# XMC1000 / XMC4000 Motor Control Application Kit

Getting Started 01 v1.0

**Induction Motor V/F Control App** (ACIM\_FREQ\_CTRL)





# Induction Motor V/F Control App





# Induction Motor V/F Control App





#### Kit composition – XMC 1300 Boot Kit





#### Kit composition – PMSM LV 15W Card





## Kit composition – connection XMC1300





## Kit composition – XMC4400 Enterprise Kit



Copyright © Infineon Technologies AG 2015. All rights reserved.



# Kit composition – General Purpose Motor Drive



ACT connector to CPU Card (e.g. CPU\_44A)



### Kit composition – connection XMC4400





# Induction Motor V/F Control App





#### Development Tool: DAVE<sup>™</sup> version 4

- > DAVE<sup>™</sup> is a free development platform for code generation by Infineon
- The Software package: DAVE<sup>™</sup>, Examples, Videos, Apps, XMCLib... can be downloaded from
- http://www.infineon.com/DAVE
- > This Getting started is based on DAVE<sup>TM</sup> v. 4.1.2





# Induction Motor V/F Control App





# Getting started limitations

- > The following example shows the default usage of the App.
- This Getting Started shows how to create an example with the default settings.
   Only the used App configurations are described. More information about the spectrum of the App can be found in the Help or an Application Note.
- The creation is described in steps. If a step is specific to XMC1300 or XMC4400 it is mentioned in the title and a sub-step e.g. 2.a, 2.b. Variation of the example (e.g. with adjustable speed) based on the main example.
- The following examples based on ACIM\_FREQ\_CTRL/ACIM\_FREQ\_CTRL APP v.
   4.0.5 beta

# Step 1: create new project

- **Open Dave** >
- Select a workspace or use the default workspace >
- Click "OK" >

New

Close Close All

Save

Open File...

File  $\rightarrow$  New  $\rightarrow$  DAVE Project... >

DAVE IDE - DAVE<sup>™</sup> - C:\Workspaces\DAVE-4.1\Motor

File Edit Source Refactor Navigate Project Search Run Da

Alt+Shift+N ► 💣

Ctrl+W

Ctrl+S

Ctrl+Shift+W

Workspa	ice Launcher
Select a v DAVE™ st Choose a	<b>workspace</b> tores your projects in a folder called a workspace. workspace folder to use for this session.
Workspac	e: C:\Workspaces\DAVE-4.1\Motor   Browse
▸ Copy Set	tings
c <mark>h Run DAVE W</mark> i	ndow I-
DAVE Project	
Project	
Example	



Ctrl+N

Example...

Other...

P



#### Step 1: create new project

- Enter project name: e.g. GT\_ACIM\_XMC44\_Example1\_v1\_0 >
- Select "DAVE CE Project" for Project Type >
- Click "Next >" >
- Select your microcontrol >
  - **XMC1300**: XMC1302 \_

- XMC4400: XMC4400
- Click "Finish" >

	W New DAVE Project				
	DAVE Project				
troller:	Create a new C/C++ project for Infineor	n tool chains			
302-TO38X0200	Project Name: GT_ACIM_XMC44_Examp	ple1_v1_0			
400-F100x512	Location: C:/Workspaces/DAVE-4.1/Motor				
New DAVE Project	Project Type:	Tool Chain:			
icrocontroller Selection Page elect the microcontroller for which the project has to be created Microcontrollers Microcontrollers Microcontrollers XMC4000 XMC4000 XMC4000-F100x256 XMC400-F100x	<ul> <li>Infineon Projects</li> <li>ARM-GCC Application</li> <li>Easy Start Project</li> <li>Simple Main Project</li> <li>DAVE CE Project</li> <li>Empty Project</li> <li>ARM-GCC Library</li> <li>Empty Project</li> <li>Show project types and tool chains of</li> </ul>	ARM-GCC Application			
Inter Option  Remove unused sections  untime Library  Ubrary Newlib-nano  Add floating point support for printf  Add floating point support for scanf	? < Back	Next > Finish Cancel			

Copyright © Infineon Technologies AG 2015. All rights reserved.

### Step 2: add APP

> Click "Add New App"



- > Deactivate "Hide beta versions"
- > Enter in search filter "Motor Control"
- Select "ACIM\_FREQ\_CTRL"
- > Click "Add"
- Read the warning regarding beta versions and Click "OK" to confirm.
- Add in a new APP takes a few seconds
- Click "Close" to hide the "Add new APP" window







Copyright © Infineon Technologies AG 2015. All rights reserved.

nfineon

#### Step 3: APP configuration

- Open "ACIM\_FREQ\_CTRL" by double click or right click → "Configure App instance"
- Open "Basic Control Scheme" tab
- Select "FB\_RAMP\_0"
- This will add the AUTOMATION APP. This can take a few seconds.





Infineon

Step 3



#### Step 3: APP configuration

- > Open "Power Board" tab
- Set "Dead time rising edge[ns]" to 1100
- > Set "Dead time falling edge[ns]" to 885

Control Algorithm	Basic Cont	rol Scheme	Control Parameters	Task Scheduler	Power Board	Mea
Power Board Con	figuration –					
DC link voltage [V	]:	24				1 1
Dead time rising e	edge [ns]:	1100		PWM Tim Compare		~
Dead time falling	edge [ns]:	885		value	/	
Switch delay [ns]:		500		High Side	·	Ц
Inverter enable pir	n:	Active Hig	h 👻	PWM		1
Bootstrap time [m	ns]:	0		Low Side PWM		H
Output polarity				Phace	:	
High side switch	nes:	Active Low	· •	Voltage	: →	
Low side switch	es:	Active Low	· ·			
Current Amplifier	Configurati	ion				
VADC reference	[V]:	3.3			Amplifier Bia	5
Rshunt [mOhms]	]:	10		⊒↓	Y	
Amplifier gain:		21			<pre>{</pre>	
		45.74.4005			1	

#### Step 4: Pin assignment

- > The pin allocation can be done in two ways:
  - 1) table view 🚺

Filter PWM_SVM_0 ·			
APP Instance Name	APP Pin Name	Pin Number (Port)	
# PWM_SVM_0			
	PhaseU_High Pin	Not Selected	*
	PhaseV High Pin	Not Selected	
	PhaseW High Pin	Not Selected	*
	PhaseU Low Pin	Not Selected	*
	PhaseV Low Pin	Not Selected	
	PhaseW Low Pin	Not Selected	
	Trap Pin	Not Selected	-
	Inverter Enable Pin	Not Selected	-
		Not Selected	-
		#17 ( P0.0 )	
		#18 (P0.1)	
		#19 ( P0.2 )	
		#20 ( P0.3 )	
		#21 ( P0.4 )	-
		#22 ( P0.5 )	
		#23 ( P0.6 )	
		#24 ( P0.7 )	
		#27 ( P0.8 )	
		#28 ( P0.9 )	
		#29 ( P0.10 )	
		W30 (P0.11)	
		#31 (P0.12)	
		#32 (P0.13)	
		#33 (P0.14)	
		#34 (P0.15)	
-		#16(PL0)	*







# Step 4: Pin assignment- table view

The Pin Allocation can be done in two ways:

- Table view: >
  - Click "Manual Pin Allocator"



- Table: select the corresponding pin for each pin
- Click "Save" \_

File	Edit	Navigate	Search	Project	Run	DAVE	Window	He
		28%	<b>8</b>	* 🗠	0	9	2 🌒 🔅	•
<b>E</b> (	/C++	Projects 🛛	Proje	ect Explor	er 🚺	lanual P	in Allocato	or

iter PWM_SVM_0 -			
			E
APP Instance Name	APP Pin Name	Pin Number (Port)	
PWM_SVM_0			
	PhaseU_High Pin	Not Selected	*
	PhaseV High Pin	Not Selected	-
	PhaseW High Pin	Not Selected	-
	PhaseU Low Pin	Not Selected	-
	PhaseV Low Pin	Not Selected	+
	PhaseW Low Pin	Not Selected	-
	Tran Pin	Not Selected	-
	Inverter Enable Pin	Not Selected	-
	Inverter Endore Fin	Not Selected	<b>^</b>
		#17(P0.0)	
		#18 ( PO.1 )	
		#19 ( PO.2 )	
		#20 ( P0.3 )	=
		#21 ( P0.4 )	-
		#22 ( P0.5 )	
		#23 ( P0.6 )	
		#24 (P0.7)	
		#28 ( P0.9 )	
		#29 (P0.10)	
		#30 (P0.11)	
		#31 (P0.12)	
		#32 (P0.13)	
		#33 (P0.14)	
		#34 ( P0.15 )	
-		#16 ( P1.0 )	*



# Step 4: Pin assignment- graphical view

- Graphical view:
  - Click "Pin Mapping Perspective"
  - Select pin in the left table
  - Right click on a colored pin
  - Click "Assign"

🦆 DAVE CE - DAVE™ - C:\Wo	orkspaces\DAVE-4.1\Motor
File Edit Navigate Searc	h Project Run DAVE Window Help
🚊 🛍 🔺 🔁 🌌 🌽 👌 🕯	★ ≥ =
🗟 C/C++ Projects 🖾 🏠 Pro	oject Explorer 🔅 🗢 🖗 Pin Mapping Perspective
GT_ACIM_XMC44_Ex	ample1_v1_0 [ Active - Debug ]
FinMapping - DAVE™ - C:\W	orkspaces\DAVE-4.1\Motor
File Edit Navigate Search	Project Run DAVE Window Help
🔁 🕶 🖬 🖬 🛍 🗠 🎽 🖄	/> R 👩 🎋 👈 🚍 🗊 🗷 🛛 🕏 🔷 🌣 🕶 🚾
👯 Virtual Pin View 🛛 🖓 🗖	# Package View
Virtual Pin List	
PWM_SVM_0	
Invertor Enable Din	P2.4 1
PhaseU_High Pin	$p_{25} = 2$
PhaseV High Pin	$P_{2,5} = 2$
PhaseV_Low Pin	P2.7 = 4
PhaseW_High Pin	P2 8 5
PhaseW_Low Pin	$P29 \square 6$
Trap Pin	P2 10
	Assign 130
	D1 C 11

Note: See legend color code for additional information



# Step 4a: Pin assignment - XMC1300

APP Instance Name	APP Pin Name	Pin Number (Port)	
PWM_SVM_0			
	PhaseU_High Pin	#17 ( P0.0 )	Ψ.
	PhaseV_High Pin	#24 ( P0.7 )	Ψ.
	PhaseW_High Pin	#27 ( P0.8 )	Ψ.
	PhaseU_Low Pin	#18 ( PO.1 )	*
	PhaseV_Low Pin	#23 ( P0.6 )	Ψ.
	PhaseW_Low Pin	#28 ( P0.9 )	Ψ.
	Trap Pin	#31 ( P0.12 )	~
	Inverter Enable Pin	#30(P0.11)	<b>~</b>





# Step 4b: Pin assignment- XMC4400

	APP Pin Name	Pin Number (Port)		
PWM_SVM_0				
	PhaseU_High Pin	#97 (P0.5)	*	
	PhaseV_High Pin	#98 (P0.4)	*	
	PhaseW_High Pin	#99 ( P0.3 )	~	
	PhaseU_Low Pin	#100 ( P0.2 )		
	PhaseV_Low Pin	#1 ( PO.1 )		
	PhaseW_Low Pin	#2 ( P0.0 )	*	
	Trap Pin	#89 (P0.7)	*	
	Inverter Enable Pin	#68 (P1.15)	*	
			,	P0.2 P0.5 P0.5 P0.5 P0.12 P0.12 P0.7 P0.7 P0.7 P0.7 P0.7 P0.7 P0.7 P0.7
0		Deret	Close	01 5 8 6 6 9 5 7 5 5 7 5 8 6 8 8 8 8 8 8 8 9 9 9 9 9 9 9 9 9 9 9
)		dve	Close	P0.0 3 P0.9 4
the second second second				P31 6 P30 7

P2.5 P2.4 P2.3 PORS VDDO VSSO XTAL XTAL VDDF

#### Step 5: Generate code

- > Click "Generate Code"
- Code Generation can take a few seconds.

۵ 🦆	DAVE C	E - GT_ACIN	1_XMC44	00_Exam	ple1_v	1_0/Da	/e/Model//	APPS/A	CIM_F	REQ_CT	RL/v4_0_	5/Uimodel
File	Edit	Navigate	Search	Project	Run	DAVE	Window	Help				
R	6 8	22%	D (0)	<b>.</b> 19 9.	198 - j	. <del>1</del> . 1	<b>1</b> 11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	<b>t</b> 🚵	= 0	•	8	\$≉ <b>•</b> [ <u>k</u> ] •
品	C/C++	Projects 22	Proje	ect Evolor	er	¢		1 😘	~ -		Gener	ate Code



# Step 6: Add function



 Edit main.c by adding the following function call: ACIM\_FREQ\_CTRL\_MotorStart(&ACIM\_FREQ\_CTRL\_0);

```
23 int main(void)
24 {
25
     DAVE STATUS t status;
26
627
     status = DAVE Init();
                                     /* Initialization of DAVE APPs */
28
29
     if(status == DAVE_STATUS_FAILURE)
30
     {
31
       /* Placeholder for error handler code. The while loop below can be replaced with an user error handler. */
32
       XMC_DEBUG("DAVE APPs initialization failed\n");
33
34
       while(10)
35
       {
36
37
        }
38
     }
39
40
     ACIM_FREQ_CTRL_MotorStart(&ACIM_FREQ_CTRL_0);
41
     /* Placeholder for user application code. The while loop below can be replaced with user application code. */
42
     while(10)
43
     {
44
45
     }
46 }
47
```

#### Step 7: Build project



> Build Project



File	Edit	Navigate	Search	Run	Project	DAVE	Window
	R L	201	C 15	1	a 📕 🙃		2 🔹 🎋 🔹
6	C/C++	Pro Build A	ctive Proj	ect kp	olorer	¢	⇔ @   ⊡

#### Step 8: Debug – create debug session



6

- > Click "Debug":
- > Double click "GDB SEGGER J-Link Debugging" Sector Sec
- > Click "Debug"
- The debugger is downloading the program

(See next slide)

ate manage and run configurations		Create, manage, and run configurations					
ate, manage, and run comigurations		)					
er matched 1 of 18 items	Configure launch settings from - Press the 'New' button to - Press the 'Duplicate' button - Press the 'Delete' button - Press the 'Filter' button to - Edit or view an existing of Configure launch perspective spage.	Section 25 - Constraints of the section 25 - Constraints	Name: GT_ACIM_XMC44_E Main © Debugger) Project: GT_ACIM_XMC44_Exampl C/C++ Application: Debug/GT_ACIM_XMC44_ Build (if required) before Build configuration: Sele © Enable auto build © Use workspace setting	xample1_v1_0 Del > Startup > Sour e1_v1_0 Example1_v1_0.el Variables launching sc Cc	bug ce TC Common f Search Project ) Disable auto build onfigure Workspace Se	Browse., Browse., • ttings.,	
)		Filter matched 2 of 19 items			Apply	Revert	



DAVE CE - GT\_ACIM\_XMC44\_Example1\_v1\_0/Dave/Model/APPS/ACIM\_FREQ

File Edit Navigate Search Project Run DAVE Window Help



#### Step 8: Debug – start program

- > Switch to debug perspective. Confirm with "YES"
- > To start the program click "Resume (F)



File	Edit	Source	Refactor	Navigate	Search	Window
	6.0		I 🔲 🕅 🕄	. 👁 i•	表式	۰ ا



#### Behavior

> The Motor slowly ramps up to 1500rpm





## Induction Motor V/F Control App





# Getting started limitations

- > The following example shows the default usage of the App.
- This Getting Started shows how to create an example with the default settings.
   Only the used App configurations are described. More information about the spectrum of the App can be found in the Help or an Application Note.
- The creation is described in steps. If a step is specific to XMC1300 or XMC4400 it is mentioned in the title and a sub-step e.g. 2.a, 2.b. Variation of the example (e.g. with adjustable speed) based on the main example.
- The following examples based on ACIM\_FREQ\_CTRL/ACIM\_FREQ\_CTRL APP v.
   4.0.5 beta
- > Example 2 with adjustable speed based on example 1. Only the delta is discribed in this cheptar. The target speed is selected by adjusting the potentiometer.

#### Step 1: APP configuration



- > open "ACIM\_FREQ\_CTRL" by double click or right click → "Configure App instance"
- Open the "Measurements" tab
- Click "Enable speed set via analog input"
- This will add the ADC APP. This can take a few seconds.

Control Algorithm	Basic	Control Sche	me	Control Paramete	rs   Task Sch	eduler	Power Board	Measurements	Err
Measurement									
Current measurer	ment:	None					Ψ.		
Enable over current detection									
Enable voltag	e comp	pensation							
V Enable speed	set via	analog inpu	t						
ADC Carfin metia									
Enable measurem	n nent	Request sou	rce	Queue position	Refill	Exter	nal trigger		
I_Average		Queue A	Ŧ	0	Enable	E	nable		
V_DCLink		Queue A	Ŧ	1	Enable	E	nable		
Analog_Inp	ut	Queue A	Ŧ	2	✓ Enable	E	nable		
User_Define	ed 🛛	Queue A	Ŧ	3	Enable	E	nable		





## Step 1: APP configuration – XMC4400

The V/f control is less efficient than FOC control. To reduce the maximum power consumption the default values is be changed. This only applies to **XMC4400** kits.

- Open the "Control Parameters" tab
- Reduce "No load speed [rpm]" to 2000
- > Enable "User defined"
- > Set "V/f constant" to 70
- > Set "V/f offset" to 300

Control Algorithm	Basic Control S	Scheme Control Parameters		ters	Task Scheduler	Power Board	Measurem	ents	Error Handler	Interrupt Settings	
Control Panel Para	ameters					Motor Paramet	ers				
Motor direction:	[	Clockw	ise		-	Nominal voltag	e [V]:	24			
User speed set [rp	om]:	1500				No load speed [	[rpm]:	200	)		
Over current limi	t [mA]:	500				Pole pair:	4				
Maximum voltag	e limit [%]:	100									
V/f Configuratio	on	Defau	lt 🗸	User o	lefined						
V/f constant [m	nV/Hz]:	180	70								
V/f offset [mV]:	: [	1200	300								

Copyright © Infineon Technologies AG 2015. All rights reserved.

# Step 2: Pin assignment

- > Assign the ADC pin in table or graphical view:
  - 1) table view 💷

Filter PWM_SVM_0 •				
APP Instance Name	APP Pin Name		Pin Number (Port)	
# PWM_SVM_0				
	PhaseU_High Pin		Not Selected	*
	PhaseV High Pin		Not Selected	
	PhaseW High Pin		Not Selected	
	PhaseU Low Pin		Not Selected	
	PhaseV Low Pin		Not Selected	
	PhaseW Low Pin		Not Selected	
	Trap Pin		Not Selected	-
	Inverter Enable Pin		Not Selected	-
			Not Selected	~
			#17 ( P0.0 )	
			#18 (P0.1)	
			#19 ( P0.2 )	
			#20 ( P0.3 )	
			#21 ( P0.4 )	-
			#22 ( P0.5 )	
			#23 ( P0.6 )	
			#24 (P0.7)	
			#27 ( P0.8 )	
			W28 (P0.9)	
			W29 (P0.10)	
			#30 (P0.11)	
			#32(0012)	
			#33 ( P0.14 )	
			#34 (P0.15)	
			#16(PL0)	-
(2)		Chief.	Porot	Closes







Note: Pin assignment is explained in example1 step 4







# Step 2a: Pin assignment - XMC1300

> Allocate the "Analog\_Input pin" to the potentiometer input pin

			E F
PP Instance Name	APP Pin Name	Pin Number (Port)	
ACIM_FREQ_CTRL_0			
	Analog_Input pin	#2 ( P2.5 )	*
PWM_SVM_0			
	PhaseU_High Pin	#17 ( P0.0 )	*
	PhaseV_High Pin	#24 ( P0.7 )	
	PhaseW_High Pin	#27 ( P0.8 )	-
	PhaseU_Low Pin	#18 ( P0.1 )	*
	PhaseV_Low Pin	#23 ( P0.6 )	*
	PhaseW_Low Pin	#28 ( P0.9 )	*
	Trap Pin	#31 ( P0.12 )	*
	Inverter Enable Pin	#30 ( P0.11 )	*





# Step 2b: Pin assignment- XMC4400

> Allocate the "Analog\_Input pin" to the potentiometer input pin

			(□	Ē
APP Instance Name	APP Pin Name	Pin Number (Port)		
PWM_SVM_0				
	PhaseU_High Pin	#97 ( P0.5 )	*	
	PhaseV_High Pin	#98 ( P0.4 )	*	
	PhaseW_High Pin	#99 ( P0.3 )	*	
	PhaseU_Low Pin	#100 ( P0.2 )	~	
	PhaseV_Low Pin	#1 ( P0.1 )	~	
	PhaseW_Low Pin	#2 ( P0.0 )	Ψ.	
	Trap Pin	#89 (P0.7)	*	
	Inverter Enable Pin	#68 ( P1.15 )	*	000 000 000 000 000 000 000 000
				P0.0 = 7 P0.0 = 7 P0.0 = 3 P0.9 = 4 P3.2 = 5 P3.1 = 6 P3.0 = 7 P3.0 = 7 P3.0 = 7

P14.15 P14.14 P14.13 P14.12

# Step 3: Generate, build, debug

- > Repeat following steps from example 1:
  - Step 5: Generate code
  - Step 7: Build code
  - Step 8: Debug



7



# infineon

#### **Behavior**

- > The target speed is selected by potentiometer
- > The target speed can vary from 0rpm to "No load speed"
- Motor slowly ramps up or down to the target speed





# Induction Motor V/F Control App





#### App help

This will show helpful information regarding to the APP:

- Right click on ACIM\_FREQ\_CTRL\_0
- > Select "App Help"
- > This will show the help contents this App

	а0 <b>ж</b>
e Locate Back Torward Stop	LE 1.n." A& CCP7 LET Refresh Home Fort Part Conce
ents   Igdax    Search   Favorites	ACIM_FREQ_CTRL
Apps = () Ucense Terms and Copyright Inform = () Ucense Terms and Copyright Inform	Home
Abbreviations and Definitions     Overview     Disordiary	Overview
Architecture Description     APP Configuration Parameters     Enumerature	Overview
	ACIM_EREQ_CIRL APPI implements open loop V/f control algorithm to drive three phase AC Induction Hores. This is scalar control technique which involves controlling the magnitude of voltage or frequency of the induction motor. Constant V/f method maintains the too constant flux density by changing the voltage in programmeter of the V/f control. This APP also facilitates the use of the AUTOMATION APP for runtime parameter checking, error logging and connecting to the Ramg Generator Function Block. It uses MOTOR_LIB which is math library used for common algorithms.
	The ACIM_FREQ_CTRL APP provides the following features:
	1. Basic Control Scheme: This includes mandatory V/F control algorithm parameters. 2. Avavanced Control Scheme:
	This gives option for adding offset values, inserting user code via call back function and Position control algorithm. 3. Task Scheduler:
	Lasks can be scheduled from various othe sources, and cask execution time can be adjusted. 4. State Machine: Motor state machine controls the execution of the motor control algorithm. This APP also supports Drive State Machine (optional) which interacts with the external system and decides the motor state.
	<ol> <li>Measurements: It supports the average current measurement (overcurrent protection), DC link voltage measurement (Voltage compensation) and analog speed control measurement.</li> <li>Parameter Monitoring:</li> </ol>
	Romane secyse of one motor parameters. 7. Error Handler: Logging the error, warnings reported by the APP which can be communicated to the central system. 8. Ram Drundton Block Connection:
	Speed of the motor can be ramped in linear or s-curve by connecting the user end speed to the ramp function block.
	Hardware and software connectivity of APP Figure 1, shows how the APP is structured in DAYE. The LLD layer provides abstraction for these hardware modules. Control algorithm is built on top of the basic building blocks like PVIM (PWM_SVM) and ADC (ADC_QUEUE). It makes use of low level drivers for CCU8, ADC, SCU and GPIO.
.11. 1	user application (main.c) USER CODE





## Where to buy - XMC1300

<b>Development Boards</b>	5	Order Number
XMC1300 Boot Kit		KIT XMC13 BOOT 001
XMC1000 Motor Control Application Kit		KIT XMC1x AK Motor 001



#### Where to buy – XMC4400

<b>Development Boards</b>	Order Number
XMC4400 Enterprise Kit	<u>KIT XMC44 EE1 001</u>
General Purpose Motor Drive Kit	KIT XMC4x MOT GPDLV 001
XMC4400 Motor Control Application Kit	<u>KIT XMC44 AE3 001</u>



# General information

- Information about all available XMC Motor Control Application Kits:
   <u>LINK</u>
- For latest updates, please refer to: <u>http://www.infineon.com/xmc1000</u> <u>http://www.infineon.com/xmc4000</u>
- > DAVE<sup>™</sup> development platform: <u>http://www.infineon.com/DAVE</u>
- > For support:

http://www.infineonforums.com/forums/8-XMC-Forum



Part of your life. Part of tomorrow.

