<br/>display)</li> <li>Default: 697,5 1 0</li> </ul> |   |
| EPLP    | Pollution level parameters                   | <ul> <li>1st datum: Effective length for<br/>Pollution Level [mm]<br/>(10 32767)</li> </ul>  | Remote when<br>data backup<br>deactivated |
|         |  | <ul> <li>2nd datum: Negative offset for<br/>Pollution Level [%]</li> <li>(0 20)</li> </ul>   |   |
|         |  | <ul> <li>3rd datum: Correction value for<br/>soot concentration<br/> 32.767)</li> </ul>  |   |
|         |  | <ul> <li>Default: 380 3,5 1.000</li> </ul>   |   |
| EMMP    | Measurement<br>parameters for<br>manual mode | <ul> <li>1st datum: Number of<br/>measurement cycles for manual<br/>mode<br/>(1 5)</li> </ul>  | Remote                                    |
|         |  | <ul> <li>2nd datum: Sampling time for<br/>manual mode [s]<br/>(1 180)</li> </ul>   |   |
|         |  | <ul> <li>3rd datum: Nominal volume [ml]<br/>(50 30000)<br/>This value is also used for the<br/>volume calibration.</li> </ul>  |   |
|         |  | • Default: 1 6 1000  |   |



| Command | Function                  | Parameters   | Permitted                                 |
|---------|---------------------------|--|---|
| ECPA    | Calibration<br>parameters | <ul> <li>1st datum: Dead volume (sampling line 1) [ml]</li> <li>(0 2000)</li> </ul>                                      | Remote when<br>data backup<br>deactivated |
|         |                           | <ul> <li>2nd datum: Dead volume (sampling line 2, for two-channel measurement)</li> <li>ml]</li> <li>(0 2000)</li> </ul> |   |
|         |                           | <ul> <li>3rd datum: Ambient pressure<br/>[mbar]<br/>(100 2000)</li> </ul>  |   |
|         |                           | <ul> <li>4th datum: Correction factor volume</li> <li>(0.01 32.767)</li> </ul>   |   |
|         |                           | • Default: 42 42 980 1.000   |   |
| EGPA    | Global parameters         | <ul> <li>1st datum: Minimum sampled<br/>volume in auto-range [ml]<br/>(100 max.)</li> </ul>                              | Remote                                    |
|         |                           | <ul> <li>2nd datum: Maximum sampled<br/>volume in auto-range [ml]<br/>(min 20000)</li> </ul>                             |   |
|         |                           | <ul> <li>3rd datum: Extended purge time [s]<br/>(0 60)</li> </ul>  |   |
|         |                           | <ul> <li>4th datum: External trigger delay [s]<br/>(0 10)</li> </ul>   |   |
|         |                           | <ul> <li>5th datum: Tolerance for Special<br/>Measurement [FSN]</li> <li>(0 1)</li> </ul>                                |   |
|         |                           | • Default: 500 5000 0 0 0.20   |   |

# 6.1.6 Query Commands

Query commands are permitted during any mode. These commands apply to an AVL 415 smoke meter which is used as the example test device in this procedure. Because there may be differences between smoke meter models, refer to the manufacturer's manual for a comprehensive list of commands.



#### *()* Important:

A # sign before the % sign in the return data format specification implies that the returned parameter is optional. If the # sign is not used for an optional return, an error will be flagged indicating that the device is not properly responding to a query.

| Command | Function  | Parameters  |
|---------|---|---|
| AKEN    | Device ID   | Variable data block: 415S Vx.xx<br>(x: software version)  |
| AMZY    | Definition of the<br>measurement cycle<br>for remote mode | <ul> <li>Variable data block for measurement cycle with preselected sampled volume</li> <li>1st datum: V</li> <li>2nd datum: Sampled volume [ml]<br/>(50 20000 or 0 = Auto-Range with pre-<br/>measurement)</li> <li>3rd datum: Number of samples<br/>(1 5 or 0 = Special Measurement, i.e. with tolerance monitoring)</li> <li>Variable data block for measurement cycle with preselected sampling time</li> <li>1st datum: Z</li> <li>2nd datum: Sampling time [s]</li> </ul> |
|         |   | <ul> <li>(1 120, i.e. 1 s 2 min or 0 = Auto-Range online)</li> <li>3rd datum: Number of samples <ul> <li>(1 5 or 0 = Special Measurement, i.e. with tolerance monitoring)</li> </ul> </li> </ul>  |
| ASYS    | System parameters   | <ul> <li>1st datum: Effective filter area [mm<sup>2</sup>]<br/>(1 3276.7)</li> <li>2nd datum: Temperature conditioning<br/>0: Standard version without heating or heating off<br/>1: Heated version</li> <li>3rd datum: Gray value error evaluation<br/>0: Standard monitoring<br/>1: Gray value &lt; white value is permitted (that also<br/>activates the<br/>negative measurement value display)</li> </ul>  |
| APLP    | Pollution Level<br>parameters                             | <ul> <li>1st datum: Effective length for Pollution Level [mm]<br/>(10 32767)</li> <li>2nd datum: Negative offset for Pollution Level [%]<br/>(0 20)</li> <li>3rd datum: Correction value for soot concentration<br/>(0.01 32.767)</li> </ul>  |

Table 10: Query Commands



| Command | Function  | Parameters   |
|---------|---|--|
| AMMP    | Measurement<br>parameters for<br>manual mode<br>(These parameters<br>are used as<br>defaults, for<br>example, when the<br>device is powered<br>up.) | <ul> <li>1st datum: Number of measurement cycles for manual mode<br/>(1 5)</li> <li>2nd datum: Sampling time for manual mode [s]<br/>(1 180)</li> <li>3rd datum: Nominal volume [ml]<br/>(50 30000)</li> </ul>   |
| ACPA    | Calibration<br>Parameters   | <ul> <li>1st datum: Dead volume (sampling line 1) [ml]<br/>(0 2000)</li> <li>2nd datum: Dead volume (sampling line 2, for two-<br/>channel measurement)<br/>[ml]<br/>(0 2000)</li> <li>3rd datum: Ambient pressure [mbar]<br/>(100 2000)</li> <li>4th datum: Correction factor for volume<br/>(0.01 32.767)</li> </ul>   |
| AGPA    | Global parameters   | <ul> <li>1st datum: Minimum sampled volume in Auto-Range<br/>[ml]<br/>(100 max.)</li> <li>2nd datum: Maximum sampled volume in Auto-Range<br/>[ml]<br/>(min 20000)</li> <li>3rd datum: Extended purge time [s]<br/>(0 60)</li> <li>4th datum: External trigger delay [s]<br/>(0 10)</li> <li>5th datum: Tolerance for Special Measurement [FSN]<br/>(0 1)</li> </ul> |
| ASTF    | Current error code  | Refer to the manufacturer's manual for a description of the<br>error code displayed, and the action required to correct<br>the error.<br>If no error has occurred or the error has already been<br>remedied, 0 is transmitted.   |



| Command                                      | Function   | Parameters   |
|--|--|--|
| ASTZ   | Current status   | <ul> <li>1st datum: Identification as to whether interface is<br/>active (SREM)<br/>SMAN, SREM</li> </ul>  |
|  |  | <ul> <li>2nd datum: Current status, described by one of the<br/>following function codes:<br/>SRES, SRDY, SASB, SMES, SVOP, SFPF</li> </ul>                        |
|  |  | <ul> <li>3rd datum: Paper economy mode<br/>SPSE or SPSA</li> </ul>   |
|  |  | Single-channel version:  |
|  |  | <ul> <li>4th datum: Only exists when purge active<br/>SPUL</li> </ul>  |
|  |  | Two-channel version:   |
|  |  | <ul> <li>4th datum: Sampling point<br/>SEX1 or SEX2</li> </ul>   |
|  |  | <ul> <li>5th datum: Only exists when purge active<br/>SPUL</li> </ul>  |
| AMWE Query of last<br>measurement<br>results | Query of last<br>measurement   | <ul> <li>1st datum: Sampled volume at 25 °C/1 bar [ml]<br/>(1 decimal place)</li> </ul>  |
|  | results  | <ul> <li>2nd datum: Effective sampling length [mm]<br/>(1 decimal place)</li> </ul>  |
|  |  | <ul> <li>3rd datum: Paper blackening [–]<br/>(3 decimal points)</li> </ul>   |
|  |  | <ul> <li>4th datum: Filter Smoke Number [FSN]<br/>(3 decimal points)</li> </ul>  |
|  |  | <ul> <li>5th datum: Soot concentration [mg/m<sup>3</sup>]<br/>(2 decimal points)</li> </ul>  |
|  |  | <ul> <li>6th datum: Pollution Level [%]</li> </ul>   |
| АРАР   | Queries how much paper is left                                       | <ul> <li>1st datum: Number of available single measurements<br/>rounded to the nearest 50 (without paper economy<br/>mode) available after a paper feed</li> </ul> |
| AKON   | Last soot<br>concentration<br>measured [mg/m³]<br>(2 decimal places) | <ul> <li>1st datum: Number of measurement results</li> <li>0: no data available</li> </ul>   |
| m<br>(2                                      |  | <ul> <li>2nd datum: Mean value or measurement value when<br/>number = 1</li> </ul>   |
|  |  | <ul> <li>3rd datum: 1st measurement value (if number &gt; 1)</li> </ul>  |
|  |  | <ul> <li>4th datum: 2nd measurement value (if number &gt; 1)</li> </ul>  |
|  |  | <ul> <li>5th datum: 3rd measurement value (if number &gt; 2)</li> </ul>  |
|  |  | • 6th datum: 4th measurement value (if number > 3)   |
|  |  | <ul> <li>7th datum: 5th measurement value (if number 5)</li> </ul>   |




| Command   | Function   | Parameters   |
|---|--|--|
| AFSN Last Filter Smoke<br>Number measured<br>(3 decimal places) | Last Filter Smoke<br>Number measured<br>(3 decimal places)   | <ul> <li>1st datum: Number of measurement results</li> <li>0: no data available</li> </ul>                         |
|   |  | <ul> <li>2nd datum: Mean value or measurement value when<br/>number = 1</li> </ul>                                 |
|   |  | <ul> <li>3rd datum: 1st measurement value (if number &gt; 1)</li> </ul>  |
|   |  | <ul> <li>4th datum: 2nd measurement value (if number &gt; 1)</li> </ul>  |
|   |  | <ul> <li>5th datum: 3rd measurement value (if number &gt; 2)</li> </ul>  |
|   |  | <ul> <li>6th datum: 4th measurement value (if number &gt; 3)</li> </ul>  |
|   | <ul> <li>7th datum: 5th measurement value (if number 5)</li> </ul>   |  |
|   | Important: For special measurements, $2^{nd}$ datum = mean of the last two measurements or # if outside tolerance. |  |
| APOL Last Pollution Leve<br>[%] measured (2<br>decimal places)  | Last Pollution Level<br>[%] measured (2  | <ul> <li>1st datum: Number of measurement results</li> <li>0: no data available</li> </ul>                         |
|   | decimal places)  | <ul> <li>2nd datum: Mean value or measurement value when<br/>number = 1</li> </ul>                                 |
|   |  | <ul> <li>3rd datum: 1st measurement value (if number &gt; 1)</li> </ul>  |
|   |  | <ul> <li>4th datum: 2nd measurement value (if number &gt; 1)</li> </ul>  |
|   |  | <ul> <li>5th datum: 3rd measurement value (if number &gt; 2)</li> </ul>  |
|   |  | <ul> <li>6th datum: 4th measurement value (if number &gt; 3)</li> </ul>  |
|   |  | <ul> <li>7th datum: 5th measurement value (if number 5)</li> </ul>   |
|   |  | Important: For special measurements, $2^{nd}$ datum = mean of the last two measurements or # if outside tolerance. |



| Command | Function     | Parameters   |
|---------|--------------|--|
| ASER    | Service data | Important: For special measurements, 2 <sup>nd</sup> datum = mean of the last two measurements or # if outside tolerance.                          |
|         |              | Important: These following query commands are primarily intended for purposes of service.  |
|         |              | <ul> <li>1st datum: Relative pressure [mbar]<br/>(2 decimal places)</li> </ul>   |
|         |              | <ul> <li>2nd datum: Differential pressure [mbar]<br/>(2 decimal places)</li> </ul>   |
|         |              | <ul> <li>3rd datum: Reflectometer head signal [mV]<br/>2 decimal places)</li> </ul>  |
|         |              | • The 2nd and 3rd datum are multiplexed; the 3rd datum is only relevant when the reflectometer head bulb is switched on.                           |
|         |              | <ul> <li>4th datum: Lamp current [mA]</li> <li>1 decimal place)</li> </ul>   |
|         |              | <ul> <li>5th datum: Electronics temperature [°C]<br/>(1 decimal place)</li> </ul>  |
|         |              | <ul> <li>6th datum: Electronics temperature [°C]<br/>(1 decimal place)</li> </ul>  |
|         |              | <ul> <li>7th datum: Heating fan temperature [°C]<br/>(1 decimal place)<br/>(Heating Option)</li> </ul>   |
|         |              | <ul> <li>8th datum: Air temperature (mechanics) [°C]<br/>(1 decimal place)<br/>(Heating Option)</li> </ul>   |
|         |              | <ul> <li>9th datum: Heating hose temperature [°C]<br/>(1 decimal place)<br/>Sample 1<br/>(Heating Option)</li> </ul>                               |
|         |              | <ul> <li>10th datum: Heating hose temperature [°C]<br/>(1 decimal place)<br/>Sample 2<br/>(Heating and Two-Chappel Measurement Options)</li> </ul> |
|         |              | <ul> <li>11th datum: Ventilator current for electronics chamber<br/>[mA]<br/>(1 decimal place.)</li> </ul>   |
|         |              | <ul> <li>12th datum: Ventilator current for hot-air ventilator [mA]<br/>(1 decimal place)<br/>(Heating Option)</li> </ul>                          |
|         |              | <ul> <li>13th datum: Supply voltage 24 V [V]<br/>(2 decimal places)</li> </ul>   |
|         |              | • 14th datum: Subsystem status byte (decimal 0 127)  |



| Command | Function   | Parameters   |                            |                         |                               |                       |       |        |       |     |  |
|---------|--|--|----------------------------|-------------------------|-------------------------------|-----------------------|-------|--------|-------|-----|--|
|         |  | Bit  | 7                          | 6                       | 5                             | 4                     | 3     | 2      | 1     | 0   |  |
|         |  | Description  |                            | v4                      | v3                            | v                     | v     | Р      | m     | Ι   |  |
|         |  | Bit Description  |                            |                         |                               | Meaning<              |       |        |       |     |  |
|         |  | 0  | Ι                          | I                       |                               |                       |       |        |       |     |  |
|         |  | 1  | m                          | m                       |                               |                       |       |        |       |     |  |
|         |  | 2  | Ρ                          | P                       |                               |                       | Pump  |        |       |     |  |
|         |  | 3 6  | V1                         | V2                      | 1                             |                       | Valve | 91     | 4     |     |  |
|         |  | 7  |                            |                         |                               |                       | No fu | nctior | ١     |     |  |
|         |  | 0 deactivat<br>Important: Th<br>if the heating   | ted, 1<br>e valu<br>optior | ac<br>les of<br>h is no | tivated<br>the da<br>ot insta | d<br>ata fo<br>alled. | r 7   | 10 ar  | e >13 | 0°C |  |
| AOPT    | Last values<br>measured by the<br>optical system | <ul> <li>1st datum: Paper blackening [-]<br/>(3 decimal points)</li> <li>2nd datum: White value [mV]<br/>(1 decimal place)</li> <li>3rd datum: Gray value [mV]<br/>(1 decimal place)</li> <li>4th datum: Black value [mV]<br/>(1 decimal place)</li> </ul>   |                            |                         |                               |                       |       |        |       |     |  |
| AEVL    | Last effective<br>volume and length<br>measured  | <ul> <li>1st datum: Sampled volume [ml] referred to 1 bar and 25 °C</li> <li>2nd datum: Effective length [mm]</li> </ul>   |                            |                         | and                           |                       |       |        |       |     |  |
| AIZU    | Internal status data                             | <ul> <li>1st datum: Active operating point<br/>M: manual (no interface)<br/>Neither interface COM1 nor interface COM2 is active.<br/>F: remote control (AVL 4210) active (own interface)<br/>This interface in <i>F</i> status is the active master/control<br/>interface.<br/>R: Remote active (other interface)<br/>The other interface is active, the interface in <i>R</i> status is<br/>acting as the "slave". Only query commands are<br/>permitted.</li> <li>2nd datum: Sampling point<br/>0: one-channel<br/>1: sampling point 1<br/>2: sampling point 2</li> <li>3rd datum: Internal status number<br/>0: power up</li> </ul> |                            |                         | tive.<br>e)<br>rol<br>itus is |                       |       |        |       |     |  |



| Command | Function | Parameters  |  |
|---------|----------|---|--|
|         |          | <ol> <li>error</li> <li>set ready</li> <li>ready</li> <li>ready with purge</li> <li>device test, reflectometer head calibration or leak<br/>check</li> <li>volume check</li> <li>determination of paper blackening</li> <li>standard measurement</li> <li>set ready for sampling</li> <li>ready for sampling</li> <li>synchronized measurement</li> <li>external trigger delay</li> <li>paper feed</li> </ol> |  |
|         |          | <ul> <li>4th datum: Preselected operating mode</li> <li>V: preselected volume</li> <li>Z: preselected time</li> <li>5th datum: Preselected (setpoint) sampled volume [m]]</li> </ul>  |  |
|         |          | with preselected volume or preselected (setpoint)<br>sampling time [s] with preselected time  |  |
|         |          | <ul> <li>6th datum: Setpoint number of samples</li> <li>7th datum: Actual number of samples (for single measurement always 0)</li> </ul>  |  |
|         |          | <ul> <li>8th datum: Result type</li> <li>0: no measurement result available</li> <li>1: paper blackening</li> <li>2: sampled volume after volume check</li> <li>3: single measurement</li> <li>4: multiple measurement</li> <li>5: multiple measurement with mean value</li> </ul>  |  |
|         |          | <ul> <li>9th datum: Service mode (data backup off)</li> <li>0: deactivated</li> <li>1: activated</li> </ul>   |  |
|         |          | <ul> <li>10th datum: Paper economy mode</li> <li>0: deactivated</li> <li>1: activated</li> </ul>  |  |



| Command | Function   | Parameters  |
|---------|--|---|
| AIDA    | Internal data  | <ul> <li>1st datum: Paper blackening [-](3 decimal points)</li> <li>2nd datum: White value [mV](1 decimal place)</li> <li>3rd datum: Gray value [mV](1 decimal place)</li> <li>4th datum: Black value [mV](1 decimal place)</li> <li>5th datum: Sampled volume [ml] referred to 1 bar and 25 °C</li> <li>6th datum: Sampled volume [ml] referred to parameterized ambient pressure and temperature measured internally</li> <li>7th datum: Effective length [mm]</li> <li>8th datum: Leak [ml/100 mbar * s] (1 decimal place)</li> </ul>  |
| AADC    | ADC data   | <ul> <li>1st datum: Differential pressure [mbar]</li> <li>2nd datum: Relative pressure + "parameter-izable"<br/>(can be a parameter) ambient pressure [mbar]</li> <li>3rd datum: Measured orifice plate temperature [°C]<br/>(1 decimal place)</li> <li>4th datum: Supply voltage V [V]<br/>(2 decimal places)</li> </ul>   |
| AVKF    | Volume correction<br>factor (new value<br>refers to the ambient<br>temperature after<br>completed volume<br>check) | <ul> <li>Only permissible after previous volume check (SVOP).<br/>This query command requires two input parameters: <ul> <li>1st datum: Externally measured volume [ml]</li> <li>2nd datum: Externally measured temperature [°C]</li> </ul> </li> <li>Return parameters: <ul> <li>1st datum: Externally measured volume [ml]</li> <li>(same as first datum entered)</li> </ul> </li> <li>2nd datum: Internally measured volume [ml] referred to "parameter-izable" (can be a parameter) ambient pressure and entered ambient temperature</li> <li>3rd datum: Recalculated volume correction factor [-] (3 decimal places) If necessary, save this value with SKOR or enter it with ECPA.</li> </ul> |
| ABSZ    | Operating hour<br>counter – counts the<br>operating hours of<br>the pump   | <ul> <li>1st datum: Total operating time [h]<br/>(cannot be reset)</li> <li>2nd datum: Operating time [h] since the last reset<br/>(SBSZ)</li> </ul>  |



## 6.1.7 Example

In the example below, two measurements are carried out with a preselected time (sampling time 6 s), in remote mode. This example applies to an AVL 415 smoke meter, which is used as the example test device in this procedure.

| Computer  | Acknowledgment   |  |  |  |  |
|---|--|--|--|--|--|
| The following applies in general: if the<br>error status > 0, query the current error<br>number:<br><stx> ASTF<etx></etx></stx> | <stx> ASTF 1 30<etx><br/>Error code 30 – filter paper installed incorrectly or<br/>no paper</etx></stx>  |  |  |  |  |
| Switch AVL 415S to Remote mode:<br><stx> SREM<etx></etx></stx>  | <stx> SREM 0<etx></etx></stx>  |  |  |  |  |
| Query device status:<br><stx> ASTZ<etx></etx></stx>   | <stx> ASTZ 0 SREM SRDY SPSA<etx></etx></stx>   |  |  |  |  |
| Set measurement cycle:<br><stx> EMZY Z 6.0 2<etx></etx></stx>   | <stx> EMZY 0<etx></etx></stx>  |  |  |  |  |
| Set device to Ready for Sampling<br>status:<br><stx> SRDY<etx></etx></stx>  | <stx> SRDY 0<etx></etx></stx>  |  |  |  |  |
| Sample:<br><stx> SMES<etx></etx></stx>  | <stx> SMES 0<etx></etx></stx>  |  |  |  |  |
| Check whether sampling is still running<br>(query periodically):<br><stx> ASTZ<etx><br/><stx> ASTZ<etx></etx></stx></etx></stx> | <stx> ASTZ 0 SMES SPSA<etx><br/><stx> ASTZ 0 SRDY SPSA<etx><br/>Measurement finished, device is ready for<br/>measurements again; it is purged for up to 60 s.</etx></stx></etx></stx> |  |  |  |  |
| Query the measured soot concentration in<br>FSN:<br><stx> AFSN<etx></etx></stx>   | <stx> AFSN 0 2 3.205 3.224<br/>3.186<etx><br/>Two samples taken.<br/>Mean value: 3.205 FSN<br/>1st measurement value: 3.224 FSN<br/>2nd measurement value: 3.186 FSN</etx></stx>       |  |  |  |  |

## Table 11: Example Measurements

## 6.2 General ASCII Synchronous Protocol

If your test device uses the GENSYNC or another protocol, refer to the device's manufacturer's manual for details about the protocol and commands. Most test cell configurations using DevCom use that protocol. Refer to *Section 6.1 Generic Communication AK Protocol* on page 55.