# *PCschematic ELautomation*

# Project Generator

## *Version 10.0*

## *15/7 - 2007*

*© Copying this manual is not allowed without the prior written permission of DpS CAD-center ApS*

**News** and **Update** on **www.pcschematic.com**

**Developed by: DpS CAD-center ApS, Bygaden 7, DK-4040 Jyllinge, Denmark**

**Phone: +45 4678 8244 Homepage: www.pcschematic.com**
**Fax: +45 4673 0215 E-mail: dps@dps.dk**

# TABLE OF CONTENTS

PREFACE ..... 5
OTHER MANUALS ..... 5

GENERATE AND UPDATE PROJECTS AUTOMATICALLY ..... 9
ON AUTOMATIC PROJECT GENERATION ..... 9
POINTS TO CONSIDER IN PROJECT GENERATION ..... 10
ON PROJECT DEFINITION FILES ..... 12
GENERATING PROJECT DEFINITION FILES FROM PCSCHEMATIC ..... 13
GENERATE PROJECTS BASED ON PROJECT DEFINITION FILES ..... 18

SYNTAX FOR PROJECT DEFINITION FILES ..... 23
WHAT CAN BE CONTROLLED VIA THE PROJECT DEFINITION FILE ..... 23
THE STRUCTURE AND CONTENTS OF PROJECT DEFINITION FILES ..... 24
PROJECT TITLE AND PROJECT DATA IN PROJECT GENERATION ..... 28
REFERENCE DESIGNATIONS IN PROJECT GENERATION ..... 28
SPECIFYING DRAWING TEMPLATES AND SUB DRAWINGS ..... 30
CONTROLLING TEXTS IN DRAWING TEMPLATES / SUB DRAWINGS ..... 31
FRONT PAGES, DIVIDERS AND LISTS IN AUT. GENERATED PROJECTS ..... 36
COMMENTS IN THE PROJECT DEFINITION FILE ..... 36
INSERTING NEW PAGES WHEN UPDATING EXISTING PROJECTS ..... 37
COMMA SEPARATED FILES FOR AUTOMATIC PROJECT GENERATION ..... 37

MERGING PROJECTS VIA THE PROJECT GENERATOR ..... 41
BENEFITS OF MERGING PROJECTS ..... 41
HOW TO MERGE PROJECTS ..... 42
KEEPING THE ORIGINAL PAGE NUMBERS WHEN MERGING PROJECTS ..... 43

APPENDICES ..... 45
VARIABLES ..... 47
STRUCTURE FOR PROJECT DEFINITION FILES ..... 49
INDEX ..... 51

# PREFACE

This manual describes how to generate projects automatically in ***PCschematic ELautomation***. Together with this manual it is recommended to read the "Sub drawings" chapter in the ***PCschematic ELautomation*** manual.

## OTHER MANUALS

For ***PCschematic ELautomation*** the other manuals are:

— The ***PCschematic ELautomation*** manual, which is the main manual for the electro-technical documentation program ***PCschematic ELautomation***.

— The ***PCschematic tutorial*** manual, which contains step by step introductions on how to create projects in ***PCschematic ELautomation***.

— The ***PCschematic Tools*** manual, which describes how you can document symbols in projects or libraries automatically, how to set up parameters for DWG/DXF import/export, how to use the ***PCschematic Translator*** program, for translating project texts, as well as how you automatically create graphical ***terminal plans***, graphical ***cable plans*** and graphical ***connection plans***.

— The ***PCschematic Database*** manual, which describes how to use the ***PCschematic Database*** program for creating and maintaining your database for ***PCschematic ELautomation***.

# Generate and Update Projects

This section describes how to generate and update projects automatically in ***PCschematic ELautomation***.

# GENERATE AND UPDATE PROJECTS AUTOMATICALLY

THIS CHAPTER DESCRIBES

— ON AUTOMATIC GENERATION OF PROJECTS .....(P. 9)
— POINTS TO CONSIDER IN PROJECT GENERATION .....(P. 10)
— ON PROJECT DEFINITION FILES .....(P. 12)
— GENERATING PROJECT DEFINITION FILES FROM PCSCHEMATIC .....(P. 13)
— GENERATING A PCSCHEMATIC PROJECT FROM A PROJECT DEFINITION FILE .....(P. 18)

— Please also refer to “Syntax for project definition files” on page 23 and “Merging projects via the Project generator” on page 41.

## ON AUTOMATIC PROJECT GENERATION

**POSSIBILITIES IN AUTOMATIC GENERATION OF PROJECTS**

— GENERATING PLC AND CIRCUIT DIAGRAMS FROM A PROJECT DEFINITION FILE FROM E.G. EXCEL

— PROJECTS ON HUNDREDS OF PAGES CAN BE GENERATED FROM A SINGLE PROJECT DEFINITION FILE

— CORRECTIONS CREATED IN ***PCschematic ELautomation*** CAN BE TRANSFERRED BACK TO THE PROJECT DEFINITION FILE, JUST LIKE CORRECTIONS IN THE PROJECT DEFINITION FILE CAN BE TRANSFERRED BACK TO ***PCschematic ELautomation***

Before you can generate projects automatically, you must do this:

1) Create all relevant *sub drawings* for automatic placing — see "Creating Project pages for aut. placing of sub drawings" in the ***PCschematic ELautomation*** manual.

2) Create all relevant *drawing templates* for inserting sub drawings.
— See “Creating drawing templates for automatic project generation” on page 10 and “Points to consider in Project generation” on page 10.

3) Create all other complete pages and/or projects — also including front pages, chapter front pages, tables of contents and all types of lists —, that you wish to include in the generated project.

4) Specify in which folders all of these sub drawings, drawing templates and project pages / projects are placed. — See “Settings for the Project Generator” on page 14, “Specify where sub drawings and drawing templates are placed” on page 14 in particular.

After this, you can generate a *project definition file* — see “On Project definition files” on page 12. Based on this project definition file you can then automatically generate ***PCschematic ELautomation*** projects. — See “Generate projects based on Project definition files” on page 18.

**On projects created via automatic project generation**

When you have created a project via automatic project generation, you can continue working on the project, as with any other ***PCschematic ELautomation*** project. If you want to, you can update the project definition file at a later stage, so that it matches the changes you have made in the project.

**Creating drawing templates for automatic project generation**

The pages, which the sub drawings are placed on during the automatic project generation, are called *drawing templates*.

Drawing templates are created just like you create sub drawings, as described in "Preparing Sub drawings for Drag’n Draw / Aut. placing" in the ***PCschematic ELautomation*** manual.

On drawing templates you can — as opposed to sub drawings — also place a drawing header.

On drawing templates you can — just as for sub drawings — insert *variable markers*, so that the texts in the drawing template can be controlled from the project definition file.

An example on creating a drawing template can be found in "Creating Project pages for aut. placing of sub drawings" in the ***PCschematic ELautomation*** manual.

## POINTS TO CONSIDER IN PROJECT GENERATION

When you wish to create sub drawings for automatic project generation, it is necessary that you spend some time considering exactly how to apply the sub drawings, and in which context they are going to be used. — See also "Preparing Sub drawings for Drag’n Draw / Aut. placing" and "Placing the insertion point symbols carefully" in the ***PCschematic ELautomation*** manual.

**CREATING SUB DRAWINGS GRADUALLY**

TO MAKE STARTING TO APPLY AUTOMATIC PROJECT GENERATION EASIER, YOU CAN START BY CREATING THE SUB DRAWINGS GRADUALLY, AS YOU GET TO NEED THEM IN YOUR DAILY WORK.
— TO GET THE MAXIMAL BENEFIT FROM THE SUB DRAWINGS IN THE LONG RUN, YOU MUST FIRST CONSIDER THE ISSUES MENTIONED BELOW.

You should for instance consider:

### Paper size and Reference system

One of the things you should consider, is of course which paper size the sub drawings should be designed for, because this has an influence on the size of the sub drawings.

If you are applying a reference system on the pages, there must also be space available for current path numbers on the page.

### Using variables

When you are creating the sub drawings, you should also consider which data are fixed, and which are changed in the repeated use of the sub drawings. For those data that changes, you should create variables.

At the same time, you should also decide whether you wish to apply automatic naming or not. If you wish not to apply automatic naming, you will have to create more variables in the sub drawings.

Please note that it is possible to place variable markers for e.g. symbol names, and in the project definition file specify not to change the symbol names from there — see “Project generation with no symbol names in the project definition file” on page 35.

### Creating sub drawings based on existing projects

As a starting point, it is a good idea to look closely on the already existing projects, and discover which of the existing pages, that can be used for creating sub drawings. When examining the projects, you can also discover where the insertion points should have been placed, if you had generated the projects automatically.

When you have done this, you can for instance start by placing the insertion points for a coming sub drawing already in the existing project. You can then copy the desired area containing symbols, lines and insertion points etc., and *then* create a sub drawing based on the copied area.

— See "Creating sub drawings by Copying from projects" in the ***PCschematic ELautomation*** manual.

### Creating sub drawings as Modules

To use the area of the page to full extend in project generation, it is a good idea to consider the size of the sub drawings.

It is here recommended to create the sub drawings in different sets of fixed module sizes, in multipla that fits to the page size — or more precisely: In multipla that fits to the room between the insertion points on the page.
— See "The function of the insertion points" in the ***PCschematic ELautomation*** manual.

As the largest module size — when disregarding sub drawings that fills a full page — you can for instance create sub drawings, which you can have exactly two of on a page.

As the next module size you can create sub drawings, of which you can have exactly four, eight and even sixteen on a page.

Designed this way, you can for instance have one half page sub drawing and two quarter page sub drawings on the same page, which gives you a better use of the pages.

**Drawing template + 4 sub drawings of 1/4 page = Full use of the page**

**Model data in automatic project generation**
— See “Create model variable” on page 34.

## ON PROJECT DEFINITION FILES

To generate projects automatically in ***PCschematic ELautomation***, there must first be generated a *project definition file*.

Together with the created *sub drawings*, *drawing templates* etc. — see “On automatic project generation” on page 9 —, this file defines the contents of the automatically generated project.

You can either create the project definition file directly in Excel, or you can let ***PCschematic ELautomation*** generate a project definition file based on an existing project — see “Generating Project definition files from PCschematic” on page 13. You can then adapt the generated project definition file according to your needs.

— See “Syntax for project definition files” on page 23.

## GENERATING PROJECT DEFINITION FILES FROM PCSCHEMATIC

To generate a *project definition file* in Excel format — see “On Project definition files created from PCschematic” on page 14 — from a drawing template, do the following:

---

**WHEN USING DRAG’N DRAW: KEEP TRACK OF THE FOLDERS**

WHEN YOU USE DRAG’N DRAW FOR DRAGGING SUB DRAWINGS AND PROJECT PAGES INTO A PROJECT, AND *THEN* GENERATE A PROJECT DEFINITION FILE, INFORMATION ABOUT THE *NAMES* OF THESE SUB DRAWINGS, PROJECT PAGES ETC. ARE INCLUDED, BUT THE *PATH* LEADING TO THESE SUB DRAWINGS ETC. IS *NOT* INCLUDED.

WHEN YOU EXECUTE THE AUTOMATIC PROJECT GENERATION BASED ON THE PROJECT DEFINITION FILE, YOU MUST THEREFORE HAVE CREATED AN *ALIAS* FOR EACH OF THE FOLDERS, WHICH YOU NEED TO FETCH SUB DRAWINGS ETC. FROM.

OTHERWISE, THE PROJECT WILL NOT BE GENERATED CORRECTLY!

— READ MORE ABOUT ALIASES IN “WHAT IS AN ALIAS?” ON PAGE 15.

---

1) Choose **Tools => Project generator**.

2) In the dialog box **Project generator**: Click on **Options** if you need to change the settings for the project definition file — see “Settings for the Project Generator” on page 14.

3) Click on ***Create project definition file from the active project*** and click **OK**.

4) In the dialog box **Save as**: Assign a name to the project definition file, and specify where to save the file, and click **Save**.

5) You now get a status of the generated project definition file:

Click **OK**.

You have now generated a project definition file from the active project.

---

**ON PROJECT DEFINITION FILES CREATED FROM *PCSCHEMATIC***

WHEN YOU HAVE GENERATED A *PROJECT DEFINITION FILE* IN THE EXCEL FORMAT, YOU CAN EDIT THE FILE FROM MICROSOFT EXCEL, AS DESCRIBED IN "SYNTAX FOR PROJECT DEFINITION FILES" ON PAGE 23.

SUCCEEDINGLY YOU CAN UPDATE THE ACTIVE PROJECT — OR GENERATE A NEW PROJECT — BASED ON THE PROJECT DEFINITION FILE. READ MORE ABOUT THIS IN "GENERATE PROJECTS BASED ON PROJECT DEFINITION FILES" ON PAGE 18.

---

**Settings for the Project Generator**

When you have chosen **Tools => Project generator**, and clicked on **Options**, you can specify the settings for the project definition file you wish to generate from ***PCschematic ELautomation***:

— See also "The significance of the first project page in the project definition file" on page 26.

*Specify where sub drawings and drawing templates are placed*

Below ***Folders for sub drawings / drawing templates*** you can specify from where to fetch the sub drawings, drawing templates and other project pages for the generated project.

This is done by creating an *alias* for each folder from which ***PCschematic ELautomation*** needs to fetch drawing templates / sub drawings during the project generation.

**WHAT IS AN ALIAS?**

AN ALIAS CONSISTS OF AN *ALIAS NAME*, AND A *PATH* TO A FOLDER ON YOUR PC — OR ON THE NETWORK THAT YOUR PC IS CONNECTED TO.

WHEN YOU CREATE A *PROJECT DEFINITION FILE* FROM A PROJECT, *ONLY THE NAMES* OF THE USED SUB DRAWINGS, DRAWING TEMPLATES ECT. ARE INCLUDED IN THE FILE. THE *PATH* TO THE FOLDERS WHERE THESE ELEMENTS COMES FROM, ARE NOT INCLUDED IN THE PROJECT DEFINITION FILE.

WHEN YOU WISH TO GENERATE A PROJECT BASED ON A PROJECT DEFINITION FILE, ***PCschematic ELautomation*** FIRST EXAMINES THE FOLDER AT THE TOP OF THE ALIAS LIST, TO E.G. FIND AN APPLIED SUB DRAWING. IF THIS SUB DRAWING CAN NOT BE FOUND IN THIS FOLDER, THE PROGRAM EXAMINES THE CONSECUTIVE FOLDERS IN THE ALIAS LIST, UNTIL A SUB DRAWING WITH THE USED NAME IS FOUND. THIS PROCESS IS REPEATED FOR EACH INDIVIDUAL SUB DRAWING, DRAWING TEMPLATE ETC. LISTED IN THE PROJECT DEFINITION FILE.

THEREFORE IT IS IMPORTANT IN WHICH ORDER THE ALIASES APPEARS IN THE ALIAS LIST. — SEE ALSO “ALIASES/FOLDERS CAN ALSO BE SPECIFIED IN THE PROJECT GENERATOR DIALOG BOX” ON PAGE 24 AND “ALIASES RELATIVE TO OTHER ALIASES” ON PAGE 25.

You can create or delete aliases by right-clicking in the alias window, as described in "Directories" in the ***PCschematic ELautomation*** manual, in particular in "Show directory alias". If you click on the **Select folder** button for one of the aliases, you can click your way to a new folder for the selected alias.

— See also “On automatic project generation” on page 9.

**WHEN USING DRAG’N DRAW: KEEP TRACK OF THE FOLDERS**

WHEN YOU USE DRAG’N DRAW FOR DRAGGING SUB DRAWINGS AND PROJECT PAGES INTO A PROJECT, AND *THEN* GENERATE A PROJECT DEFINITION FILE, INFORMATION ABOUT THE NAMES OF THESE SUB DRAWINGS, PROJECT PAGES ETC. ARE INCLUDED, BUT THE *PATHS* LEADING TO THESE SUB DRAWINGS ETC. ARE *NOT* INCLUDED — AS DESCRIBED ABOVE.

WHEN YOU GENERATE A PROJECT, YOU MUST THEREFORE HAVE CREATED AN *ALIAS* FOR EACH OF THE FOLDERS, WHICH YOU NEED TO FETCH SUB DRAWINGS ETC. FROM, IN ORDER TO GENERATE THE PROJECT.

OTHERWISE, THE PROJECT WILL NOT BE GENERATED CORRECTLY!

*Updating lists automatically when using drawing templates*

This only applies for generating projects: Select ***Update lists after generation*** when you wish to have the project lists updated automatically after generating/updating a project.

This is the same as executing the command **Lists => Update all lists** after the project is generated.

*Updating component data automatically when using drawing templates*
This only applies for generating projects: Select ***Update component data from database after generation*** to update component data automatically after generating/updating a project.

This causes that for instance the type descriptions matches the article number — without including the type descriptions in the project definition file.

*Auto name symbols during generation*
This only applies for generating projects: If you choose ***Auto name symbols during generation***, the symbols are named automatically in the generated project. If you choose this option, you will therefore not have to specify symbol names in the project definition file.

— See “Current path numbers Or Automatically generated names” on page 26.

— See also “Project generation with no symbol names in the project definition file” on page 35.

*Add page reference designation to symbols in subdrawings*
This only applies for generating projects: If you choose ***Add page reference designation to symbols in subdrawings***, symbols with no function and/or location aspect in the reference designation are automatically assigned these from the page, on which the subdrawing is placed when generating the project.

If a symbol has the name **-Q1**, and the page reference designation is **=G1+A2**, the symbol will get the reference designation **=G1+A2-Q1**, if you have chosen ***Add page reference designation to symbols in subdrawings***. — If not, the symbol will just keep the name **-Q1**.

If a symbol in a subdrawing has a location and/or a function aspect attached, the reference designation will *not* be changed, even though you have chosen ***Add page reference designation to symbols in subdrawings***.

*PLC synchronization*
When you in the project generation use pages from different projects, where pages with PLC symbols comes from one project, and pages with PLC reference symbols comes from another, ***PCschematic ELautomation*** can not determine how to synchronize the PLC symbols and the PLC reference symbols.

Below ***PLC synchronization*** you have the following options:

- ***None***: No synchronization is made.
- ***PLC reference >> PLC***: Data is transferred from the PLC reference symbols to the PLC symbols. This is equal to executing the command **Functions => Special Functions => Transfer data from PLC reference symbols to PLC symbols** after generating the project.
- ***PLC reference << PLC***: Data is transferred from the PLC symbols to the PLC reference symbols. This is equal to executing the command **Functions => Spe-**

**cial Functions => Transfer data from PLC symbols to PLC reference symbols** after generating the project.

*Invisible nonreferenced Cables and Signals*

When you create a project by applying sub drawings or project pages, where one end of a cable is illustrated on one page, and the other end is illustrated on another page, it may occur that one end of a cable is not applied in a project. When the cable connection points have ***With reference*** activated, nonreferred cables will occur. The same phenomenon can be observed for signals.

These cables and signals can be set invisible by selecting ***Unreferenced cables*** and ***Unreferenced signals*** respectively. This is equal to executing the command **Functions => Special Functions => Set unreferenced cable symbols invisible** and **Functions => Special Functions => Set unreferenced signal symbols invisible** after generating the project.

*Include Project data*

When you select ***Include project data***, the applied Project data datafields are included in the project definition file.

*Include reference designations*

When you select ***Include reference designations***, the project reference designations are included in the project definition file.

*Creating column headings in project definition files*

This only applies for generating project definition files: Select ***Insert column headings for variables*** when you want ***PCschematic ELautomation*** to create column headings describing the variables in the generated project definition file.

Here you can choose between ***Names of variables*** and ***Text types of variables***.

*Format for comma separated files*

For comma separated files you can specify ***Delimiter character*** and ***Quote character*** at the bottom of the dialog box.

## GENERATE PROJECTS BASED ON PROJECT DEFINITION FILES

Project definition files can be generated based on an existing ***PCschematic ELautomation*** project — see "Generating Project definition files from PCschematic" on page 13 — or be created directly in e.g. Excel — see "Syntax for project definition files" on page 23.

To generate a project, or to update the active project based on an existing *project definition file* in e.g. Excel format, do the following:

1) Choose **Tools => Project generator**.

2) In the dialog box **Project generator**: Click on **Options** and check whether the setup is correct for the project generation — including that the ***Default folder for drawing templates and sub drawings*** are set to the folder where you place everything you use for the automatic project generation. — See ***"On automatic project generation" on page 9*** and "Settings for the Project Generator" on page 14.

3) In the dialog box **Project generator**: Click on ***Generate new project*** when you wish to generate a new project, or click on ***Update the active project*** when you wish to update the active project.

4) Click on the **Browse** button, select the project definition file in the dialog box **Open**, and click **Open**.

*See the project definition file*

5) In the dialog box **Project generator**: If you wish to see the contents of the file, click on the **Display in Excel** button, and Excel is opened — if Excel is installed on your PC — and the contents of the selected file is displayed. Here you can make changes in the project definition file before generating the project. (Please remember to save the project definition file before returning to ***PCschematic ELautomation***).

6) In the dialog box **Project generator**: Click **OK**, and the project is updated/ generated based on the project definition file.

*The dialog box Generate new project*

7) You now enter the dialog box **Generate new project** (if you chose ***Update the active project*** you enter the dialog box **Update the active project** — see "The dialog box Update the active project" on page 20):

| Project definition file | | | Project | | |
|---|---|---|---|---|---|
| Row | Template | Variable | Page | Template | New value |
| 24 | DPSFRONTPAGE | | F1 | DPSFRONTPAGE | |
| 25 | DPSA4-T | | I1 | DPSA4-T | |
| 27 | L1L2L3NDEMO | @1 | 1 | L1L2L3NDEMO | Motor power circuits |
| | | @2 | | | A1 |
| 30 | ~MOTORDEMO1 | @1 | | ~MOTORDEMO1 | =G1+A2-Q1 |
| | | @2 | | | 4022903075387 |
| | | @3 | | | -F1 |
| | | @4 | | | 4022903085584 |
| | | @5 | | | -X1 |
| | | @6 | | | -W1 |
| | | @7 | | | 5702950410537 |
| | | @8 | | | -M1 |
| | | @9 | | | 1723410403 |
| 31 | ~MOTORDEMO1 | @1 | | ~MOTORDEMO1 | =G1+A2-Q2 |
| | | @2 | | | 4022903075387 |
| | | @3 | | | -F2 |

In the left-hand side of the **Generate new project** dialog box, you can see the project definition file, and in the right-hand side you can see the project pages and variable values, which will be inserted in the automatically generated project.
If you double-click on a row in the dialog box, the project definition file is opened in Excel, where the row you clicked on is selected.
If you make changes in the project definition file in Excel, before generating the project, please remember to save the Excel file before returning to ***PCschematic ELautomation***.

8) Click on **Generate**, and the project is generated.

— Please also see "The dialog box Update the active project" on page 20.

**Editing text fields with variables after generating projects**

When you wish to edit text fields with variables after generating a project, you can also do this by choosing **Edit => Variables**, when you subsequently work with the project in ***PCschematic ELautomation***. — See "Editing variables for inserted sub drawings" in the ***PCschematic ELautomation*** manual.

You can also change data in the generated project, just like when working with any other ***PCschematic ELautomation*** project.

### The dialog box Update the active project

If you in the **Project generator** choose ***Update the active project*** and click **OK**, you enter the dialog box **Update the active project**:

| Project definition file | | | | Project | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Row | Template | Variable | | Page | Template | Section | Text type | Current value | New value |
| 24 | DPSFRONTPAGE | | | F1 | DPSFRONTPAGE | | | | |
| 25 | DPSA4-T | | | I1 | DPSA4-T | | | | |
| 27 | L1L2L3NDEMO | @1 | | 1 | L1L2L3NDEMO | 1 | Title | Motor power circuits | Motor power circuits |
| | | @2 | | | | | Revision | A1 | A1 |
| 30 | ~MOTORDEMO1 | @1 | | 1 | ~MOTORDEMO1 | 2 | S.Name | =G1+A2-Q1 | =G1+A2-Q1 |
| | | @2 | | | | | S.Arti | 4022903075387 | 4022903075387 |
| | | @3 | | | | | S.Name | -F1 | -F1 |
| | | @4 | | | | | S.Arti | 4022903085584 | 4022903085584 |
| | | @5 | | | | | S.Name | -X1 | -X1 |
| | | @6 | | | | | S.Name | -W1 | -W1 |
| | | @7 | | | | | S.Arti | 5702950410537 | 5702950410537 |
| | | @8 | | | | | S.Name | -M1 | -M1 |
| | | @9 | | | | | S.Arti | 1723410403 | 1723410403 |
| 31 | ~MOTORDEMO1 | @1 | | 1 | ~MOTORDEMO1 | 3 | S.Name | =G1+A2-Q2 | =G1+A2-Q2 |
| | | @2 | | | | | S.Arti | 4022903075387 | 4022903075387 |
| | | @3 | | | | | S.Name | -F2 | -F2 |

Here you have the same options as described in “The dialog box Generate new project” on page 19.

*New values displayed as blue text*

Beyond this, you can see the present values of the variables in the project, and the values these variables will be assigned by generating the project again.

New values are displayed as blue text.

*Mismatch and syntax errors displayed as red text*

If there is a mismatch between the project and the project definition file — e.g. if the project definition file does not contain enough variables — the mismatch is pointed out in red writing.

Syntax errors in the project definition file are also written in red.

# Syntax for Project definition files

This section describes the syntax for project definition files.

Please note the appendix “Structure for Project definition files” on page 49 and the appendix “Variables” on page 47.

# SYNTAX FOR PROJECT DEFINITION FILES

To make it possible for ***PCschematic ELautomation*** to interpret the contents of a *project definition file* —see “On Project definition files” on page 12 —, the contents of this file must follow the syntax described below.

This syntax is automatically followed when project definition files are created directly from ***PCschematic ELautomation*** — see “Generating Project definition files from PCschematic” on page 13.

THIS CHAPTER DESCRIBES


— WHAT CAN BE CONTROLLED VIA THE PROJECT DEFINITION FILE.....(P. 23)
— THE STRUCTURE AND CONTENTS OF PROJECT DEFINITION FILES.....(P. 24)
— SYNTAX FOR PROJECT TITLE AND PROJECT DATA.....(P. 28)
— SYNTAX FOR REFERENCE DESIGNATIONS .....(P. 28)
— SPECIFYING DRAWING TEMPLATES AND SUB DRAWINGS.....(P. 30)
— CONTROLLING TEXTS IN DRAWING TEMPLATES AND SUB DRAWINGS.....(P. 31)
— FRONT PAGES, DIVIDERS AND LISTS .....(P. 36)
— COMMENTS IN PROJECT DEFINITION FILES.....(P. 36)
— INSERTING NEW PAGES WHEN UPDATING PROJECTS .....(P. 37)
— COMMA SEPARATED FILES AND AUTOMATIC PROJECT GENERATION .....(P. 37)


## WHAT CAN BE CONTROLLED VIA THE PROJECT DEFINITION FILE

The chapter “Generate and Update projects automatically” on page 9 describes how to create ***PCschematic ELautomation*** projects based on a *project definition file*. This file contains all necessary information for generating a project automatically.

In this file you can for instance specify:

- Project data and Project title
- Reference designations for projects
- The name of a project with front pages, declarations, project tables of contents etc., which your company for instance use when creating projects
- Names on the ***PCschematic ELautomation*** projects containing ready made diagram pages, which shall be part of the project — e.g. projects containing circuit diagrams for relays, and projects containing circuit diagrams for PLCs
- Drawing template pages, prepared for inserting sub drawings
- Sub drawings to insert automatically on the drawing templates — see "Sub drawings" in the ***PCschematic ELautomation*** manual and “Generate and Update projects automatically” on page 9
- Information on which data values to assign to the sub drawings, such as article numbers — see “Controlling texts in drawing templates / sub drawings” on page 31

- The name of a list project, e.g. containing all of the list types, that your company normally use for projects

... and much more, which is described in the following.

---

**CREATE AN ALIAS FOR EACH APPLIED FOLDER**

IN ORDER TO BE ABLE TO GENERATE A PROJECT CORRECTLY FROM THE PROJECT DEFINITION FILE, YOU MUST HAVE CREATED AN *ALIAS* FOR EACH FOLDER FROM WHICH YOU FETCH PROJECT FILES, DRAWING TEMPLATES AND SUBDRAWINGS WHEN GENERATING THE PROJECT.

— PLEASE ALSO SEE "WHAT IS AN ALIAS?" ON PAGE 15 AND "WHEN USING DRAG'N DRAW: KEEP TRACK OF THE FOLDERS" ON PAGE 13.

---

## THE STRUCTURE AND CONTENTS OF PROJECT DEFINITION FILES

A *project definition file* can contain up to three different sections (please see the figure in "Project definition file example" on page 25):

### 1. FOLDERS: Specifying applied folders / Aliases

This section is optional. The section is applied to specify — in the project definition file — from which folders to fetch files, drawing templates and sub drawings when generating the project.

This is done by creating a so-called *alias* for each of these folders. A such alias consists of an *alias name*, and a *path*, that specifies the exact location of a folder.
— See also "What is an Alias?" on page 15.

To create aliases in the project definition file, write *FOLDERS* in the field **A1** in the Excel spreadsheet containing the project definition file. In the field **A2** you type in the name of the first *alias*, and in the field **B2** you type in the folder (including the complete *path*), from which sub drawaings etc. are to be fetched. In the fields **A3** and **B3** you specify the next alias — etc.

*Aliases/Folders can also be specified in the Project generator dialog box*

When you generate projects, you must specify at least one alias, from which sub drawings, drawing templates and other project files must be fetched when generating the project. However, you have two options for specifying this information:

- When you specify the aliases *in the project definition file*, the sub drawings, drawing templates and other project files will be fetched from these folders.
- If you do *not* write this information in the project definition file, you must specify the aliases in the **Project generator** dialog box in ***PCschematic ELautomation*** — see "Generate projects based on Project definition files" on page 18.

When aliases are specified both in the **Project generator** dialog box and in the *project definition file*, ***PCschematic ELautomation*** first examine the aliases specified in the **Project generator** dialog box, and then examine the aliases specified in the project definition file.

**2. PCSCHEMATIC DATA: Project data and Reference designations**
This section is optional. The section can be used to include Project data and/or reference designations in the project definition file.

To include this section, you must write *PCSCHEMATIC DATA* at the top of column **A** (below *FOLDERS*, if this option is applied — see "FOLDERS: Specifying applied folders / Aliases" on page 24) — to tell that the lines below specifies general data for the generated project.

— See "Project title and Project data in Project generation" on page 28 and "Reference designations in Project generation" on page 28.

**3. PCSCHEMATIC PAGES: Files, Drawing templates, Sub drawings etc.**
If you have specified Project data or reference designations at the top of the project definition file, you must write *PCSCHEMATIC PAGES* in column **A** to specify that the consecutive rows in the project definition file contains names on drawing templates, sub drawings and other project files. — See the figure in "Project definition file example" on page 25.

Read more about this in "Specifying Drawing templates and Sub drawings" on page 30, "Controlling texts in drawing templates / sub drawings" on page 31 and "Front pages, Dividers and Lists in Aut. generated projects" on page 36.

---

**ALIASES RELATIVE TO OTHER ALIASES**

WHEN YOU SPECIFY THE *PATH* TO THE FOLDER, WHICH AN ALIAS POINTS AT, YOU CAN ALSO START BY SPECIFYING ANOTHER ALIAS BETWEEN TWO **%** CHARACTERS, AND THEN WRITE THE SUBSEQUENT PATH:

| | |
|---|---|
| **FOLDERS** | |
| **ALIAS1** | C:\PCS\SUBDRAWINGS |
| **ALIAS2** | **%ALIAS1%**\CUSTOMER14 |

IN *THE PROJECT DEFINITION FILE* YOU CAN BOTH REFER TO ALIASES SPECIFIED IN THE **PROJECT GENERATOR** DIALOG BOX, AND TO ALIASES SPECIFIED ABOVE THE RELEVANT ALIAS IN THE PROJECT DEFINITION FILE. (IN THE EXAMPLE ABOVE, ALIAS2 CAN THEREFORE BE RELATIVE TO ALIAS1 — BUT NOT THE OTHER WAY AROUND).

YOU CAN ALSO APPLY RELATIVE ALIASES IN THE **PROJECT GENERATOR** DIALOG BOX, BUT YOU CAN NOT REFER TO ALIASES IN A PROJECT DEFINITION FILE FROM THERE.

---

**Project definition file example**
A project definition file could for instance look like this:

| | A | B | C |
|---|---|---|---|
| 1 | **FOLDERS** | | |
| 2 | Folder1 | C:\AutoGenProjects\Templates | |
| 3 | Folder2 | C:\AutoGenProjects\Customer39 | |
| 4 | | | |
| 5 | **PCSCHEMATIC DATA** | | |
| 6 | | | |
| 7 | PROJECTDATA | | |
| 8 | Title | Project definition file demo | |
| 9 | Customer name | DpS CAD-center ApS | |
| 10 | Date | 06-06-06 | |
| 11 | | | |
| 12 | REFERENCE DESIGNATIONS | | |
| 13 | FUNCTION | | |
| 14 | =G1 | Conveyer belt control | |
| 15 | | | |
| 16 | LOCATION | | |
| 17 | +A1 | Controlpanel for conveyer belt | |
| 18 | +A2 | Main switchboard for conveyer belt | |
| 19 | +W1 | Conveyer belt | |
| 20 | | | |
| 21 | **PCSCHEMATIC PAGES** | | |
| 22 | | | |
| 23 | TEMPLATE | PAGE | |
| 24 | | | |
| 25 | DPSFRONTPAGE | F1 | |
| 26 | DPSA4-T | I1 | |
| 27 | | | Title |
| 28 | L1L2L3NDEMO | 1 | Motor power |
| 29 | | | |
| 30 | | | S.Name |
| 31 | MOTORDEMO1 | ~ | =G1+A2-Q1 |
| 32 | MOTORDEMO1 | ~ | =G1+A2-Q2 |
| 33 | MOTORDEMO1 | ~ | =G1+A2-Q3 |
| 34 | MOTORDEMO1 | ~ | =G1+A2-Q4 |
| 35 | MOTORDEMO1 | ~ | =G1+A2-Q5 |
| 36 | MOTORDEMO1 | ~ | =G1+A2-Q6 |
| 37 | | | |
| 38 | | | S.Name |
| 39 | MOTORDEMO2 | ~ | -X7 |
| 40 | MOTORDEMO2 | ~ | -X8 |
| 41 | MOTORDEMO2 | ~ | -X9 |
| 42 | MOTORDEMO2 | ~ | -X10 |
| 43 | | | |
| 44 | LISTS | | |

— Please note that the use of variables in project definition files is displayed in detail in “Controlling texts in drawing templates / sub drawings” on page 31.

— See also the appendices “Structure for Project definition files” on page 49 and “Variables” on page 47.

**The significance of the first project page in the project definition file**
Some project settings are fetched from the first project page / the first project in the project definition file. This is the case for:

*Current path numbers Or Automatically generated names*
If the first page — or the first project — in the project definition file uses a symbol name format containing current path numbers, this format is applied in all of the generated project. If this is not the case, automatic numbering of symbols with consecutive names are applied.

— However, the symbol naming is only changed if you in **Tools => Project Generator => Options** have selected ***Auto name symbols during generation***. If this is not the case, the names will be fetched from the project definition file during the automatic project generation — if no names are specified in the project definition file, no changes are made.

— See “Settings for the Project Generator” on page 14, “Auto name symbols during generation” on page 16 in particular.

*Text visibility*
The general settings for which texts are visible in the generated project is determined by the first project — or by the first 1-page project — in the list of projects in the project definition file.

**Information not necessarily specified in the project definition file**
Other information *can* be specified in the project definition file, but does not necessarily have to be specified there. If these types of information are not specified in the project definition file, the information is fetched from the first page / project occurring in the project definition file.

This is the case for:

- Project title and Project data — see “Project title and Project data in Project generation” on page 28
- Reference designations — see “Reference designations in Project generation” on page 28

## PROJECT TITLE AND PROJECT DATA IN PROJECT GENERATION

When you create ***PCschematic ELautomation*** projects via a project definition file, you can specify project title and Project data at the top of the project definition file in Excel.

This is done by writing *PCSCHEMATIC DATA* at the top of column **A** (below *FOLDERS*, if this option is applied — see "FOLDERS: Specifying applied folders / Aliases" on page 24) in the file — to tell that the lines below specifies the general data for the created project.

Below this write *ProjectData* in column **A**, also.

If you wish to specify a project title, write *Title* below this in column **A**, and write the project title next to this in column **B**.

If you wish to specify Project data, you can write the names of the Project data datafields in column **A**, and their contents in column **B**.

| | |
|---|---|
| PCSCHEMATIC DATA | |
| | |
| ProjectData | |
| Title | Datatemplate demo |
| Customer name | DpS CAD-center ApS |
| Date | 03-21-03 |

— Also see the figure in "Project definition file example" on page 25 and "The significance of the first project page in the project definition file" on page 26.

## REFERENCE DESIGNATIONS IN PROJECT GENERATION

In automatic project generation you can specify the reference designations for the project at the top of the project definition file.

If you have not written *PCSCHEMATIC DATA* at the top of column **A** (below *FOLDERS*, if this option is applied — see "FOLDERS: Specifying applied folders / Aliases" on page 24) in the project definition file, you must do this — see "Project title and Project data in Project generation" on page 28.

Below this write *Reference Designations* in column **A**.

Below this write *FUNCTION* in column **A**.

Below this write the names of the function aspects for the project in column **A**, and their attached descriptions in column **B**.

Below this write *LOCATION* in column **A**.

Below this write the names of the location aspects for the project in column **A**, and their attached descriptions in column **B**.

| | |
|---|---|
| Reference Designations | |
| FUNCTION | |
| =G1 | Conveyor belt control |
| | |
| LOCATION | |
| +A1 | Controlpanel for conveyor belt |
| +A2 | Main switchboard for conveyor belt |
| +W1 | Conveyor belt |

— Also refer to the figure in “Project definition file example” on page 25 and “The significance of the first project page in the project definition file” on page 26.

**Reference designations in the individual projects when generating projects**

If there exists reference designations in the different projects / project pages, these will also become a part of the generated project.

If the same reference designations exists in more than one project, and these reference designations have different attached descriptions, the descriptions for the first project / project page, in which they occur, will be attached to the generated project.

*Reference designations in applied sub drawings when generation projects*

When ***PCschematic ELautomation*** via applied sub drawings (see “Generate and Update projects automatically” on page 9) meets reference designations, which have not been created in the project, these reference designations will be created in the generated project.

Because sub drawings can not contain descriptions for reference designations, these reference designations will have no descriptions in the generated project.

Please note that the program will issue no warning about this.

**Controlling reference designations for individual pages**

— See "Sub drawings" in the ***PCschematic ELautomation*** manual, "Variables for Page title and Page data" in particular.

## SPECIFYING DRAWING TEMPLATES AND SUB DRAWINGS

The first project information you must specify, is the name of the drawing template, which must be placed in column **A** (in the column with the heading TEMPLATE).

After this you write the name of the drawing template in column **B**. When you do not write anything in column **B**, it is assigned the next available page name.

Under this — in column **A** — you place the names of the sub drawings, which shall be placed on the specified drawing template. In column **B** you write the character ~, which tells the program that this is a sub drawing. If there is no ~ character, the program knows it is a drawing template.

When more than one project page with the specified drawing template is required, these are inserted automatically — see "Sub drawings" in the ***PCschematic ELautomation*** manual, "Insertion points and automatic project generation" in particular.

When you need a drawing template or a sub drawing from another folder than the one that is specified in the **Options** dialog box — see “Settings for the Project Generator” on page 14 — (or under *FOLDERS* in the project definition file, see “FOLDERS: Specifying applied folders / Aliases” on page 24), you must type in the path to this folder in front of the drawing template or the sub drawing.

**MERGE WHEN MERGING PROJECTS**

IN STEAD OF **TEMPLATE** YOU CAN ALSO WRITE **MERGE** IN THE PROJECT DEFINITION FILE. THIS IS USED FOR MERGING PROJECTS VIA THE **PROJECT GENERATOR**. — SEE “KEEPING THE ORIGINAL PAGE NUMBERS WHEN MERGING PROJECTS” ON PAGE 43.

## CONTROLLING TEXTS IN DRAWING TEMPLATES / SUB DRAWINGS

In the automatic project generation you can control *all* texts and names in the project. This chapter describes how to control the texts in drawing templates and sub drawings.

To be able to control the texts in a sub drawing or a drawing template via the project definition file, this requires that *variable markers* have been inserted, and that a *variable list* have subsequently been created for the sub drawings / drawing templates — see "Sub drawings", and in particular "Creating a variable list for a sub drawing / drawing template" in the ***PCschematic ELautomation*** manual.

THIS CHAPTER DESCRIBES


— VARIABLES IN THE PROJECT DEFINITION FILE ..... (P. 31)
— EXAMPLE ON A PROJECT DEFINITION FILE ..... (P. 33)
— INVISIBLE SYMBOLS IN SUB DRAWINGS ..... (P. 35)
— PROJECT GENERATION WITH NO SYMBOL NAMES SPECIFIED ..... (P. 35)
— VARIABLE VALUES IN VALUE STRINGS ..... (P. 35)


### Variables in the project definition file

In the columns immediately after the drawing template / sub drawing — typically in column **C** — the values for the *variables* on the drawing template / sub drawing are specified. — See "Inserting variables in Sub drawings, Pages and Projects" in the ***PCschematic ELautomation*** manual.

**Sub drawing (or drawing template)** — **When the variables in the sub drawing are inserted in the *project definition file***

In the project definition file you must write the name of the sub drawing for each time the sub drawing shall be placed, followed by the *variable* values for this specific sub drawing.

When the project is generated, the values of the variables are fetched from the project definition file, the values are inserted in the sub drawing, and the sub drawing is inserted in the project.

**The sub drawing is inserted in the drawing template, and the *variable* values from the project definition file are inserted. *Variables* can also be inserted in drawing templates.**

**Sub drawing (*Variables* /@1 and /@2)**

**Part of project definition file (*Variable* /@1 and *variable* /@2)**

**Generated project (*Variable* values inserted)**

— See also the figure below “Example on project definition file with drawing templates and sub drawings” on page 33.

**SEE APPENDICES FOR VARIABLES AND PROJECT GENERATION**

PLEASE NOTE THE APPENDIX “VARIABLES” ON PAGE 47 AND THE APPENDIX “STRUCTURE FOR PROJECT DEFINITION FILES” ON PAGE 49.

**Example on project definition file with drawing templates and sub drawings**
A project definition file specifying drawing templates and sub drawings, could for instance look like this:

**Drawing template *L1L2L3DEMO* with page name *1***

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | | | | | | | |
| 2 | TEMPLATE | PAGE | | | | | |
| 3 | L1L2L3NDEMO | 1 | | | | | |
| 4 | | | S.Name | S.Name | S.Name | S.Name | S.Name |
| 5 | MOTORDEMO1 | ~ | -Q1 | -F1 | -X1 | -W1 | -M1 |
| 6 | MOTORDEMO1 | ~ | -Q2 | -F2 | -X2 | -W2 | -M2 |
| 7 | MOTORDEMO1 | ~ | -Q3 | -F3 | -X3 | -W3 | -M3 |
| 8 | MOTORDEMO1 | ~ | -Q4 | -F4 | -X4 | -W4 | -M4 |
| 9 | MOTORDEMO1 | ~ | -Q5 | -F5 | -X5 | -W5 | -M5 |
| 10 | MOTORDEMO1 | ~ | -Q6 | -F6 | -X6 | -W6 | -M6 |
| 11 | | | | | | | |
| 12 | | | S.Name | S.Func | S.Name | S.Name | S.Name |
| 13 | MOTORDEMO2 | ~ | -M7 | Main Conveyor belt | -F7 | -W7 | -X7 |
| 14 | MOTORDEMO2 | ~ | -M8 | Left Conveyor belt | -F8 | -W8 | -X8 |
| 15 | MOTORDEMO2 | ~ | -M9 | Right Conveyor belt | -F9 | -W9 | -X9 |
| 16 | MOTORDEMO2 | ~ | -M10 | Upper Conveyor belt | -F10 | -W10 | -X10 |

**The sub drawing *MOTORDEMO1***
**- with the *variable* values @1 (-Q1) to @5 (-M1)**
**The sub drawing is inserted (here: 6 times) on the last mentioned drawing template (here: *L1L2L3DEMO*)**

— Please note that automatic naming during project generation is not applied in the figure above, because the symbol names are controlled directly from the project definition file.

**Naming connectionpoints in the project definition file**
When you wish to specify connection names for a symbol, you just write **:** (colon) after the symbol name in the project definition file, followed by the connection name.

Example: If the name of a terminal is specified as **-X1:1** in the project definition file, the symbol gets the name **-X1**, and the connection name is set to **1**.

For symbols with more than one connection name, the connection names are separated by commas.

Example: If the name of a symbol with three connectionpoints is specified as **-M1:U,V,W** in the project definition file, the symbol is assigned the name **-M1** and the connection names are set to **U**, **V** and **W** respectively.

### Create model variable

For use in automatic project generation, you can create a so-called *model variable*, so that you from the project definition file can specify which of the *models* for a subdrawing, that you wish to insert.

To create a model for a subdrawing, do this:

1) In the **Subdrawings** window: Right-click on the subdrawing, and choose **Edit**.

2) Right-click on the page and select **Variables**.

3) In the dialog box **Variables**: Right-click and select **Model variable**.

4) Type in an unused variable name, and click **OK**:

5) The name of the model variable is then displayed in the title bar of the dialog box **Variables**:

6) In the *project definition file* you then only have to specify the **Model title** for the desired model in the subdrawing. You do this in the **Model variable** field (here: **/@10**). Hereby the values for the other variables for the subdrawing are fetched automatically from the model data in the subdrawing definition. (If you just write *55 kW* in the figure below, the model ***55 kW 40 - 100 A*** will be inserted. This is because ***PCschematic ELautomation*** selects the first model title that matches the text you write in the model variable field).

### Variable independant of model selection

As described above — see “Create model variable” on page 34 —, it is possible to specify in the project definition file that a subdrawing gets its variable values from one of the created models for the subdrawing.

In the definition of a subdrawing, it is also possible to specify that the contents/ value of a variable in a model is fetched from the project definition file.

To do this, write **/?** in the first model for a variable (here: for the variable **/@6**) in the dialog box **Variables**:

This specifies that all models in the subdrawing fetches the value for this variable from the project definition file.

### Invisible symbols in sub drawings

If a symbol and its connecting lines shall be invisible, you must set the variable for the symbol name to ***!*** in the project definition file.

If the connecting lines are cable wires, and you wish to stop the invisibility at the cable symbol, you must set the variable for the symbol name to ***!C*** in the project definition file.

### Project generation with no symbol names in the project definition file

If you do not wish to specify a symbol name from the project definition file — and a variable marker has been created for the symbol in the sub drawing —, you must set the variable for the symbol name to the value ***?*** in the project definition file.

If ***Auto name symbols during generation*** is not selected, the symbol will then be assigned the name specified in the original sub drawing.

### Variable values in Value strings in Project generation

Variable markers in sub drawings / drawing templates / project pages can be indexed, so that they can fetch values from a comma separated value string in the project definition file.

For instance will **/@1[2]** (on the project page / sub drawing) fetch the *second* value from e.g. the value string ***Brown, Black, Blue*** (from the project definition file).

*Without* the index, the variable **/@1** would be assigned the value ***Brown, Black, Blue***. *With* the index it gets the value ***Black*** — see below:

*Fetching multiple values simultaneously from the value string*

A variable can contain more than one index number, so that multiple values can be fetched from the string.

For instance will **/@1[1,3]** fetch the value ***Brown, Blue***. When multiple values are selected in a string, the values are separated by a comma.

## FRONT PAGES, DIVIDERS AND LISTS IN AUT. GENERATED PROJECTS

When you wish to insert pages, which does not use the intelligence from the project definition file, you just write the name of the page in a line in the project definition file.

The name of the project / drawing template is written in column **A**, and the name of the page can be specified in column **B**.

These projects / drawing templates could for instance include front pages, dividers or pages containing lists.

## COMMENTS IN THE PROJECT DEFINITION FILE

To place comments in the project definition file, you just have to let the first field in a row be empty. Please note, that this is only possible below the code word TEMPLATE in the project definition file.

In the figure in “Example on project definition file with drawing templates and sub drawings” on page 33, the lines with the texts ***S.Name*** and ***S.Func*** are comments, just like the blank line is a comment line. Comment lines can be inserted anywhere you want in the file.

## INSERTING NEW PAGES WHEN UPDATING EXISTING PROJECTS

When you wish to insert new pages when updating existing projects, you can specify this in the **TEMPLATE** column.

Here you write **NEWPAGES>>** *before* the list of new pages, type in the new pages, and write **<<NEWPAGES** *after* the inserted pages.

Example:

... Existing pages ...
**NEWPAGES>>**
... New pages ...
... New pages ...
**<<NEWPAGES**
... Existing pages ...

---

**REMEMBER TO DELETE NEWPAGES**

PLEASE REMEMBER TO DELETE THE **NEWPAGE** TEXTS FROM THE PROJECT DEFINITION FILE, IN CASE YOU AT A LATER STAGE WISH TO GENERATE THE PROJECT BASED ON THE PROJECT DEFINITION FILE ONCE AGAIN.

---

## COMMA SEPARATED FILES FOR AUTOMATIC PROJECT GENERATION

You can also use comma separated files for automatic generation of ***PCschematic ELautomation*** projects. These can for instance be generated directly from a company database.

Using comma separated files can be done similar to using Excel files. Here you just have to select a comma separated file (****.csv***) in stead of an Excel file in the description “Generate projects based on Project definition files” on page 18.

The syntax for the comma separated files is the same as described in “Syntax for project definition files” on page 23.

Remember to specify the settings for the comma separated file as described in “Settings for the Project Generator” on page 14.

# Merging Projects

This section describes how to merge ***PCschematic ELautomation*** projects via the **Project generator**.

Please note the appendix “Structure for Project definition files” on page 49.

# MERGING PROJECTS VIA THE PROJECT GENERATOR

In ***PCschematic ELautomation*** it is also possible to use the **Project generator** for merging projects.

THIS CHAPTER DESCRIBES

— BENEFITS OF MERGING PROJECTS ..... (P. 41)
— HOW TO MERGE PROJECTS ..... (P. 42)
— KEEPING THE ORIGINAL PAGE NUMBERS ..... (P. 43)

## BENEFITS OF MERGING PROJECTS

When you merge projects via the **Project generator**, it is possible to let more than one employee work on the same project at the same time.

When you, as the constructor of a part of the project, need to know exactly how far the other constructors have come with their parts of the project — or e.g. need to know how another constructor has named a specific component —, you can create the full project via the **Project generator**. When this project is generated, this is done based on the last saved version of each constructors part of the project. Any of the constructors will at any time be able to generate the full project, as described in “How to merge projects” on page 42.

This makes it possible for each constructor to work on much smaller project files.

### Naming when merging projects

When merging ***PCschematic ELautomation*** projects via the **Project generator**, the program does *not* rename symbols, which appears with the same name in the different projects.

However, this is not the case if you have selected ***Auto name symbols during generation*** under **Options** in the **Project generator** — see “Auto name symbols during generation” on page 16.

When merging projects it is required that the constructors agrees on a convention for naming the symbols in the project. — For instance by assigning different series of numbers to the different constructors, or by using reference designations. — See "Reference designations" in the ***PCschematic ELautomation*** manual.

## HOW TO MERGE PROJECTS

A simple example on how e.g. four project files can be merged via Excel:

1) Save the four project files — e.g. the project files **frontpages.pro**, **relays.pro**, **plcs.pro** and **lists.pro** — in the same folder. (You can also fetch the files from different folders — see "FOLDERS: Specifying applied folders / Aliases" on page 24).

2) Open the program Microsoft Excel, and create an Excel file (the *project definition file*) with the following contents:
   - Write FOLDERS at the top of column **A**, and specify the path to the folder in the next line
   - In the row below: Write TEMPLATE in column **A** and PAGE in column **B**
   - Below this the project names are written on separate lines in column **A** — according to the order in which you want the project files to be merged:

| | A | B |
|---|---|---|
| 1 | FOLDERS | |
| 2 | C:\Pcselcad\Project\Excel_Merge | |
| 3 | | |
| 4 | TEMPLATE | PAGE |
| 5 | frontpages | |
| 6 | relays | |
| 7 | plcs | |
| 8 | lists | |
| 9 | | |

3) Save the Excel file in the same folder as the project files.

4) In ***PCschematic ELautomation***: Choose **Tools => Project Generator**:

5) Click on ***Generate new project from project definition file***, choose the Excel document at the bottom of the dialog box below ***Project definition file***, and click **OK**. (If you wish to change settings, click on **Options** — see "Settings for the Project Generator" on page 14).

6) The project files are merged in a new project, with the same name as the Excel file.

7) If you wish to update the lists in the project, choose **Lists => Update all lists**. (This can be done automatically — see “Settings for the Project Generator” on page 14).

---

**OTHER OPTIONS WHEN MERGING PROJECT FILES:**
— “PROJECT TITLE AND PROJECT DATA IN PROJECT GENERATION” ON PAGE 28
— “REFERENCE DESIGNATIONS IN PROJECT GENERATION” ON PAGE 28
— “GENERATE AND UPDATE PROJECTS AUTOMATICALLY” ON PAGE 9

---

**Specifying page number start when merging**
When merging projects, you can use the **PAGE** column to specify the start page number for the individual projects, as displayed below.

| | A | B |
|---|---|---|
| 1 | TEMPLATE | PAGE |
| 2 | frontpages | 100 |
| 3 | relays | 200 |
| 4 | plcs | 300 |
| 5 | lists | 400 |

Otherwise, the pages will be numbered consecutively — see also “Keeping the original page numbers — for individual projects” on page 43.

**Keeping the original page numbers — for individual projects**
If you wish to keep the page number for a specific project, you type in the text **#REMAIN** in the **PAGE** column. The page number(s) which the project is created with, are hereby transferred unchanged to the automatically generated project.

## KEEPING THE ORIGINAL PAGE NUMBERS WHEN MERGING PROJECTS

When you wish to keep the original page numbers from the project files (or *drawing templates*) during automatic project generation, you can do this for all projects in the project definition file, or for individual projects:

**Keeping the original page numbers — for all created project pages**
If you create a **MERGE** column in stead of a **TEMPLATE** column — see “Specifying Drawing templates and Sub drawings” on page 30 —, the page numbers remains unchanged on all merged project pages.

Therefore you only need a project definition file / Excel file which only consists of the following lines, which merges three projects to one new project:

**MERGE**
**ProjectA**

**ProjectB**
**ProjectC**

The page numbers from the three projects are then transferred unchanged to the automatically generated project.

# Appendices

Here you will find the following:

- Appendix of options for variables .....(p. 47)
- Appendix for project def. file syntax .....(p. 49)
- An Index .....(p. 51)

## VARIABLES

**Variable definition: /@n**

| | |
|---|---|
| /@n | n is a number between 1 and 64000. Variables containing small numbers will be processed faster than variables containing larger numbers. Therefore smaller numbers are recommended in variables. |
| /@1 Variable 1 | /@10 Variable 10 |

**Normal variable: /@n**

| | |
|---|---|
| /@1 | The variable has no default value. Fields in the project definition file, where the variable value will be typed in later, are empty. |

**Variable with default value: xxxx/@n**

| | |
|---|---|
| 3389110610000/@1 | 3389110610000 is displayed in field in the project definition file, where the variable value will be typed in later. |
| -P1/@3 | The variable has the default value –P1.<br>The value that is typed in later, overwrites the default value. |

**Variable that inserts a value in a text: xxxx</@n>**

| | |
|---|---|
| -P</@3> | The text (name) has a fixed part (-P) and a variable part </@3> |
| </@4> out | |
| </@4> in | Can be applied with benefit where the same value shall be inserted into texts, where the fixed part are different from text to text. Here the values could for instance be 'Weighing' in one case, and 'Casting' in another. |
| -P</@3> | |
| -S</@3> | Can be applied where all — or some — names in a sub drawing always have the same serial number.<br>E.g. –P1 and –S1 |

**Insertion with default value: xxxx<yyyy/@n>**

| | |
|---|---|
| -P<1/@3> | The text (name) has a fixed part (-P) and a variable part </@3>, which has the default value '1'. |
| <Weighing/@4> out | The variable has the default value 'Weighing'. |

| **Indexed variables: /@n[i] or /@n[i1,i2,i3,....]** | |
|---|---|
| /@4[1] | The variable takes the value from the first parameter in a comma separated text.<br>E.g.: BK,BL,BU,GR The variable gets the value BK |
| /@4[2] | E.g.: BK,BL,BU,GR The variable gets the value BL |
| /@4[2,3] | E.g.: BK,BL,BU,GR The variable gets the value BL,BU<br>This function is very useful for cables.<br>Below this is shown with cable wire colors. |
| < BK,BL,BU,GR/@4[1]> | => BK |
| < BK,BL,BU,GR/@4[2]> | => BL |
| < BK,BL,BU,GR/@4[3]> | => BU |
| < BK,BL,BU,GR/@4[4]> | => GR |

## STRUCTURE FOR PROJECT DEFINITION FILES

A project definition file consists of three parts, of which the first two are optional.

### Part 1: Specifying aliases / applied folders (Optional)

| **FOLDERS** | |
|---|---|
| Name for Alias 1<br><br>E.g.: **FOLDER1** | Path to the *first* folder to examine for drawing templates, sub drawings and projects.<br>E.g.: **C:\PCschematic\MySubDrawings** |
| Name for Alias 2 | Path to the *second* folder to examine for drawing templates, sub drawings and projects. |
| Name for Alias 3<br>(Relative to other alias) | E.g.: **%FOLDER1%\SubDrawings**<br>— Can both be relative to aliases defined in the **Project generator** dialog box, and to aliases defined on a previous line here in the project definition file. |
| ... | ... |

### Part 2: Specifying Project data and/or Reference designations (Optional)

| **PCSCHEMATIC DATA** | |
|---|---|
| | |
| **PROJECTDATA** | |
| Datafield from Project data | Value for datafield |
| Datafield from Project data | Value for datafield |
| ... | ... |
| | |
| **REFERENCE DESIGNATIONS** | |
| **FUNCTION** | |
| Function aspect | Description |
| Function aspect | Description |
| ... | ... |
| **LOCATION** | |
| Location aspect | Description |
| Location aspect | Description |
| ... | ... |

**Part 3: Specifying Drawing templates, Sub drawings and Projects**

| | | | | |
|---|---|---|---|---|
| **PCSCHEMATIC PAGES** | | | | |
| | | | | |
| **TEMPLATE** | **PAGES** | | | |
| Multipage project | Page no. / Empty | | | |
| Drawing template / Single-page project | Page no. / Empty | @1 | @2 | ... |
| Sub drawing | ~ | @1 | @2 | ... |
| Project / Page | **#REMAIN** | | | |

*When Updating projects*

| | | | | |
|---|---|---|---|---|
| **NEWPAGES>>** | | | | |
| Multipage project | Page no. / Empty | | | |
| Drawing template / Single-page project | Page no. / Empty | @1 | @2 | ... |
| Sub drawing | ~ | @1 | @2 | ... |
| Project / Page | **#REMAIN** | | | |
| **<<NEWPAGES** | | | | |

*When Merging projects*

**MERGE**
Project 1
Project 2
Project 3
...
Project n

*Possible control via variables in symbol names*

| | |
|---|---|
| /@1=-M1 | => The symbol gets the name **-M1** |
| /@1=! | => The symbol and connecting lines becomes invisible. |
| /@1=!C | => The symbol and connecting lines — up to a cable symbol — becomes invisible. |
| /@1=? | => The symbol name is not controlled from the variable. |

# INDEX


**A**

Alias
- specify in project definition file . . . . . 24

Automatic project generation
- about . . . . . 9

Automatically generated names . . . . . 26

**C**

Cables
- invisible, nonreferenced . . . . . 17

Comma separated files . . . . . 37

Comments . . . . . 36

Create
- projects via project def. file . . . . . 9

Create project def. file from active project . . . . . 13

Current path numbers . . . . . 26

**D**

Default vaule
- for variable . . . . . 47

Dividers . . . . . 36

Drawing templates
- create
  - for project generation . . . . . 10
- specifying in project def. file . . . . . 30

**E**

Edit =>
- Variables . . . . . 19

Excel
- create and update projects via . . . . . 9
- syntax for project def. files . . . . . 23

**F**

Files
- comma separated files
  - for automatic project generation . . . . . 37

FOLDERS . . . . . 49
- specifying aliases . . . . . 24

Front pages . . . . . 36

FUNCTION . . . . . 28, 49

**G**

Generate new project, dialog box . . . . . 19

**I**

Indexed variables . . . . . 48

Inserting new pages . . . . . 37

Invisible
- cables and signals . . . . . 17

**L**

Lists . . . . . 36

LOCATION . . . . . 28, 49

**M**

MERGE . . . . . 43, 50

Merge projects . . . . . 41
- how to . . . . . 42
- naming when merging projects . . . . . 41
- secifying page number start when merging . . . 43

**N**

NEWPAGES . . . . . 37, 50

**P**

Page numbers
- keep . . . . . 43

PAGES . . . . . 50

PCSCHEMATIC DATA . . . . . 28, 49
- Project data and Reference designations . . . . . 25

PCSCHEMATIC PAGES . . . . . 50
- files, drawing templates, sub drawings etc. . . . 25

PLC
- synchronization . . . . . 16

PLC reference . . . . . 16

PLC reference >> PLC . . . . . 16

Project data, syntax . . . . . 28

Project definition file
- example with variables . . . . . 33
- generate and update projects . . . . . 9
- specify aliases . . . . . 24

Project definition files
- about . . . . . 12
- appendix of syntax . . . . . 49
- comments . . . . . 36
- example . . . . . 25
- front pages, dividers and lists . . . . . 36
- generate from PCschematic . . . . . 13
- generate projects based on . . . . . 18
- Project title and Project data . . . . . 28
- reference designations . . . . . 28
- significance of the first project page . . . . . 26
- structure and contents . . . . . 24
- syntax . . . . . 23
- the purpose of project def. files . . . . . 14
- what can be controlled . . . . . 23
- without symbol names . . . . . 35

Project generation
- about . . . . . 9
- drawing templates . . . . . 10
- editing variables after generating project . . . . . 19
- model data in automatic project generation . . . 12
- paper size and Reference system . . . . . 11
- points to consider . . . . . 10
- sub drawings as Modules . . . . . 11
- sub drawings based on existing projects . . . . . 11
- using variable . . . . . 11

Project generator

settings . . . . . 14
Project pages
the first page in project def. files . . . . . 26
Project title, syntax . . . . . 28
PROJECTDATA . . . . . 49
Projects
create and update projects . . . . . 9
generate based on project def. files . . . . . 18
merge . . . . . 41

**R**

REFERENCE DESIGNATIONS . . . . . 49
Reference designations
in individual projects . . . . . 29
syntax . . . . . 28
REMAIN . . . . . 43, 50

**S**

Signals
invisible, non referenced . . . . . 17
Structure
of project definition files . . . . . 24
Sub drawings
invisible symbols in . . . . . 35
specifying in project def. file . . . . . 30
Symbol names
current path numbers / auto generated names . . 26
Symbols
without name in project def. file . . . . . 35

**T**

TEMPLATE . . . . . 30, 50
Texts
in drawing templates / sub drawings . . . . . 31
visibility . . . . . 27
Tools =>
Project generator . . . . . 18
Project generator => . . . . . 13
Options . . . . . 14

**U**

Update projects
inserting new pages . . . . . 37
via project def. file . . . . . 9
Update the active project, dialog box . . . . . 20

**V**

Value strings, for variables . . . . . 35
Variables . . . . . 47
in value strings . . . . . 35
indexed variables . . . . . 48
insertion with default value . . . . . 47
normal variable . . . . . 47
possible control via var. in symbol names . . . . 50
that inserts a value in a text . . . . . 47
with default value . . . . . 47