

#### Hands-on Workshop: MQX<sup>™</sup> Lite APF-ENT-T1286

Susan Su IMM FAE

October 2013

Treasons, the Pressoale loga, ARWA: C-3, ColeXIST, DoaMMerris, CrebPris, Centhres, F. Weis, the Immige ERG end Schumen loga, Network, working PR 20, Paver 2004/DCC, Presidence Depert, Dord O, Colvon, Schlebreich, He Jackswan loga, Sterices, Symphering end Nat XXa are transmiss for financial Schlebreichams, Pre-Big, US-Pat. & Tri, OH, Almos Deeth, Riedlack, Cashine, Flexis, Lywincase, MagaN, MOC, Riedbrin III on Pockase, Canit D Dervers, G 2015, Chingin, Baay Proj. 2014/2015, Town: Tablicha Aytherin and Networks are trademarks of Freescole Sonocombacto, Inc. All other postation or service names are the preparety of their imagention owners, IC 2013 Thansale Servicesbactory, Inc. All other postation of the postation of the preparety of their imagention owners, IC 2013 Thansale Servicesbactory, Inc. All other postation, Inc. 2016, Distribution, Inc. 2016, Distri





#### **Agenda and Intro**

- A brief intro to MQX Lite
  - Overview, Main features and Code Size
- Real work: hands-on labs
  - Create a new MQX-Lite project, add ConsoleIO and BitIO components
  - Create tasks, watch the flashing lights







#### Abstract

 You will use Processor Expert (PEx) to configure MQX Lite (a PEx component) and initialize multiple tasks on the Kinetis E series processor. Light the LED and print message from UART.



Pressure, the Freesolds logs, AVNex, D.S. Color/EEF, CadeMarris, Diriffris, ColeFris, C.Wass, Intellight, Freesold, Konta, and Addit, PDD, PrevenDUCC, Processor Rayer, Card, Carris, EarthAssai, the SateMaraer logs Stachure, Symptrate and Vordia and statemarks of Freezold Interconductor, the Antar, Stelfs, BedStack, Conduct, Rass, Layersage, Mayor, WICC, Partner, a Factory, Card, Convey, D.YCC, Enger, Rady Fact, Stelfs, Personal and Tamica and Additional Interconductor and Tamica and Additional Sectionations (in Vision). Vision and Tamica and Administration of Freezold Interconductor, for Advisor, and the section of the Interconductor and Sectionations (in Technology, Vision).

### Real Time Operating Systems

- A real-time operating system (RTOS) manages the time of a microprocessor or microcontroller
- Features of an RTOS:

□Allows multi-tasking

□Scheduling of the tasks with priorities

□Synchronization of the resource access

□Inter-task communication

□Time predictable

□Interrupt handling



Presents, the Freenants logs, MVNex, D.S., Code/EEF, OpdeRasino, OrdErix, OxdErixe, Todaya, the Energy Ethient Solutions legs, Nexts, mobile(2), PDS, Preer/QUCC, Processor Disper, QuOD, Darins, Earlehause Ing, StarCare, Styratory and Vorsillo and unablendor of Freezinki Benconductor, too, Reg. U.S. Pot, S.P., Ott, Antara, Beel/R. Bedlick, CareWar, Beill, Layerough, Heyer, WCP, Felderer e # Pentage, Ord Dishere, DuOD Etyper, Beel M. War(WDS), Three, Tubolek, Vyther and War(WDS), Three, Tubolek, Vyther, Tubolek, Versilla, Starkov, Starker, Felder, March MCP, Berlander e # Pentage, Ord Dishere, DuOD Etyper, Beel M. War(WDS), Three, Tubolek, Vyther, and Threec an stadenosts of Freezink Derivorductor, Inc. 30 offer product or server annia are to property within respective wave. O 2013 Freezink Derivorductor, Inc.

# General Impact of MQX RTOS

Adding ~450 new users every month







#### Very light MQX kernel for resource-limited MCUs

- Targeted at the Kinetis L family initially
- Packaged as a Processor Expert component
- I/O capability provided by Processor Expert
  - USB via FSL bare-metal stack, also a Processor Expert component
  - No file access
- Programming model allows upward code migration
  - Code built with MQX Lite should move to full MQX RTOS easily



#### NP MQX Lite – Main Features

#### Scheduler

- Priority pre-emptive schedule
- Support for lightweight semaphore, and mutex (with polling)
- Task Management will not support dynamic task creation
   All task resources allocated at compile time
- Lightweight events and messaging only
- Dynamic memory management not allowed
- Lightweight timer included (one shot, and periodic notification)



#### NP How Small Is MQX Lite?

- Minimal App Hello Task, Idle task, interrupt stack
  - Code = 10.4K
  - Data = 3.7K (including 1.5K for stacks)
- Typical App 7 tasks + idle, lightweight events, queues
  - Code = 27K
  - Data = 10K (5K for stacks)
- Your mileage will vary



hearcons, the Franceski logs, AMNo, D.S., Cole/EDF, Cadultance, OHFins, OHFins, Differe, Differe, Differe, Balance Iog, Kineta, Intellight, P.S., Prewold, C.C., Processor Ecsen, Corti, Johns, Balakasani, Kai Salakasani Kogi, SauCani, Bandraray and Vintila variabinatica of Franceski Bancconductor, no., Reg. U.S. Pat. S.T., DR. Anna, Shelli, Badoka, Contek, Enseis, Layanopa, Lawy, MCC, Paterione a Pantaga, Cartil Generga, U.O.C. Expire, Basis, Paterio, Balanco, S., Franceski, A., Salari, Salari, Badok, Paterio, Ress, Layanopa, K., Hochek, Nyetti and Etrinic are statements of Franceski Bancooductor, Inc. Al other product or earlier names and the graperity of their suspective averas. O 2013 Franceski Servicenductor, Inc.

#### MQX Lite and Processor Expert Integration

- MQX Lite delivered as an RTOS adapter
  - Interrupt mechanism in MQX is unchanged
  - Processor Expert LDDs work with the RTOS
- The entire I/O from standard MQX removed

- I/O provided by LDD components

- Set up and configure tasks in Component Inspector
- Easy to add MQX Lite to existing app
   Just drop in the MQX Lite component





# Enough Talk, Time to Work

- Board for work FRDM-KE02Z
  - Freescale Freedom development platform, ideal for rapid prototyping.

- Featuring a Kinetis E series MCU, the industry's first 5-volt MCU built on the ARM® Cortex<sup>™</sup>-M0+ core.

- MKE02Z64VQH2 - 20MHZ, 64KB Flash, 4KB SRAM, 64QFP

- Easy access to MCU I/O

- OpenSDA debug interface, easy to download and debug applications.





MCU with limited resources for the application, Ideal to verify MQX Lite !



rescale, the Freesole logs AVVex, C.S., Color/EEF, Color/Fan, Colo

# Enough Talk, Time to Work

- Work for today
   IDE: CodeWarrior 10.5
  - Create a workspace
  - Create a new MQX Lite project
  - Add and configure ConsoleIO component to send messages through UART
  - Add and configure three BitIO components to control the tricolor LED
  - Configure the MQX Lite component
  - Define tasks and write the code
  - Build it and run it watch the blinking lights



#### Set CodeWarrior to a new workspace

- This makes sure we don't have any confusion
- File→Switch Workspace→Other

🥬 Workspace Launcher	<b>×</b>
Select a workspace	
CodeWarrior Development Studio stores your projects in a folder called a workspace. Choose a workspace folder to use for this session.	
Workspace: E:\MQXLiteWorkspace	Browse
Copy Settings	
ОК	Cancel

- Type in E:\MQXLiteWorkspace
- For the sake of others, turn off "Use this as the default..."
- Then you can hide the Welcome Screen that appears.



Presents, the Freerole logs, Milvio, D.S., Color/EEF, OsbiHanice, Ord/File, Osb/Files, O.Hara, the Energy Ethinet Solations legs, Kawas, endedolf, PEG, Preve/DICC, Processor Equipt, 2010, Damas, EstMassar, Int Statificanae logs, EstCare, Strattering and Vorsilla van salematik of Freecola Exercicedulato: the , Hey U.S. File S.T., Dit. Antar, Berlit, Berlitot, Carelon, Fast, Layersage, Magril, WIC, Pathorn is Pantage, Quint Genovage, DUCC Engin, Range free, Statificana Scholaro, Vysnil and Estimic an Internet Internetion Environmentary, Int. 30 one product or annual and to protein the Internetion Environmenta Scholarozovice lines. D 2011 Freedoat Scholarozovice lines

#### Create a new MQX Lite project

**~** · · · ·

<ul> <li>Click Import Project in Commander view</li> </ul>	
	Devices
<ul> <li>File→new→MQX-Lite Project</li> </ul>	Select the derivative you would like to use
12 New MOX-Lite Project	Device to be used:
Create an MQX-Lite Project	▲ Kinetis E Series
Choose the location for the new project	KEUX Family     KEU27 (20 MHz) Family
	MKE02Z32
	MKE02Z64
Project name: Mqx_starter	Kinetis K Series
	Kinetis L Series
Use default location	
Location: E:\MQXLiteWorkspace\Mqx_starter Browse	
	Creates project for MKF02764 (20 Mbz) derivative
	ereates project for mileoczow (zo mile) derivative
() Canad	
Cancel	
	-
	· · · · · · · · · · · · · · · · · · ·
	Cancel
	Presents, the Freesede logs, Athlian, C.S., Code/TEST, Code/Marrier, Cold/Fire, Cold/Fire, C.Ware, InvErways Efficient Solutions legis, Kiteste, incolneCT, PEC, PowerCIRCC,

New MOX-Lite Project



Processor Right, Cortl, Darina, Elektroni, Intel Metanov, Springer, and Schuler, Springer, and Korta and Schuler, Schule

### Create a new MQX Lite project

Choose openSDA as connection interface

🥬 New MQX-Lite Project	- • •
Connections	
Choose the connection to use for this project	
Connection to be used:	
P&E USB MultiLink Universal [FX] / USB MultiLink	
P&E Cyclone MAX	E
P&E TraceLink	
Open Source JTAG	
🗹 OpenSDA	
Segger J-Link / J-Trace / SWO (SWD based)	
Connect to OpenSDA.	^ _
Sack Next > Finish	Cancel



# Open Processor Expert project

- This takes a few second on first launch
- Double click the .pe file
- The Components view opens

🔂 CodeWarrior Projects 🛛	
	5 🔎
File Name Buil	d
a 🚰 Mqx_starter : FLASH	
Documentation	
🗁 FLASH	
b 🔁 Generated_Code	
D Dev MQXLITE	
NrocessorExpert.pe	
🕞 Project_Headers 🛛 🧡	
Project_Settings	
b > > Sources	
Dou	ble
Dou	ble
Dou	ble k
Dou Clic & Components - Mqx_starter	ble k □
Clic Clic Clic Clic Components - Mqx_starter	ble k □ ☎ ₽
Clic Clic Clic Clic Clic Clic Clic Clic	ble k ⊫ ≌ ≞
Clic Clic	ble k ि 🛱 🖥
Clic Clic	ble k ⊫ ≅ ∎
Clic Components - Mqx_starter & Generator_Configurations FLASH Coss MQX1:MQXLite Processors	ble k ⊡ ≌ ≞
Clic Components - Mqx_starter ☆ Generator_Configurations FLASH CSS MQX1:MQXLite Processors Cpu:MKE02Z64VLH2	ble k ⊫ ≅ ₿
Clic Components - Mqx_starter ☆ Clic Clic Components - Mqx_starter ☆ Configurations Con	ble k ⊫ ≌ ∎



Presents, the Freenake logs, Mirkey, D.S., Code/EST, Cade/Harrier, Old/Fire, Ox/Fires, O.Mara, Ine Grangy Ethiant Soldions legs, Kenta, model/07, PDS, PreverQUCC, Processor Gaper, ZorD, Carina, Earlehauer, Inis Sate/Auare Ing, StatCare, Straffer y and VortiSa are statisticated of Freezok Barcconductor, too, Eng. U.S. Fire, S.T., Ott Antar, Stelfit, BedSach, Carelon, Fast, Layersage, Mager/, MRC, Pathor et a Partage, OxfC Gonerage, OxfC Cargon, Rady Foy, SMX19005, Trave, Tubolcek, Vand and Timic are balancessor Elemenol. Statistics, Inc. 30 One product or advoce name are to proper, at the respective areas or 2011 Freedox Soldionaries, Inc. 2011 Freedox Freedox

# Examine CPU Component

- Provided and configured in starter project
- Select CPU, go to component inspector, look at System Clock
- 16.7MHZ

E CodeWarrior Projects 🛛 🗖		Scomponents Library Scomponent Inspec	tor - Cpu 🔀
🎛   🛃 📄 🔄 🔑	~	Properties Methods Events Build options	Resources
File Name Build	-	Name	Value
⊿ <sup>™</sup> Mqx_starter : FLASH		CPU type	MKF02764VOH2
b 🐉 Binaries	н	Clock settings	Mine De Lot V Que
Documentation		▲ Internal oscillator	
FLASH		Slow internal reference clock [kHz]	32.768
b 🧀 Generated_Code		System oscillator	Disabled
MQXLITE		▲ Clock source settings	1
N ProcessorExpert.pe		▲ Clock source setting 0	
Project_Headers	Ŧ	ICS settings	
<ul> <li>&gt;</li></ul>		Initialization priority	interrupts enabled
		Watchdog disable	yes
🕒 Commonwetz May status 😒 🛛 🗖 🛱 🔊 🛡	н	CPU interrupts/resets	
to. Components - Miqx_starter 23		▲ Clock configurations	1
<ul> <li>▲ Generator_Configurations</li> <li>▲ FLASH</li> <li>▲ OSs</li> <li>▲ MQX1:MQXLite</li> <li>▲ Processors</li> <li>▲ Cpu:MKE02Z64VQH2</li> </ul>		⊿ Clock configuration 0	
		▲ Clock source setting	configuration 0
		ICS mode	FEI
		⊿ System clocks	
		Core clock	16.777216
		Bus clock	16.777216
Components	-		
<i>≸ freescale</i> ™		16 Processor Reserved Config. Sonna, Estebasiani, Na Cabilizzari Koji, Schlörer Spran Arian, Savidi, Beetler, Carillo, Tarian, Layeringer, Magrid, MC, Patrieri et al. est Tritola va patientia et l'Estebasia Services (Estebasia).	any and Yorilla are trademate of Freetak Terrocodutter, to , Reg. U.S. Par. 87n. Off antige, Garroll Converge. QUICE Empre. Ready Rey, Statistical Services, Tetolute, Vytml we name are the streams of their neocotice areasy. C 2011 Freetake Servicestoper Inc.

#### Add and configure ConsolelO component

- Open Components Library
- CPU External Devices -> Display -> ConsoleIO
- Double click the component to add it into project

The CodeWarrior Projects	 Components Library S2 🔊 *Component Inspector - CsIO1
File Name Build	Categories Alphabetical Assistant Processors
<ul> <li>Mqx_starter : FLASH</li> <li>Documentation</li> <li>FLASH</li> <li>Generated_Code</li> <li>MQXLITE</li> <li>ProcessorExpert.pe</li> <li>Project_Headers</li> <li>Project_Settings</li> <li>Sources</li> </ul>	Component
<ul> <li>Components - Mqx_starter Image: Components - Mqx_starter Image: Configurations</li> <li>FLASH</li> <li>Coss</li> <li>MQX1:MQXLite</li> <li>Processors</li> <li>Cpu:MKE02Z64VLH2</li> <li>Components</li> <li>CsIO1:ConsoleIO</li> <li>PDD</li> </ul>	<ul> <li>EVM</li> <li>EVM</li> <li>CPU Internal Peripherals</li> <li>Logical Device Drivers</li> <li>Operating Systems</li> <li>SW</li> </ul>
escale "	Filtering disabled, active project Mqx_starter Artar; BwRik BwStack, Carviar, Plate, Layences, Marris, WRC, Parfers e a Package, Card Onivega, CUCC Engine, Ready Per, SWMTMOR, Trave-

respective awners. © 2013 Freescale Samisonaturatic Inc.

Add and configure ConsolelO component

🖬 CodeWarrior Projects 🛛 🗖 🗖	
File Name Build	
<ul> <li>▲ Solution</li> <li>▷ Documentation</li> <li>▷ FLASH</li> <li>▷ Generated_Code</li> <li>▷ MQXLITE</li> <li>♥ ProcessorExpert.pe</li> <li>▷ Project_Headers</li> </ul>	
<ul> <li>Project_Settings</li> <li>Sources</li> </ul>	
<ul> <li>Components - Mqx_starter X</li> <li>Generator_Configurations</li> <li>FLASH</li> <li>Components</li> <li>MQX1:MQXLite</li> <li>Processors</li> <li>Cpu:MKE02Z64VLH2</li> <li>Components</li> <li>SciOl:ConsoleIO</li> <li>IO1:Serial_LDD[ConsoleIO\ConsoleIO_Serial_LDD]</li> <li>PDD</li> </ul>	

Components Library Stomponent Inspector - IO1 & Properties Methods Events				
Device	UART1	UART1		
Interrupt service/event	Disabled			
⊿ Settings				
Data width	8 bits			
Parity	None			
Baud rate	115200 baud	116508.444 baud		
⊿ Receiver	Enabled			
RxD	PTC6/UART1_RX	PTC6/UART1_RX		
▲ Transmitter	Enabled			
TxD	PTC7/UART1_TX	PTC7/UART1_TX		
Auto initialization	yes			



Presents, the Freetook logs, AVNvs, C 3, Cob/TEST, CadeMarcin, ColdFire, Col

Logical Device Drivers -> Port I/O -> BitIO\_LDD •



Presecute, the Freesole logs. ADVec, C.S. Code/ESF, Code/Fast, Code/Fast, C.Mare, IN-Energy Ethilent Solutions legs, Xineta, mobile/37, PSG, PreveGUCC. Processor Expert. Confil. Gorina, EditAcaute, the Solid-Acaute loss StaCare Scrutture and Vorilla are trailerated of Freedoate Sereconductor, to: Res. U.S. For & 7m. Off Antari, Beef R, BeeSteck, CareNet, Fleets, Layersnaps, MagnV, MRC, Parkner, e. a Package, GorG Ocnverge, GUICC Empre. Ready Play, SWARWOS, Trave, TurboLok, Vytmit and Ethtels are badenorie of Pressue Remiconductor, Inc. A2 other product or solves names are the progenty of their respective evenes. C 2011 Pressue as Service during the

• Rename the component BLUE and configure it.

CodeWarrior Projects 🔀		Scomponent Inspector - BLUE	🖇 🗞 Components Library
		Properties Methods Events	
File Name Build		Name	Value
KE-MQXLite : FLASH		Pin for I/O	PTE7/FTM2_CLK/FTM1_CH1
Binaries		Direction	Input/Output
Documentation		⊿ Initialization	
FLASH		Init. direction	Output
		Init. value	1
		Auto initialization	yes
	-		
<ul> <li>Components - KE-MQXLite S</li> <li>Generator_Configurations</li> <li>FLASH</li> <li>OSs</li> <li>MQX1:MQXLite</li> <li>Processors</li> <li>Cpu:MKE02Z64VLH2</li> <li>Components</li> <li>CsIO1:ConsoleIO</li> <li>BLUE:BitIO_LDD</li> </ul>			



 Add the second BitIO component, rename the component GREE and configure it.

CodeWarrior Projects 🔀	Scomponents Library Scon	nponent Inspector - GREE 🔀
File Name Build     File Name Build     Image: Second Secon	Name       Pin for I/O       Direction       Initialization       Init. direction       Init. value       Auto initialization	Value       PTH2/BUSOUT/FTM1_CH0       Input/Output       Output       J       yes
<ul> <li>Components - Mqx_starter X</li> <li>FLASH</li> <li>OSs</li> <li>MQX1:MQXLite</li> <li>Processors</li> <li>Cpu:MKE02Z64VLH2</li> <li>Components</li> <li>CsIO1:ConsoleIO</li> <li>BLUE:BitIO_LDD</li> <li>GREE:BitIO_LDD</li> </ul>		



*|| CCSLaIC* 

Add last BitIO component, rename the component RED and configure it.

🔂 CodeWarrior Projects 🔀		💊 Components Library 💊 Co	mponent Inspector - RED 🕱
	₽ ▽	Properties Methods Events	
File Name Build	^	Name	Value
⊿ 😂 Mqx_starter : FLASH		Pin for I/O	PTH1/FTM2 CH1
b 🐝 Binaries		Direction	Input/Output
Documentation	E	⊿ Initialization	
FLASH		Init. direction	Output
Generated_Code		Init. value	1
MQXLITE		Auto initialization	yes
>>> ProcessorExpert.pe			
Project_Headers	-		
Project_Settings			
😪 Components - Mqx_starter 🔀	🖻 💼 🔮 🔻 🗖		
🔺 🗁 OSs			
MQX1:MQXLite			
Processors			
Cpu:MKE02Z64VLH2			
Components	-		
CsIO1:ConsoleIO	=		
BLUE:BitIO_LDD			
GREE:BitIO_LDD			
RED:BitIO_LDD			
⊳ 🗁 PDD	Ŧ		
<i>F</i> roocolo™		Presecute. the Presecute logs. ArWeo, C.S. Cod Processor Expert, Corff, Corff, Corff, Strans, B	YESS, OxdeWarter, OxdeFree, DoubFree, OxMann, Net Energy Efficient Solutions lugs, Kineta, Incolaidoll, PEO, Presenducci, a Solutionare logis, StatiCare, Symptomy and Vortila are trainerable of Francesch Energical Academic Inc. Rep. U.S. Fer. 177: DR Mathematical Solutions and Provide and Provide and Provide and Provide and Provide Academic Inc. Rep. U.S. Fer. 177: DR Mathematical Solutions and Provide Academic Inc. 2010;10(1):10(1

and Etymolic are packenesis of Freescale Services during the All other product or service names are the property of their Aspective averages O 2013 Freescale Services during free

#### **Configure the MQX Lite Component**

- Double-click MQXLite
- Go to the Component Inspector, look at MQX1.

📎 Component Inspector - MQX1 🛛	🚫 Components Library
Properties Methods Events	
Name	Value
Component name	MQX1
MQX Lite version	V1.1.0
Copy source files to project	yes
System timer	SystemTimer
Task Template List	1
⊿ Task1	Enabled
Task settings	Task
Configuration parameters	
	Component settings
Lightweight Semaphores	yes
Lightweight Events	no
Lightweight Message Que	no
Mutexes	no
Lightweight Timer	no
Kernel Logging	no
Lightweight Memory Allo	no
⊿ Interrupts	Interrupt settings
Interrupt stack size	256 D
⊿ Idle Task	Idle Task settings
Use Idle Task	yes
Enable sleep in idle	no
Enable idle counters	no
⊿ User settings	
Definitions	1 line(s) Select to view/edit
Manage allocated interrupts	Automatically
First allocated interrupt	INT_SysTick
Last allocated interrupt	INT_LVD_LVW



Presents, the Freezele logs, Athles, D.S., Code/EET, Oxde/Hartin, OxtPine, OxtPine, D.Hara, Into Every Ethiom Soldiers logs, Konta, endedd, PEG, PreveOUCC, Processor Gaser, Carlo, Garna, Barkawa, the Salekawa logo, Statum, Braytony and Vinitalia an trademake of Freezel Marcondum, too, Hey U.S. et al. Th. Ot Antar, Berlin, Berlin, Barkawa, Freezel, Jaymage, Mayni, WIC, Pathers e A Pantage, OxtG Growage, D.KOC Engine, Rady Ny, SMMITMOS, Trave, Tudouts, Ver, Tu

# Configure the Component

- We will have three tasks, so increase by two.
- Task Template
   List = 3
- Other values
   are default

☐ CodeWarrior Projects SX     □   □   SX     □   □   SX     □   □   SX     □   □   SX   File Name			📎 Components Library 🚫 *Component Inspector - MQX1 🛛				
			Properties Methods Events				
te	File Name	Build	Name	Value	Details		
	😂 Mqx_starter : FLASH		Component name	MQX1			
	🗁 Documentation		MQX version	mqx_1_0			
	🔁 FLASH		Copy source files to project	yes			
	🗁 Generated_Code		System timer	SystemTimer			
	📎 ProcessorExpert.pe		▲ Task template list	3			
	🔁 Project_Headers		⊿ Task1	Enabled			
	🔁 Project_Settings		Task settings	Task			
	🔁 Sources		⊿ Task2	Enabled			
			Task settings	Task			
			⊿ Task3	Enabled			
			Task settings	Task			
	ធ- Components - Mqx_starter 🕴 🔪		Configuration Parameters				
	E 4	~ 🖻 🗀 🕻	Synchronization objects				
	MQX1:MQXLite	*	Semaphores	yes			
	💭 SystemTimer1:TimerUni	t_LDD[MQ>	Mutexes	no			
► DI Task1:MQXLite_task[MQXLite\Task]			Lightweight Events	no			
	▶ 📴 Task2:MQXLite_task[MQXLite\Task]		⊿ Interrupts				
	Task3:MQXLite_task[MQ	XLite\Task	Interrupt Stack size	1024	D		
	💹 _int_default_isr		Use Idle Task	yes			
			A Manage allocated interrunts	Automatically			



## Configure the First Task

- Select the task
- Examine the properties
- Actually, default values for all properties
- Task 1
  - Name = Task1 (default)
    - Case matters! Code depends on this
  - Entry point function
  - Stack size = 400







Components - Migx\_starter 🚲

# Configure the Second Task

- Task2
  - Name = Task2 (Default)
    - Case matters! Code depends on this
  - Priority = 8
  - AUTO\_START\_TASK =

#### Enable

 we will instantiate and start the task in our code



🗞 Components Library 💊 Co	mponent Inspector - Task2 🔀	
Properties Methods Events	;]	
Name	Value	
Name	Task2	
Entry point function	Task2_task	
Stack size	400	D
Priority	8	D
Creation parameter	0	
⊿ Attributes		
MQX_AUTO_START_TA	ASK Enabled	
MQX_FLOATING_POIN	T_TASK Disabled	



# Configure the Last Task

- Task3
  - Name = Task3 (Default)
    - Case matters! Code depends on this
  - Priority = 8
  - AUTO\_START\_TASK = Disable
    - we will instantiate and start the task in our code



🗞 Components Library 💊 Component Ins	pector - Task3 🔀	
Properties Methods Events		
Name	Value	
Name	Task3	
Entry point function	Task3_task	
Stack size	400	D
Priority	8	D
Creation parameter	0	
▲ Attributes		
MQX_AUTO_START_TASK	Disabled	
MQX_FLOATING_POINT_TASK	Disabled	





Click <sup>a</sup> to generate PE code





Presents: the Freenake logs. MVNex, D.S. Code/EST, GoldManice, OldFire, OldFire, OldFire, Nelline, Ne Swegy Ethiots Soldions legs. Knots, mobileDL PGS, PreverQUCC, Processor Royer, CorD, Carina, Earlehauer, Int Satel-Aurer Roy, Estic Im, Shyrtenitry and Vocalia and statematics of Freenace Manicovation, the JUS Res. Eth. Ok. Antas, Beellis, BeelStack, Carelus, Fest, Layersage, Magel, MIC, Pathors in a Panlage, OxfO Gonverg, DUCC Expres. Royer, Poly 2007/WOS, Trave, Tubolcek, Vania and Tamica and Antanna and Panaceak Barrusonbarry. Int 30 One product or advece means and the property at the Angelorate and the Scholarek. 2011 Freedool Scholarek. 2010 Freedool Scholarek. 2010

#### Code Walk Through: Initialize the OS

- Open ProcessorExpert.c
- RTOS initialization auto generated code

```
118 /*** RTOS startup code. Macro PEX_RTOS_START is defined by the RTOS co
119 #ifdef PEX_RTOS_START
120 PEX_RTOS_START(); /* Startup of the selected RTOS.
121 #endif
122 /*** End of RTOS startup code. ***/
```

That Macro = a call to \_mqxlite() – which sets up the OS





This code creates the tasks – mqxlite.c

```
h MQX1.h
          🔂 maxlite.c 🔀
266 mqx uint mqxlite(void)
267 { /* Body */
268
        KERNEL DATA STRUCT PTR
                                             kernel data;
269
        MQXLITE TASK TEMPLATE STRUCT PTR
                                             template ptr;
270
        TD STRUCT PTR
                                             td ptr;
271
272
        _GET_KERNEL_DATA(kernel_data);
273
274
        /* Start Tick timer */
275
        MQXLITE RTOS ADAPTER SYSTEM TIMER START(NULL);
276
277
        /* Create the idle task */
278 #if MQX USE IDLE TASK
279
        td ptr = task init internal((MQXLITE TASK TEMPLATE STRUCT PTR)&kernel data->IDLE TASK TEMPLATE,
280
                                      kernel data->ACTIVE PTR->TASK ID,
281
                                      (uint 32)0,
282
                                      FALSE,
283
                                      (pointer)kernel data->IDLE TASK TEMPLATE.TASK STACKADDR,
284
                                      ( mem size)kernel data->IDLE TASK TEMPLATE.TASK STACKSIZE);
285 #if MQX CHECK ERRORS
286
        if (td ptr == NULL) {
287
             mqx exit(MQX OUT OF MEMORY);
288
        } /* Endif */
289 #endif
290
        task ready internal(td ptr);
291 #endif
292
293
        /* Check here for auto-create tasks, and create them here */
294
        template ptr = kernel data->INIT.TASK TEMPLATE LIST;
295
        while (template ptr->TASK TEMPLATE INDEX) {
296
            if (template ptr->TASK ATTRIBUTES & MQX AUTO START TASK) {
297
                td ptr = task init internal(template ptr,
298
                                              kernel data->ACTIVE PTR->TASK ID,
299
                                              template ptr->CREATION PARAMETER,
300
                                              FALSE,
301
                                              (pointer) template ptr->TASK STACKADDR,
302
                                              ( mem size) template ptr->TASK STACKSIZE);
```



Presents, the Freeholds logs, AVNex, D.S., Colar/EST, CadeMarcin, OxtFine, OxtFine, OxtFine, C.Mass, Nex-Evergy Ethioth Solidons legic Xineta, incided/31, PEG, Preve/GUCC, Processor Ecsev, Confl, Darina, EditAkatar Kos, StacCere, Spripting and Vorlida with tablenda of Freezenka Interconductin, the Antari, Self-R, Beddach, Canifer, Faste, Layersean, Magni, MCC, Pathone a & Pathaga, OxtG Gomerga, OXCC Entrye, Read, Play, SMATMOS, Treve, Turbolmi, Vymit and Thinks are associated. Freezenka Interconducting. Int. Although conduction on and are to property of their respective entrye. 5 (30) Freezenka Relacionation Exand Thinks and Sandonatia all-Resonation Entry Conducting. Int. Although conduction on the to entry of their respective entrye. 5 (30) Freezenka Relacionation Ex-



#### **Task templates generated from component**

- Open task\_template\_list.c
- in Generated\_Code folder
- Here are our tasks
  - Based on the properties set in the component
  - One is an auto-start task as we specified

```
const TASK_TEMPLATE_STRUCT MQX_template_list[] =
  /* Task: Task1 */
  {
  /* Task number
                                       */ TASK1_TASK,
  /* Entry point
                                       */ (TASK_FPTR)Task1_task,
  /* Stack size
                                          TASK1 TASK STACK SIZE,
  /* Task priority
                                       */
                                           8U,
                                          "task1",
  /* Task name
                                       */
  /* Task attributes
                                       */ (MQX AUTO START TASK),
                                       */ (uint32 t)(0)
  /* Task parameter
  },
  /* Task: Task2 */
                                       */ TASK2_TASK,
  /* Task number
                                       */ (TASK_FPTR)Task2_task,
  /* Entry point
  /* Stack size
                                       */ TASK2 TASK STACK SIZE,
  /* Task priority
                                           8U,
                                           "task2",
  /* Task name
  /* Task attributes
                                       */ (MQX AUTO START TASK),
                                       */ (uint32 t)(0)
  /* Task parameter
  },
    Task: Task3 */
  /* Task number
                                       */ TASK3 TASK,
  /* Entry point
                                       */ (TASK FPTR)Task3 task,
                                         TASK3 TASK STACK SIZE,
  /* Stack size
  /* Task priority
                                           8U,
                                           "task3",
  /* Task name
  /* Task attributes
                                           (0),
                                          (uint32 t)(0)
  /* Task parameter
                                       */
  },
  TASK TEMPLATE LIST END
};
```





- In mqx\_tasks.c
  - in Sources folder
- Function header automatic
  - You will still need to create body of function obviously

 Loops endlessly flashing BLUE LED

```
38 **
                      : Nothing
         Returns
39 **
                                    _____
40 */
41 void Task1 task(uint32 t task init data)
42 {
43
    int counter = 0;
    printf("task 1 start running!\n");
44
45
    while(1) {
46
      counter++;
47
      /* Write your code here ... */
48
49
      BLUE ClrVal(0);
50
          _time_delay_ticks(50);
51
          BLUE SetVal(0);
52
         time delay ticks(50);
53
    }
54 }
55
```



Presents: The Freesewite logs: MONex, D.S. Code/EST, Code/Este, Oxf/Fire, Oxf/Fire, Oxf/Fire, Different Soldions legs: Monex, mode/s01, PEG, PreverQUCC, Processor Royer, CorD, Carros, Earlehauer, Na Salekauer Rog, StatCare, Strafferey and Vorlab are trailereade of Freezewite Reserved Samcconducts; to: Deg U.S. Fire, Thr. Oth. Antar, Bel/R, BedStack, Careker, Rest, Layrenage, Migry, MRC, Pathon et a Pantage, OxfO Gonverg, UXCC Teger, Robin Pay, StatCare, Tabolan, Vytral and Tamica as before a File-Reserved Reserved Astron. Science Samo are to anyone means are to anyone; if the respective areas as 2011 Freedak Samcconducts; to: Science.com



- Loops endlessly flashing GREE
- Setup task3

```
MQX task routine. The routine is generated into mqx_tasks.c
62 **
63 **
             file.
64 **
         Parameters :
65 **
             NAME

    DESCRIPTION

66 **
             task init data
67 **
         Returns
                     : Nothing
68 **
     _____
                                    _______
69 */
70 void Task2 task(uint32 t task init data)
71 {
72 int counter = 0;
    task id task id;
73
74 printf("task 2 start running!\n");
   task id = task create_at(0,TASK3_TASK,10,Task3_task_stack,TASK3_TASK_STACK_SIZE);
75
    if ( task id == MQX NULL TASK ID )
76
77
    {
        printf("task3 create fail!\n");
78
79
    }
80
    while(1) {
81
82
      counter++;
83
      /* Write your code here ... */
84
      GREE SetVal(0);
85
              _time_delay_ticks(50);
86
87
              GREE ClrVal(0);
88
              _time_delay_ticks(50);
89
90
    }
91 }
```

The OS handles task priority and switching



Presents, the Freenance logs, AVWe, D.S. Code/EEF, Code/Harris, DiefFre, Code/Free, C. Haus, Inv Energy Ethinet Solutions logs, Kanta, endeledit, PSD, Preve/DUCC, Processon Raper, Cardi, Carna, Earlwane, Na Salekaner logs, Rachure, Sprytherey and Vorlib, and trainistande of Freezak Benccarkum; the USL Rip. E. Tro. OR. Antar, Stel/R, Beddack, Cardwar, Ress, Layersage, Mayri, WRC, Pathore is a Pathoga, Cardi Cohneya, DUCC Engin, Rady Ney, SMITHOS, Tree, Tubolick, Vysnit and Tamica and Indexnity. If Research Research Research and the second and the second second



 Loops endlessly flashing RED

```
98 **
         Description :
99 **
             MQX task routine. The routine is gen
00 **
             file.
.01 **
         Parameters :
02 **
             NAME

    DESCRIPTION

.03 **
             task init data
04 **
         Returns
                     : Nothing
05 **
     -----
.06 */
.07 void Task3_task(uint32_t task_init_data)
.08 {
.09 int counter = 0;
    printf("task 3 start running!\n");
.10
.11
    while(1) {
.12
      counter++;
.13
    /* Write your code here ... */
.14
.15
      RED_ClrVal(0);
.16
              _time_delay_ticks(50);
.17
              RED SetVal(0);
              time delay ticks(50);
.18
.19 }
.20 }
.21
.22 /* END mqx_tasks */
```

The OS handles task priority and switching



Presents, the Freenance logs, AVWe, D.S. Code/EEF, Code/Harris, DiefFre, Code/Frei, C. Haus, Inv Energy Ethinet Solutions logs, Kanta, endeledit, PSD, Preve/DBCC, Processon Raser, Card, Darina, Earbhauan, Ma Salekaran Kog, RacZue, Spripturg, and Vorlib, and tambianak of Freezak Barccodusts, the JUS. Fig. 57, DR. Antar, Stel/R. Beddack, Cardwar, Ress, Layersaga, Mayol, MCC, Patters et a Patieting, Card Converga, DJCC Engen, Rado Nat, Stel/Mittal, Structure, Vandula, Vysnil and Tamica and Status, and Hannes Reservoirsate. In Active product or server name at the inspective and the Internative Solutional Schoolastic Schoolastic, Nat. 2013; File Schoolastic, Cardwar, Ress, Layersaga, Mayol, MCC, Patters et a Patieting, Card Converga, DJCC Engen, Rado Nat, Status, Schoolastic, Visnil and Tamica and Status, and Schoolastic Schoolastic Reservoirsate. 300; File Schoolastic Reservoirsate. 301; File Schoolastic Reservoirsate. 302; File Schoolastic Reservoirsate. 302; File Schoolastic Reservoirsate. 302; File Schoolastic Reservoirsate. 303; File Schoolastic Reservoirsate. 304; File Schoolastic Reservoirsate. 304; File Schoolastic Reservoirsate. 304; File Schoolastic Reservoirsate. 304; File Schoolastic Reservoirsate. 305; Fi

### Build the Code

- Just click Build in the Commander view OR...
- Select the project
- Project→Build Project
- Should be no errors





Anaecetis, the Financelo logit, AlVino, D.S., Code/EST, Cade/Marrier, Ord/Fine, Oxd/Fine, Oxd/Fine, Diffuser, Solidions logit, Xinata, modeleGT, PGG, PreverGUCC, Processor Equipm, Cardi, Qurina, Barkasawa ing Sate/Anae Sage Sate/Ana, Barytanya and Vintilla are statemated or Financek Tamicconductor, the U.S. Har, S. Fin, Ott. Antari, Sate/A, Barddack, Canduct, Finas, Lagrenages, Magny, MRC, Parthern et a Pantage, Quring Generage, Quice Chigms, Rado Pay, State/Finas, Sate/A, Sate/A,

# Getting start with FRDM-KE02Z

#### Installing Drivers

1 Optional: Download and Install the P&E OpenSDA USB Drivers found at <u>www.pemicro.com/opensda</u>

2 Plug in a USB cable (not included) from a USB host to the OpenSDA mini-B USB connector. The FRDM-KE02Z will be powered by this USB connection.

- 3 a. Open Device Manager
- b. Locate and right-click on "OpenSDA CDC Serial Port"
- c. Select "Update Driver Software"
- d. "Browse" and select the FRDM-KE02Z drive
- e. Click "Next" to complete the installation



Presents, the Freerolds logit, MVNov, D.S., Code/EEST, Osdolfarrice, OshFire, OshFire, OshFire, Nellises, Nei Sarey, Ethionet, Soldiones legit, Kineta, modeleGT, PGS, PreverQUCCC, Processor Royer, CodrQ, Carrina, Tarbahawa, Hai Satel-Asara Kag, Staticitine, Strateging and Vorsilla van statematice of Freezolek Terriconducture, Inc., Hey, U.S. Res, Ern., Ott. Antari, Stellit, Berdland, Carrieva, Fanni, Layersonge, Magnel, WCC, Peterleer e A Pantage, QUICC Designs. Ready (2000) Design Market (2000) Vorsilla van Statematica (2000) Statematica (2000) Antario, Ant

# Introduction to openSDA

- Open-standard serial and debug adapter.
- Bridges serial and debug communications between a USB host and an embedded target processor.
- OpenSDA software includes a flash-resident USB massstorage device (MSD) bootloader and a collection of OpenSDA Applications. FRDM-KE02Z preinstalled with the MSD Flash Programmer OpenSDA Application.





# Using the MSD Flash Programmer

 MSD Flash Programmer - a composite USB application that provides a virtual serial port and an easy and convenient way to program applications into the KE02Z MCU. It emulates a FAT16 file system, appearing as a removable drive in the host file system with a volume label of FRDM-KE02Z.



1 Locate the Precompiled Examples folder in the FRDM-KE02Z Quick Start Package.

2 Copy & paste or drag & drop one of the .srec files to the FRDM-KE02Z drive.

The new application should now be running on the FRDM-KE02Z. Starting with the MSD Flash Programmer, you can program repeatedly without the need to unplug and reattach the USB cable before reprogramming.





#### Introduction to openSDA

#### **Bootloader mode**

- 1. Unplug the USB cable if attached.
- 2. Press and hold the Reset button.(SW1).
- 3. Plug in a USB cable between a USB host and the OpenSDA USB connector (labeled "SDA").
- 4. Release the Reset button.

A removable drive should now be visible in the host file system with a volume label of BOOTLOADER.

	BOOTLOADER (G:)	
D	107 MD free of 107 MD	
	12/ IVID TREE OF 12/ IVID	

#### Load OpenSDA Application

- Locate the OpenSDA Applications folder in the FRDM-KE02Z Quick Start Package.
- Copy & paste or drag & drop the Debug Application (DEBUG-APP\_Pemicro\_v106.SDA)to the BOOTLOADER drive.
- 3. Unplug the USB cable and plug it in again. Open device manager.
  - PEMicro OpenSDA Debug Drives Now we can
  - WinDriver download and debug!
    - ECP Printer Port (LPT1)

OpenSDA - CDC Serial Port (http://www.pemicro.com/opensda) (COMD)

Enter bootloader mode and copy & paste or drag & drop the MSD Flash Programmer Application (MSD-FRDM-KE02Z\_Pemicro\_vXYZ.SDA) to the BOOTLOADER drive to use the MSD Flash Programmer again.



#### Enable OS Awareness to debug MQX Lite tasks

- Choose the project, right click and choose Properties -> Run/Debug Settings -> OpenSDA ->Edit
- Edit Configuration -> Debugger -> OS Awareness





- Run→Debug As→CodeWarrior Download
  - Pick the OpenSDA connection
- Debug perspective appears
- Code stops at first line of main()



Set a breakpoint at line 132 of mqx\_tasks.c





#### NP Debug MQX Lite tasks

Click resume to run the application, the application will stop at the break point.

🥦 Debug - MQXLitedemo/Sources/mqx_tasks.c - CodeWarrior Development Studio	
File Edit Source Refactor Search Project MQX Tools RTCS MQX PEMicro Run	Window Help
<u> <u> </u> </u>	
<ul> <li>Debug S</li> <li>MQXLitedemo_FLASH_OpenSDA [CodeWarrior]</li> <li>MQXLitedemo_FLASH_OpenSDA [CodeWarrior]</li> <li>ARM Processors, MQXLitedemo.elf (Suspended)</li> <li>Thread [ID: 0x10001] (Suspended: Signal 'Process Suspended' received. Description: 2 _mqx_idle_task() idletask.c:56 0x00002139 1 _task_exit_function_internal() task.c:2980 0x0000144c</li> <li>Thread [ID: 0x10003] (Suspended: Signal 'Process Suspended' received. Description: 2 Task2_task() mqx_tasks.c:85 0x00002605 1 _task_exit function internal() task.c:2980 0x0000144c</li> <li>Thread [ID: 0x10002] (Suspended: Breakpoint nit.) 2 Task1_task() mqx_tasks.c:60 0x000025e0 1 task_exit_function_internal() task.c:2980 0x000144c</li> <li>E:\MQXLiteWorkspace\MQXLitedemo\FLASH\MQXLitedemo.elf (9/29/13 1:23 PM)</li> </ul>	Image: Second Secon
<	F C F
idletask.c	□ 🗳 Disassembly 🛛 🗖 🗖
<pre>*/     void Task1_task(uint32_t task_init_data)     {         int counter = 0;         printf("task 1 start running!\n");         wnile(1) {             counter++;             /* Write your code here */             &lt;</pre>	<pre>     Enter location here     @      @</pre>

# Debug MQX Lite tasks

 Click MQX in the menu, MQX debug options appears. Including Task summary, Stack usage.....





# Debug MQX Lite tasks

- Click task summary to watch the task status.
  - When MQX Lite RTOS starts, three tasks are created.

📃 Console 📳 Problems 📲 MQX Task Summary 🔀						
Task Name	Task ID	TD	Priority	State	Task Error Code	
_mqx_idle_task	0x10001	0x20000388	9	Ready	OK (0x0000)	
⊳ task1	0x10002	0x200006b4	8	Active	OK (0x0000)	
b task2	0x10003	0x200004a8	8	Ready	OK (0x0000)	

- · υποι στασι παία το παιστι της στασι ποαλε οι απ τασισ.
  - Stack size = 400 byte

📮 Console  🔒 Problems	MQX Stack Usage 🛛				
Task	Stack Base	Stack Limit	Stack Used	% Used	Overflow?
_mqx_idle_task	0x200004a0	0x20000400	0x20000444	57 %	No
⊳ task1	0x200008b0	0x20000720	0x2000083c	29 %	No
⊳ task2	0x200006b0	0x20000520	0x20000654	23 %	No
Interrupt	0x20000070	0x1fffff70	0x20000070	0 %	No



# Debug MQX Lite tasks

Click to continue running application –four tasks.

🥦 Debug - MQXLitedemo/Sources/mqx_tasks.c - CodeWarrior Develop	🥦 Debug - MQXLitedemo/Sources/mqx_tasks.c - CodeWarrior Development Studio						
File Edit Source Refactor Search Project MQX Tools RTCS	MQX PEMicro Run W	indow Help					
Fî - 🛛 🕅 🤣 🦑 i 🛪 🕷 🎫		🔳 💦 🏇 🕶 🔗 🕶 🌛 😓		াৰ্শ			
					-		
			Quick Acc	ess 🗄 🗄	暄 C/C++   珍 Debug		
🏇 Debug 🔀 🖳 🗖	🗱 🔍 🕒 🗱 🗱	kpoints 🚻 Registers 📋 Me	emory 🛋 Modules   🕅 🏘	i 🖂   🚱 🕶   🖋 🗙	*		
🍇 🖉 😌 🕹 📴 🖉 🖤 👖 👖 ▾ 🌉 🗟 ▽	Name	Value	Lo	ocation			
MQXLitedemo FLASH OpenSDA [CodeWarrior]	(x)= task init data	0	0x	20000884			
ARM Processors, MQXLitedemo.elf (Suspended)	C counter	1	0x	2000088c	<u> </u>		
A 🖓 Thread [ID: 0x10001] (Suspended: Signal 'Process St		-					
2_mqx_idle_task() idletask.c:78 0x00002147							
1_task_exit_function_internal() task.c:2980 0x00							
🕢 🔊 Thread [ID: 0x10003] (Suspended: Signal 'Process S	0				-		
4 _time_delay_internal() time_ticks.c:180 0x0000					-		
3 _time_delay_ticks() time_ticks.c:232 0x00000d1	•				Þ		
2 Task2_task() mqx_tasks.c:98 0x0000264c					_		
1 _task_exit_function_internal() task.c:2980 0x00	🛛 🕌 MQX Task Summary 🛛	3		s~ [	] ਵਿ ♡		
Thread [ID: 0x10002] (Suspended: Breakpoint hit.)	Task Name	Task ID	TD	Priority	State		
2 Task1_task() mqx_tasks.c:60 0x000025e0	▶ max idle task	0×10001	0v20000388	9	Ready		
1 _task_exit_function_internal() task.c:2980 0x00	▷ task1	0x10002	0x20000566	8	Active		
🔺 🔐 Thread [ID: 0x10004] (Suspended: Signal 'Process Sເ	⊳ task2	0x10003	0x200004a8	8	Ready		
4 _time_delay_internal() time_ticks.c:180 0x0000	⊳ task3	0x10004	0x2000007c	8	Time delay blocked		
3_time_delay_ticks() time_ticks.c:232 0x00000d1	<i>v</i>			-	,,		
2 Task3_task() mqx_tasks.c:127 0x0000268a							
1_task_exit_function_internal() task.c:2980 0x001							
4 III >>	•				ŀ		
					_		
i mqx_tasks.c ☆ ime_ticks.c □	🛛 🚟 Disassembly 🕌 MQX	Stack Usage 🔀		] %	] ਵਿ ♡		
<pre> ovoid Task1_task(uint32_t task_init_data) </pre>	Task	Stack Base	Stack Limit	Stack Used	% Used		
{	b max idle task	0x200004a0	0x20000400	0x2000042c	72 %		
<pre>int counter = 0; printf("task 1 start supping[\p");</pre>	▶ task1	0x200008b0	0x20000720	0x200007f4	47 %		
while(1) {	⊳ task2	0x200006b0	0x20000520	0x20000560	84 %		
counter++;	⊳ task3	0x20000280	0x200000f0	0x200001c4	47 %		
	Interrupt	0x20000070	0x1fffff70	0x1ffffdc	57 %		
/* Write your code here */							
time delay ticks(50):							
< III +	•		III		•		
IIGGGGGG		and Through an approved	a of Freedom Serviceschuter, inc. 42 ofter sector of	a polytery reprode the strategy of their a	espectad investors (C 2013 Press at Surviv		



Click the Resume button

- Resume each time you stop
  - You'll see the count variable increase, and the lights on the board flash
- Remove the breakpoint to run the app without interference
  - Right click, Toggle breakpoint



🝽= Variables 🔀 💁 Breakpoints 👫	Registers	📋 Memory 🛋 Modules 🛛 🗖
		😓 🍕 🕞   🚱 🔻 🗳   📬 🎽
Name	Value	Location
(x)= task_init_data	0	0x20000884
(x)= counter		0x2000088c
. (		





Click the Terminate button



- You have built an app using
  - the MQX Lite component and PEx
  - Adding a PEx component template
- You should understand some of the key differences between MQX RTOS and MQX Lite



hreecels, the Freesele logs. AdVec, D.S., ColorEST, Cadultario, CadFire, OxfFire, C.Mass, Inv Energy Ethiant Soldions logs. Knets, included. PSG, PreedOJCC, Possess Essen, Cord, Jonna, Bahkasan, Ini Sakkasar logs SeaCare, Smythery and Vorlab are trailereade of Freezel Initiation Soldions, Ing. K. R. B. (20. First, Tr., Ok Antar, Selfik, Beddack, Carlwar, Free, Layersage, Mayri, MIC, Partner, et a Partage, Gird Generge, GUCC Engine, Reid Part, Per, Statular, Vand and Stress are tradeworks of Freezel Instructures. In: Although conductor solver name are to anyon and and any source of the process. Excitorious Colling and Stress are tradeworks of Freezel Instructures. In: Although collection of solver name are to anyon and an anyon of the solution of the Instructures. Coll J. Freezel. Solverson and Stress are to anyon of the solution of the Instructures. Coll J. Freezel. Solverson and Stress are to anyon of the solution of the Instructures. Coll J. Freezel. Solverson and the sources of the solution of the Instructure. Coll J. Freezel. Solverson and the source of the solution of the Instructures of the Instructures. The Instructure of the Instructures of



