

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Department of Electrical Engineering and Computer Science
6.897 Parallel Processing Mechanisms & Applications

Class Project
Beamformer Contest

Issued : 3/21/02
Due : 5/10/02

Project parameters

Number of antennae	As many as you can manage
Angle of interest	-0.4 rad off of broadside
Antenna spacing	Linear array with distance of $\frac{\lambda}{2}$
Transmission medium	Free space
Input data rate	50MHz
Signal Frequency	10MHz
Input data value type	IEEE Single Precision Float
Input data representation	Complex
RAW mesh	4×4
RAW clock rate	300MHz
RAW port usage	As you see fit

Project :

The challenge is to design and simulate a beamformer with as many antennae as possible using the RAW simulation environment. This beamformer will be the same as the one presented in class (see Handout 10) except it will operate on complex, not just real, data. The goal is to map the beamformer application onto a multiprocessor, in this case RAW. We will give you a complex input stream for each antenna in your beamformer and you will have to perform an appropriate weighting function on each stream and then combine them into a beamformed output. You will have to figure out how you would like to take the data in, operate on it, and then ship out the result. Which operations occur on which processors and how data is moved between processors is all up to you. The more efficient your implementation, the more input data streams it should be able to handle, the more antennae can be used in the system, and the more directed and tight the resultant beam. We will provide you with input data files representing the data seen at each antenna. Raw has 14 I/O ports each of which will be running at 6 times the data rate. As such, there is a theoretical maximum of 42 antennae. See if you can hit it. Please note that we will provide off-chip manipulation of data (i.e. multiplexing) to allow multiple data streams to share the same I/O port. The working parameters for the system are given above.

You will have to figure out the best, most clever, spiffiest mapping of the beamformer application onto the RAW multiprocessor. Doing this will require persistence, fortitude, and an intimate knowledge of how communication and processing occurs on RAW. We can't give you the first two, but we can help get you started on the last.

All the information for the project, and attendant links, can be found on the project web page at:

<http://www.cag.lcs.mit.edu/classes/6.897/info.html>

Information which is/will be available from the web site:

- This document
- A tutorial on how to get started
- A brief overview of RAW
- The RAW specification
- A brief introduction to starsearch, the RAW simulation environment
- The class WiKi

In addition, all of the necessary files and docs can be found in the class home directory:

`/home/bits6/classes/6.897`

Be sure to check out the class **WiKi**. It is a web-based message board where you can find useful information and post questions and comments for the staff and your fellow students.

There is a link to the Wiki on the class web page, or it can be found at:

<http://www.cag.lcs.mit.edu/classes/6.897/twiki/bin/view/Six897/WebHome>

RAW is a single chip multiprocessor with both nearest neighbor statically scheduled and message passing communication. An overview of the RAW processor can be found at:

<http://www.cag.lcs.mit.edu/raw/documents/RawSpec99.pdf>

The Appendages section at the end of the document is particularly useful. It includes an instruction set list of available opcodes. Any opcodes which are not described in the appendage are equivalent to those found in the MIPS 4000 architecture. A description of those opcodes can be found at:

http://www.cag.lcs.mit.edu/raw/documents/R4400_Uman_book_Ed2.pdf

We will be using the RAW starsearch infrastructure for simulation. Preliminary documentation on the starsearch system can be found at:

`/home/bits6/classes/6.897/info/README.external`

Please note that you will need a CAG computer account for simulation. If you do not already have an account please see one of the TAs who will be happy to get you set up with one.

We are using a simulation system separate from the general starsearch system, so please don't mess with that system. Instead make a copy of the class starsearch directory in `/home/bits6/classes/6.897/starsearch`

There is also a bin directory with useful tools, such as a program to called `antint` which creates test input streams.