

Programs, from PG3 to PG5

Content :

1) Introduction

2) Import of user's programs from PG3 to PG5

1.1) Procedure

1.2) Importation of options

3) Structure of program conversion for PCD2/4.H310

2.1) Structure of a PG3 program

2.2) Structure of a PG5 program

Attached :

Example of a program PCD2/4.H310 :

A) with PG3

B) with PG5

SAIA-Burgess Controls Ltd. 2001. All rights reserved.
October.2002

Subject to technical changes

1) Introduction

This document will describe how to import programs from PG3 to PG5.

Depending how old are your modules, how the programs are written, the full success without manual correction is possible. If no, you will have to make manually some adaptations.

The chapter 2 is a general description for all PG3 programs, the chapter 3 as well the attached files are dedicated to PCD2/4.H310 modules.

2) Import of user's programs from PG3 to PG5

2.1) Procedure

If each PG3 project is already written and saved in different directory, the Pg3 program can be imported with the followings manipulations:

Import (Add files) all the src and sfc file in one PG5 CPU from PG3 project

- all the files *.src must be imported in PG5

Remarks concerning the H modules :

- The libraries for new H modules are already installed in the folder ../PG5/FB and must not be imported.
- Import the files H3def.src and H3fb.src from the folder ../PG5/FB/D4h3_old
- More information are described in the file Readme.txt of this folder.

Separate the cyclics and graftecs programs

With PG3 the program blocks COB,PB,FB,XOB and SB could be written in one file *.src

With Pg5 we must separate the cyclic and graftec programs, to do that :

- duplicates the file
- delete the cyclic part in the first file
- delete the graftec part in the second file. This file must have at least the XOB16, COB0.
- the cyclic and graftec files must have a different file name
- the cyclic file have an extension *.src
- the graftec file have an extension *.sfc with pg5
- the graftec file must start with the instruction SB. Comments or equates table or DOC table can make problems and must be deleted

Remark concerning the H modules :

Verify if the IST include the definitions for H modules:

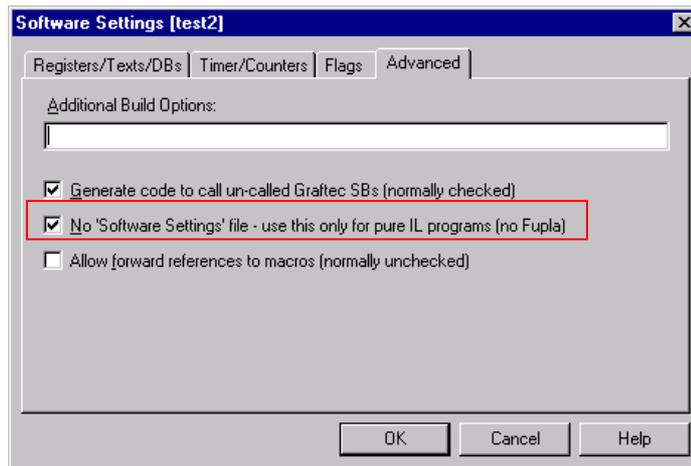
userdef.src	for PCD4.H3
.....	for PCD4. H2
.....	for PCD4.H1

Adjust the hardwares settings PG5

- Open the project PG3
- In menu "Configure", all the settings hardware are defined under "SAIA PCD CONFIGURATION"
- Then put the parameters of PG3 in the "Setting Hardware" of PG5.

Adjust the Software Settings PG5

The easiest solution is to disable the software settings (Advanced) with “No software settings” :



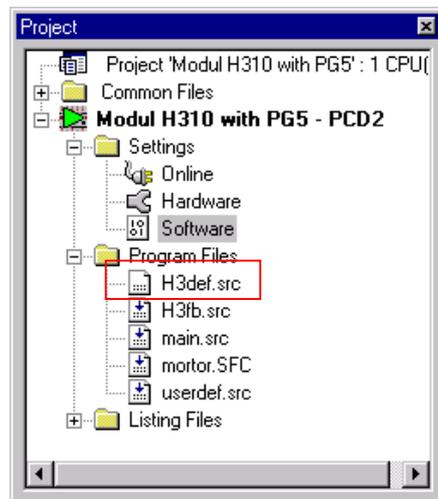
So we continue to work under PG5 without to use any software settings.

Remarks :

- Dynamic symbols, timer/counter definitions, time base definition and volatile flags definition can't be modified from software settings. We must support these definitions from IL program with the DEFTC,DEFTB,DEFVM instructions in a xob 16.
- When the software setting are disable, the DB/TXT 3999 must always be free for PG5 programming tool.
- You must verify that your program don't use this DB.
- Concerning H modules :
 - For PCD2/4/.H3 application a modification is necessary in file userdef.src
 - For PCD4.H1/2 it is probably the same thing

Adjust file which must be linked.

It can happen that some files are not used or include in an other file, for these reasons, we must open the project under PG3 and see in menu “Assembler” and “Link” which files are to assembled and linked. Then we have to select the files who must be linked under PG5.



Correct the src file layout.

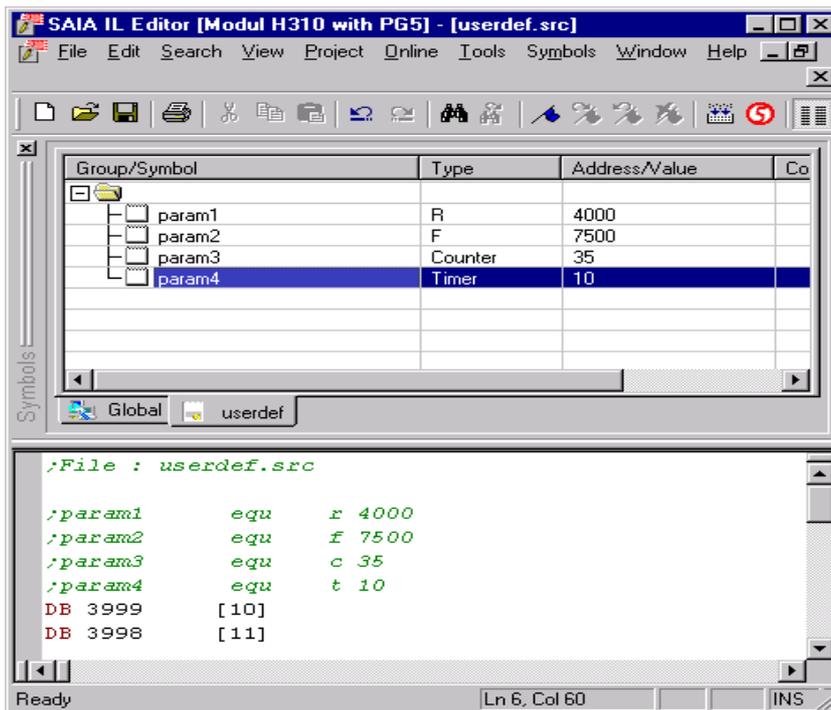
- Open each src file
- Mark all the program and select the menu: Tool, auto format

2.2) Importation of options

Move the resources equates table in the symbol editor

- Open each src file
- Mark all the EQU table and move it in symbols editor, menu Tools → Move EQU to Symbol Table

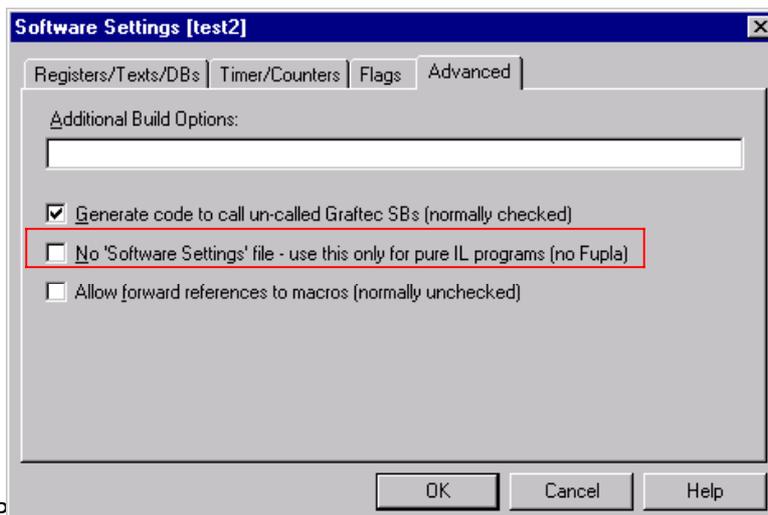
The Text and DB definitions will not moved in the symbols editor. You can let it with the program code or rewrite it manually in symbols editor.



If you want to use dynamics symbols

Enable the software settings and you will have the possibility to use dynamic symbols and to define the number of Timer/Counter, the time base, the volatile flags from software settings.

- Disable the software settings with “No software settings” :



- Adjust the software settings to the values defined in the XOB 16 with the definitions instructions.
- The definitions instructions must be removed from XOB 16

```

XOB 16

;DEFTB
;DEFTC
;DEFVM

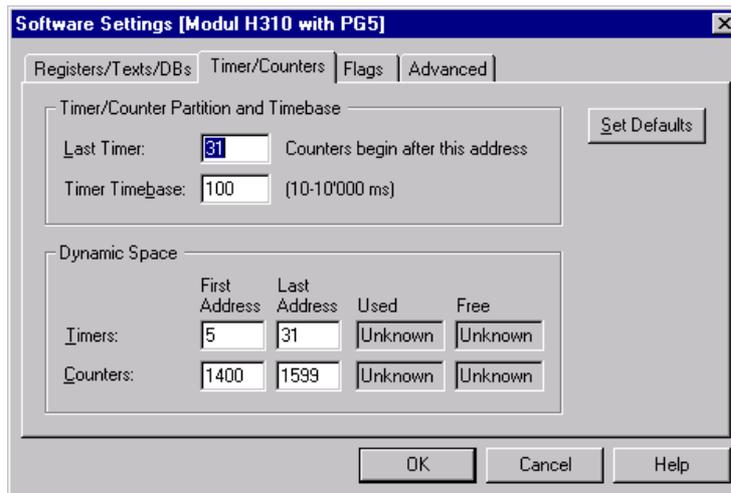
EXOB
  
```

- The dynamics resources ranges must be adjust to the application.

Comment :

The main problem is that user resources are certainly defined in the dynamic space.

- If it is possible, we have to define this dynamic space in an other range.
- If is not possible, we have to renumber the R/T/C/F,X,DB addresses. This renumbering could disturb the program if the user don't use symbol names and when a supervision or terminal must modify registers, flags,....



Remark :

For programs with H modules, the library definition must certainly be renumbered.
 Example for PCD2/4.H3: Adjust the basis addresses for the library H310 in the file :

File : H3def.SRC

```

MNA EQU 2 ; Max. Number of Axes
;;===== Memory Media:
BAF EQU 100 ; Base Address Flags
BAR EQU 100 ; Base Address Registers
BAC EQU 200 ; Base Address Counters
;;===== Functionblocks
BAFB EQU 900 ; Base Address Functionblocks
;;===== Axesconstants specification
  
```

Build the program and track the last errors listed in the errors message window.

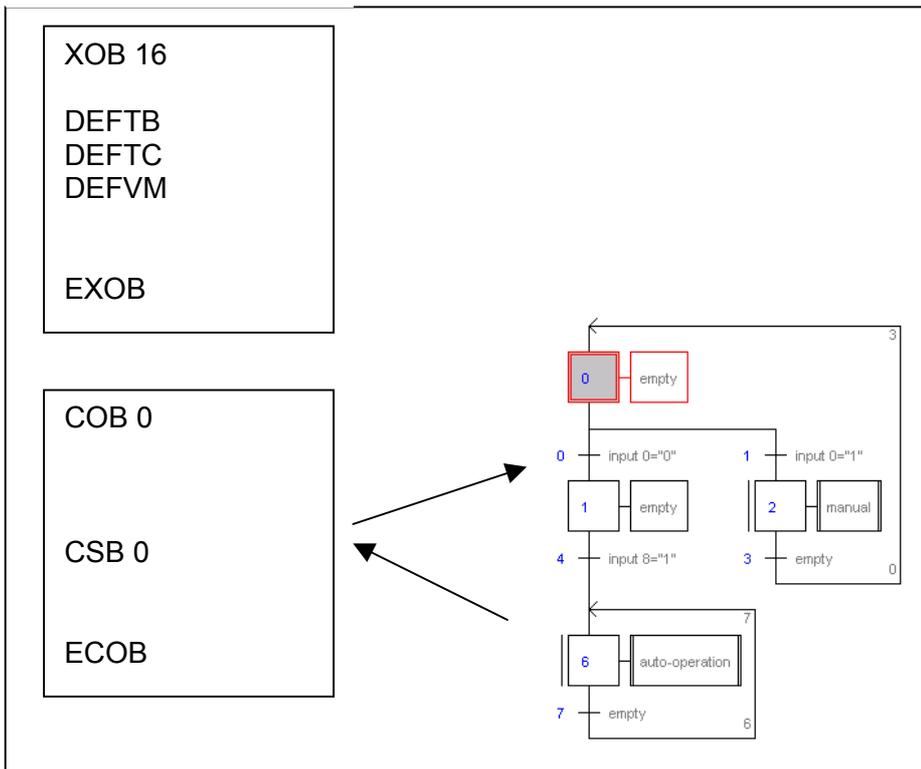
3) Structure of program conversion for PCD2/4.H310

3.1) Structure of a PG3 program

Fichier : userdef.src

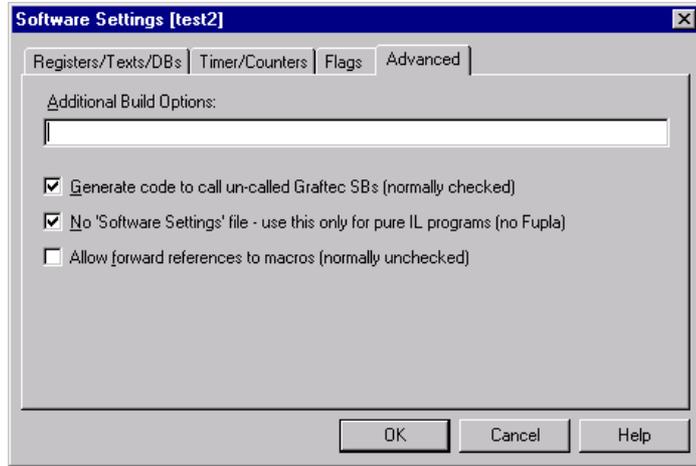
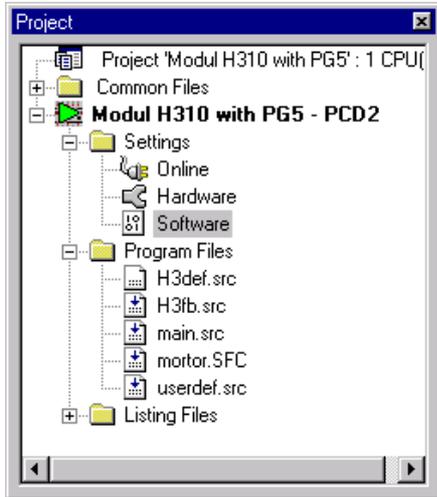
```
param1 equ r 4000
param2 equ f 7500
param3 equ c 35
param4 equ t 10
DB 3999 [10]
DB 3998 [11]
```

Fichier : BSP03.SRC



3.2) Structure of a PG5 program

Base structure :

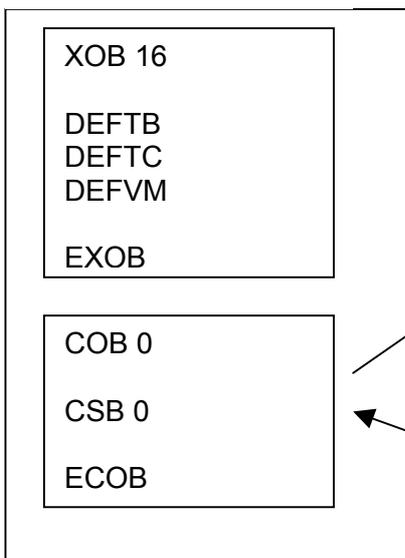


Fichier : userdef.src

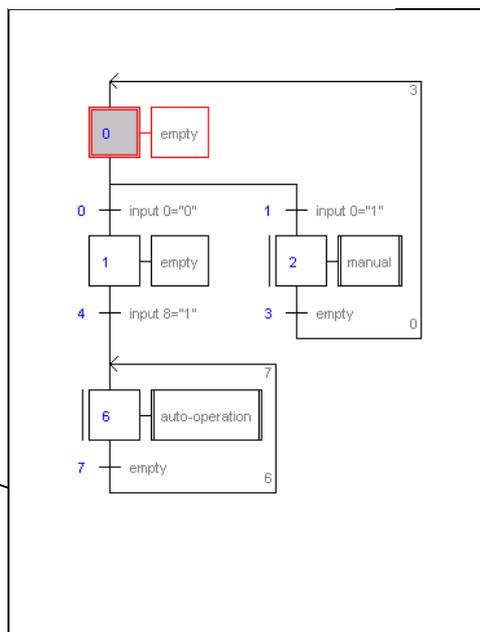
```

param1    equ    r 4000
param2    equ    f 7500
param3    equ    c 35
param4    equ    t 10
DB 3997  [10]
DB 3998   [11]
  
```

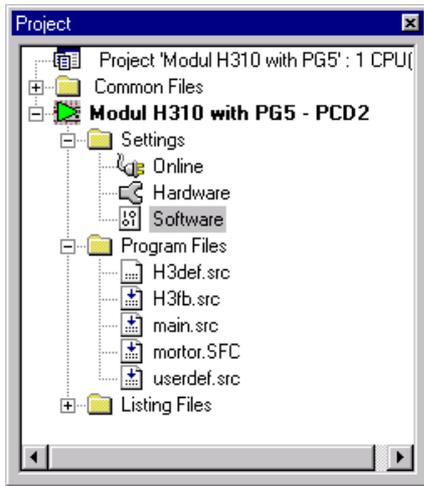
Fichier : main.SRC



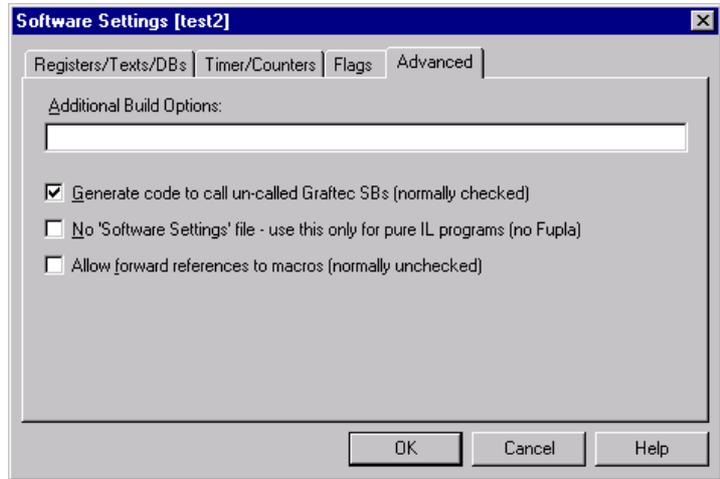
Fichier : sequence.SFC



Importation of options :

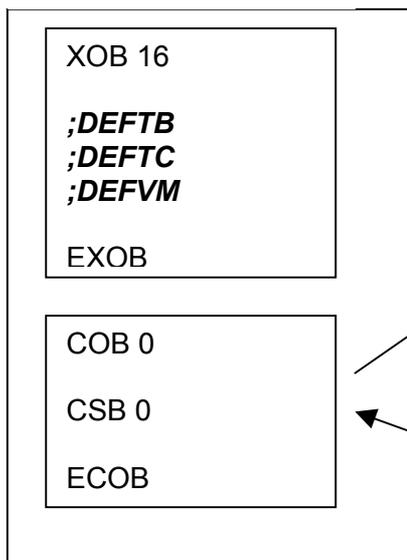


Global resources

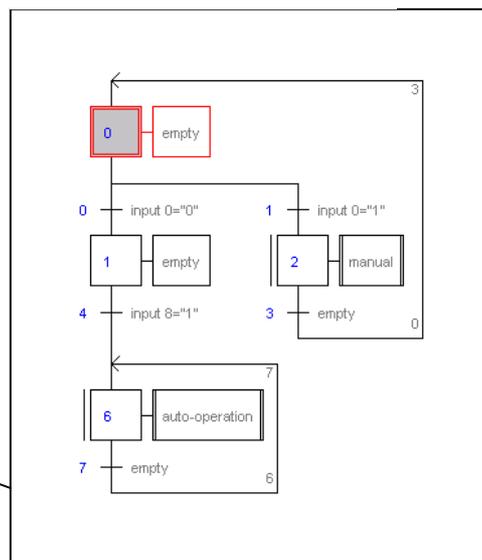


Group/Symbol	Type	Address/Value	Comment
param1	R	1000	
param2	F	750	
param3	Counter	100	
param4	Timer	2	

Fichier : main.SRC



Fichier : sequence.SFC



Fichier : H3def.SRC

```
MNA EQU 2 ; Max. Number of Axes
;===== Memory Media:
BAF EQU 100 ; Base Address Flags
BAR EQU 100 ; Base Address Registers
BAC EQU 200 ; Base Address Counters
;===== Functionblocks
BAFB EQU 900 ; Base Address Functionblocks
;===== Axesconstants specification
```