# **Microcontroller Aware Hardware Design**

June 2015, Freescale Technology Forum

Ben Jordan, Product Manager, Altium.

00

# First, a story...



# At 259,000 Km

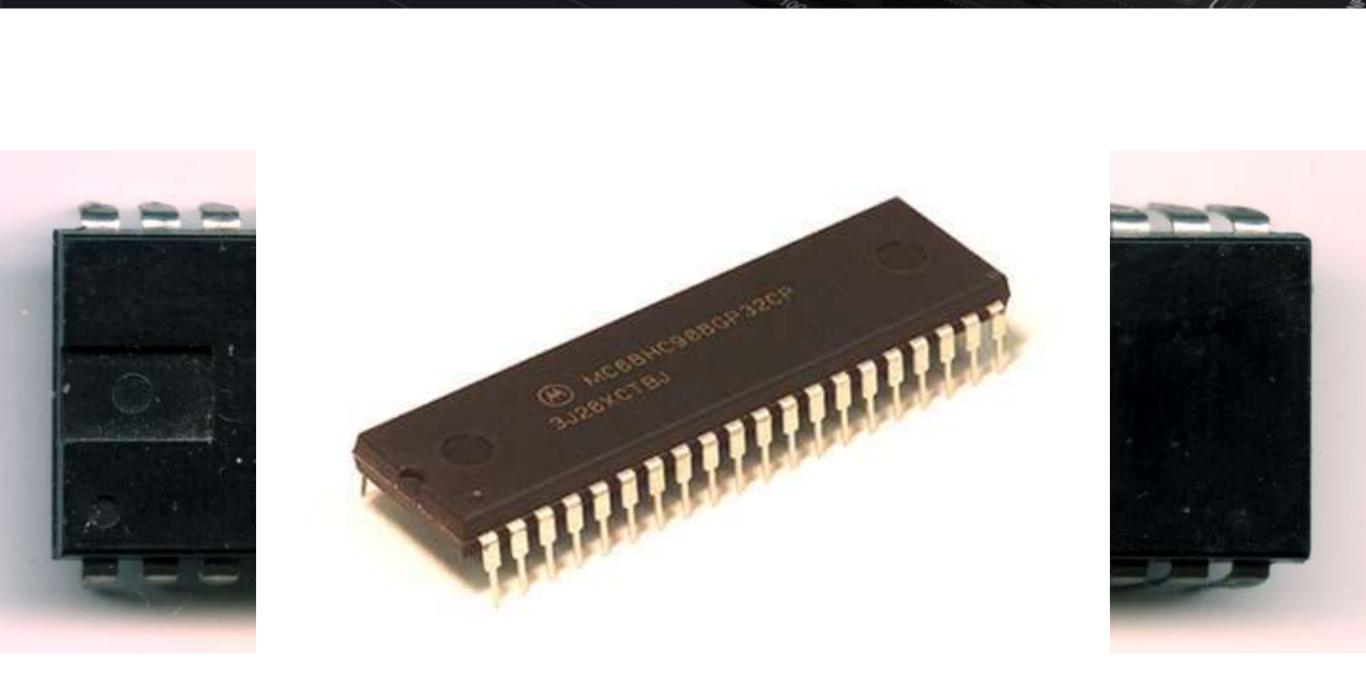


# What to do?







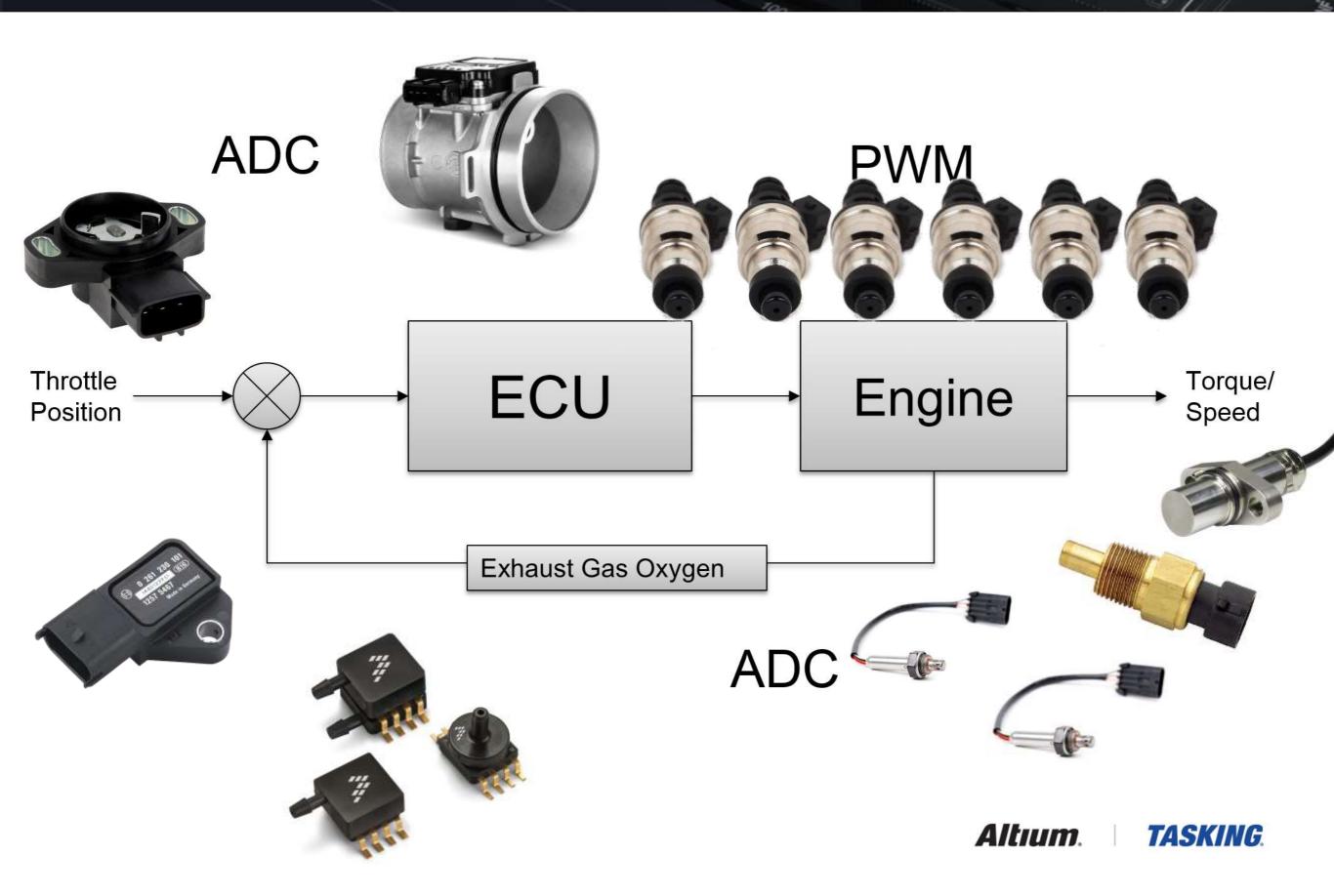


10

100 CO



#### With Great Power Comes Great Complexity



#### With Great Power Comes Great Complexity







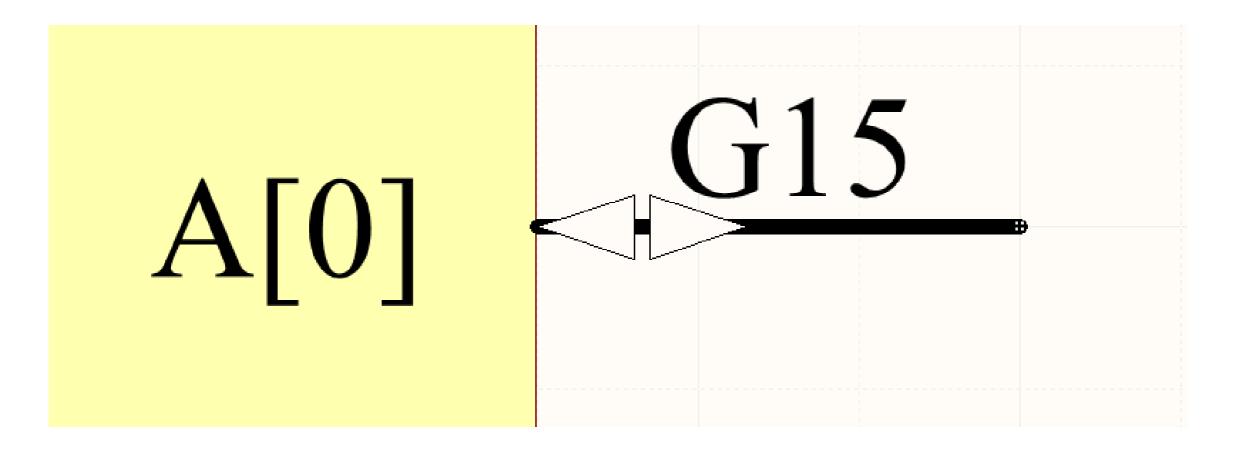




Typical systems now consist of many inputs and control signals:

- Several ADC Channels for Throttle, MAP, Airflow, Oxygen, Temperature, Altitude Pressure, Pedal, Fuel Pressure, etc.
- Several Schmitt / Digital inputs for Speed, Gear, Vane-type Airflow, etc.
- Multiple PWM Channels and Timers for Injection control (esp. multi-point), Fuel Pump, Transmission and more...
- Inputs from other vehicle systems such as ABS, Gear Selector, Security, Air-Bag, Collision Detect/Avoidance.

# The Humble Microcontroller and its GPIO pin...



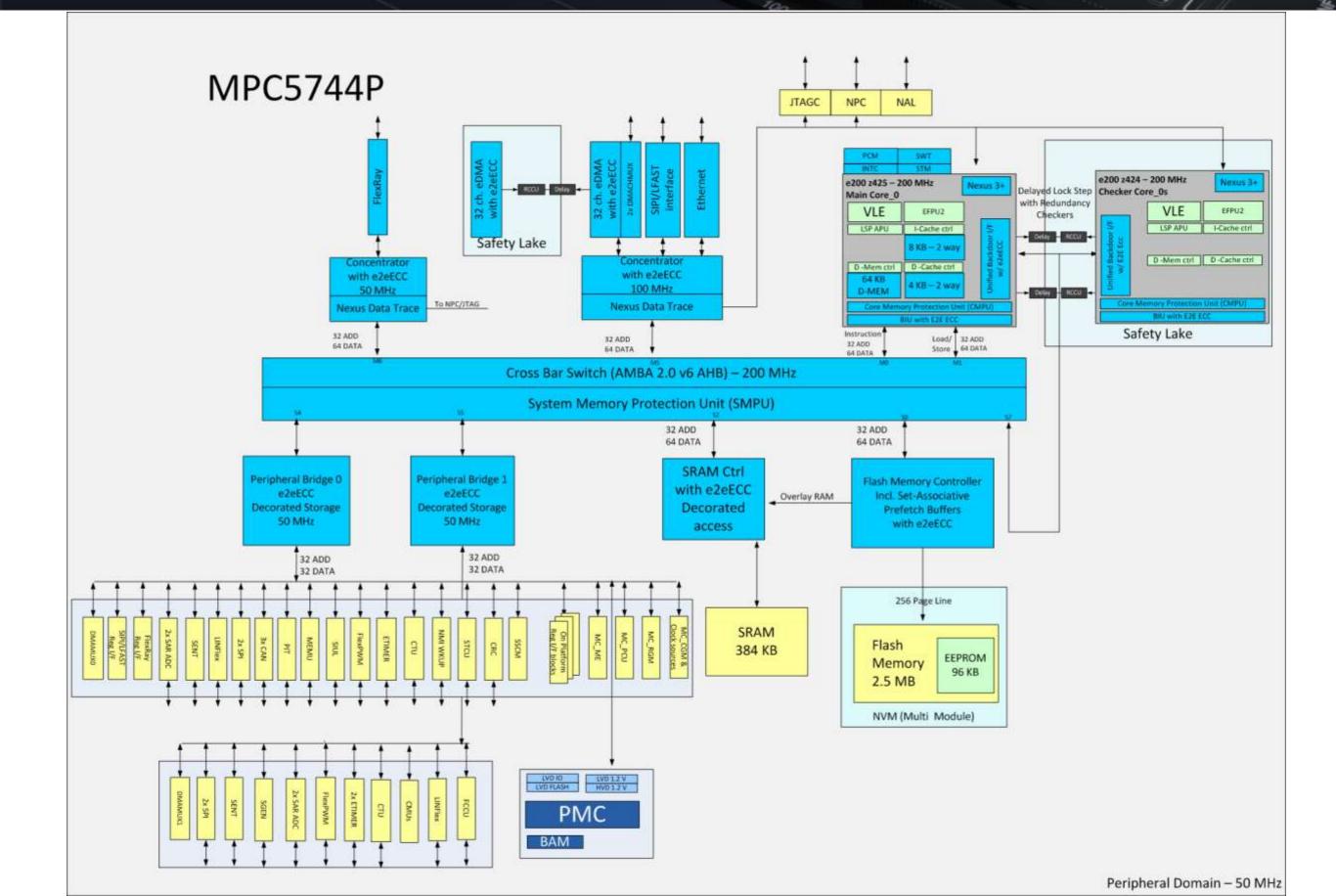
## ... is not so humble any more!

Table 8. Pin muxing (continued)

Port Pin	SIUL2 MSCR/ IMCR Number	MSCR/ IMCR SSS Value <sup>1</sup>	Signal	Module	Short Signal Description	Dir	LQFP144	BGA257
A[3]	MSCR[3]	0000 (Default)	GPIO[3]	SIUL2-GPIO[3]	General Purpose IO A[3]	I/O	92	G15
		0001	ETC3	eTimer_0	eTimer_0 Input/Output Data Channel 3	I/O		
		0010	CS0	DSPI2	DSPI 2 Peripheral Chip Select 0	I/O		
		0011	B3	FlexPWM_0	FlexPWM_0 Channel B Input/ Output 3	I/O		
		0100-1111	-	Reserved	_			
	IMCR[171]	0000 (Default)	ABS2	MC_RGM	RGM external boot mode 2	I	с. И	
	IMCR[62]	0010	ETC3	eTimer_0	eTimer_0 Input Data Channel 3	I/O		
	IMCR[49]	0001	CS0	DSPI2	DSPI 2 Peripheral Chip Select 0	I/O		
	IMCR[98]	0001	B3	FlexPWM_0	FlexPWM_0 Channel B Input/ Output 3	I/O		
	IMCR[176]	0001	REQ3	SIUL2	SIUL2 External Interrupt Source 3	I		



## ...With Great Power Comes Great Complexity



# How can we manage the complexity?

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
A	VSS_HV_ IO	VSS_HV_ IO	A[14]	A[9]	D[3]	JCOMP	H[12]	C[15]	VDD_HV	<b>I</b> [3]	E[13]	J[1]	F[15]	H[13]	F[13]	VSS_HV_ IO	VSS_HV _IO
в	VSS_HV_ IO/VSS_LV_ COR	VDD HV	F[3]	D[2]	B[6]	F[0]	D[4]	D[0]	VSS_HV _IO	E[14]	A[10]	B[3]	H[9]	C[10]	J[3]	VDD HV	VSS_HV _IO
с	I[15]	[0]L	VSS_HV _IO	FCCU_ F[1]	A[13]	i[0]	H[10]	E[15]	H[11]	I[14]	J[2]	B[2]	H[6]	B[1]	VSS_HV_	B[0]	H[15]
D	A[6]	[7]	A[15]	C[6]	N/C	EXT_POR _B	A[12]		VSS_HV _IO	A[11]	I[2]	F[14]	J[4]	VDD HV	VPP_TE ST	A[4]	F[12]
Е	F[4]	F[6]	D[1]	NMI_B	\$1).	- 24	121	13	10	8	- 19	- 22	12	N/C	C[13]	G[3]	D[14]
F	F[5]	H[7]	H[5]	H[4]	6	VDD_LV_ COR	VDD_LV _COR	VDD_LV _COR	VDD_LV _COR	VDD_LV _COR	VDD_LV _COR	VDD_LV _COR	14	C[14]	D[12]	G[4]	G[2]
G	MDO0	VDD_HV	C[5]	A[7]	23	VDD_LV_ COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VDD_LV _COR	12	B[4]	A[3]	<b>J</b> [8]	G[6]
н	A[8]	VSS_HV_ IO	C[4]	A[5]	-	VDD_LV_ COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VDD_LV _COR	12	G[12]	TMS	VDD_HV _FLA	тск
J	C[7]	<b>I</b> [4]	F[8]	F[7]	25	VDD_LV_ COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VDD_LV _COR	12	G[13]	H[1]	VDD_LV_ NEXUS	B[5]
к	<b>J</b> [9]	F[10]	F[9]	I[8]	-	VDD_LV_ COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VDD_LV _COR	14	G[15]	H[0]	VSS_LV_ NEXUS	J[10]
L	H[8]	F[11]	I[9]	D[8]	13	VDD_LV_ COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VSS_LV _COR	VDD_LV _COR	12	A[2]	G[14]	N/C	J[11]
м	VDD_HV _OSC	VDD_HV	I[10]	D[5]		VDD_LV_ COR	VDD_LV _COR	VDD_LV _COR	VDD_LV _COR	VDD_LV _COR	VDD_LV _COR	VDD_LV _COR	11	C[12]	(6)	G[7]	G[5]
N	XTAL	VSS_HV_ IO	D[9]	VSS_LV_ PLL	23	12	141	84	10	15	- 49	12	12	G[8]	((5)	VDD_LV_ LFAST	VSS_LV _LFAST
P	VSS_HV_ OSC	RESET_ B	D[6]	VDD_LV_ PLL	[[12]	I[13]	B[8]	J[5]	J[6]	J[7]	B[14]	A[0]	H[14]	G[9]	N/C	C[11]	D[11]
R	EXTAL	FCCU_ F[0]	VSS_HV _IO	D[7]	<b>B</b> [7]	E[6]	VDD_HV _ADRE0	B[10]	VDD_HV _ADRE1	B[13]	B[15]	C[0]	BCTRL	N/C	VSS_HV_ IO	D[10]	G[10]
т	VSS_HV_ IO	and the second se	I(1)	C[1]	E[5]	E[7]	VSS_HV _ADRE0	B[11]	VSS_HV _ADRE1	VDD_HV _ADV	E(10)	E[12]	E[0]	A[1]	G[11]	VDD_HV	VSS_HV _IO
U	VSS_HV_	VSS_HV_ IO	[[11]	E[4]	C[2]	E[2]	B[9]	B[12]	VSS_HV _ADV	E[9]	E(11)	N/C	N/C	VDD_HV PMU/IO		VSS_HV_ IO	VSS_HV _IO

#### **Configurable Devices**

# The microcontroller IDE allows us to visually map these IO functions and peripherals to the device pins.

# **IDE Pin Configuration**

TASKING Pin Mapper - myproje File Edit Navigate Search P	An experience of the party of the second states of the second	and the second second second	and a state of the state of the state																		0	•
□ • □ □ □ □ □ □ □ * •	Contraction of the second se	Contraction of the Contraction o	and the second second						Q	uick A	ccess			. 8		TASK	ING C	/C++	5	TASKI	NG Pir	n Mappe
a myproject.pincfg 23				- 0	6	Packa	ige 🛛		_								(	⊇, ⊕	-		Ab	[
TASKING Pin Mapper -	MPC574xP / M	APBGA25	7	2 / 2 - 6		1	2	1		ŧ	6	,	8	Q	10	11	12	13	14	15	ы	17
Pin Selection	Pin Configuration				A	VSS_HV	VSS W	A(14)	A[9]	미개	JCOMP	HIS	Q15]	WE DON	(3	E[13]	41	F[15]	H[13]			VSS_HV
type filter text 🖉 🗵 🖽		đ	Use tag:	type new tag 🔹	8	VALW, EVELUS	NUCC IN	F[3]	Dj2j	BKI	FIO	D[4]	D(0)	V\$5_H/ _10	E[14]	A(10)	B[3]	HE	Cital	4[3]	100 HV	V55_W/ _30
> Ports			1.000 B	1.499	c	(H)	tül	VSS_H/	FCCU_ F[1]	<b>A(1</b> 3)	(K)	H[10]	E[15]	H[11]	<b>84</b>	42)	8(2)	HR	B[1]	VS5_HV_ IO	B[0]	H(15]
Peripherals					D	A[6]	[7]	A[15]	CMI	NC	EXT_POR	A[12]	NEW HY	VSS_HI	A[11]	121	F[14]	44	WCODW OC	VPP_TE ST	44	F[12]
D Other Pins					E	F(4)	明	D[1]	NV.B									-	NC	c[r3]	G[3]	D(14)
					F	F[5]	H[7]	H[5]	H(4)		VDD_LV_ COR			V0D_LV _COR			VDD_LV COR		CIM	D[12]	G(4)	0(2)
					G	MDO0	100 14	여왕	A[7]		VDD_LV_	VSS_LV	155_LV	V55_LV COR	V55_U		VOD_UV		明	시키	.00	Giti
					н	A(8)	VSS_IW	C[4]	A(6)		VDD_LV_	VSS_LV	VSS_LY	VS5_LV _COR	VSS_LV	VSS_LV	VOD_LV		G[12]	TMS	VOD_HV _FLA	тск
					3	C[7]	[4]	F[3]	F[7]		VDD_LV_ COR	VSS_LV	VSS_LV	VS5_LV COR	185_UV	VSS_LV	VOD_LV		G[13]	H(I)	NDD_LV	B(5)
					к	胡	F[10]	FIN	181			VSS_LV	VSS_LV	VS5_LV _COR	V55_LV	V55_LV	VOD_LV		G[15]	HQ	VSS IV	101
					L.	H(8)	F[11]	(9)	Djaj		VDD_LV_ COR	VSS_LV	VSS_LV	VS5_LV COR	VSS_LV	VSS_LV	VDD_UV		A[2]	G[14]	NC	3(22)
					N	VED H	UDD HI	<b>8:0</b>	D(5)	1	VDD_LV_ COR	VDD_LV	VOD-LV	VOD LV COR	VDD_LV	VDD_LV	VDD_UV		C[12]	(0]	G[7]	G[5]
					N	XTAL	VSS, HV	Diał	VSS_LV_		LUR		- COURT	von	_ un	- Von	UNIX		G[E]	駒	VDD LV	VIIS_LV LEAST
					P	VSS HV	RESET_	DIS	VDD_LV_ PLL	(12)	(13)	8(0)	相	周	.[7]	8[14]	A(0)	H[14]	ાંગ	NC	C[11]	0[11]
					R	EXTAL	FCCU	VSS_HI		8(7)	E[6]	VDD_HV ADRED	1.1	VDO_HV		B[15]	C[0]	BCTRL	_	VSS_HV_	D[10]	G[10]
					,	VSS_HV	FIC H	(1)	CII	E	87	VSS HV ADRED	and the second data in the second data is a second data in the second data in th	State of State of State	VDD HV		E[12]	EIO	A[!]	6[11]		VBS_HV
Pin Conflicts				▽ □ □		174	VSS_HV	101	E[4]	qa	82	B[9]	6(12)	VSS_H/	EIM	EIII	NC.		VOD_HV PMLHO		VSS_HV_	VSS HV
0 items		Module	Pin	Location		10	0		511	MM	alet	els1	stiet	ADV	-	50.1			PMLIO		0	0
		module	FIL	Location		Conne	iction sta	tusi	Tw.	arring	l	MAP		257 (	Top V	iew)						
+ [	m																					
					-																	

#### Pin Configuration in the PCB Design

How can we manage this complexity for good PCB Design?

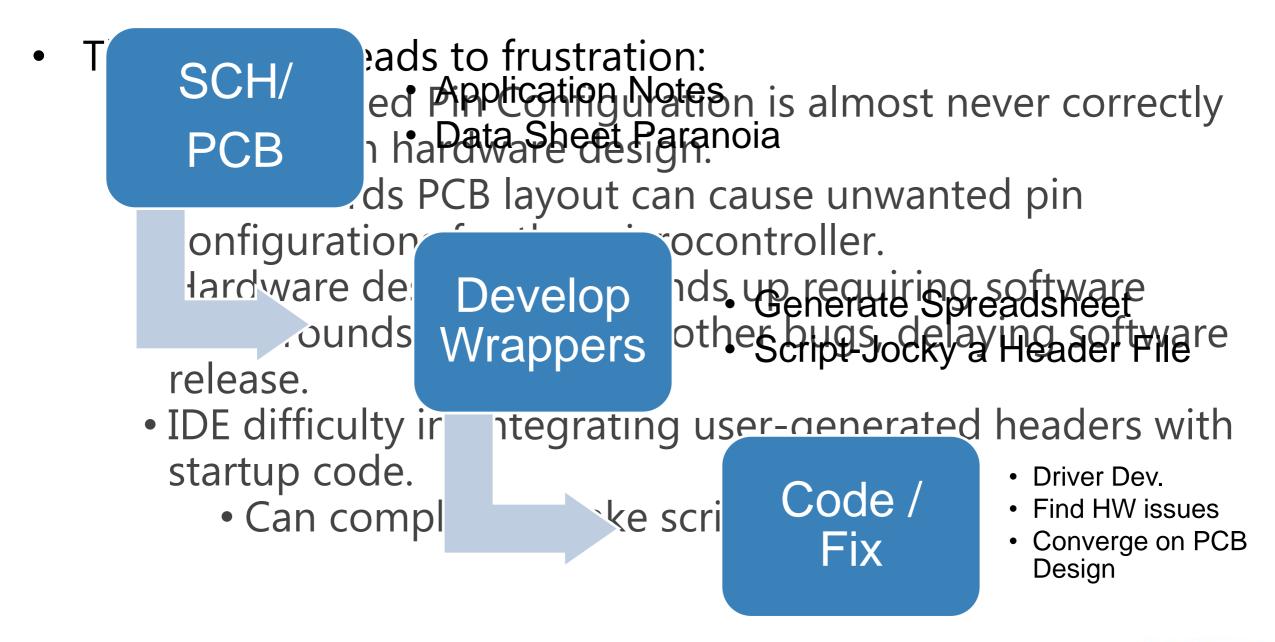
#### The traditional approaches:

Reference of Schelegattles Reveloted of Kitsights: ned Pin Configuration is often not correctly Design enected in hardware design. 'ariants of F signs often include un-needed Develop Merge Reference Examples explai Software • Add Custom Functions usually optimally designed for Reference B arget application your particula Modify Custom Reference PCB Designs

#### Pin Configuration in the PCB Design

How can we manage this complexity for good PCB Design?

The traditional approaches:



#### Introduction to the TASKING Pin Mapper

The TASKING Pin Mapper will accelerate:

- Peripheral IO Selections
- Conflict Resolution
- Pin and Port Symbolic Naming
- Peripheral and GPIO wrappers generation

Δltuum

• PCB / Hardware design.

# Introduction to the TASKING PinMapper

TASKING Pin Mapper - myproject/r		100-15-P-1																	-	
File Edit Navigate Search Proje							Q	uick A	Access			. 83	<b>V</b> 7	TASK	ING C	:/C++	6	TASKI	VG Pin	n Mapp
a) myproject.pincfg			- 0	D Pack	age 🛛										(	⊇ ⊛			Ab	• -
TASKING Pin Mapper - MF	PC574xP / MAPBGA257	1	8 & • 6		2	1		L		,	4	0	10	11	12	13	14	15	N	17
Pin Selection P	in Configuration		-	A VSS H	VSS W	A[14]	A(9)	D(3)	JCOMP	HIS	Q15]	VICE DON	(i)	E[13]	41	F[15]	H[12]		VSS_HV_	VSS_HV
type filter tex 🖉 🗐 🖽	👘 Use	tag: ty	/pe new tag 👻	E KIVSEL		F[3]	Dj2j	Bļoj	FIQ	D[4]	D(D)	V\$5_M/ _30	E[14]	A(10)	B[3]	HE	C[10]	7[3]	NDD IIN	V\$5_W/
> Ports			2011 (2)	C 415	tol	VSS_H/	FCCU_ F[1]	4(13)	(0)	H(10)	E[15]	H[11]	हम्ब	42	8[2]	HR	B[1]	V\$5_HV_ 10	B(0)	H(S)
Peripherals				D 4(6)	(7)	A[15]	CMI	NC	EXT_POR	A(12)	NEW HY	VSS_HI	A[11]	2	F[14]	44	wears or	VPP_TE ST	44	F[12]
D Other Pins				E F[4]	刑罚	0[1]	NN_B			1	12	7		12	2	-	NC	C[r3]	G[3]	0(14)
				F F[5]	H[7]	H[5]	H(4)		VDD_LV_ COR			V00_LV _COR				-	CIM	D[12]	G(4)	0[2]
				G MDO	100.14	대왕	A(7)		VOD_LV_ COR	VSS_LV	155_LV	VSS_LV COR	V55_{V	VSS_LV	VOD_UV		BHJ	A[3]	J0)	GHEI
				H A(8)	VSS_W	ભા	A(6)		V0D_LV_ COR	VSS_LV	VSS_LY	VS5_LV _COR	VSS_LV	VSS_LV	YOD LY		G[12]	TMS	VED_HV FLA	TCK
				2 C[7]	[4]	F[8]	मग		VDD_LV_	VSS_LV	VSS_LV	VS5_LV COR	185_LV	the second s			G[13]	H(1)	VDD_LV_	B(5)
				K 期	F[10]	FIN	181		VOD_LV_ COR	VSS_LV	VSS_LY	VS5_LV COR	V55;LV	V55_LV	VOD_LV		G[15]	HO	VSS 1V	101
				L H(8)		(9)	D(R)		VOD_LV_	VSS_LV	VS8_1.V	VS5_1V	VSS_LV	VSS_LV	VDD_LV		A[2]	6[14]	NC	J[11]
				N VIO H	N UND H	8:01	D(5)			VDD_LV	VOD-LV	VOD_LV	VDD_LV	VDD_LV	VDD_LV		C[12]	彻	G(7)	615
				N XTAL	VSS_HV	-	VSS_LV_		UUK	COR	COR	COR	CON	CUN	COR		대리	啊	VDD_UV	V85_1V
					RESET,	Dig	VOD_LV_	(12)	(13)	8(8)	桐	相	.[7]	B[14]	A(0)	H[4]	6[9]	NC	C[11]	UFAST O(11)
				R EXTA	FOCU_ FIG	-	PLL D(7)	8(7)	E[6]	VDD_HV ADRED	1.1	VDO_HV ADRE1		B[15]	C[0]	BCTRL	NC	VSS_HV_	DINI	G[10]
				VSS H	-	(1)	CII	-45 日初	87	ADREO VSS_HV ADREO	and the second second second	And in case of the local division of the loc	VDD HV		E[12]	EIN	A[!]	IO G[11]		VBS_HV
b Pin Conflicts			~	V9S_H	L V55 HV							ADRE1					VDD_HV PMUID		0	10 VSS HV
0 items		2	1	0	10	[11]	E[4]	OS!	티외	8(9)	6(12)	ADV	EM	E[11]	NC:	NC	PMLIO	NC	0	0
Description	Module P	in	Location	~	ection stu					MAPI	BGA	257 (	Top Vi	iew)						
					Bror Bror		W	erning		0	ĸ									
•	Ш	1																		

# Peripherals

TASKIN				_ 🗆 ×										
ile Edit Navigate Search Project Debug Pin Mapper Window Help														
3 ▼ 🗒 🖻   🖬 ! 🅸 ▼ ! 🖋 ▼ ! ½ ▼ 💱 ▼ 🏷 ← ▼ ↔ ▼   🗃									Q	uick Acce	ess	🖻 🛛 💌 T	ASKING	C/C++ TASKING Pin Mapper
a K64_Boo.pincfg 🛛		🗖 Pa	ckage 8	3 📮 C	onsole									@ @ ▼   🏢 ▼ 🔤 ▼ 🗆 🗖
TASKING Pin Mapper - MK64FN1M0CAJ12R / WLCSP-142	🖻 🖆   🍐 🔻 🗟   🧸	1	2	3	4	5	6	7	8	9	10	11	L	
Pin Selection Pin Configuration		A	PTC4/ LLWU_P8	PTC7	PTC9	PTC12	PTC15	PTC17	PTD1	PTD5	PTD7	PTD9	PTD14	A
type filter text 🖉 🖳 🖽	🧃 Use ta 🛛 new tag 🗸	в	PTC3/	PTC6/	PTC8	PTC11/	PTC14	PTC18	PTD3	PTD6/	PTD8	PTD12	PTD15	в
PA     Module name: PA3			LLWU_P7	LLWU_P10	FICO	LLWU_P11	1014	FICID	FIDS	LLWU_P15	FIDS	FIDIZ	FIDIS	в
<ul> <li>PA0</li> <li>Symbolic name:</li> </ul>		с	PTC2	PTC5/ LLWU_P9	PTC10	PTC13	PTC16	PTD2/ LLWU_P13	PTD4/ LLWU_P14	PTD11	PTD13	PTE0	PTE3	с
PA2	~	D	PTB23	PTC0	PTC1/	PTB22	PTC19	PTD0/	PTD10	PTE1/	PTE2/	PTE4/	PTE5	D
PA3 PA3 Configuration					LLWU_P6			LLWU_P12		LLWU_PO	LLWU_P1	LLWU_P2		
Mode:     Alternate function       PA5     PA6       Speed:     High		E	PTB18	PTB19	PTB20	PTB21	VDD	VSS	PTE6	PTE7	PTE8	PTE9	PTE10	E
PA7 Output type: Push-pull V		F	PTB16	PTB17	VDD	VSS	VSS	VDD	VDD	ADC0_DP1	PTE11	PTE12	VSS	F
PA8 PA9 Pull-up/Pull-down: No pull-up/no pull-dc ✓		G	PTB10	PTB11	PTB9	PTB8	VDD		vss		ADC0_DP0 ADC1_DP3	VOUT33	USB0_DP	e
PA10 Chip input/output			PIBIO	FIBI	1100	100	100		100	AD00_DIMI	ADC1_DP3	100133	0300_01	3
✓ PA11 PA3: ✓ ADC1_IN3 ✓	⇔	н	PTB7	PTB6	PTB5	PTB4	VSS	VSS	VDD	VDD	ADC0_DM0 ADC1_DM3	VREGIN	USB0_DM	н
PA12 None PA13 ADC1_IN3 PA14 ADC2_IN3			PTB3	PTB2	PTB1	PTB0/	✓ PTA14	✓ PTA11	✓ PTA2	PTE27	RTC_	ADC1_DP0/ ADC0_DP3	ADC1 DP1	4
✓ PA15 ADC3_IN3	Ň		✓ 100	✓	~	LLWU_P5	<ul> <li>✓</li> </ul>	~	✓					
FXTI FXTI3	► 1	к	PTA29	PTA28	PTA27	PTA26	PTA12	PTA8	PTA1	PTE25	ADC0_SE16 CMP1_IN2/ ADC0_SE21	ADC1_DM0/ ADC0_DM3	ADC1_DM1	к
Pin Conflicts       ■ Properties ≅       TIM2_CH4         Property       Valu       TIM5_CH4         TIM9_CH2       TIM9_CH2		L	RESET_b	✓ PTA24	✓ PTA25	PTA16	✓ PTA9	✓ PTA5	✓ PTA0		VREF_OUT CMP1_IN5/ CMP0_IN5/ ADC1_SE18	VREFH	VDDA	L
USART2_RX USB_OTG_HS_ULPI_D0		м	PTA19	VSS	✓ PTA17	PTA 13/ LLWU_P4	✓ PTA7	PTA4/ LLWU_P3	PTE28	VBAT	DAC1_OUT CMP0_IN4/ CMP2_IN3/ ADC1_SE23	ADC1_SE16/ CMP2_IN2/ ADC0_SE22	VREFL	м
GPIO		N	PTA18	VDD	✓ PTA15	PTA10	✓ PTA6	✓ PTA3	PTE26	EXTAL32	XTAL32	DAC0_OUT/ CMP1_IN3/ ADC0_SE23	VSSA	Ν
		Ca	1 Dnnecti Error	2 on stat		₄ 64FN1M0 ming	5 DCAJ121		7 CSP-142	8 2 (Top V	9 ïew)	10	11	
<	>													

IASKING.

AICIUM.

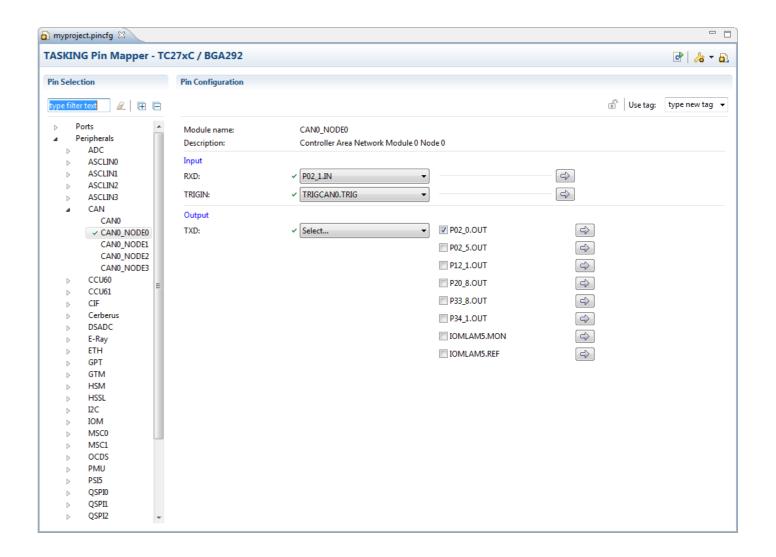
#### Package Selection

onfiguration: Debug [Active]	~	Manage Cor	figurations
		nanage eer	garations
ocessor selection			
Kinetis K53 Series		^	Expand A
Kinetis K60 Series			
Kinetis K61 Series			band Sele
Kinetis K63 Series			Collapse A
∡ ✔ Kinetis K64 Series			
✓ MK64FN1MQVLL12			
MK64FN1M09LQ12			
MK64FN1M0VMD12			
MK64FX512VLL12			
MK64FX512VLQ12			
MK64FX512VMD12		*	
PU problem bypasses and checks			
752770 LDR SP, mem may result in incorrect SP when interrupted			Select Al
			Decelect (
			Deselect A
Show all CPU problem bypasses and checks			
Add startup file(s) to the project			
Update CPU of referenced/referencing projects			
opulate cro or referenced/referencing projects			

Package Selection happens when you begin a new embedded project, but can be changed later on when targeting a different processor variant:



# Add a Pin-Mapper Document to the Project



10

Altıum. TASKING.

## **Pin Configuration - Ports**

Pin Selection	<b>Pin Configuration</b>		
type filter text 🖉  🖽 🖻			🧉 Use ta 🛛 new tag 🗸
Ports	Module name:	PA4	
⊿ PA	Symbolic name:	STATUSLED	
✓ PA0	Comment:	Open Drain to Turn On Status LED	~
✓ PA1	commenta	Open Drain to Turn On Status LED	× .
✓ PA2	PA4 Configuration		
✓ PA3	Mode:	Output mode	
PA4	wode.	Output mode +	
✓ PA5	Speed:	Low 👻	
PA6	Output type:	Open drain 🗸	
PA7			
PA8	Pull-up/Pull-down:	No pull-up/no pull-dc ♥	
PA9	Chip input/output	No pull-up/no pull-down が	
PA10		Pull-up	
✓ PA11	PA4:	Pull-down	
✓ PA12			
	Sympholi	Accian a use	ar defined aurob

**Symbolic name** - Assign a user-defined symbol name to the port pin PA13 **Comment** - Any user comments can be added here PA14 ✓ PA15 **Mode** - This field sets the port mode register PB **Speed** - Sets the port output speed register (2, 25, 50 or 100 MHz) PC PD **Output type** - Sets the port output type register (push-pull or open drain) PE **Pull-up/Pull-down** - Sets the port pull-up/pull-down PF Chip input/output. Here you can make a pin connection PG

## **Pin Configuration - Peripheral**

Module name - The name of the selected peripheral Configuration. The configuration depends on the peripheral you selected Input / Output - A list of port I/O functions (virtual pins)

Altıum.

TASKING

Pin Selection		<b>Pin Configuration</b>	ı						
type filter text 🖉 🖳						í	Use ta	new tag	~
TIM14	^	Module name:		USB_OTG_HS					^
TIM2		USB_OTG_HS Con	fiau	ration					
TIM3		External Phy:	5	Host_Only	~				
TIM4		-							
TIM5		Internal Phy:		Host_Only	~				
TIM8		Activate_SOF:		Disabled	~				
TIM9 UART4		Activate_VBUS:		Disabled	~				
UART5		Input/Output							
USART1				PB14	~			$\Rightarrow$	
USART2		DM:	*	PD14	•				
✓ USART3		DP:	~	PB15	~			$\Rightarrow$	
USART6		ID:		None	$\sim$			$\Rightarrow$	
✓ USB_OTG_FS		SOF:		None	$\sim$				
<ul> <li>USB_OTG_HS</li> <li>Other Pins</li> </ul>		ULPI_CK:	~	PA5	~			$\Rightarrow$	
VBAT		ULPI_D0:	×	* PA3	~			$\Rightarrow$	
VDD		- ULPI_D1:		PBO	~			$\Rightarrow$	
PDR_ON								$\Rightarrow$	
VSS_SA		ULPI_D2:	~	PB1	~				
BOOT0		ULPI_D3:	~	PB10	~			$\Rightarrow$	
VSS VCAP_2	~	ULPI_D4:	~	PB11	~			$\Rightarrow$	5
< >	Ť	<						>	Ť

#### **Pin Configuration - Other**

Module name - The name of the selected other pin Boot domain - The boot domain the pin uses Power domain - The power domain the pin uses Reset domain - The reset domain the pin uses

Pin Selection		Pin Configuration		
type filter text 🖉 🖉	Ð 🖻			🗈 Use ta 🛛 new tag 🗸
> PH	^	Module name:	NRST	
> PI		Power Domain:	NRST	
Peripherals				
<ul> <li>Other Pins</li> </ul>				
VBAT				
VDD				
PDR_ON				
VSS_SA				
BOOT0				
VSS		R		
VCAP_2		. //		
NRST				
BYPASS_REG				
VSSA				
VCAP_1				
VREF-				
VDDA				
	~			

# **Package View**

The Package view shows a graphical representation of a BGA

<b>D</b>	Packa	age 🛛	X												0	€ -		•	Ab 🔻	· -	
	1	2	з	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
A	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	₽14_1	P15_6	P15_4	P15_1	VDDP3	V <u>SS_</u> 18	A
в	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	VDDP3	V <u>SS_</u>	P15_0	в
с	P02_2	P02_1																	P15_2	P20	с
D	P02_4	P02_3		VSS	VFLEX	P11	P11_ 14	P11_5	P11_6	P11_4	P14_	P14_9	P14_7	P15_8	P15_7	VDDFL	V <u>SS_</u>		P20_	P20_ 13	D
E	P02_6	P02_5		P02_9	VSS	P11	P11_8	P11_7	P11_1	P11_0	<u> </u>	P12_0	P14_2	P15_5	VDDEL	V <u>SS_</u> 18	P20_9		P20_ 10	P20_ 11	E
F	P02_8	P02_7		P02_	P02_ 10	13									3	18 ESR0	P20_6		10 P20_7	11 P20_8	F
G	¥00_0	P00_1		11 <sup></sup> P01_4	10 <sup></sup>			VDDVD	V <u>SS_</u>	V <u>SS_</u>	V <u>SS_</u>	V <u>SS_</u>	VDD			ESR1	PORST		P20_1	P20_3	G
														 		<u> </u>	<u> </u>		$\vdash$		-
н	P00_2	P00_3	 	P01_6	P01_5		VDDVD		V <u>SS</u> _ 24	V <u>SS</u> _ 24	V <u>SS</u> _ 24	V <u>SS</u> _ 24	ļ	VDD		P21_7	P21_6		P20_2	P20_0	H
	P00_4	P00_5		P00_6	P01_7		V <u>55</u> -	V <u>SS</u> _		V <u>55</u> _	V <u>SS</u> _		V <u>55</u> _	V <u>SS</u> _		тск	P21_1		P21_3	P21_5	3
к	P00_7	P00_9		P00_8	P00 10-		V <u>55</u>	V <u>55</u> _	V <u>55</u> _	N.C.		TMS	P21_0		P21_2	P21_4	к				
L	P00_ 11	P00_ 12		AN43	AN42		V <u>55</u> - 27-	V <u>SS</u> _ 21	V <u>SS</u> _ 21	V <u>SS_</u>	V <u>SS</u> _ 21	V <u>SS</u> _	V <u>SS_</u>	V <u>55</u> -		P22_ 10	P22_ 11		TRST	VSS	L
м	AN46	AN47		AN41	AN40		V <u>SS</u> _ 20	V <u>SS</u> _		V <u>SS</u> _ 20	V <u>SS</u> _		V <u>SS_</u> 20	V <u>SS</u> _		P22_8	P22_9		XTAL2	XTAL1	м
N	AN44	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	VDDP3	N
P	P40_9	P40_7		P40_4	AN34			VDD	V <u>SS</u> _	V <u>55</u> _	V <u>55</u> _	V <u>SS_</u> 19	VDD		1	P22_4	P22_5		P22_1	P22_0	P
R	P40_5	AN35		AN31	AN23											P23_7	P23_6		P22_3	P22_2	R
т				AN30	AN22	AN15	AN12	AN6	AN4	AND	VEVRS	P34_2	P34_4	P33_	P32_5	VSS	P23_5		P23_3	P23_4	т
U	AN29	AN28		N.C.	AN17	AN14	AN9	AN7	ANB	AN1	P34_1	P34_3	P34_5	P33	P32_6	P32_7	VSS		P23_1	P23_2	U
v	P40_3	P40_2																	VEXT	P23_0	v
w	P40_1	P40_0	AN19	AN18	AN16	AN13	AN11	ANB	AN2	P33_0	P33_2	P33_4	P33_6	P33_8	P33_	P33_	VGATE	P32_4	VSS	VEXT	w
Y	N.C.	AN21	AN20	VSSM	VDDM	VAREF		AN10	AN5	P33_1	P33_3	P33_5	P33_7	P33_9	P33_ 11	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	
								TC2	7xC -	BGA	292 (1	Гор V	iew)								
		ection s Error	status:	Г		rning			ОК												

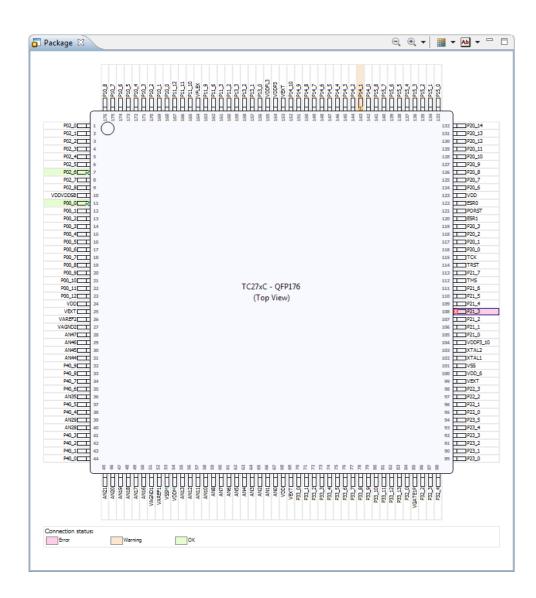
- A square around a pin marks a selected pin; In the shown package pin P00\_0 is selected
- A green check mark indicates that the pin has a valid connection; here P00\_0 and P01\_6
- A red cross indicates an error; P21\_3 as an example
- A triangle with exclamation mark indicates a warning; P14\_1 as an example
- When you click on a pin, the pin appears in the editor

Altum

TASKING

# **Package View**

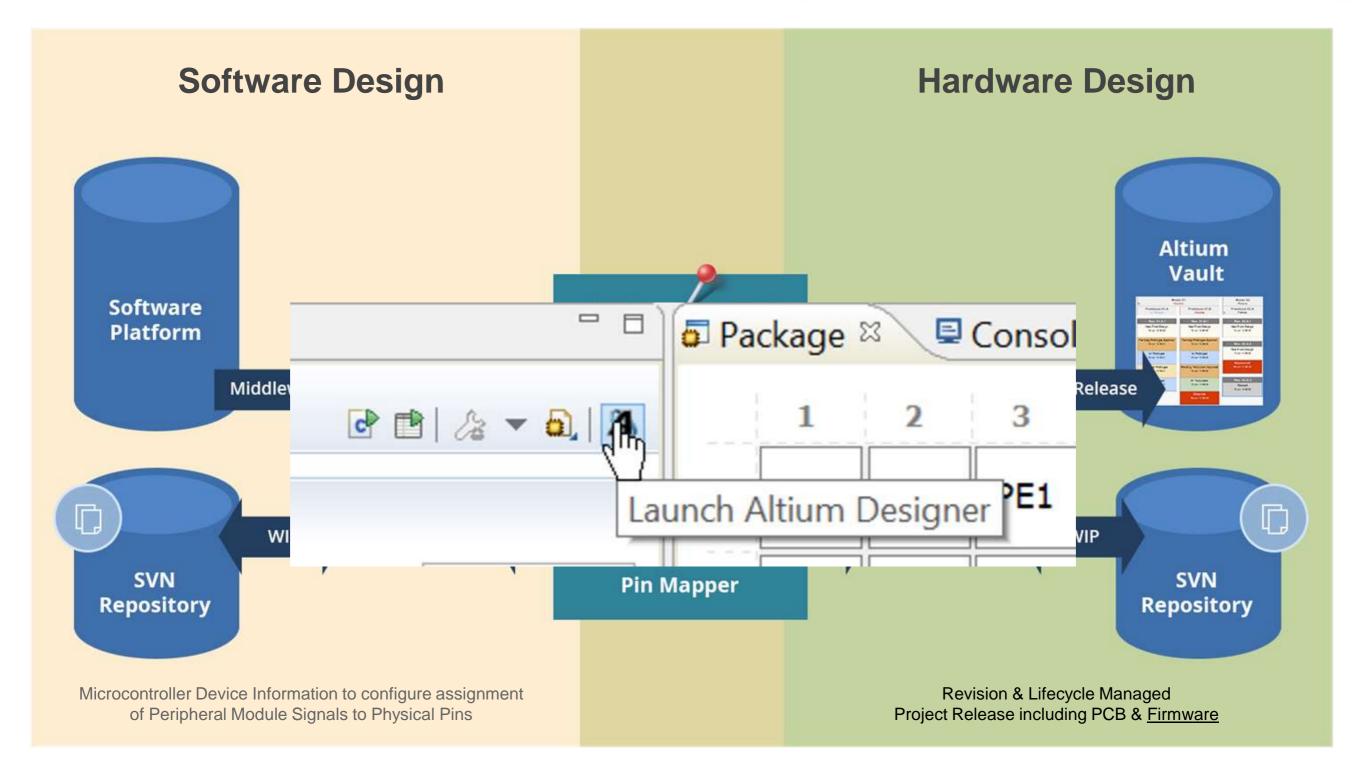
The Package view shows a graphical representation of a QFP



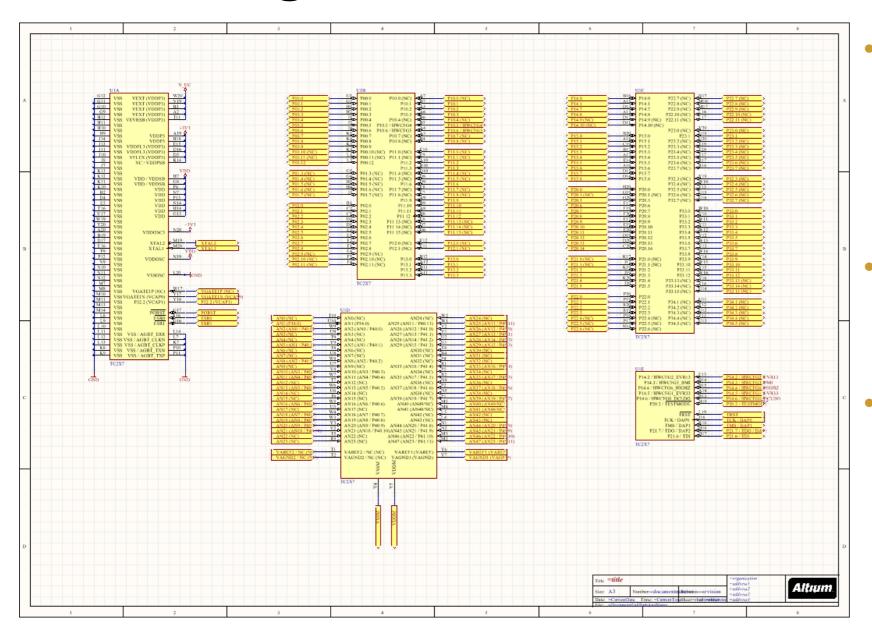
- A square around a pin marks a selected pin; In the shown package pin P21\_3 is selected
- A green check mark indicates that the pin has a valid connection; here P00\_0 and P02\_6
- A red cross indicates an error; P21\_3 as an example
- A triangle with exclamation mark indicates a warning; P14\_1 as an example
- When you click on a pin, the pin appears in the editor



## We can no longer work in Isolation



# **Push changes to Embedded Schematic**



- The Pin Mapper will Autogenerate the Schematic with the correct Pin signal names and Port grouping
- Power Pins and Boot options are also inserted
- Package view shows a graphical representation of the QFP package

Microcontroller Aware Hardware Design

# Pin Swapping

# Once the bane of Embedded Developers...

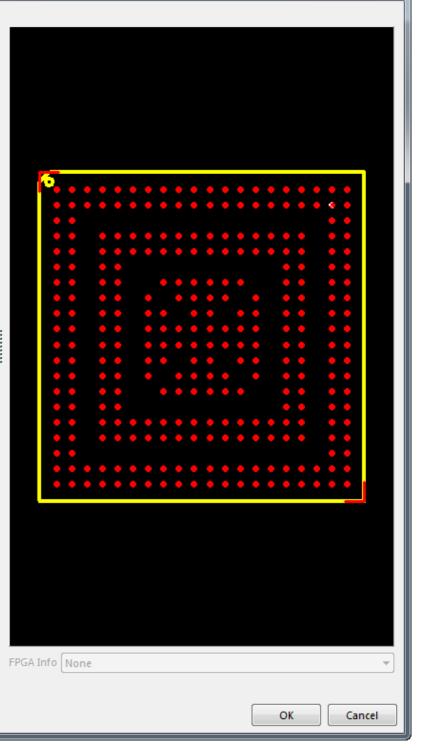


## Pin Swapping at the PCB Level

#### Configure Pin Swapping For [U101 in Appl Kit TC2X7.PrjPcb / (No Configuration)]

orag a column header	here to group by that column				
	Pin Status				Swap D
Designator	Pin Name	Net	Туре	Part No.	Pin Group /
V13	P33.6	P33.6	I/O	3	7
13	P33.7	P33.7	I/O	3	7
V14	P33.8	P33.8	I/O	3	7
14	P33.9	P33.9	I/O	3	7
V15	P33.10	P33.10	I/O	3	8
15	P33.11	P33.11	I/O	3	8
1	AN43 (NC)	NetU101_L4	Input	4	9
5	AN42 (NC)	NetU101_L5	Input	4	9
11	AN46 (AN22 / P41.10)	NetU101_M1	Input	4	9
12	AN47 (AN23 / P41.11)	NetU101_M2	Input	4	9
14	AN41 (AN48/NC)	NetU101_M4	Input	4	9
15	AN40 (AN49/NC)	NetU101_M5	Input	4	9
4	AN36 (NC)	NetU101_N4	Input	4	9
5	AN38 (NC)	NetU101_N5	Input	4	9
4	AN32 (NC)	AN32(15)	Input	4	9
5	AN34 (NC)	NetU101_P5	Input	4	9
4	AN31 (NC)	NetU101_R4	Input	4	9
5	AN23 (NC)	NetU101_R5	Input	4	9
10	ANO (NC)	ANO	Input	4	9
4	AN30 (NC)	NetU101_T4	Input	4	9
5	AN22 (NC)	VDD	Input	4	9
5	AN15 (NC)	NetU101_T6	Input	4	9
7	AN12 (NC)	NetU101_T7	Input	4	9
3	AN6 (NC)	AN6(5)	Input	4	9
)	AN4 (NC)	NetU101_T9	Input	4	9
5	AN17 (NC)	AN17(14)	Input	4	9
6	AN14 (NC)	NetU101_U6	Input	4	9
7	AN9 (NC)	NetU101_U7	Input	4	9
8	AN7 (NC)	NetU101_U8	Input	4	9
9	AN3 (NC)	AN3	Input	4	9
/2	AN24 (NC)	AN24(12)	Input	4	9
20	P22.0	P22.0	I/O	3	9
19	P22.1	P22.1	I/O	3	9
20	P22.2	P22.2	I/O	3	9
19	P22.3	P22.3	I/O	3	9

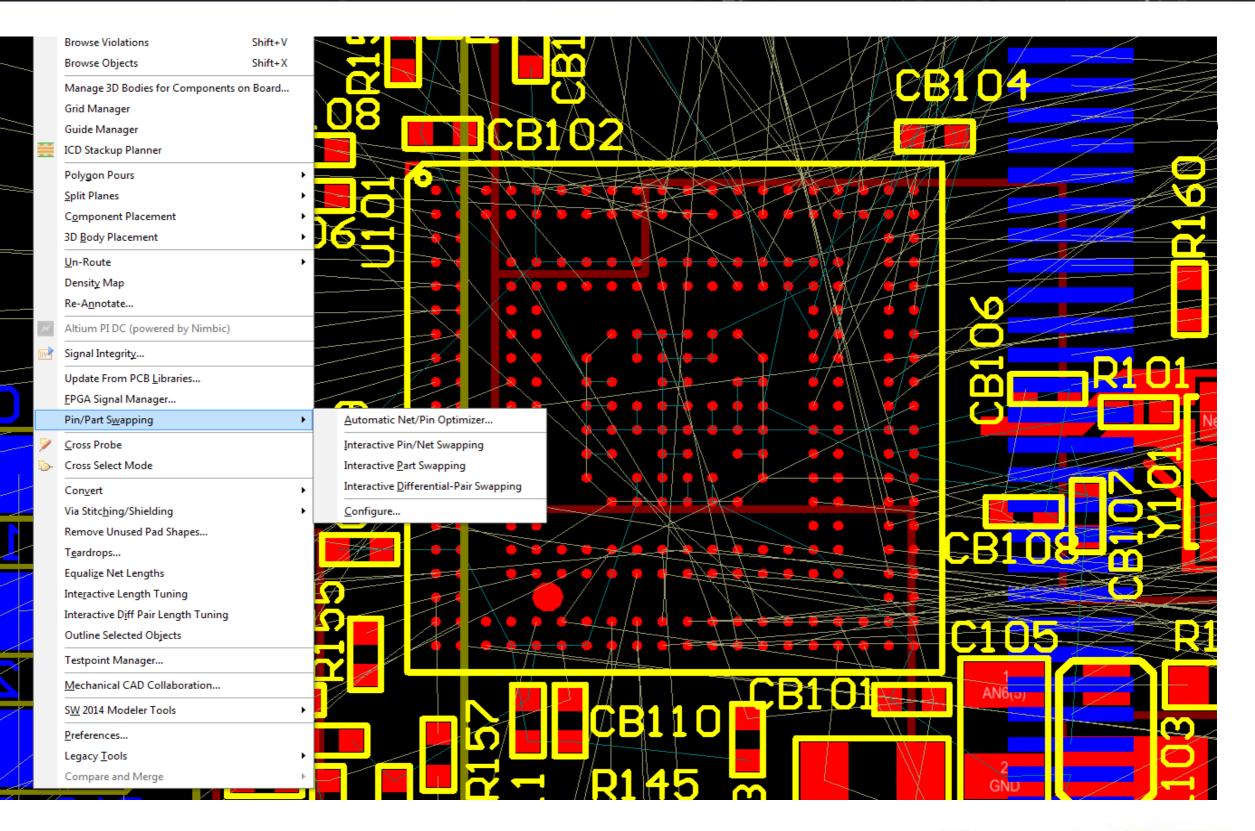
.....



×



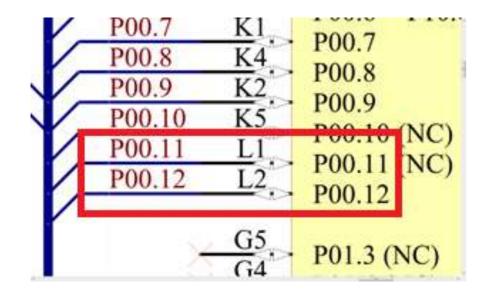
#### Pin Swapping at the PCB Level

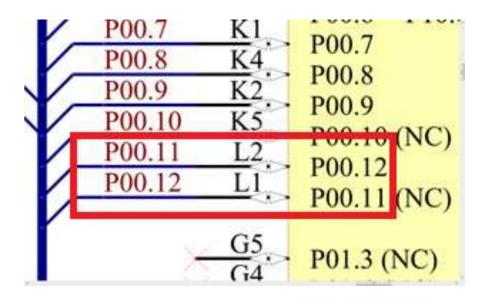


Altıum. TASKING.

#### **Pin Swapping Back-Annotates to Schematic**

When you Configure the swap groups in the *Pin Swapping* dialog, the edits you make are applied to the schematic components. Pin swaps in the PCB are passed back to the schematic in through a reviewable, controlled dialog known as the Engineering Change Order.





Altıum. TASKING.

#### **PCB Signal Integrity Analysis**

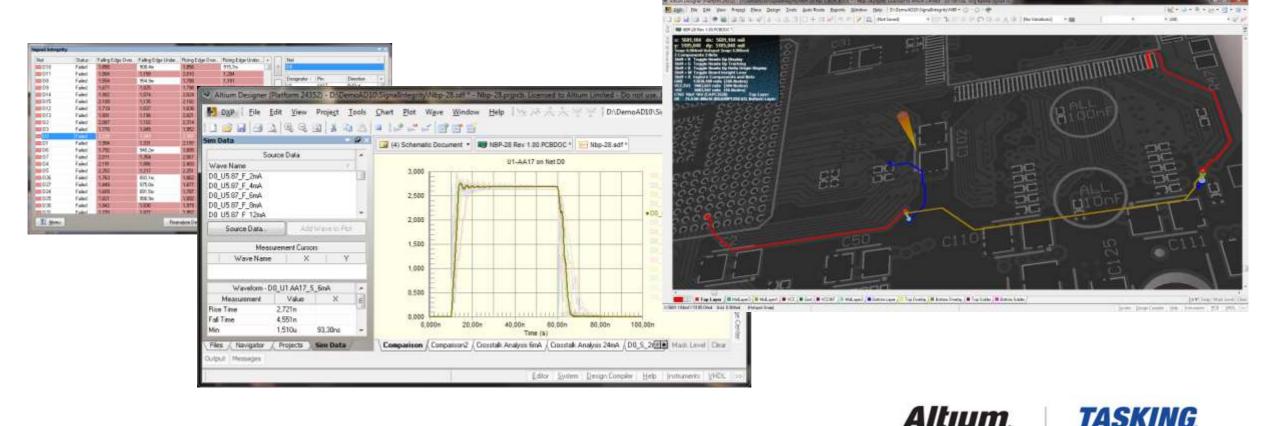
# Proven Technology

- Transmission line calculations and I/O buffer models (IBIS)
- Industry-proven algorithms (Field-Solver)

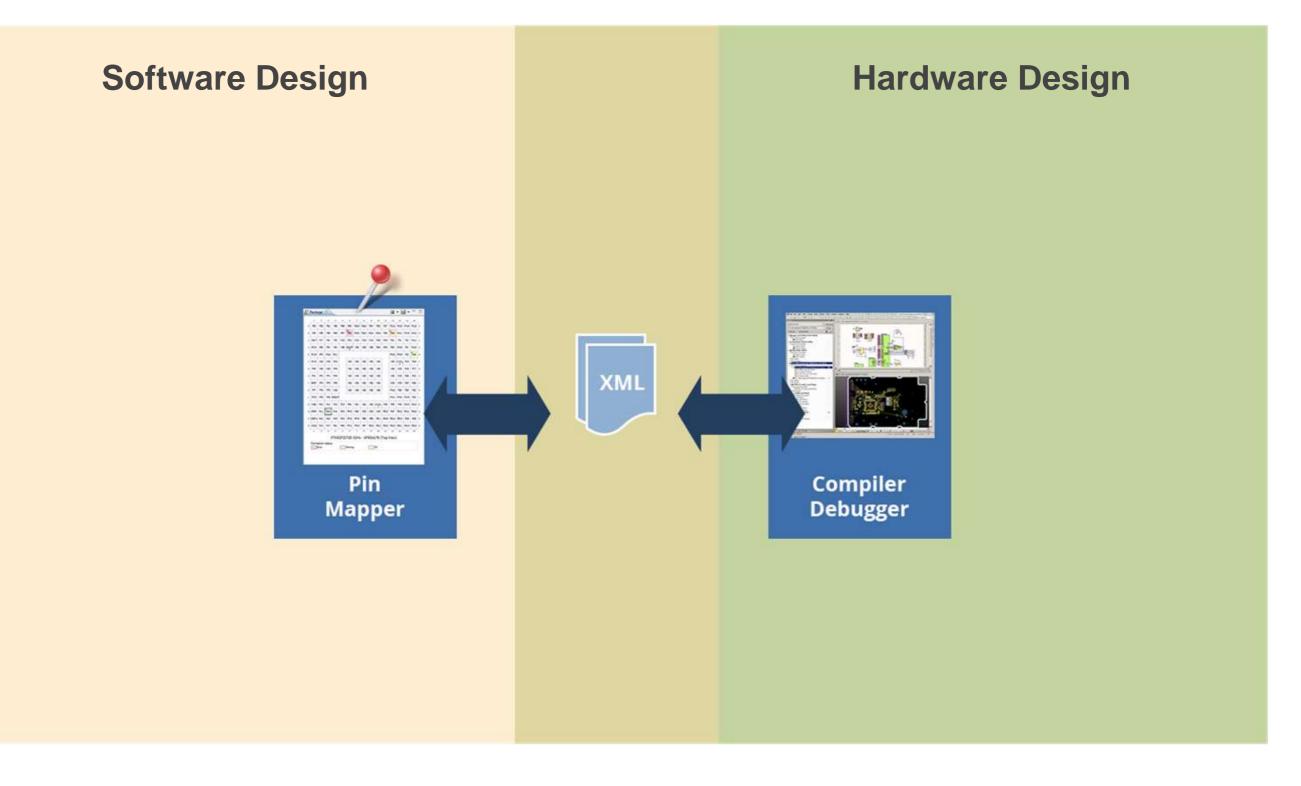
# Quickly explore potential solutions

Cross-Probing / Highlighting SCH/PCB/SIM

# Takes advantages of Unified Database



# The Tasking Pin Mapper can be EXTENDED!



Altıum. TASKING.

	≣ *K64_boo.swpxmi 🛛					
Software	TASKING Software Platfo	<b>e</b> =				
	Software Services		Device Stacks			
	Software Platform Builde	er ^				
Talkin	Cortex-M Architecture					gless if
you d	Interrupt Management					code,
you d driver	JSON Parser					Jless if code, yly use
	POSIX Message Queues		Graphics Services			
	POSIX Multi-threading		GRAPHICS_1			
	POSIX Signals		SPFD5408 LCD Module Driver (parallel)	JPEG D	ecoder	
	Software Timing Service	s 🗸	DRV_SPFD5408_PAR_1	JPGDEC	CODER_1	
ĺ	🏖 Pin Conflicts 💷 Properties 🛛				r 🕈 🗸	
	POSIX Multi-threading				ŧ	
	Property	Value	Тур	e	Range / Prototype	^
	Software Service Options					
	Shared Memory	false	BOC	LEAN		
	DESTRUCTOR_ITERATIONS	4	UIN	8	>= 4	
	KEYS_MAX	16	UIN	16	16128	
	THREADS_MAX	8	UIN	8	>= 8	
	MAIN_STACK_SIZE	2048	UIN	[32	>= 512	
	MAIN_PRIORITY	32	UIN	[8	162	-
	IDLE_STACK_SIZE	512	UIN		>= 512	
	DEFAULT_STACK_SIZE	2048	UIN		>= 512	-
	SPORADIC_SERVER	false		DLEAN		-
	CPUTIME	false		DLEAN		
		1024	UIN			TASKING.
	CANCEL_SUPPORT	false	ROC	IEAN		<b>~</b>
	<				2	→

# **Software Platform Builder**

- the Software Platform Builder is used to manage your Software Platform
- It is both a graphical editor and a code generator
- Collections of software modules are delivered as Software Platform repositories

# **Software Platform repository**

 A Software Platform repository may contain any kind of software, but typical modules include interrupt services, timers, peripherals (hardware wrappers), drivers, kernel services (such as POSIX multithreading), device I/O, file system (FAT-Fs), networking (TCP/IP), graphical user interface (GUI), etc.





An address of the second set.

\* blocking read from

- /\* without echo
- /# IE #2
- /# ionose th
- 1= 12 1
- 74 August 12.00



# **Support for the GTM-IP MCS**

CERT I

Copyright © 2015 Altium BV, Confidential

- CORDE

### **Programming the GTM / MCS**

- The GTM is a timer module (containing MCS cores), present on several high-end microcontrollers for automotive, including Qorivva Power Architecture
- Traditionally the GTM is programmed in assembly, using assembler solutions from BOSCH (GCC based) or TASKING (VX-technology)
- 2013: Altium and BOSCH AE have investigated enhancements to the GTM core for making the development of an efficient C compiler possible
- 2014: Altium and BOSCH AE started the co-operation to develop a C compiler for the new GTM core level V3.x
- 2015: Altium releases C compiler for GTM V3.x



## New TASKING C compiler for GTM / MCS

- Supports GTM-IP MCS core version V3
- *Truly* ISO-C99 compliant
  - Supports all ISO-C99 language features and C-library functions
  - Facilitates cost-efficient tool qualification against safety standards such as ISO 26262 and DO-330
- Based on TASKING's well known Viper "VX" compiler technology:
  - Similar to the C compilers for ARM (Kinetis) and Power Architecture (Qorivva), with industry proven stability
  - Perfectly suited for specialized cores, as proven through C compilers for other coprocessor cores, targeting both efficiency and safety
  - ISO 26262 Support Program facilitates certification against various safety standards
- Data type characteristics match the MCS ISA
  - 24-bit integers, 48-bit longs, 32-bit float, ...
- Integrated static code analysis for MISRA C and CERT



TASKING

ISO 26262



#### New TASKING C compiler for GTM / MCS (cnt'd)

- Each MCS channel can be associated with an execution thread
  - Default register usage is limited to OREG registers
    - No need to suspend a subsequent channel
  - Optional use of XOREG registers
    - Improves execution speed at the cost of suspending a channel
- Emits debug information in DWARF 3 format
  - Facilitates symbolic C-level debugging
  - Supported by third party (in-circuit) debugger vendors





#### GTM / MCS Compiler Intrinsic Functions

- Provide access to MCS ISA features not addressable via ISO-C99
- Designed to optimize throughput over MCS busses
  - Compiler optimizations work efficiently with intrinsic functions
  - Function prototypes permit bus data to be accessible through function parameters and return values
- Supported operations:
  - GTM Time Base Unit Access
  - Trigger Registers Access
  - Bus Master Addressing





- Initial level of tool support for the GTM/MCS has been rolled out for multi-core MCU toolsets in 2011
- TASKING offers support for all versions of the GTM/MCS

Taal	GTM version				
Tool	V1	V2	V3		
Eclipse IDE	$\checkmark$	$\checkmark$	$\checkmark$		
Assembler	$\checkmark$	$\checkmark$	$\checkmark$		
Multi-core Linker	$\checkmark$	$\checkmark$	$\checkmark$		
C compiler			$\checkmark$		
Simulator / Debugger	$\checkmark$	$\checkmark$	$\checkmark$		

- C compiler support available in 2015
  - Initial (standalone) release available per April 21, 2015
  - Available prior to HW/silicon availability





#### **Interoperability of MCS Tools**

- MCS Embedded Applications Binary Interface
  - Facilitates interoperability between tools
  - DWARF-3 debug support for compatibility with debug solutions
- Assembly conversion utility
  - TASKING provides a utility to convert GNU based BOSCH MCS assembly to syntax used by the TASKING assembler
- Advanced Multi-core Linker
  - Supporting a multitude of output formats
    - ELF Facilitates C-level symbolic debugging
    - Intel HEX or Motorola S-record format For PROM programming
    - C-file with C-array Supports "old-style" MCS to CPU interfacing





#### **MCS Tool Support Availability**

- All MCS tools are accessible from within the Eclipse IDE, as well as from the command line
- Full featured C and assembly level debugger
  - Runs within an Eclipse perspective
  - Interacts with fast instruction set simulator
- Support will be included in TASKING VX-toolsets:
  - Freescale Power Architecture Qorivva MPC5xxxx series
  - STMicroelectronics Power Architecture SPC5xx series
- MCS EABI is available from Bosch AE or Altium





### MCS Tool Support Roll-out Plan

- April 2015:
  - Standalone C compiler for GTM / MCS
    - Intermediate solution until integration in the VX-toolsets for the appropriate CPUs
- Q3-4 2015:
  - Integration in the TASKING VX-toolset for Power Architecture

Subject to change. Proposed features may be implemented in earlier intermediate revisions, as well as postponed to later releases or rejected due to lack of customer interest or resources.





#### **TASKING Information**

TASKING is an Altium brand

#### TASKING Development Tools are available from Altium

- Altium is a global organization, with 400+ employees
  - Sales and Support offices in Americas, Australia, China, Germany, Japan
- TASKING product development is located in The Netherlands
- Altium is listed on the Australian Stock Exchange: ASX:ALU
- TASKING Development Tools support many automotive and multimarket architectures from the world's leading semiconductor vendors, including:
  - ARM, Atmel, Cypress, Freescale, Infineon, NXP, ON Semiconductor, Renesas, Silicon Labs, STMicroelectronics, Texas Instruments
- More information: <u>www.tasking.com</u>
  - or: <u>iwantacompiler@tasking.com</u>



And a state of the second s

\* blocking read from

- /= without echo
- /# IE #2
- /# Londad th
- 1= 12 1
- 74 Passes Land



# Thank you for your attention

(1))

Copyright © 2015 Altium BV, Confidential

- CARL