

Klippel QC Output Switching Using TN02 Automation API and VBScripting

Technical Note for the KLIPPEL QC SYSTEM (Document Revision 1.0)

1 Application

In this particular example application a status lamp for QC application shall be controlled for use with Klippel QC. The lamp shall indicate the pass/fail result for 1 second right after the test.

Although it is easy to set the output pins of the PA hardware unit directly from the QC software depending on the test result (using *IO-Task*), it is not possible switching it off after a defined time using the built in output control.

In this example, pin 17 shall be on for 1 second if PASS, pin 3 shall be on for 1 second if FAIL.

2 Solution

The solution is the control of the output pins using a direct access of the *Production Analyzer* hardware by Klippel *Automation API*.

This TechNote shows an approach using VBS. VBScripting is just an example; the automation interface may also be used from C++, C# and other languages.

Basically the control software directly uses the memory mapped control interface of the *Production Analyzer* hardware. Be careful with individual extensions beyond output control. You may easily corrupt the functionality of the hardware and make it dysfunctional with Klippel QC software.

Note that switching off the output is done in background; it does not extend the test time!

3 Details on Output Access

3.1 Enumeration

All connected Klippel devices can be listed using the command

```
Dim devenum
Set devenum = CreateObject("KlAutomation.KlDeviceEnum")
```

3.2 Device Type:

Klippel *QC Production Analyzer* exists in two versions:

```
const eDevTypeKGate      = 5      ' Production Analyzer Firewire+USB
const eDevTypePA         = 10     ' Production Analyzer USB only
```

3.3 Output Pin Reference

| QC Label | GPIO connector Pin | Address | Value (Bitmask) |
|----------------|--------------------|---------|-------------------------|
| | | | Info: C: ClearS: Set |
| STATE | 13 | 0x00A8 | {xxxxxxCS} |
| FINISH | 25 | 0x00A8 | {xxxxCSxx} |
| OUT 1 / Result | 24 | 0x00A8 | {xxCSxxxx} |
| OUT 2 / Temp | 11 | 0x00A8 | {CSxxxxxx} |

| | | | |
|-------|----|--------|------------|
| Login | 18 | 0x00A9 | {xxxxxxCS} |
| OUT 3 | 5 | 0x00A9 | {xxxxCSxx} |
| OUT 4 | 17 | 0x00A9 | {xxCSxxxx} |
| OUT 5 | 4 | 0x00A9 | {CSxxxxxx} |
| OUT 6 | 16 | 0x00AA | {xxxxxxCS} |
| OUT 7 | 3 | 0x00AA | {xxxxCSxx} |
| OUT 8 | 15 | 0x00AA | {xxCSxxxx} |
| OUT 9 | 2 | 0x00AA | {CSxxxxxx} |

3.4 Output Control Example:

Switch Pin 3 on:

Write at Address = 0x00A8 the Value = 0x04 (binary value = 0000 0100)

Switch Pin 3 off:

Write at Address = 0x00A8 the Value = 0x08 (binary value = 0000 1000)

3.5 Example VB Script

A full example of software is included in the tech note zip-archive. If the software is not available, find a reduced example here.

This is a batch file for VBS launching. It is required for using vb scripts on 32 and 64 bit computer systems:

```
@echo off
set OPTS="%~dp0dev.vbs"
if "%PROCESSOR_ARCHITECTURE%"=="AMD64" goto 64BIT
cscript.exe %OPTS%
goto END
:64BIT
%SystemRoot%\SysWOW64\cscript.exe %OPTS%
:END
```

The actual VBScript file controlling output pins (may be modified to achieve different Pass/Fail setting):

```
' Declare constants KlDeviceType
const eDevTypeKGate = 5 ' Klippel Gate (Production Analyzer)
const eDevTypePA = 10 ' PA USB only

Function RunDevCommand(dev, command)
    Dim result, errmsg
    dev.RunCommandVBS command, result, errmsg
    If Not (errmsg & "" = "") Then
        result = "ERROR: " & errmsg
    End If
    RunDevCommand = result
End Function

'----- MAIN PROGRAM -----
Dim devenum, devFound
Set devenum = CreateObject("KlAutomation.KlDeviceEnum")
devFound = 0
For Each dev In devenum.Devices
    If dev.Type = eDevTypeKGate Or dev.Type = eDevTypePA Then
```

```

devFound = devFound + 1
r = RunDevCommand(dev, "write 0x00AA 0x04") ' Set digital output pin 3
r = RunDevCommand(dev, "write 0x00A9 0x10") ' Set digital output pin 17
WScript.Sleep 1000
r = RunDevCommand(dev, "write 0x00AA 0x08") ' Clear digital output pin 3
r = RunDevCommand(dev, "write 0x00A9 0x20") ' Clear digital output pin 17
End If
Next
If devFound=0 Then
    WScript.Echo "No Production Analyzer Connected"
End If
    
```

4 Integration Into Klippel QC

In the test sequence an *IO-Task* shall be inserted at the end. Configure it to run the batch-file for either PASS or FAIL verdict:

| | |
|--|--|
| | <ul style="list-style-type: none"> • Define “if (condition)” as “All tests passed” • Define batch file for PASS in the „Then (Action)” • Uncheck „Wait for completion” • Check option „run silent” |
| | <ul style="list-style-type: none"> • Define batch file for FAIL in the „Else (Action)” • Uncheck „Wait for completion” • Check option „run silent” |

5 Possible Extensions

5.1 Timing Issues

In the given example the duration of the lamp shall be less than the cycle time of the production. If the duration is longer, you may get interference pattern. That means that a pending command may switch off a lamp that was switched on by a following test. It results in unpredictable and short duration of PASS/FAIL indication.

Since the called batch files are fully independent and have no common state information, an external synchronization is required to avoid such interference.

One option to solve this problem is using the registry. The following functions may be added to the VBS command files:

- When switching the PASS lamp on, get a time stamp and store it in a local variable in VBS and also in a registry key. Do not check if there is any other time stamp stored.

- When switching the PASS lamp off, check, if the above mentioned registry key still holds the stored time stamp.
 - If not, the PASS lamp was switched on in between from another run (batch file). In this case nothing shall be done!
 - If so, switch off the PASS lamp. You may now delete the time stamp from registry indicating that there is no switched on lamp pending.
- Do the same with different registry key for the FAIL lamp.

5.2 Defined State of Lamps at Measurement Start

It is good practice to switch off all lamps when a new test is triggered. This can be easily done using the direct output access of the digital interface. Please refer to the *QC User Manual* section *Setting Output Pins and Delay*.

6 References

| | |
|--------------|---|
| Help | Klippel Automation Documentation Package Manual IO-Task Manual QC-System |
| Software | Batch files, VBS files Example test database: SetOutput.kdbx For QC4.1 only: da1cc.dll must be copied into "C:\Program Files (x86)\Klippel\QC\bin" |
| Requirements | QC 4.1 or higher New da1cc.dll must be copied into "C:\Program Files (x86)\Klippel\QC\bin" |

Find explanations for symbols at:

<http://www.klippel.de/know-how/literature.html>

Last updated: May 20, 2022

