Experiment 2: MCSDK demos

August 18, 2011



Emulation environment

- Shannon EVM
- XDS100v1 USB Emulator



Demo 1: PA Simple Example



Flow Chart



PA Simple Project:

- Import CCS project from PDK example
- Build project
- Specify emulation configuration
 - Target select
 - Add evmc6678l.gel file
- Launch target configuration
- Load program and run



Import PDK example

- Import project
 - Project ->Import existing CCS eclipse project





• Browse to the directory

(C:\Program Files\Texas Instruments\ pdk_C6678_1_0_0_11\packages\ ti\drv\exampleProjects\ PA_simpleExample_exampleProject)

- PA Simple Example
- Select 'Finish' when done

😵 Import CCS Eclipse P	Projects	
Select Existing CCS Ecl Select a directory to search	ipse Project for existing CCS Eclipse projects.	
 Select search-directory: Select archive file: Discovered projects: 	C:\Program Files\Texas Instruments\p	Browse
PA_simpleExam	ole_exampleProject (C:\Program Files\T	<u>S</u> elect All Deselect All R <u>e</u> fresh
	space	Cancel



Build project

 Right-click the project and select 'build project'





Specify emulation configuration

- Add a new target configuration
 - File->New->Target configuration File

\$ (💱 CCS Edit - Code Composer Studio												
File	Edit	View	Source	Refactor	Navigate	Search	Pro	oject	Tools	Run	Scripts	Window	Help
	New				AltH	FShift+N	►	1	CCS Pro	ject			
	Open F	ile							Project.				
	Close				Ctrl	+W		C [°]	Source F	=ile			
1	Close A	411			Ctrl	+Shift+W	/	ĥ	Header	File			
IJ	Save				Ctrl	+S		6	Class				
E.	Save A	s							File from	n Temp	late		
R	Save A				Ctrl	+Shift+S			Folder				
	Revert							च	Target (Ionfigu	ration Fil	e	
	Move								DSP/BIC)S v5.x	: Configui	ation File	



• Provide a name for the new target

🕸 New Target Configuration	
Target Configuration	
Create a new Target Configuration file.	
Eile name: c6678_XDS100v1_emu.ccxml	
✓ Use <u>s</u> hared location	
Location: C:/Documents and Settings/x0145164/user/CCSTargetConfigurations	Eile System Workspace
?	Einish Cancel

- Select 'Use shared location'
- Select 'Finish' when done



Target select

- Use drop down menu to select the connection type
- Select the device
- Save the file when done
- Select 'Advanced' tab

	🕄 c6678_XDS1	00v1_emu.ccxml 🖾				
	General Setup					
	This section (describes the general configuration about the target.				
	Connection	Texas Instruments XDS100v1 USB Emulator	*			
	Device	c6678				
		✓ TM5320C6678				
\setminus						
	\ \		>			
	\backslash	C66x core	~			
	\backslash					
	\		V			
	Basic Advance	d Source				



Add evmc6678l.gel file

🖹 *c6678_XD5100v1_emu.ccxml 🔀 💽 paExample.c	
Target Configuration	
All Connections	Cpu Properties C66xx CGEM+FP CPU Set the properties of the selected cpu. Bypass initialization script\\emulation\boards\evmc6678l\gel\evmc6678l.get Browse Slave Processor
 Select core 0 Browse to select evm gel file (C:\Program Files\Texas Instruments\ccsv5\cc emulation\boards\evmc6678l\gel\evmc6678l.g Save the file when done 	s_base_5.0.3.00028\ el)



Launch target configuration



- Switch to target configuration window
- Right-click on c6678_XDS100v1_emu.ccxml, select 'launch selected configuration'



Load program and run

- Select core 0
- Click 'connect target'
- Load program
- Browse for *.out file



📬 • 🖫 💼 🕘 💼 🗄 🖉 • 🖓 • 🍫 • 🔶 • 🕂 🐅 • 🗄 💻

😑 🜍 c6678_XDS100v1_emu.ccxml [Code Composer Studio - Device Debugging]

🍇 🕩 🗉 🔳 M | R. O. R. O. R. 🗐 🖬 🕱 🌰 🗸

Texas Instruments XDS100v1 USB Emulator 0/C66xx 0 (Disconnected : Unknown)

Paras Instruments XDS100v1 USB Emulator_0/C66xx_1 (Disconnected : Unknown)

Connect Targel

🎋 Debug 🔀



When program is loaded,
 Select the 'run' button to
 execute application





	c6678_XD5100v1_emu.ccxml:C66xx_0:CIO
	function findPacket: Correct packet found
	PA STATS
	C1 number of packets: 4
	C1 number IPv4 packets: 1
	C1 number IPv6 packets: 0
	C1 number custom packets: 0
	C1 number non IP packets: 0
	C1 number llc/snap fail: 0
	C1 number table matched: 2
	C1 number failed table matched: O
	C1 number IP frags: 0
	C1 number IP depth overflow: 0
Core0 CIO output →	C1 number vlan depth overflow: O
I	C1 number gre depth overflow: O
	C1 number mpls packets: 0
	C1 number of parse fail: 0
	C1 number invalid IPv6 opts: 0
	C1 number of command failures: O
	C1 number invalid reply dests: O
	C1 number of silent discard: 0
	C1 number of invalid control: 0
	C1 number of invalid states: 0
	C1 number of system fails: 0
	C2 number of parse fail: 0
	C2 number invalid headers: U
	C2 number of UDP packets: 1
	C2 number of TCP packets: 0
	C2 number of custom packets: U
	C2 number of command failures: U
	C2 number of invalid reply dests: 0
	C2 number of silent discard: 0
	C2 number of invalid control: U
	Modify number of command file: O
	Common number of ID allocation fail: 0
	Test Completed successfully



Demo 2: PA Multi-core Example



Flow Path



Flow Path

• Initialize the components required to run the example:

- Queue Manager (QM) Subsystem
- Packet Accelerator (PA) CPPI DMA
- Ethernet Subsystem (Ethernet switch + SGMII + MDIO)
- PA Subsystem + PDSP
- Sets up the CPPI descriptors and Queues required for sending and receiving data using Ethernet.
 - Uses Host descriptors
 - Uses High Priority Accumulation interrupts
- Sets up the example application's configuration (MAC address it uses to send/receive data; IP address and port number it's listening on) in PA Subsystem so as to enable the PASS to forward all packets matching this configuration onto the application for processing.
 - Switch MAC address configured = 0x10:0x11:0x12:0x13:0x14:0x15
 - Example's IP address = 192.168.1.10
 - Example App's listening port = 0x5678 + corenum



Flow Path (continued)

- Sends packets onto wire (constructed manually in code here with following settings):
 - Source MAC = 0x00:0x01:0x02:0x03:0x04:0x05
 Destination MAC = 0x10:0x11:0x12:0x13:0x14:0x15
 - Source IP = 192.168.1.1
 - Destination IP = 192.168.1.10
 - Source Port = 0x1234
 Destination Port= 0x5678 + corenum
 - Payload Data (80 bytes)

The packets sent by the application are sent onto wire and since the destination MAC on the packet is the Ethernet Switch MAC address, the packets are received by simulator and passed back up to the example application for processing.

• Application receives all packets using QM High priority interrupt registered; Validates received packet against data sent.



PA Multi-core Project:

- Import CCS project from PDK example
- Build project
- Launch target configuration
- Load program and run



Import PDK Example

- Import project
 - Project ->Import existing CCS eclipse project





• Browse to the directory

(C:\Program Files\Texas Instruments\ pdk_C6678_1_0_0_11\packages\ ti\drv\exampleProjects\ PA_multicoreExample_exampleProject)

- PA_multicore example
- Select 'Finish' when done

😵 Import CCS Eclipse P	Projects	
Select Existing CCS Ecl Select a directory to search	ipse Project for existing CCS Eclipse proje ^L ts.	
 Select search-directory: Select archive file: Discovered projects: 	Projects\PA_multicoreExample_examp	Browse
PA_multicoreExa	ample_exampleProject (C:\Program File	<u>S</u> elect All Deselect All R <u>e</u> fresh
☐ Copy projects into works	space	Cancel



Build project

 right-click the project and select 'build project'





Launch target configuration



- Switch to target configuration window
- Right-click on c6678_XDS100v1_emu.ccxml, select 'launch selected configuration'



Load program and run

• Group core 0~3







Select the 'run' button to execute application on Group 1





Result

• Open 4 consoles to show 4 cores result together

🗱 Variables 🔗 Expres 👯 Registers 📮 Console 🖾 🖵 🗖	🐨 Disassembly 🚺 Memory Bro 📮 Console 🖾 🖵 🗖	💁 Breakpoints 📮 Console 🛛 👘 🗖			
c6678 XDS100v1 emu.ccxml:C66xx 0:CIO	c6678 XDS100v1 emu.ccxml:C66xx 1:CIO	c6678 XDS100v1 emu.ccxml:C66xx 2:CIO			
	🔓 🚮 🛃 ד 📬 ד	🔓 🚮 📑 🖬 ד 📬 ד			
******	d an entry in receive queue with swir				
i Core Example Started on Core O ***	**************************************				
*****	Example Started on Core 1 ***				
g Free Descriptors.	**************************************				
sfully initialized	config	/ initialized			
sfully initialized	fully initialized	lone			
sfully initialized	lly done	7 done ;o reach the barrier before transmissi art			
ubsystem successfully initialized	fully done				
ccessfully done	res to reach the barrier before trans				
ccessfully done	n Start	12.			
successfully done	n Done.) be Received			
all cores to reach the barrier before t	ts to be Received	= 10			
smission Start	nt = 10	·d = 10			
ismission Done.	ceived = 10	***************			
l mackets to be Received	*****	ule Ended on Core 2 ***			
kets Sent = 10	Example Ended on Core 1 ***	******			
kets Received = 10	****				
I mackets to be Received in all cores V	~	~			
	< > >	< X X X X X X X X X X X X X X X X X X X			
E Console 22					
céé78 YDS100ut emu combicéévy 3iCIO					
**************************************	****	2			
*** D) Multi Core Evemple Started on Core 3	***				
**************************************	****				
Weiting for globel config					
OMSS Local successfully initialized					
Unas Lucessfully initialized					
AN SECUR SUCCESSIVILY AURE					
reas setup successfully upper					
Martering for an optical on the particle prove transmission states					
Parket Transmission Done					
Wait for all markets to be Received					
VerifyBacket: Found an entry in receive queue with swinfod = 0x00000000, expected 0xaaaaaaaa					
VerifyBacket: Found an entry in receive queue with swinfol = 0x00000000, expected 0xaasaaaaa					
VerifyBacket: Byte 37 expected 0x7b, found 0x79					
VerifyBacket: Byte 37 expected 0.72h, found 0.79					
WarifuBacket: Bute 37 expected 0x7b, found	0x1-2 0v70				
- · · · · · · · · · · · · · · · · · · ·		× .			



Thanks!

August 17, 2011

