

## **CS8803 High Performance Communication, Spring 2005**

### **Assignment #1**

**DUE:** Monday, January 24

hard copy in class, or via email by 2:05pm

#### **Part I.**

Answer the following questions. You can use the IXP2400 manuals to find your answers.

1. Your application uses the ENP-2611 board to support line rates of up to 2Gbps in a Gigabit Ethernet network. What is the cycle budget that can be allocated for each Ethernet packet in order to make the throughput guarantees?
2. How can you measure the per-packet processing time for a set of operations executed by a single microengine? What if the set of operations is executed by two pipelined microengines? Describe your solution by referring to specific registers, microcode instructions, or macros and functions provided with the SDK. How would you make this information available outside of the application?
3. How do you implement a critical section (e.g., with a mutex) between threads on different microengines? How do you implement a critical section between a microengine thread and the XScale?
4. How is a microengine notified that a DRAM write request has completed? What if the request is split across DRAM banks?

#### **Part II.**

For this part, you will use the SDK Workbench and the Architecture Tool.

- You can get the installation CDs for the SDK from me (SDK3.51), or download them from `/net/hp31/ixpdev/exports-sdk3.1` (SDK3.1). The Workbench is also available on CoC Windows machines.
- The AT is at `/net/hp31/ixpdev/exports-sdk4.0/SRC/nassaupr8_noncrypto.zip`.

In addition you will need the example code from Chapter 5 from Johnson & Kunze's Coding Guide (`/net/hp31/ixpdev/CD-Images/Programming_2400_2800/Chapter05/`).

The Chapter05 code is a counting application built for the IXP2800. The files include two project files (.dwp) built for the microcode and microC version of the application.

1. Open, build and run the application in the Workbench simulator. You may work with the microcode or microC version of the project. Understand what is going on in the application (e.g., Step through and follow the processing of one packet). Then answer the following:
  - a. How many cycles does it take to transmit the first packet? What percentage time is microengine 1 idle during this time?

- b. What does this application count? Where is the counter stored (add a Data or Memory Watch to verify this)?
- c. What are the packet rates Rx and Tx packet rates recorded in the Packet Simulation?
- d. Modify the input data stream to contain min-sized packets. How does this affect the answers to a. and c.?
- e. What would it take to modify the project for the IXP2400? (*Optional: Go ahead with it*)

*Note1:* The sample code is built with SDK3.0 – you will need to include and link all macros and files from the newer version. Update the Assembler (for .uc) or Compiler (for .c) include files in Build -> Settings...-> General.

*Note2:* For .c you need to explicitly include intrinsic.c. Insert this file in the project (Project->Insert Compiler Files...) and explicitly include it in Build-> Settings...-> Compiler-> Sources to compile... for each of the three output target lists in the applications.

2. Using the Architecture Tool build a project representing the counting application. Follow Section 2 of the AT User Guide to create a simple AT project, then modify appropriately. Use the Microblocks provided with the AT to represent the packet Rx and Tx tasks.

- a. Analyze this application for different packet sizes and rates? Pick 5 representative [size, rate] values and report the throughput rates at each pipeline stage.
- b. Change the complexity of the 'counter' by adding/removing memory references or increasing/decreasing the amount of computation (this does not have to correspond to a meaningful task), or change the number of threads or microengines executing this task. How is the throughput affected – report any observations you find interesting.