Project documentation

CAQ 2002 Production monitor

for Windows 9x, 2000 and NT 4

Author:	**Lars Graeve**
Department:	**APB 3.4** **System development & technical software**

Table of contents

History of this document

Version 1.0 25.04.2003
Old documentation from August 2002

Version 2.0 25.04.2003
New format
FTP section
Overview of the record fields (in English)

Version 2.1 02.02.2004
`CAQ.CountRec Parameter correction`
`fTestTime As Single to dTestTime As Double`

Version 2.2 03.02.2004

27	test program index	At moment integer 32767, for future dll release: long integer (32 bit) version of the test program and the INI file in the format MMmmRRRIII (“316372051”) =	[if test status “N_” or “W_”]
	or OFD (opportunity for defects)	program version: 3.16.372, INI version: 51 number of possible errors (“137”)	[if test status “P_”]

Version 2.3 09.02.2004

- Protocol update discussed and fixed with HDT5

27	test program index	long integer (32 bit) version of the test program and the INI file in the format MMmmRRRIII (“316372051”) = program version: 3.16.372, INI version: 51 numbers up to 2147483647 are supported!!!	[if test status “N_” or “W_”]
	or OFD (opportunity for defects)	Number of possible errors (“137”)	[if test status “P_”]

- Entry correction Tester_Mode to Test_Mode (Page 10 + 14).

`TEST_MODE=PRE` (Pretest)
`TEST_MODE=END` (Endtest)
`TEST_MODE=RIH` (Runin hot)

- Build update from 1.0.07 to 1.00.0011

  Current Version of CAQ

  Version No: 1.00.0011
  Date: 11.02.2004

Chapter 1 – About CAQ 2002

Current Version of CAQ

Version No: 1.00.0011
Date: 11.02.2004

Description

The module CAQSERV.DLL contents routines for sending production data from PC-controlled test machines to a network server. The ActiveX-DLL can be integrated into a Visual Basic project since version 5 by selecting the pull-down menu “Project → Reference” (“Projekt → Verweise”). CAQ 2002 can be used under Windows 95, 98, 2000 and NT 4.

Modules and files

CAQSERV.DLL	(Class file)
CAQSERV.INI	(Configuration file)
CAQSHIFT.INI	(Configuration file)
CAQSECU.INI	(Temporary File)

Properties

FailNo As String	(Failure number)
FailText As String	(Failure description)
FailValue As String	(current value)
FailValInfo As String	(additional information; e.g. limits, units)
FailItemNo As String	(Test item number)

Functions

Init	(for further information see Chapter 2 – Functions)
SetPar	
CountRec	
WriteRec	

1.2 – Example for using CAQ

```
Public CAQ As New CAQ2002

Sub MAIN()
    Call CAQ.Init(sArtNo, sCAQIniFile)

    Do
        fStart = Timer                                  'Calculate necessary test time
        Call TestProgram                                'considering the day change
        fTestTime = Timer - fStart
        If fTestTime < 0 Then fTestTime = fTestTime + 86400)

        nIniFileVersion = ...                           'Version of ini file form article.ini
        sProgIndex = Trim$(Str$(App.Major * 100000000 + App.Minor * 1000000 + _
          App.Revision * 100 + nIniFileVersion))        'compose program index
        CALL CAQ.SetPar("PROGRAM_INDEX", sProgIndex)

        CALL CAQ.SetPar("SER_NUM", "00004711")          'Optional

        If part = False Then                            'Part is faulty
            nOKFlag = False
            CAQ.FailNo = "failure number"
            CAQ.FailText = "failure description"
            CAQ.FailValue = "actual value"
            CAQ.FailValInfo = "add. inform."            'Optional (e.g. limits, units)
            CAQ.FailItemNo = "test item number"
        Else                                                    'Part is OK
            nOKFlag = True
        End If

        Call CAQ.CountRec(nOKFlag, dTestTime)
    Loop Until bExitProgram = True

    Call CAQ.WriteRec
End Sub
```

Chapter 2 – Functions

Following chapter describes the functions of the CAQ-2002-DLL.

2.1 – CAQ.Init

Description

Reads parameters from configuration files and initialisesthe CAQ module

Syntax

```
Call CAQ.Init(sArtNo, sCAQIniFile)
```

Source code

CAQSERV.BAS

Parameters passed to the routine

`sArtNo As String`	Article number
`sCAQIniFile As String`	Name of the configuration file including the path

Parameters returned from the routine

None

2.2 – CAQ.SetPar

Description

Sets special parameters in the CAQ module

Syntax

```
Call CAQ.SetPar(sTarget, sValue)
```

Source code

CAQSERV.BAS

Parameters passed to the routine

`sTarget As String`	Name of the parameter to be set, possible values are: ARTICLE_NAME CUSTOMER_ART_NO LOCATION COST_CENTRE PROD_LNE TESTER_NO TESTER_TYPE TEST_MODE TEST_STATUS IDENT SER_NUM PROGRAM_INDEX READINGS_EXIST DOC_LEVEL TEST_ITEM_NO RESERVE_1 RESERVE_2 (in future:) GOODPARTS BADPARTS CAQCOUNT DATE SHIFT
`sValue As String`	Value to be set

Parameters returned from the routine

None

Example

```
Call CAQ.SetPar("COST_CENTRE", "2821")
Call CAQ.SetPar("TEST_MODE", "ICT")
Call CAQ.SetPar("TEST_STATUS", "NR")
Call CAQ.SetPar("SER_NUM", "08154711")
Call CAQ.SetPar("PROGRAM_INDEX", "208013432")
```

2.3 – CAQ.CountRec

Description

Buffers CAQ information and sends data to the network server if one of the sending conditions is fulfilled. It is usually used after each performed test.

Syntax

```
Call CAQ.CountRec(nTestOK, dTestTime)
```

Source code

CAQSERV.BAS

Parameters passed to the routine

`nTestOK As Integer`	True: test passes False: test failed
`dTestTime As Double`	Necessary test time (pure test time without handling time)

Parameters returned from the routine

None

2.4 – CAQ.WriteRec

Description

Send actually buffered data to server and resets all counters in the CAQ module. Usually used when program is terminated

Syntax

```
Call CAQ.WriteRec()
```

Source code

CAQSERV.BAS

Parameters passed to the routine

None

Parameters returned from the routine

None

Chapter 3 – Configuration file CAQSERV.INI

Description

Contains the configurations for CAQSERV.BAS. If it is only in the root directory it will be valid for all programs on the test machine. If it is found in the program directory (default) this file will have a higher priority. Those setting will be used. Most of the parameters can be overwritten by using the function CAQ.SetPar.

The file consists of a common section, a FTP section and optional article specific sections. Setting made in the article specific sections will overwrite those from the common section.

The structure of the CAQSERV.INI is like that:

```
[COMMON]
...
...
[FTP]
...
...
[03750010]
...
...
[NEXT ART_NO]
...
...
```

3.1 – Common section

This section is introduced by the `[COMMON]` tag. It contains following settings.

CAQ_CFG_FIL

Pointer to another configuration file. It makes it possible to hold all configuration files in one central directory. Example:

```
CAQ_CFG_FILE=F:\PC-TEST\CAQDATA\CAQSERV.INI
```

SHIFT_X_START

Set the start time of the shifts. `SHIFT_1_START` defines the start time of the first shift, `SHIFT_2_START` that one of the second shift and `SHIFT_3_START` the start time of third. Example:

```
SHIFT_1_START=6:00
SHIFT_1_START=14:00
SHIFT_1_START=22:00
```

SHIFT_CFG_FIL

Pointer to another configuration file containing shift information. If file does not exist the setting from this file will be used (see above). Example:

```
SHIFT_CFG_FILE=F:\PC-TEST\CAQDATA\CAQSHIFT.INI
```

LOCATION

Sets the location of the production area. Limited on 5 characters. Example:

```
LOCATION=BELL
```

All possible abbreviations are `BELL`, `HALV`, `MZHGN`, `BBLGN`, `KOI`, `KOB`, `KITA`, `KOSPA`, `KOCR`, `KOBRA`, `KOMEX`, `KOCHI`, `KOA`.

COST_CENTRE

Defines the cost centre of the production line. Limited on 4 characters. Example:

```
COST_CENTRE=1234
```

PROD_LINE

Number or ID of the production line. Limited on 4 characters. Example:

```
PROD_LINE=1
```

TESTER_NO

Number of the test machine. 3 (default) to 7 characters. Example:

```
TESTER_NO=001
```

TESTER_TYPE

That identify the type of the test system. Possible values are for example:

```
TESTER_TYPE=PCPG3
TESTER_TYPE=PCPG4
TESTER_TYPE=EMPPC
```

Limited on 5 characters. All possible abbreviations are PGPG3, PGPG4, PCPGM, EMPPC, ICT, INTBX, MCDNT, WFCPC, SPSPC, OPTIC.

TEST_MODE

Sets the test mode. Limited on 3 characters. Examples:

```
TEST_MODE=PRE          (Pretest)
TEST_MODE=END          (Endtest)
TEST_MODE=RIH          (Runin hot)
```

(*also see 4.1 – Overview of the record fields)*

IDENT

Identification, e.g. charge number. Limited on 5 characters. Example:

```
IDENT=ABCDE
```

CAQ_COUNT

Number of good resulting tests between writing the data. Example:

```
CAQ_COUNT=1
```

CAQ_TIMEOUT

Time in seconds for writing data on the network server or on local hard disk. Example:

```
CAQ_TIMEOUT=2
```

CAQ_PATH

Path on the network server or on local hard disk. Example:

```
LOCATION=F:\PC-TEST\CAQDATA
```

CAQ_EXTENSION

Extension used for the data files. Example (default):

```
CAQ_EXTENSION=PDT
```

CAQ_DAILY_SERNUM

Enabling of daily serial numbering. The two possible values are:

`CAQ_DAILY_SERNUM=ALL`	(A serial number will be generated automatically for each part)
`CAQ_DAILY_SERNUM=BAD`	(A serial number will be generated only for each bad part)

Any other value disables the daily serial numbering.

TEMP_FILE

Name and path of the temporary file used to store data if no connection to the network server is available. Example:

```
TEMP_FILE=C:\CAQTEMP.INI
```

Set `NO` for no temporary storing.

SECU_ENABLED

Flag for enabling buffering of module data on local hard disk after each test.

```
SECU_ENABLED=1     (Enable, default)
SECU_ENABLED=0     (Disable)
```

SECU_FILE

File in for buffering module data. Example (default):

```
SECU_FILE=C:\CAQSECU.INI
```

3.2 – FTP section

This section is introduced by the `[FTP]` tag. It contains following settings.

FTP_PATH

Path on the FTP server. Example (default):

```
FTP_PATH=./
```

TRANSMITCOUNTER

Number of writing proceedings between sending the local file to the server. Example

```
TRANSMITCOUNTER=5
```

NUMDESTINATIONS

Number of possible FTP destinations on the network server. If primary destination is not available, other destinations can be used. Example:

```
NUMDESTINATIONS=2
```

DESTINATIONX

Write the number of destination instead of X. The first destination has the keyword `DESTINATION1`, the second one `DESTINATION2`, and so on. Example:

```
DESTINATION1=ftp://linectrl
DESTINATION2=ftp://linectrl2
```

USERX

X must be substituted by the number of the user. Example:

```
USER1=CAQ          (standard user for ftp connections)
```

PASSWORDX

Substitute X by the number of the password. Example:

```
PASSWORD1=CAQ      (standard password for ftp connections)
```

FASTLOGINX

Sets for the destination number X that the login should be fast.

```
FASTLOGIN1=xzy
FASTLOGIN2=xzy
```

3.3 – Article specific sections

In these section you can make settings for a specific article. The section name is the article number.

ARTICLE_NAME

Name of the article. Limited on 50 characters. Example:

```
ARTICLE_NAME=door module
```

CUSTOMER_ART

Set the customer article number here. Limited on 20 characters. Example

```
CUSTOMER_ART=218 821 10 51
```

TEST_MODE

Sets the test mode. Limited on 3 characters. Examples:

```
TEST_MODE=RIC          (Runin cold)
TEST_MODE=PAN          (Panel test)
```

(*also see 4.1 – Overview of the record fields*)

IDENT

Identification (e.g. charge number) for distinguishing of records (for future use). Limited on 5 characters. Example:

```
IDENT=VWXYZ
```

CAQ_COUNT

Number of good resulting tests between writing the data. Example:

```
CAQ_COUNT=2
```

CAQ_DAILY_SERNUM

> **ATTENTION:**
> Will not be supported in future! Use “SerPar”-Function for setting serial numbers.

Enables or disables the daily serial numbering. The two possible values are:

```
CAQ_DAILY_SERNUM=ALL    (A serial number will be generated automatically for each part)
CAQ_DAILY_SERNUM=BAD    (A serial number will be generated only for each bad part)
```

For disabling the creation of daily serial numbers set any other value.

3.4 – Example of the configuration file CAQSERV.INI

For further information about the settings please read chapter 3.1 to 3.3.

```
'*************************************************************************
'
'                 configuration file for module CAQSERV.BAS
'
'*************************************************************************

[COMMON]

CAQ_CFG_FILE=                          'pointer to other configuration file
                                       'e.g. on file server

SHIFT_1_START=6:00                     'shift 1 start time
SHIFT_2_START=14:00                    'shift 2 start time
SHIFT_3_START=22:00                    'shift 3 start time

SHIFT_CFG_FILE=C:\PC-TEST\CAQDATA\CAQSHIFT.INI
                                       'pointer to other configuration file with
                                       'shift information if file does not exist,
                                       'information in this file will be used

LOCATION=HALV                          'location        (max. 5 char.)
COST_CENTRE=2821                       'cost centre     (max. 4 char.)
PROD_LINE=0                            'production line (max. 4 char.)

TESTER_NO=300                          'tester no.      (max. 3 char.)
TESTER_TYPE=PCPG3                      'tester type     (max. 5 char.)
TEST_MODE=END                          'test mode       (max. 3 char.)
IDENT=                                 'identification  (max. 5 char.)

CAQ_COUNT=2                            'tests between writing CAQ data
CAQ_TIMEOUT=2                          'timeout for error free writing of data on
                                       'the network server or on local hard disk

CAQ_SERVER=BENFV-S                     'CAQ server name for automatic login
CAQ_PATH=C:\PC-TEST\CAQDATA                'path on network server or on local hard
disk
CAQ_EXTENSION=PDT                      'extension of data files

CAQ_DAILY_SERNUM=ALL                   'ALL: daily serial number is activated for all
                                       'parts
                                       'BAD: activated only for the bad parts
                                       'other: no daily serial number

TEMP_FILE=C:\CAQTEMP.INI               'temporary file, if no connection to server
                                       ' =NO -> no temporary storing

SECU_ENABLED=0                         'enable flag
SECU_FILE=C:\CAQSECU.INI               'security file, where actual records are
                                       'buffered


[FTP]
FTP_Path=./                            'path on the FTP server
TransmitCounter=5                      'interval for sending data

NumDestinations=1                      'number of destinations
Destination1=ftp://linectrl            'parameters for the destinations
User1=caq
Password1=caq
FastLogin1=1
```

```
[03750010]
ARTICLE_NAME=door module
                                    'Article name
CUSTOMER_ART_NO=123 456 78 90       'customer article number

TEST_MODE=END                       'test mode       (max. 3 char.)

CAQ_COUNT=2                         'tests between writing CAQ data

CAQ_DAILY_SERNUM=ALL                'ALL: daily serial number is activated for all
                                    'parts
                                    'BAD: activated only for the bad parts
                                    'other: no daily serial number


[03750011]
ARTICLE_NAME=window lifting module
CUSTOMER_ART_NO=000 815 47 11
...
```

Chapter 4 – Record format for production monitor

For successful importing of the CAQ data to the production monitor it is important to keep strict on this rules:

1. Data must be enclosed in quotation marks ("..." / ASCII 34).
2. If quotation marks are used in the data (e.g. error description), it must be doubled.
3. The fields are separated by semicolon (; / ASCII 59)
4. If a field is empty, write two semicolons (...;...;;... / field 3 is empty)
5. Each record is to be terminated by carriage return and line feed (CR/LF, ASCII 13, 10)
6. It is not allowed to exchange fields.

4.1 – Overview of the record fields

No.	Name / Using	Format Contents (Example)
1	date	exact 10 characters date of record in the Format DD.MM.YYYY ("30.01.2003")
2	article number	at most 15 characters ("08154711")
3	article name	at most 50 characters ("Seat memory")
4	customer article number or user ID	at most 20 characters ("218 821 10 51") [if test status "N_" or "P_"] user who enters test results ("Miller01") [if test status "W"]
5	location	at most 5 characters ("BELL") = Bellmerei
6	cost centre	at most 4 digits ("1234")
7	production line	at most 4 digits ("1")
8	tester number	at most 5 digits ("301")
9	tester type	at most 10 characters ("PCPG3") = PC-controlled testing machine 3
10	test mode	at most 3 characters ("PRE") = pretest [test status is "N"] ("END") = endtest [test status is "N"] ("RIH") = runin hot [test status is "N"] ("RIC") = runin cold [test status is "N"] ("ICT") = in-circuit-test [test status is "N"] ("VIS") = visual inspection [test status is "N"] ("AUD") = audit inspection [test status is "N"] ("AX1") = axial inspection [test status is "P"] ("RA1") = radial inspection [test status is "P"] ("CB1") = combined inspection [test status is "P"] ("GL1") = glue inspection [test status is "P"] ("PA1") = paste inspection [test status is "P"] ("RE1") = reflow inspection [test status is "P"] ("SE1") = sealing inspection [test status is "P"] ("WA1") = wave inspection [test status is "P"]
11	test status	at most 2 characters first character: ("N_") = normal test ("P_") = process supervision ("W_") = weekly report ("A_") = Analysis second character: ("_R") = retest ("_T") = test error ("NR") = retest of a normal test
12	faulty test step number	at most 10 characters

No.	Name / Using	Format Contents (Example)	
		according test specification (“80.1”)	
13	faulty test step description	at most 80 characters according test specification (“Establishing CAN bus communication”)	
14	identification	at most 5 characters (“46512”)	
15	measured value or component position or comment (part 1/3)	at most 50 characters measured values without units (“13.2”) (“C12”) any kind of comment (“...”)	 [if test status “N_”] [if test status “P_”] [if test status “W”]
16	additional information for measured value or comment (part 2/3)	at most 50 characters units, limits, etc. (“≥ 15.0 & ≤18.2 [V]”) any kind of comment (“...”)	 [if test status “N_” or “P_”] [if test status “W”]
17	flag marking existence of measured values	at most 1 digit (“0”) = yes or (“1”) = no	
18	level of documentation required	at most 1 digit (“0”) to (“5”)	
19	number of good parts (shift 1)	long integer (32 bit) (“29834”)	
20	number of good parts (shift 2)	long integer (32 bit) (“98743”)	
21	number of good parts (shift 2)	long integer (32 bit) (“61248”)	
22	number of faulty parts (shift 1)	long integer (32 bit) (“341”)	
23	number of faulty parts (shift 2)	long integer (32 bit) (“490”)	
24	number of faulty parts (shift 3)	long integer (32 bit) (“398”)	
25	sum of test time for good parts	double (“31.541”)	
26	sum of test time for faulty parts	double (“43.193”)	
27	test program index or OFD (opportunity for defects)	long integer (32 bit) version of the test program and the INI file in the format MMmmRRRIII (“316372051”) = program version: 3.16.372, INI version: 51 numbers up to 2147483647 are supported!!! number of possible errors (“137”)	 [if test status “N_” or “W_”] [if test status “P_”]
28	serial number	at most 20 characters (“12345678912345”)	
29	number of inspection feature	at most 10 characters (“39”)	
30	reserved 1	at most 30 characters additional information to the record (“$F”) = Record contains follow-up error, in the production monitor only the error but not another faulty part is counted (“$Q”) = fields 15, 16 and 31 used as comment	 [if test status “P_”] [if test status “W_”]
31	reserved 2 or comment (part 3/3)	at most 30 characters no use defined yet any kind of comment (“...”)	 [if test status “W_”]