

MA160-870 (v1.0) February 12, 2016

Copyright © 2016 Altium BV.

All rights reserved. You are permitted to print this document provided that (1) the use of such is for personal use only and will not be copied or posted on any network computer or broadcast in any media, and (2) no modifications of the document is made. Unauthorized duplication, in whole or part, of this document by any means, mechanical or electronic, including translation into another language, except for brief excerpts in published reviews, is prohibited without the express written permission of Altium BV. Unauthorized duplication of this work may also be prohibited by local statute. Violators may be subject to both criminal and civil penalties, including fines and/or imprisonment. Altium, TASKING, and their respective logos are trademarks or registered trademarks of Altium Limited or its subsidiaries. All other registered or unregistered trademarks referenced herein are the property of their respective owners and no trademark rights to the same are claimed.

Table of Contents

1. Introduction	
1.1. Port Pin	1
1.2. Peripheral	1
1.3. Connection	1
1.4. Package	2
2. Getting Started with the TASKING Pin Mapper	3
2.1. Adding a Pin Mapper Document to an Existing Project	3
2.2. TASKING Pin Mapper Perspective	4
2.2.1. TASKING Pin Mapper Editor View	5
2.2.2. Package View	9
2.2.3. Pin Conflicts View	12
2.3. Configuring the Pins	13
2.4. Solving Pin Conflicts	
2.5. Generating and Using the Source Code	16
2.6. Generate CSV File	
2.7. Launch Altium Designer	18
2.8. Pin Mapper Preferences	18
2.9. Errors and Warnings	18

Chapter 1. Introduction

Nowadays many microcontrollers are equipped with a large number of on-chip peripheral modules. These microcontrollers are made available in surface mount packages with various numbers of physical pins. The number of pins usually does not allow all peripheral modules to be used simultaneously. Hardware registers on the microcontroller allow for configurable assignment of peripheral module signals to physical pins. This means that you have to select the proper device for your application and properly initialize hardware registers from software. The purpose of the TASKING Pin Mapper is to assist you in performing those tasks.

1.1. Port Pin

The basic entity of the TASKING Pin Mapper is the port pin. Port pins are physical pins of the chip. A port pin name typically consists of a port identifier and, if the port identifier is shared by multiple pins, a sequential pin number, for example TMS, AN4 or P10_2. A port pin name is not package specific. Each package has its own specific assignment of port pin names to package pin numbers.

The basic configuration item of a port pin is its function (which in most cases is determined by the signal assigned to it). Some ports have dedicated functions, mostly related to power supply or chip infrastructure. Other ports have configurable functionality (ultimately defined by run-time register settings). The required function is application specific.

A port function definition consists of a symbolic name, a direction and a description. The TASKING Pin Mapper distinguishes between input functions and output functions. Functions are listed by their symbolic names (which are mostly signal names, for example ASCLK3).

Port pins can have a number of other chip specific properties. Such properties can be fixed (for example the type of pad the port is connected to) or configurable (for example the pad state). Port pins are visible in the Package view. You can assign a user-defined symbol name to the port pin and you can make the symbol names visible in the Package view. You can use filters in pin selection views or you can use property values as background color codes in the Package view. You can change the configurable property values.

1.2. Peripheral

AURIX devices, such as the TC27xC, have support for several on-chip peripherals, such as serial controllers, timer units, and analog-to-digital converters.

1.3. Connection

The TASKING Pin Mapper allows you to define connections between modules, where a module is either a port pin or a peripheral.

You can connect port pin modules to peripheral modules by selecting the pin's Direction and its Chip input function or its Chip output function. The peripheral module side of a connection is referred to as a virtual pin. Some peripheral modules can also be connected to other peripheral modules. Such a connection

is made between two virtual pins. Virtual pins are internal to the chip and are not visible in the Package view. However, virtual pin conflicts are reported in the Pin Conflicts view, for example when you try to connect a GTM virtual output pin to a physical port pin and at the same time to a virtual input pin of an ADC peripheral module.

If you have defined a connection (either valid or invalid) the "follow the connection" button ()) becomes available and you can navigate from one side of the connection to the other side. This applies to virtual pins as well as physical pins. Virtual pins usually do not have a configurable direction. They are either Input or Output (and some virtual pins are bidirectional).

See Section 2.3, Configuring the Pins for a step-by-step example.

1.4. Package

When you create a new pin mapper document, as explained in Section 2.1, Adding a Pin Mapper Document to an Existing Project, you first select a processor (family), for example TC27XC, and a package, for example BGA292. This selection defines the package pin numbers and the linking of package pins to port pins. It is possible that a port pin is linked to a single package pin multiple times by means of port pin aliases, for example AN39 and P40_9.

For QFP (Quad Flat Package) type packages the pin number consists of the package identifier and a sequential pin number. For BGA (Ball Grid Array) type packages the pin number consists of the package identifier and a matrix cell number.

Chapter 2. Getting Started with the TASKING Pin Mapper

The TASKING Pin Mapper is the graphical interface to configure the pins and modules in a package.

2.1. Adding a Pin Mapper Document to an Existing Project

To configure the pins and modules of a package you need to create a project and add a Pin Mapper document.

1. If you have no existing C project, follow the steps to create a new C project, as explained in the *Getting Started with the TASKING VX-toolset for TriCore*.

In the following steps we assume you have an existing TriCore TC27xC C project named myproject.

2. From the File menu, select New » TASKING Pin Mapper Document.

The New TASKING Pin Mapper Document wizard appears.

- 3. Select the **Project** folder for the Pin Mapper document: type the name of your project (myproject) or click the **Browse** button to select a project.
- 4. In the File name field, enter a name for the Pin Mapper document, for example myproject.pincfg and click Next.

The Device and Package page appears.

New TASKING Pin Mapper Document	- • •
Device and Package	
Select device and package for the new pin mapper document	
Processor Settings	
Only show items for TC27xC ○ Show all	
▲ ■ Infineon ▲ ■ AURIX Family ▲ ■ TC27xC ▼ BGA292 ♀ QFP176	
Rack Next > Finish	Cancel

5. Select the processor, for example TC27xC, and package, for example BGA292, for which you want to make a pin configuration and click **Finish**.

A pin configuration file with extension .pincfg is added to the existing project. Because the TASKING C/C++ perspective is still active, Eclipse asks to open the TASKING Pin Mapper perspective.

6. Optionally, enable the option Remember my decision and click Yes.

The TASKING Pin Mapper perspective is opened.

2.2. TASKING Pin Mapper Perspective

The TASKING Pin Mapper perspective contains several views. Each of the views are discussed in the following sections.

C TASKING Pin Mapper - myproject/myproject.pin																						-		•	×
File Edit Navigate Search Project Debug																									
C ▼ 🗟 🕲 🗎 🐘 ▼ 🛷 ▼ 🛃 ▼		> • 🖻							Q	uick A	Acces	s			Ē	[VX] TAS	KING				ASKIN			_
a myproject.pincfg 🛛			- 0] 🗗	Pack	age	2	_											Q	٠	-	•	Ab -	• -	
TASKING Pin Mapper - TC27xC / BG	A292 🧏	🖻 🖻 🌽	<mark>} - 0</mark> , A		-	-	3 P10_7	_	-	_	-	_	9 P11_9	10 P11_2	11 P13_3		13 P14_8	14 P14_5	15 P14_1	16 P15_6	17 P15_4	18 P15_1	19 VDDP3		
Pin Selection Pin Configurat	tion				-		VEKT	-					-	-		-	î	-	1	î —	-	VDDP3		28 P15_0	
type filter text 🖉 🕀 🕀			d 🐿		-	2 902_3						12	10											-	c
> Ports				P	P02_4	P02_3		VSS	VILEX	P <u>11</u> -	P11-	P11_5	P11_6	P11_4	P14-	P14,9	P34.7	P15_8	P15_7	voger	V55_		P20- 12-	F29-	P
Peripherals				E	P02_3	8023		P02_9	VSS	-	P11_8	P11_7	P11_1	P11_0	P12_1	P12_0	P14_2	P15_5	vogeu	V55-	P20_9		P20-	P20_ 11	E
Other Pins				F	P02_3	02,7		P02_ 11	P02-											ESRO	P20_6		P20_7	P20_8	F.
				G	P00_0	P00_1		P01_4	P01_3			VDDVD	VSS-	VSS_	V55_	V55-	V00			ESR1	PORST		P20_1	P20_3	G
				н	P00_3	2 000,3		P01_6	P01_5		voovo		٧ <u>%</u> -	V <u>55</u> -	V35-	٧ <u>8</u> 5-		V00		P21_7	P21_6		P20_2	P20_0	н
				3	P00_4	P00_5		P00_6	P01_7		V55_	V55_		V55_	V55_		V55_ 23	V55_		тск	P21_1		P21_3	P21_5	3
				ĸ	P00_3	P00_9		P00_8	P98-		V.55-	V55-	V55-	V <u>55</u> -	V55- 22-	V55-	V55-	N.C.		THS	P21_0		P21_2	P21_4	к
				L	P00_	P99-]	AN43	A142		V <u>85</u> -	V55	V55	V55-	V55_ 21	V55-	V55_	V55- 29-		P22- 10-	P22- 11-		TRST	VSS	i.
				м	ANNE	A107		ANH1	ANHO		V55_	V55-		V55-	V55_		V55_	V55_		P22_8	P22_9		XTAL2	XTAL1	м
				N	ANH	ANHS]	P40_6	P40_8		VDD		VSS-	VSS-	V55. 19	V55-		V00		P22_6	922,7		(00_8	VOQP3	N
				•	P40_3	940_7		P40_4	A/84			VDD	V55-	V <u>55</u> -	V55_	V55_	VDD			P22_4	P22_5		P22_1	P22_0	•
				R	PHO_S	5 ANOS	1	ANG1	A103											P23_7	P23_6		P22_3	P22_2	R
				т	VARE	r VAGN	•	AN30	A7422	AN15	AN12	ANS	ANH	AND	vevrs	P34_2	P34_4	P33_ 14	P32_5	VSS	923_5		P23_3	P23_4	т
				U	ANDS	A1128	1	NC.	AN117	AN14	AND	AN7	AIB	ANI	P34_1	P34_3	P34_5	P33_	P32_6	932,7	VSS		P23_1	P23_2	U
Pin Conflicts			~	v	PHO_	PH0_2																	VEKT	P23_0	v
0 items				w	P40_3	PH0_0	AN19	AN18	AN16	AN13	AN11	ANS	A/Q	P33_0	P33_2	P33_4	P33_6	933_8	P33-	P33	VGATE	P32_4	VSS	VERT	w
Description	Module	Pin	Location	Y	NC.	A101	AN20	VSSM	VDDM	VAREF	VAGNE	ANIO	ANS	P33_1	P33_3	P33_5	P33_7	P33_9	P33_	P32-	P32_0	P32_2	P32_3	vss	Y
					1	2	3		5		7	8		10		12	13	14	15	16	17	18	19	20	
					C = 1							TC2	7xC -	BGA	292 (1	Гор V	fiew)								
						ection Error	status:	[Wa	rning			ок												
٠			Þ																						

2.2.1. TASKING Pin Mapper Editor View

The TASKING Pin Mapper Editor view is the main area where you can make changes to your configuration.

Port example

🔒 myproject.pincfg 🛛					- 8
TASKING Pin Mapper - TO	27xC / BGA292				9a 🖻 🖆 🋵 🕶 🔍 🤱
Pin Selection	Pin Configuration				
type filter text 🖉 🕀 🕀					a a
⊿ ✓ Ports	Module name:	P00_0			
Analog	Description:	Port 0 pin 0			
⊿ ✓ P00	Pad class:	MP			
✓ P00.0	Reset state:	PU1			
P00.1 P00.2	Power domain:	VEXT			
P00.2 P00.3	Symbolic name:				
P00.4	Comment:				
P00.5	Comment:			*	
P00.6					
P00.7	Direction:	Input	•		
P00.8	Pad level:	CMOS/Automotive	-		
P00.9 P00.10			·		
P00.10 P00.11	Pad strength:	Speed grade 4	•		
P00.12	Chip input				
> P01	Mode:	Pull-up	•		
P02					
⊳ P10	IN:	✓ Select	▼ CIF0.D9		
P11 P12			ETH0.MDI		
▷ P12 ▷ P13			GTM0_TIM2_CH0.IN		
⊳ P14			GTM0_TIM3_CH0.IN		
P15					
P20 P21			MSC0.INJ0		
▷ P21 ▷ P22			T12CCU0.SYNC		
⊳ P23			CCU61.TRAP		
⊳ P32					
P33	Chip output				
⊳ P34	Mode:	Push-pull	*		
⊳ P40	OUT:	None	•		
 Peripherals Other Pins 					
p other rifts					

Peripheral example

🔒 myproject.pincfg 🛛			- 8
TASKING Pin Mapper - TO	?7xC / BGA292		9a 🖻 🖹 🋵 - 🛍 🖄
Pin Selection	Pin Configuration		
type filter text 🖉 🗎 🕀 🖨			n 🍋
 ▷ Ports ▲ ✓ Peripherals ▷ ADC ▷ ASCLIN0 	Module name: CAN0_NOE Description: Controller / Input	DE0 Area Network Module 0 Node 0	
ASCLIN1 ASCLIN2 ASCLIN3	RXD: ✓ P02_1.IN TRIGIN: ✓ TRIGCANCE	▼).TRIG ▼	
 ∠ CAN CAN0_NODE0 CAN0_NODE1 CAN0_NODE2 CAN0_NODE3 CCU60 CCU61 CIF Cerberus DSADC E-Ray ETH GPT GTM HSM HSSL IZC IOM MSC0 MSC1 OCDS PMU PSI5 QSPI0 QSPI2 QSPI2 	Output TXD: ✓ Select	 ▼ P02_0.0UT P02_5.0UT P12_1.0UT P20_8.0UT P33_8.0UT P34_1.0UT IOMLAM5.MON IOMLAM5.REF 	

The following toolbar icons are available:

lcon	Action	Description
Sec.	Solve Pin Conflicts	Solve conflicts with pin mappings. See Section 2.2.3, <i>Pin Conflicts View</i> .
c	Generate Code	Generate the source code and add it to your project. See Section 2.5, <i>Generating and Using the Source Code</i> .
	Generate CSV File	Save the pin mapper configuration in a file with comma-separated values (CSV). See Section 2.6, <i>Generate CSV File</i> .
<u>/a</u>	Select Device/Package	Select another package for your device. For example, from BGA to QFP or vice versa.
<mark>a</mark>]	Import Pin Configuration	Import pin configurations from a file and merge it with the current configuration.
A	Launch Altium Designer	Launch Altium Designer with the pin configurations. See Section 2.7, Launch Altium Designer.

Select Device/Package

With the Select Device/Package drop-down menu (2) you change the package for your device. For example:

%	🖻 🖻 🛵 - 🚉 🗛
	✓ TC27xC/BGA292
	TC27xC/QFP176
	dî 🔁

After the package switch you still keep the current configuration settings. If the same pin mapping can be used on the new device/package, the pin mapper automatically applies the mapping. For mapping parts which cannot be applied, the pin mapper issues a warning message.

Import Pin Configuration

You can import saved pin configurations and merge it with the current configuration. This can be useful, for example, when you have a set of configurations for ports and peripherals that you want to use in several projects.

1. Click the Import Pin Configuration button (

The Import Pin Configurations from File dialog appears.

💟 Impor	t Pin Configurations from File	
File:		
?	OK Cancel Workspace	File System

2. Type the full path name of the configuration file (.pincfg), or use the **Workspace** button to select a configuration from one of your project directories, or use the **File System** button to select one from any directory.

The imported configuration will be merged with the current configuration.

If the current configuration already has settings for a pin or peripheral and the imported configuration has other settings, pin conflicts errors or warnings may occur.

Pin Selection

In the left pane the port pins, peripherals and other pins are listed. Port pins can be logically grouped in different ways. Each of these groupings defines a tree view. Groups can be logically grouped into higher

level groups as long as the overall structure can be represented as a mathematical tree. The leafs of the tree are references to port pins, peripherals or peripheral sub-modules.

The tree structure of the grouping allows you to expand and collapse the view.

In the type filter text edit field, you can add a port pin selection filter in order to reduce the number of visible port pins. The filter is case insensitive. For example, type an to only show pins that have "an" in their name. Wildcards are allowed.

Pin Configuration

When you click on a port or module in the left pane, the configuration appears in the right pane. The following information can be present:

- Module name The name of the selected module or port pin.
- Alias An alias for the module name.
- **Description** The description of the selected module or port pin.
- Pad class The assignment of a port pin to one of the pad classes.
- Reset state The state of the pin after reset. For example, PU (Pull-up) or PD (Pull-down).
- Power domain The power domain the pin uses.
- **Symbolic name** You can assign a user-defined symbol name to the port pin. You can make the symbol names visible in the Package view.
- Comment Any user comments you can add here.
- Pin function You can configure some pins as digital or analog.
- Direction You can specify if a pin must be configured as an input pin or an output pin.
- Pad level Here you can select CMOS/Automotive or TTL.
- Pad strength For an output pin you can specify the Speed grade.
- **Mode** The chip input or output mode. For chip input this can be Pull-up, Pull-down or Tri-state. For chip output this can be Push-pull or Open drain.
- Other properties allow you to make a pin connection. See Section 2.3, *Configuring the Pins* for more information.

When you make a selection a '*' can appear in front of other selectable pin names. When you select a pin name with a '*', this results in an error.

Lock/Unlock

When you want to be certain that the port settings cannot be changed anymore, you can click the in button to lock the current settings. You can always click the abutton to unlock the settings again.

Tags

You can assign a tag name to a connection. When you click the 🔄 (Edit tags) button, you can edit the

tag field to the right of the 🖾 (Follow connection) button. You can use it the way you want. For example, to tag connections of the same type.

Undo/Redo

You can undo (Ctrl+Z) or redo (Ctrl+Y) one or more actions.

Back/Forward

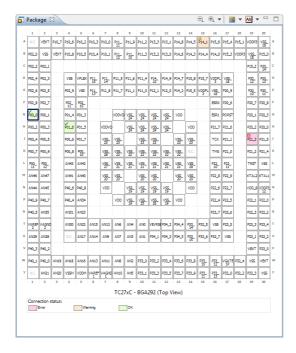
With the 🔶 (Back) and 🔿 (Forward) buttons you can navigate to a previously opened pin configuration.

Save

From the **File** menu, select **Save** (Ctrl+S) or click 🔚 to save the configuration.

2.2.2. Package View

The Package view shows a graphical representation of the package. For example, it shows the individual pins in a Ball Grid Array.



A square around a pin marks the selected pin. In the package above P00_0 is selected. A green check mark indicates that the pin has a valid connection; P00_0 and P01_6 in the package above. A red cross

indicates an error; P21_3 in the package above. A triangle with exclamation mark indicates a warning; P14_4 in the package above. When you click on a pin, the pin appears in the editor. If the pin contains an error, you can see what is wrong in the Pin Conflicts view.

Zoom in / Zoom out

With e and e you can zoom in or zoom out on the package view. Use the drop-down menu next to e to select a zoom factor directly.

Background Color

By default, the background colors of the pins indicate the connection status. But you can change the color properties of the view to show static configuration items (Pad class, Power domain or Reset state), or dynamic configuration items (Direction, Pad level, Pad strength, Mode or Pin function). Use the drop-down

menu next to I to switch color properties. Some examples:

	Packa	age 🖁	X												Q	€ -		•	Ab 🕤		1
	1	z	3	4	5	e	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
•	N.C.	VEKT	P10_7	P10_6	P10_2	P10_3	P10_0	P11-	P11_9	P11_2	P13_3	P13_1	P14_8	P14_5	P14_1	P15_6	P15_4	P15_1	VDDP3	V55_	A
	P02_0	VSS	VEXT	P10_8	P10_5	P10_4	P10_1	P11_ 12-	P11_ 10-	P11_3	P13_2	P13_0	P14_6	P14,3	P14_4	P14_0	P15_3	VDOP3	V <u>SS_</u> 18	P15_0	•
	P02_2	P02_1																	P15_2	P20_ 14	•
	P02_4	P02_3		VSS	VFLEX	P11-	P11_ 14	P11_5	P11_6	P11_4	P14_ 10-	وبه	P14_7	P15_8	P15_7	VODEL	V55 18		P20- 12-	P20-	1
	P02_6	P02_5		P02_9	V55	P <u>11</u> -	P11_8	P11_7	P11_1	P11_0	P12_1	P12_0	P14_2	P15_5	VOGFL	V <u>SS_</u> 18	P20_9		P20_ 10_	P20_ 11	E
	P02_8	P02_7		P02_ 11	P02_ 10											ESRO	P20_6		P20_7	P20_8	F
	P00_0	P00_1		P01_4	P01_3			VDDVD	V55-	V <u>55</u> _	V <u>55</u> -	V55-	VDD			ESR1	PORST		P20_1	P20_3	G
1	P00_2	P00_3		¥01_6	P01_5		voovo		V <u>SS</u> _	V <u>\$5</u> _	V <u>SS</u> _	V <u>55</u> _		VDD		P21_7	P21_6		P20_2	P20_0	•
	P00_4	P00_5		P00_6	P01_7		V <u>SS</u>	V <u>SS</u> _		V <u>55</u> _	V <u>55</u> _		V <u>55</u> _	V <u>55</u> _		тск	P21_1		P21_3	P21_5	,
ĸ	P00_7	P00_9		P00_8	P98-		VSS-	V55-	V <u>55</u> -	V <u>55</u> _	V <u>55</u> -	V <u>55</u> -	V <u>55</u> -	N.C.		THS	P21_0		P21_2	P21_4	,
L	P00_ 11	P00		AN43	ANH2		V <u>\$</u> 5-	V <u>SS_</u> 21	V <u>SS_</u>	V <u>\$5</u> _	V <u>SS_</u>	V <u>SS_</u>	V <u>SS_</u>	V <u>55</u> _		P22_ 10	P22_		TRST	VSS	ŀ
•	A1146	AN#7		AN41	ANHO		V55_	V <u>55</u> _		V <u>55</u> _	V <u>55</u> _		V55_	V <u>55</u> _		P22_8	P22_9		XTAL2	XTAL1	ŀ
N	AN44	AN45		P40_6	P40_8		VDD		V55-	V55_	V55-	V55-		VDD		P22_6	P22_7		VDD_8	VDDP3,	
P	P40_9	P40_7		P40_4	ANB4			VDD	V55_	V <u>55</u> _	V55_	V55_	VDD			P22_4	P22_5		P22_1	P22_0	ŀ
R	PH0_5	AN35		AN31	AN23											P23_7	P23_6		P22_3	P22_2	-
T	VAREF	VAGNE		A1(30	AN22	AN15	AN12	AN5	AN4	AND	VEVRS	P34_2	P34_4	P33_ 14	932_5	VSS	P23_5		P23_3	P23_4	Ī
u	AN29	AN28		NC.	AN17	AN14	AN9	AN7	ANB	AN1	P34_1	P34_3	P34_5	P33	P32_6	P32_7	VSS		P23_1	P23_2	u
v	PH0_3	P40_2				_			_						_		_		VEKT	P23_0	v
N	P40_1	P40_0	AN19	AN18	AN16	AN13	AN11	ANS	AN2	P33_0	P33_2	P33_4	P33_6	P33_8	P33-	P33_	VGATE	P32_4	VSS	VEXT	ŀ
Y	NC.	AN21	AN20	VSSM	VDDM	VAREF	VAGND	AN10	ANS	P33_1	P33_3	P33_5	P33_7	P33_9	P33_	P33_	P32_0	P32_2	P32_3	VSS	Y
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	ľ
								TC2	7xC -	BGA	292 (1	Гор V	iew)								
		ass: x2 .VDSM_I XDSM_I XDRST XTAL2	N		D LVI S	DSM_P			LP MP VGAT	'E1P			LVDSHL MP+ Vx	N		M	DSHLP RR TAL1				

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	VEXT	P10_7	P10_6	P10_2	P10_3	P10_0	P11_ 11	P11_9	P11_2	P13_3	P13_1	P14_8	P14_5	P14_1	P15_6	P15_4	P15_1	VDDP3	V55-
P02_0	VSS	VEXT	P10_8	P10_5	P10_4	P10_1	P11_ 12	P11_ 10-	P11_3	P13_2	P13_0	P14_6	P14_3	P14_4	P14_0	P15_3	VDOP3	V55_ 18	P15_0
P02_2	P02_1																	P15_2	P20_ 14
P02_4	P02_3		VSS	VFLEX	P11 15-	P11_ 14	P11_5	P11_6	P11_4	P14_ 30 ⁻	P14_9	P14,7	P15_8	P15_7	VODEL	VSS_ 18		P20_ 12	P20_ 13
P02_6	P02_5		P02_9	VSS	P11- 13-	P11_8	P11_7	P11_1	P11_0	P12_1	P12_0	P14_2	P15_5	VOOPL	V <u>SS</u> _ 18	P20_9		P20 10-	P20_ 11
P02_8	P02_7		P02_ 11	P98-											ESRO	P20_6		P20_7	P20_8
ř00_0	P00_1		P01_4	P01_3			VDOVD	V <u>SS</u> _ 24-	V <u>SS</u> _ 24-	V <u>SS</u> _ 24-	V <u>SS</u> _ 24-	VDD			ESR1	PORST		P20_1	P20_3
P00_2	P00_3		ř01_6	P01_5		VDDVC		V <u>55</u> _ 24	V55_ 24	V55_ 24	V <u>55</u> _ 24		VDD		P21_7	P21_6		P20_2	P20_0
P00_4	P00_5		P00_6	P01_7		٧ <u>%</u> -	V <u>55</u> -		V <u>35</u> -	V <u>85</u> -		V <u>8</u> 5-	V <u>8</u> 5-		тск	P21_1		P21_3	P21_5
P00_7	P00_9		P00_8	P00 30		V <u>SS</u> _ 28	V <u>95</u> _	V <u>55</u> _	V <u>55</u> _	V <u>SS</u> _ 22	V <u>55</u> _	V <u>SS</u> _ 22	N.C.		TMS	P21_0		P21_2	P21_4
P00_ 11	P00 12-		ANH3	AN42		V <u>55</u> -	V <u>55</u> _ 21	V55_ 21	V <u>55</u> _ 21	V55_ 21	V <u>55</u> _ 21	V <u>55</u> _ 21	V55-		P22- 10-	P22- 11-		TRST	VSS
AN46	AN47		AN41	AN40		V <u>55</u> -	V <u>55</u> -		V <u>8</u> 8-	V <u>55</u> -		V <u>55</u> -	V <u>55</u> -		P22_8	و_229		XTAL2	XTAL:
AN 41	AN45		P40_6	P40_8		VDD		V <u>SS</u> _ 19	V <u>SS</u> _ 19	V <u>SS</u> _ 19	V <u>SS</u> _ 19		VDD		P22_6	P22_7		VDD_8	VDDP: 12
P40_9	P40_7		P40_4	AN34			VDD	V <u>55</u> - 19	₩28-	₩.	V55- 19	VDD			P22_4	922,5		P22_1	P22_0
P40_5	AN35		AN31	AN23											P23_7	P23_6		P22_3	P22_2
VAREF 2	VAGNE		AN30	AN22	AN15	AN12	ANS	ANH	AND	VEVRS	P34_2	P34_4	Р <u>33</u> _ 14	P32_5	VSS	P23_5		P23_3	P23_4
AN29	AN28		N.C.	AN17	AN14	AN9	AN7	ANB	ANI	P34_1	P34_3	P34_5	먬.	P32_6	P32_7	VSS		P23_1	P23_2
P40_3	P40_2																	VEXT	P23_0
P40_1	P40_0	AN19	AN18	AN16	AN13	AN11	ANS	AN2	P33_0	P33_2	P33_4	P33_6	P33_8	P33 10-	P33- 12-	VGATE	P32_4	VSS	VEXT
N.C.	AN21	AN20	VSSM		VAREF	1		AN5	P33_1	P33_3	P33_5	P33_7	P33_9	P33_ 11	P33- 13-	P32_0	P32_2	P32_3	VSS
1	2	3	4	5	6	7	s TC2	。 7xC -	BGA	292 (12 Fop V	iew)	14	15	16	17	18	19	20
	r doma /DDM	in:	[VD	093			VEXT				VFLEX							

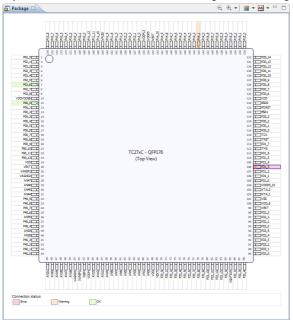
1	Packa	ige 🎖	× /												0	€ -		•	Ab 🔻	· -
	1	2	3	4	5	e	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	N.C.	VEXT	P10_7	P10_6	P10_2	P10_3	P10_0	P11_ 11	P11_9	P11_2	P13_3	P13_1	P14_8	P14_5	P14_1	P15_6	P15_4	P15_1	VDDP3	V55- 18-
	P02_0	VSS	VEXT	P10_8	P10_5	P10_4	P10_1	P <u>11</u>	P11- 10-	P11_3	P13_2	P13_0	P14_6	P14_3	P14_4	P14_0	P15_3	VOOP3	V <u>SS</u> _	P15_0
	P02_2	P02_1																	P15_2	P20_
	P02_4	P02_3		VSS	VFLEX	P11-	P11_ 14	P11_5	P11_6	P11_4	P14_ 10	P14_9	P14,7	P15_8	P15_7	VODEL	V55_		P20_	P20_
	P02_6	P02_5		P02_9	VSS	P11-	P11_8	P11_7	P11_1	P11_0	P12_1	P12_0	P14_2	P15_5	voçer	VSS_	P20_9		P20_	P20_
	P02_8	P02_7		P02_	P92-										-	ESRO	P20_6		P20_7	P20_8
	¥00_0	P00_1		P01_4	P01_3			VDOVC	VSS-	VSS_	VSS_	V55_	VDD			ESR1	PORST		P20_1	P20_3
	P00_2	P00_3		¥01_6	P01_5		VDDVD		VSS_	V55_	V55_	V55_		VDD		P21_7	P21_6		P20_2	P20_0
	P00_4	P00_5		P00_6	P01_7		V%-	VSS-		V%-	v祭-	27	V%-	٧ <u>%</u> -		тск	P21_1		P21_3	P21_5
	P00_7	P00_9		P00_8	P00_		23 V <u>55</u> -	V <u>55</u> _	V <u>\$</u> 5_	V55_	V55_	V <u>55</u> _	23 V <u>55</u> -	NC.		TMS	P21_0		P21_2	P21_4
	P00_	P99		A1143	AN42		20 V\$\$-	V55_	V <u>55</u> _	V55_	V55_	V55_	V55_	vss-		P22_	P22_		TRST	VSS
	AN46	AN47		AN41	AN40		2/ V\$\$-	V55_		V\$5-	V55-		vss-	VSS-		P22_8			XTAL2	XTAL1
	ANH	ANHS		P40_6	P40_8		VDD	20	V <u>55</u> _	V55_	V55_	V55_	20	VDD		P22_6	P22_7		VDD_8	V00P3
	P40_9	P40_7		P40_4	AN34			VDD	V55-	V%-	v <u>%</u> -	V55_	VDD	ľ		P22_4	P22_5			P22_0
	P40_5	AN35		ANB1	AN23				19	- 20		19				P23_7	P23_6		P22_3	P22_2
	VAREF	VAGNE		AN30	AN22	AN15	AN12	ANG	ANH	AND	VEVRS	P34_2	P34_4	P33_	P32_5	VSS	P23_5		P23_3	P23_4
	41(29	AN28		N.C.	AN17	AN14	ANG	AN7	ANB	AN1	P34_1	P34_3	P34_5	14 192-	P32_6	P32_7	VSS		P23_1	P23_2
	P40_3	P40_2					_										_		VEXT	P23_0
	P40_1	P40_0	AN19	AN18	AN16	AN13	AN11	ANS	AN2	P33_0	P33_2	P33_4	P33_6	P33_8	P33_	P33	VGATE	P32_4	VSS	VEXT
	N.C.	AN21	AN20	VSSM	VDDM	VAREF	VAGNO	AN10	ANS		933_3	P33_5		P33_9	10 P33_	12 P33_	1P P32_0	P32_2	P32_3	VSS
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		17	18	19	20
								TC2	7xC -	BGA	292 (Гор V	iew)							
	Reset	state: U		[EV PU	R13 SMF 1	is → Pi), GF	HighZ				OD			PC)			

Labels

With the drop-down menu next to 🖾 you can switch between module names and symbol names.

Quad Flat Package

If you have selected a Quad Flat Package, the Package view may look similar to:



2.2.3. Pin Conflicts View

Potentially conflicting pin assignments are reported in the Pin Conflicts view. For example, if multiple virtual pins are assigned to the same port pin then there is a potential conflict. Also, if the same virtual pin is assigned to multiple port pins then there is a potential conflict.

Pin Conflicts 🕴				
1 error, 1 warning, 0 others				
Description	Module	Pin	Location	Resource
O Dangling connection	P00_11	IN	N.A.	myproject.pi
Multiple selections for a 1-to-N connection	P14_1	IN	N.A.	myproject.pi

When you double-click on a conflict the corresponding pin configuration will be visible in the Editor view.

See Section 2.4, Solving Pin Conflicts for information how to solve pin conflicts.

See Section 2.9, Errors and Warnings for a list of errors and warnings.

2.3. Configuring the Pins

Basically you can connect pins in two different ways. The first way is to select a port pin and then assign a peripheral virtual pin (function) to it. The second way is to select a peripheral virtual pin (function) and then assign a port pin to it.

You can choose between input or output mode. If an application is going to use a port pin in both directions then you should select the initial run-time direction of the port. This is necessary because the pin mapper will generate port pin initialization code and needs to know whether to configure for input or for output.

To connect pins from the port pin side

- 1. In the Project Explorer view double-click on a configuration file (myproject.pincfg) to view its contents.
- 2. In the Pin selection pane, select a port pin. For example, Ports » P00 » P00_0.
- 3. In the Pin configuration pane select the Direction: Input or Output. In this example we select Input.
- 4. Select the pin **Mode**.
- 5. Connect the input pin to a peripheral virtual pin:
 - From the IN drop-down box choose GPIO or Select ...

With Select... a virtual pin appears, or with a 1-to-N connection (one source, many destinations) a list of virtual pins appears.

• Select one (or more) pins.

A green check mark ~ appears if the connection is successful.

6. (Optional) When you want to prevent that these settings can be changed, click if to lock the settings.

A little lock appears next to the port pin.

7. From the **File** menu, select **Save** (Ctrl+S) or click 🔚 to save the configuration.

When you click vous a light blue connection from source to destination or vice versa. The corresponding selection is shown in a light blue color.

To connect pins from the peripheral side

- 1. In the Project Explorer view double-click on a configuration file (myproject.pincfg) to view its contents.
- 2. In the Pin selection pane, select a module. For example, Peripherals » ETH » ETH_0.
- 3. In the Pin configuration pane select the pin you want to assign to the virtual pin.

A green check mark v appears if the connection is successful.

- 4. Click to follow the connection to the port pin side. The corresponding selection is shown in a light blue color.
- 5. (Optional) When you want to prevent that these settings can be changed, click if to lock the settings.

A little lock appears next to the port pin.

6. From the **File** menu, select **Save** (Ctrl+S) or click 🔚 to save the configuration.

2.4. Solving Pin Conflicts

It is your responsibility to solve any pin conflicts. It is important that there are no conflicts when you use the tools to generate the code. You can solve the conflicts by hand by making other connections, but when there are many conflicts or in situations where most port pins are in use, it can be quite cumbersome and complex to solve conflicts. The Pin Conflicts Solver can automate this process and solve most conflicts for you.

To solve pin conflicts

1. From the **Pin Mapper** menu, select **Solve Pin Conflicts** (¹/₂₀).

The Solve Pin Conflicts dialog appears with a list of actions that will be taken when you click OK.

	u want to continue?		olve the dangling conn		
Action	Module	Pin	Old port Pin	New port Pin	
move		GPIO	P02.2	P02.0	
move	ASCHS1	TXD	P15.1	P02.2	
connect	ERUIN7	IN	P20.9	P15.1	
move	T12CCU1	CC62	P11.10	P33.9	
connect	ASCLIN1	RXD	P15.1	P11.10	
connect	ASCLIN1	TXD	P15.1	P15.5	
connect	CAN0_NODE2	RXD	P15.1	P02.3	
move	G0_QREQ	BFL	P33.5	P33.6	
reconnect	T12CCU1	CC60	P33.5	P33.5	
move	SCU	EXTCLK1	P33.10	P11.12	
reconnect	GTM0_CMU	CLK1	P33.10	P33.10	
move	GTM0_TIM0_C	IN	P02.4	P21.6/TDI	
connect	QSPI3_S	SELIN	P33.10	P02.4	
connect	QSPI0_M	CLK	P20.13	P20.11	
move	QSPI0_M	RXD(N)	P20.12	P14.6	
	TIACCUI	COUTCI	000.10	000.10	

2. Click **OK** to accept the actions.

The configuration is updated accordingly.

3. Check the pin configuration settings of the new connections and make changes if needed.

The meaning of the columns in the Solve Pin Conflicts dialog is:

Column	Description	
Action	The action that takes place to solve the pin conflict.move - move the connection of the module pin from the Old port Pin to the New port Pin including all configuration settings. The Old port Pin is reset to the default settings.connect - connect the module pin from the Old port Pin to the New port Pin using the default configuration settings. You still need to check and configure the pin settings.reconnect - after a move of the original connection to another pin, reconnect to the same port pin. You still need to check and configure the pin settings.reset - reset the port pin to the default settings.	
Module	The name of the selected module.	
Pin	The name of the (virtual) pin of the module.	
Old port Pin	The name of the port pin the module pin was connected to.	
New port Pin	The name of the port pin the module pin will be connected to.	

Note that the solver only tries to solve dangling connections between virtual module pins and port pins, not between virtual pins (internal connections). Also, the solver does not change peripheral instances. If, for example, a connection error with peripheral ASCLIN0 exists, the solver will not try to use ASCLIN1 instead. Try to use a different instance of the same peripheral type manually. If the solver does not find a solution for all pin conflicts, then no port pins will be swapped at all. You can try to move some connections manually or unlock some pin settings (if applicable) or, as a final resort, try to switch to another device/package.

Depending on the projected calculation time, the solver either performs an exhaustive search or applies a heuristic partial search algorithm to find a solution. In case of a partial search it is possible that a solution does exist while the solver does not find it. The solver error dialog distinguishes these situations.

Locked connections are not touched. If you have locked a port pin, its configuration stays the same. The standard undo/redo functionality is available. The menu entry and button are disabled (grayed) if there are no Pin Conflicts to resolve.

2.5. Generating and Using the Source Code

When you have configured the pins you are ready to add the Pin Mapper sources to your project.

Generate source code

From the Pin Mapper menu, select Generate Code or click 🔮.

The Pin Mapper sources are generated and are added to your project in the folder PinMapper.

- myproject [Active Debug]
 - Includes
 - 🔺 📂 PinMapper
 - b aurix_pin_mappings.c
 - b in aurix_pin_mappings.h
 - cstart_tc1.c
 - b h cstart_tc1.h
 - b cstart_tc2.c
 - b h cstart_tc2.h
 - b c cstart.c
 - b h cstart.h
 - b ic myproject.c
 - b sync_on_halt.c
 - DConfig
 - MConfig
 - myproject.launch
 - Myproject.lsl
 - myproject.pincfg

Using the Pin Mapper sources

The Pin Mapper sources can generate pin mappings for peripherals, internal MUX initialization code and an initialization function for GPIO pins, depending on the configured module connections. For GPIO the following line is present in aurix_pin_mappings.c.

#include <Port/Io/IfxPort_Io.h>

For internal MUX initialization the Pin Mapper generates:

```
#include <_Lib/InternalMux/Ifx_InternalMux.h>
```

The generated code is meant to work closely together with the Infineon iLLD library. This library must be available to your project.

You can easily add the Infineon iLLD library to your project by adding a TASKING Software Platform document and generate the code.

The Software Platform sources are generated in the SoftwarePlatform folder of your project. To use the generated files, you must add a #include statement into your top-level source file (typically main.c):

```
#include "swplatform.h"
```

You can access the Pin Mapper sources from your application. To use the generated files, you must add a #include statement into your top-level source file (typically main.c):

```
#include "PinMapper/aurix_pin_mappings.h"
```

You can now use the functions of the Pin Mapper in your sources. For initialization of the GPIO pins or internal MUX settings, use the initialization functions from the Pin Mapper sources. For initialization of the peripherals and other pins use the initialization functions from the iLLD library. For example,

```
#include "PinMapper/aurix_pin_mappings.h"
#include "swplatform.h"

IfxEth *drv;
int main( void )
{
    /* initialize GPIO */
    gpio_init_pins();
    /* initialize internal MUX settings */
    internal_mux_init();
    /* initialize ETH0 */
    drv = IfxEth_Phy_Pef7071_ActInit(IFXETH_PHY_PEF7071_1);
    ...
    return 0;
}
```

2.6. Generate CSV File

You can save the pin mapper configuration in a file with comma-separated values (CSV).

• From the **Pin Mapper** menu, select **Generate CSV File** (1).

The CSV file (pincfg_*.csv) is generated in the folder PinMapper of your project.

The following is an example of such a file:

"Name","Pin","Function","Direction","Mode","Pad level","Pad class","Power domain","Reset state" "P00_0","G1","GTM0_TIM2_CH0_IN","Input","Pull-up","CMOS/Automotive","MP","VEXT","PU1" "P10_5","B5","SCU_HWCFG4","Input","Pull-up","CMOS/Automotive","LP","VEXT","PU1"

You can open the file with any text editor or in a spreadsheet program.

2.7. Launch Altium Designer

When you have Altium Designer installed with the TASKING Pin Mapper software extension, you can start Altium Designer from the TASKING Pin Mapper Editor view.

From the Pin Mapper menu, select Launch Altium Designer (A).

Altium Designer launches and a wizard guides you to import the pin configurations. This results in a schematic document with the selected device, peripherals and pins.

2.8. Pin Mapper Preferences

You can use the Preferences dialog in Eclipse to specify how the Pin Mapper should operate.

To set preferences

1. From the Window menu, select Preferences.

The Preferences dialog appears.

2. Select TASKING » Pin Mapper.

The Pin Mapper page appears.

3. Set your preferences and click OK.

You can set the following preferences:

Generate code on save

By default the TASKING Pin Mapper asks if you want to generate code when you save a document (**Prompt**). You can choose to do this automatically (**Always**) or **Never**.

2.9. Errors and Warnings

The TASKING pin mapper reports errors and warnings in the Pin Conflicts view. When you encounter an error, you can double-click on the error message to go to the conflicting situation.

Error: Dangling connection

A connection consists of two ends; a source and a destination. If the source is connected to a destination while the destination is not connected to the source then the connection is reported to be dangling. There are two situations in which this error can appear.

• Situation 1. When dealing with a 1-to-N connection (one source, many destinations), on the right-hand side of the drop-down box a list of destinations is shown. Each destination has a check box and you can make one or more selections. Initially nothing is selected and at that point the error appears.

For example, select **Ports** » **P00** » **P00_0** and change the **Chip input** configuration **IN** from **None** to **Select**. Several destinations appear, all unselected. The error appears to notify you to make a selection. Once you made a selection, the error disappears.

- Situation 2. When you attempt to connect a source to a destination that already has a connection. In this case the source is connected to the destination, while the destination keeps its old connection. For example,
 - 1. Select Ports » P00 » P00_0 and change the Direction configuration from Input to Output
 - 2. Change the configuration **OUT** from **None** to **GPIO**.
 - 3. Select Peripherals » GTM » TOUT9 and change the output OUT from None to P00_0.OUT.

The error appears since TOUT9.OUT is now connected to P00_0.OUT, while P00_0.OUT is still connected to GPIO.

Error: Incorrect settings for *name*, expected pins *name1* and *name2* to be connected

This error appears when you connect one pin, where multiple pins need to be connected.

For example,

- 1. Select Ports » P21 » P21_3 and change the Direction configuration from Input to Output.
- 2. Change the configuration **OUT** from **None** to **ETH0.MDO3**.

The error will disappear when you follow the connection and also connect input MDI from ETH0 to P21_3.IN.

Error: Modules *name1* and *name2* cannot be used at the same time, since they are mutually exclusive

This error appears when you try two use two modules of which only one module can be used at a time.

For example,

1. Select Peripherals » ASCLINO » SPI3WO and change the input RXD(N) from None to P14_1.IN.

2. Select Peripherals » ASCLINO » SPI4WO and change the output TXD from None to P14_1.OUT.

Modules SPI4W0 and SPI3W0 cannot be used at the same time.

Warning: Multiple selections for a 1-to-N connection

This warning appears when you make more than one selection for a 1-to-N connection.

For example, select **Ports** » **P00** » **P00_0** and change the **Chip input** configuration **IN** from **None** to **Select**. Several destinations appear, all unselected. Now select more than one destination. The warning disappears if you only select one destination.