



# SW-Tools TRIO

Addendum

New functionalities

02

## Table of contents

1.	Excel printer .....	4
1.1	Report page layout sections .....	4
1.1	Setting up the Excel printer .....	5
1.1.1	Output to file.....	5
1.1.2	View output with.....	5
1.1.3	Definition file.....	5
1.2	The definition file .....	5
1.2.1	[LINES] defines which print areas to include .....	5
1.2.2	[HEADLINE] column names .....	6
1.2.3	[POINT] decimal point.....	6
1.2.4	[DATE] date format .....	6
1.2.5	[PING] mark alpha fields with ' .....	7
1.2.6	[QUOTE] enclose alpha fields in “...” .....	7
1.2.7	[CRLF] defines the newline character(s) .....	7
1.3	Controlling the printer from the calculations .....	7
1.3.1	Using the Excel printer with manual control through the PRINT statement.....	7
1.3.2	Using the Excel printer as COPIES printer .....	7
1.3.3	Using the Excel printer as secondary printer .....	7
1.3.4	PIP print archiving of the Excel print.....	7
2.	Barcode and QR code printing .....	8
2.1	Barcodes.....	8
2.2	QR codes .....	8
3.	Test functions for programmers .....	9
3.1	DIALOG function.....	9
3.2	WIF- functions.....	9
3.2.1	WIFA Testprint of Alpha field .....	9
3.2.2	SSV Making a set of fields as a SSV string .....	9
3.2.3	Filename for the WIFA testprint .....	9
3.2.3	Building Excel files using SSV and WIFA .....	9
3.2.4	WIFD Testprint of Numeric field .....	10
3.3	WIF.TO Forcing internal testprint .....	10
3.4	BBJ License test.....	10
3.5	Logging system for Individual users .....	11
3.5.1	Activating the log for an user .....	11
3.5.2	Reducing the log output.....	11
3.5.3	Extending the log output .....	11
3.5.4	Logfile sample .....	11



This manual is intended to describe new (or undiscovered) functionality in the SW-Tools TRIO package.

## 1. Excel printer

The Excel printer has been completely reworked for improved functionality.

Instead of reflecting the reportlayout the resulting .CSV file will contain the actual data printed.

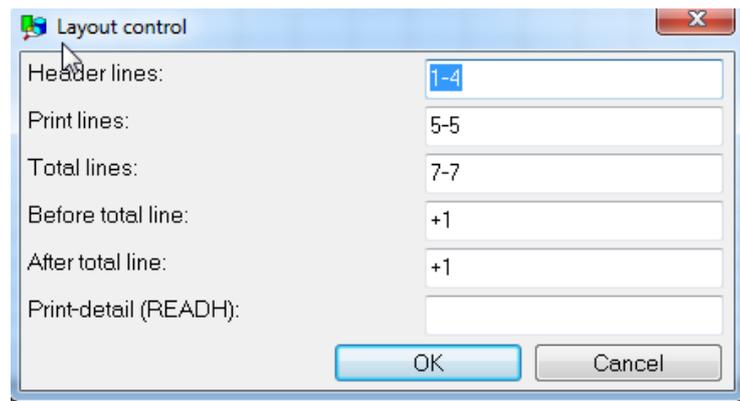
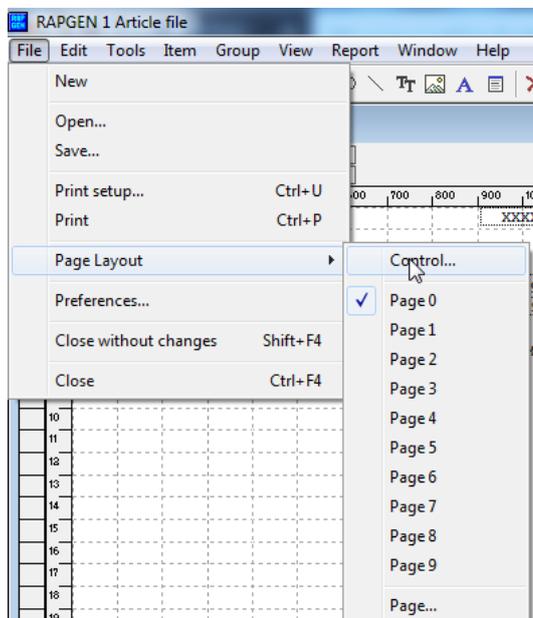
Each field printed on the report will be inserted into the Excel output file in the order it is met during print resulting in one column in Excel independent on where the field is placed in the report layout.

The field headings will be taken from the field name in the definition and not from the report layout as there may be no correspondence between the report header position and the field itself.

The total fields on a report will be named as the text TOTAL followed by the field reference as "TOTAL va#3". The Excel printer will solve this reference and place the total in the same column as the field va#3.

### 1.1 Report page layout sections

The page layout will define Heading, Normal print lines and Total lines.



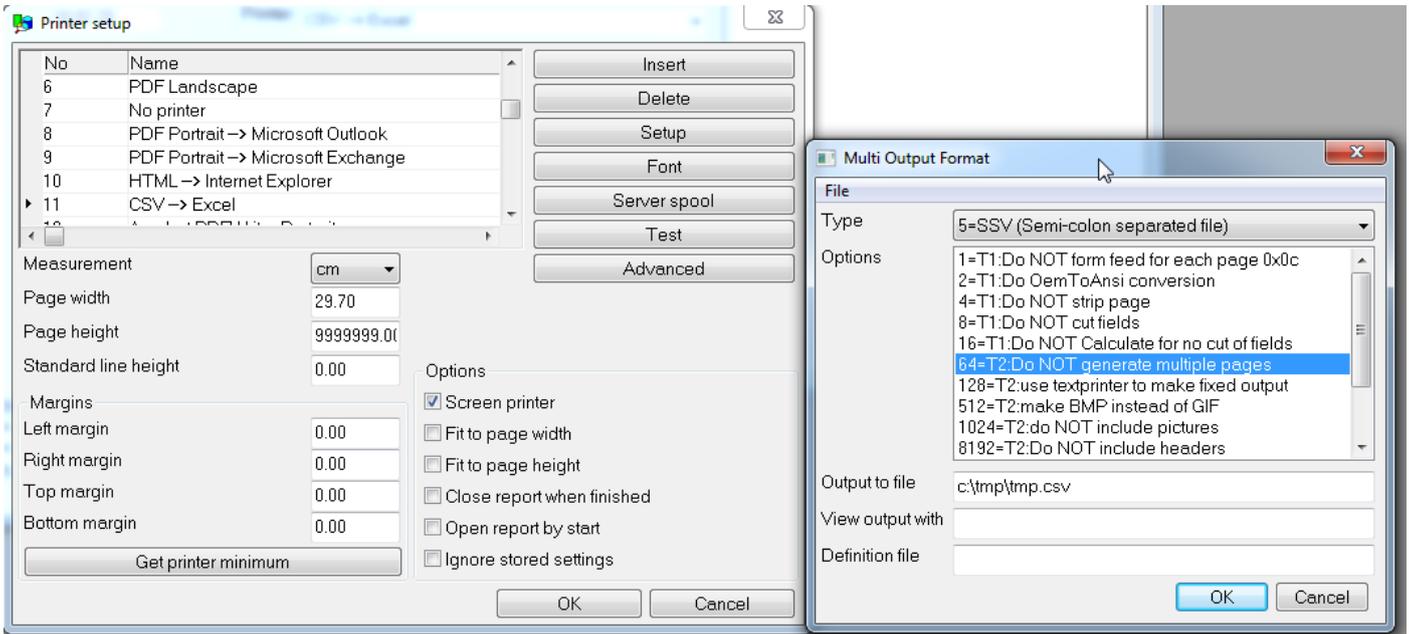
With the standard Excel printer setup all output of lines in the Header Lines range will be ignored, just fields placed outside the header area will be part of the output.

The output of demo report 9 – Margin report would look like

	A	B	C	D	E	F	G	H	I	J
1	Article no	Description	Group no	Selling price	Cost price	Margin	Margin%	Stock balance	Value	Total name
2	101	CHOCOLATE	0	2	1,5	0,5	33,33	100	150	
3	102	LARGE MACHINE	9	20000	10000	10000	100	0	0	
4	110	BUS	2	100000	60000	40000	66,67	1	60000	
5	1001	MONEY	0	1000	500	500	100	100	50000	
6	1005	MACHINE	1	2000	1500	500	33,33	10	15000	
7	2001	CREDITCARD	9	20	10	10	100	10	100	
8	2002	ID-CARD	9	25	10	15	150	200	2000	
9							70,85		127250	7 Total
10										

## 1.1 Setting up the Excel printer

The Excel printer should be set up as a screen printer of type 5 marked for no page generation:



### 1.1.1 Output to file

You should state a filename here as ....CSV. If you leave it blank the printer will use the default filename swssv.ssv and the filetype .ssv should be associated in the registry with Excel.

### 1.1.2 View output with

The CSV files usually is associated with Excel, but you may state any other preferred program here.

### 1.1.3 Definition file

The default file is named myprt005.ini and is placed on SWTOOLS path where the report generator is installed.

You may make copies of this file and state the name specific for a printer to get multiple Excel printers with different behavior.

## 1.2 The definition file

The definition file may be used to control the behavior of the printer. It may be edited using Notepad or other editor and contains a number of [sections] containing values to control the output.

For Excel the following points is used:

### 1.2.1 [LINES] defines which print areas to include

Default value is -1 that is everything but headings.

A range like 1-4 or -1,-4 where the figures corresponds to:

- 1: Heading
- 3: Normal print lines
- 4: Total lines

For reports using READH transaction control only:

- 2: READH Heading
- 5: READH Normal print lines
- 6: READH Total lines

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	SN System name	PD As of date	PP Pageno	DD Todays date	Article no	Description	Group no	Selling price	Cost price	Margin	Margin%	Stock balance	Value
2	SW-Tools	18.02.20	1	18.02.20	101	CHOCOLATE	0	2	1,5	0,5	33,33	100	150
3					102	LARGE MACHINE	9	20000	10000	10000	100	0	0
4					110	BUS	2	100000	60000	40000	66,67	1	60000
5					1001	MONEY	0	1000	500	500	100	100	50000
6					1005	MACHINE	1	2000	1500	500	33,33	10	15000
7					2001	CREDITCARD	9	20	10	10	100	10	100
8					2002	ID-CARD	9	25	10	15	150	200	2000
9													

Above [LINES] is changed to 1-3 to print headings and suppress totels.

### 1.2.2 [HEADLINE] column names

Default value is 0 –fieldname is used.

If set to 1 the field references as va#3 is used as headings.

### 1.2.3 [POINT] decimal point

Default value is . decimal point, may be set to , for decimal comma.

Excel will usally detect itself what is used inside the csv file

### 1.2.4 [DATE] date format

Dates on the report may be formatted as DD/MM/YY, YY/MM/DD, DD/MM/YYYY or other dependent on the output settings.

The output to the CSV file may be converted according to this setting to:

- Blank: No conversion - default
- 0: YYMMDD
- 1: YYYYMMDD
- 2: Excel date (number from basedate 19000101 and up)

	A	B	C	D	E	F	G	H	I
1	Article no	Description	Selling price	Cost price	Date of last purchase	Supplier no	Alternative supplier	Group no	Stock balance
2	101	CHOCOLATE	2	1,5	34700	271	270	0	100
3	102	LARGE MACHINE	20000	10000	33970	100	0	9	0
4	110	BUS	100000	60000	34318	123	100	2	1
5	1001	MONEY	1000	500	34699	205	102	0	100
6	1005	MACHINE	2000	1500	34486	100	0	1	10
7	2001	CREDITCARD	20	10	34700	205	102	9	10
8	2002	ID-CARD	25	10	34515	205	102	9	200
9									

Demo report 1 on an Excel printer with [LINE] as 3 (just printlines) and [DATE] as 2.

You will have to format the date cells yourself as Excel dates.

### 1.2.5 [PING] mark alpha fields with ‘

If set to 1 all alpha fields are prefixed with a ‘ stopping Excel to try to convert them to numeric and right justify some values.

### 1.2.6 [QUOTE] enclose alpha fields in “...”

If set to 1 all alpha fields will be quoted in the csv file.

If set to = all alpha fields will be made as: =(“xxxx”) and all “ will be replaced by ””, CR and ; will be okay inside.

### 1.2.7 [CRLF] defines the newline character(s)

Default is 0x0d as on Unix which use to be quite okay with Excel.

0:	0x0d (default)
1:	0x0a0d
2:	0x0d0a

## 1.3 Controlling the printer from the calculations

All of the above settings from the definition file may be changed from the calculations with the PRINT(CSV= command, as:

PRINT(csv=LINES=3)            changes print output to normal print lines without totals

But note that this cannot be used as a FIRST calculation as the printer is not open until the first line is printed.

If you have to override settings for the first output line you will need to change the myprt005.ini file.

### 1.3.1 Using the Excel printer with manual control through the PRINT statement

Each PRINT statement will make a line in the Excel file but just if any fields are present meaning dummy layout controls as PRINT(EH=10) does not go to the csv file.

Printing multiple layout lines in one PRINT will just make one output line in the Excel file, as will

PRINT(5-6)    just generate one line in the CSV file with the needed fields.

### 1.3.2 Using the Excel printer as COPIES printer

The Excel printer may be used just like others printers with and controlled with

COPIES(1,11) for a copy on second printer no. 11 defined as Excel CSV printer

### 1.3.3 Using the Excel printer as secondary printer

The Excel printer may be used just like others printers with and controlled with

PRINT(>2)    for output on second printer

### 1.3.4 PIP print archiving of the Excel print

The PIP print archiving will work with the Excel printer just as for other printers.

## 2. Barcode and QR code printing

The report generator can print any form or barcodes through a barcode font and it has building capabilities to print text in QR code too.

### 2.1 Barcodes

This is quite simple, just place a text field in the report layout and select a barcode font.

You may download free barcode fonts from many sites on the web, such as

<https://www.1001fonts.com/barcode-fonts.html>

The font should just be installed as font in Windows to be used in the report generator.

### 2.2 QR codes

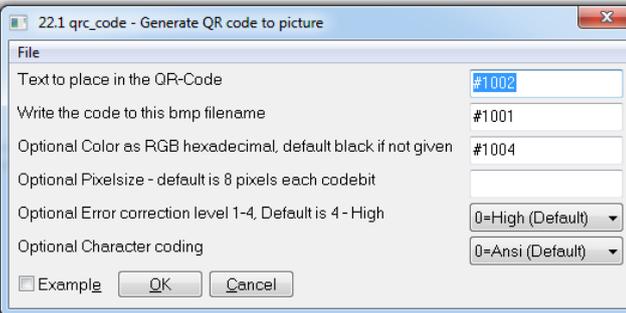
QR codes are not fixed font codes but is generated 'on the fly' by the function QRC\_CODE in the calculations

Just place a text into a free field (say #1001) and in the calculations type

QRC\_CODE(#1001,#1002) (followed by Ctrl.Q and you will get the following help)

#1002 becomes the barcode picture of the #1001 text and should just be defined as a large free text field (say 4000 characters) and placed in the report layout as a picture.

The sample below shows the calculations used to generate the QR code on the front page.



22.1 qrc\_code - Generate QR code to picture

Number qrc\_code(Text par1, Text par2, Text par3, Number par4, Number par5, Number par6)

Par1: Text to place in the QR-Code  
Par2: Write the code to this bmp filename  
if <par2> is blank an unique bmk filename will be made on the TMP directory  
this field should be inserted as a #p field on the report layout  
Par3: Optional Color as RGB hexadecimal, default black if not given  
FF0000 for red, 0000FF for blue QR-Code  
Par4: Optional Pixelsize - default is 8 pixels each codebit  
Par5: Optional Error correction level 1-4, Default is 4 - High  
0=High (Default)  
1=Low  
2=Medium  
3=Good quality  
4=High  
Par6: Optional Character coding  
0=Ansi (Default)  
1=UTF8  
2=OEM

The given text is made as QR-Code and written as .bmp file  
This may be inserted as a picture (#p) field in any report

Returnvalue:>0: Filenumber of QR code generated  
<=0: Error in generation of code

See also:barcode fonts

Example: If #20 is a #p field place on the report - 512 chars long  
and you shoud tick on the box for the #p field Keep Aspect Ratio  
#21="This is a test of\ na QUcode"  
#20=""  
qrc\_code(#21,#20)

if you want the QR-code in red add a field #22 of 10 characters:  
#22="ff0000"  
qrc\_code(#21,#20,#22)

**Calculations**

```

1 #11="SW-Tools Aps\n"
2 #11=#11,"Gilleleje Stranvej 33\n"
3 #11=#11,"DK-3250 Gilleleje\n"
4 #11=#11,"Denmark\n"
5 #11=#11,"\n"
6 #11=#11,"Tel. +45 33 33 05 56\n"
7 #11=#11,"Mail swtools@swtools.com\n"
8 #11=#11,"www.swtools.com\n"
9
10 qrc_code(#11,#10)
11

```

### 3. Test functions for programmers

You may place test functions in the calculations in different ways

#### 3.1 DIALOG function

The DIALOG functions is nice to get an output of field values on screen, but have the disadvantage that you may have to press OK many times when the report is running

#### 3.2 WIF- functions

The WIF set of functions will create a textfile C:\TMP\WIF with testprint.

This file may be edited with any editor like Notepad and you can just delete it after use for new testprint.

The report will run normal without interruption and the test can be inspected afterwards.

##### 3.2.1 WIFA Testprint of Alpha field

The WIF functions follows the C-Printf syntax, with one field defining the print layout and one data field. A common use would be like

```
WIFA("\nMy testfield: %s here",#1003)
```

%s will be replaced in the text with the contents of field 1003 which must be defined alphanumeric.

\n makes a newline in the file – if you omit this completely the file will end up having just one very long line.

Useful as there may be multiple WIFA(..) in a report section with output to the same testprint line

##### 3.2.2 SSV Making a set of fields as a SSV string

The SSV function is useful to get multiple field values to the testprint file in one go as:

```
#1003="My test:",ssv("#1-10,17")
WIFA("\n%s",#1003)
```

You may even use Excel to inspect such testprint afterwards.

##### 3.2.3 Filename for the WIFA testprint

If you want output to another file than C:\TMP\WIF you may give a third parameter to the WIFA as

```
#1010="c:/temp/myfile.csv"
WIFA("\n%s",#1003,#1010)
```

##### 3.2.3 Building Excel files using SSV and WIFA

These functions may be used together to build an CSV file completely independent of the report printout as

```
FIRST
#1010="c:/temp/myfile.csv"           /* Define the output filename
#1011="#1-10,17"                     /* State the fields in the file
#1003=SSV(#1011,"", "", "",110002)  /* Make a string will all fieldnames
WIFA("%s\n",#1003,#1010)             /* Output as heading
AFTER
#1003=SSV(#1011)                     /* Make a string with fieldvalues after selection
WIFA("%s\n",#1003,#1010)             /* Output the field values
```

### 3.2.4 WIFD Testprint of Numeric field

WIFD("%d.",#1001) may be used for output of an numeric field, integer value without decimals.

Mostly it will be more convenient to edit the numeric field as an alpha field and then use WIFA as above.

### 3.3 WIF.TO Forcing internal testprint

With notepad you can create a file named C:\TMP\WIF.TO just containing one digit with testprint level.

The presents of this file will trigger the internal testprint which may become huge dependent on the level – be carefull not to leave this file unintended on a customer production system as it definitely will slow down all report run.

The testprint level may be:

0	No testprint
+1	Trace of all start report and filenames by open/create
+2	Trace of all calculation lines step by step with intermediate results
+4	Output of data from all definition files read

The filename for output is c:\tmp\wif - the testprint level may be followed by a filename, from a command prompt:

```
echo 7,d:\test\ud.txt >c:\tmp\wif.to
```

will switch full testprint om with output to d:\test\ud.txt instead of c:\tmp\wif

Just remember to delete the c:\tmp\wif.to file after use !

### 3.4 BBJ License test

Starting the report module with BBJ option may be used just for a test of the license:

```
RAP00132.EXE –BBJ=bbjserveraddress
```

will produce a new file c:\tmp\wif with the result of the license function calls.

## 3.5 Logging system for Individual users

As of 20/09-2021 a new log system controlled via rap.ini is introduced.

### 3.5.1 Activating the log for an user

You will have to locate and edit the rap.ini file using notepad or like adding the following entries as:

```
[log]
log_USERNAME=c:\tmp\log.csv
```

This will activate logging to the file c:\tmp\log.csv for user USERNAME.

```
log_all=c:\tmp\log.csv
```

activates logging for all users.

```
log_ch=c:\tmp\log.csv!
```

activates logging for user ch and the logfile will be created everytime discarding the previous logfile.

The logfile is made as an csv textfile which can be opened and examined using Excel.

The logfile will contain entries from the RAP00132.EXE running executable only.

As usual you may deactivate such entry just by a ; in front, like ; log\_ch=c:\tmp\log.csv!

### 3.5.2 Reducing the log output

The user entry may be followed by a state specification:

```
log_USERNAME=c:\tmp\log.csv;1-3
```

as the log will be made for all the states if the parameter is not given:

- 1: Program start and Main program control flow
- 2: Early license / BBJ check
- 3: Open/close of files and load of DLL routines
- 4: Read of any file
- 5: Print output

### 3.5.3 Extending the log output

The previously known WIF.TO parameter may be added to the user entry as:

```
log_USERNAME=c:\tmp\log.csv;1-3;7
```

whereas the output will be given to the user logfile for these (maybe huge) entries too.

Using this specification will make the logfile not fully nice and CSV compatible for Excel.

### 3.5.4 Logfile sample

```
Module;Log;No;How;User;Date;Time;Function;File;Line;Text;000001
rap00132;1;0;0;ch;20210820;121923;G:\rapmdi;g:\rapmdi\rap00132.exe;0;008.081
rap00132;1;1;1;ch;20210920;121929;;rapi01;103;start:/c:/TMP/job00918.ini
rap00132;1;2;1;ch;20210920;121929;;rapi01;141;heltom
```

The first data line contains info on the CWD and real executable build date / version.