Edwin Louie, Ryan Metzger, Sohil Pokharna
"Three Don't Cares"

ECE 411 CP 2 Updates

What we have Done

This week we implemented the forwarding unit, static not-taken branch predictor, arbiter, caches, and all the corresponding stall and flush signals that come along with it.
Forwarding Unit + Verification Code– Ryan
Static Branch Predictor + Handle new connections into existing pipeline – Sohil
Arbiter + Caches + Assemble Code – Edwin

Testing

We wrote a testbench for the forwarding unit. Since we had limited time, we couldn't get the rvfi monitor and shadow cache working. We slowly tested the assembly code from checkpoint 1 to verify the baseline of our code is correct before testing our code on the checkpoint 2 provided code, where we then further closed bugs that we saw.

Next Checkpoint Plans

Next checkpoint, we plan on working on the following advanced features. Individual tasks will be subject to change based on availability. We will individually implement each advanced feature before merging it to the overall pipeline. We will ensure that we verify everything with the old code from checkpoint 1 and 2.

8-way L2 Set Associative Cache (Sohil, Edwin)

The 8-way set associative cache will be implemented using Pseudo LRU replacement policy. This algorithm is demonstrated on the datapath using a tree. The lookup will be done using combinational logic and the storage for the bits will use registers. Additionally, we will be making a 8-way cache so we will need to make 8 arrays for data tag ect. Instead of the 2 from the cache mp.

Eviction Write Buffer (Sohil)

The eviction write buffer will be used to hold the evicted lines before they are written back to MM. This unit will go between the L2 cache and MM. It is essentially a storage for a cacheline that is being written back to MM, the evict line will stay in this buffer and can be used while it is being written back to MM instead of stalling the cpu.

Tournament Branch Predictor (Ryan)

Tournament prediction will replace the current static not taken branch predictor. The basis for this functionality is shown in the data path.

Return Address Stack (Ryan)

The Return Address Stack will be designed to function with the JAL and JALR pointers, this stack will be implemented using a FIFO and the logic controlling the FIFO will be designed based off of the current opcode and where or not a jump or return is occurring.

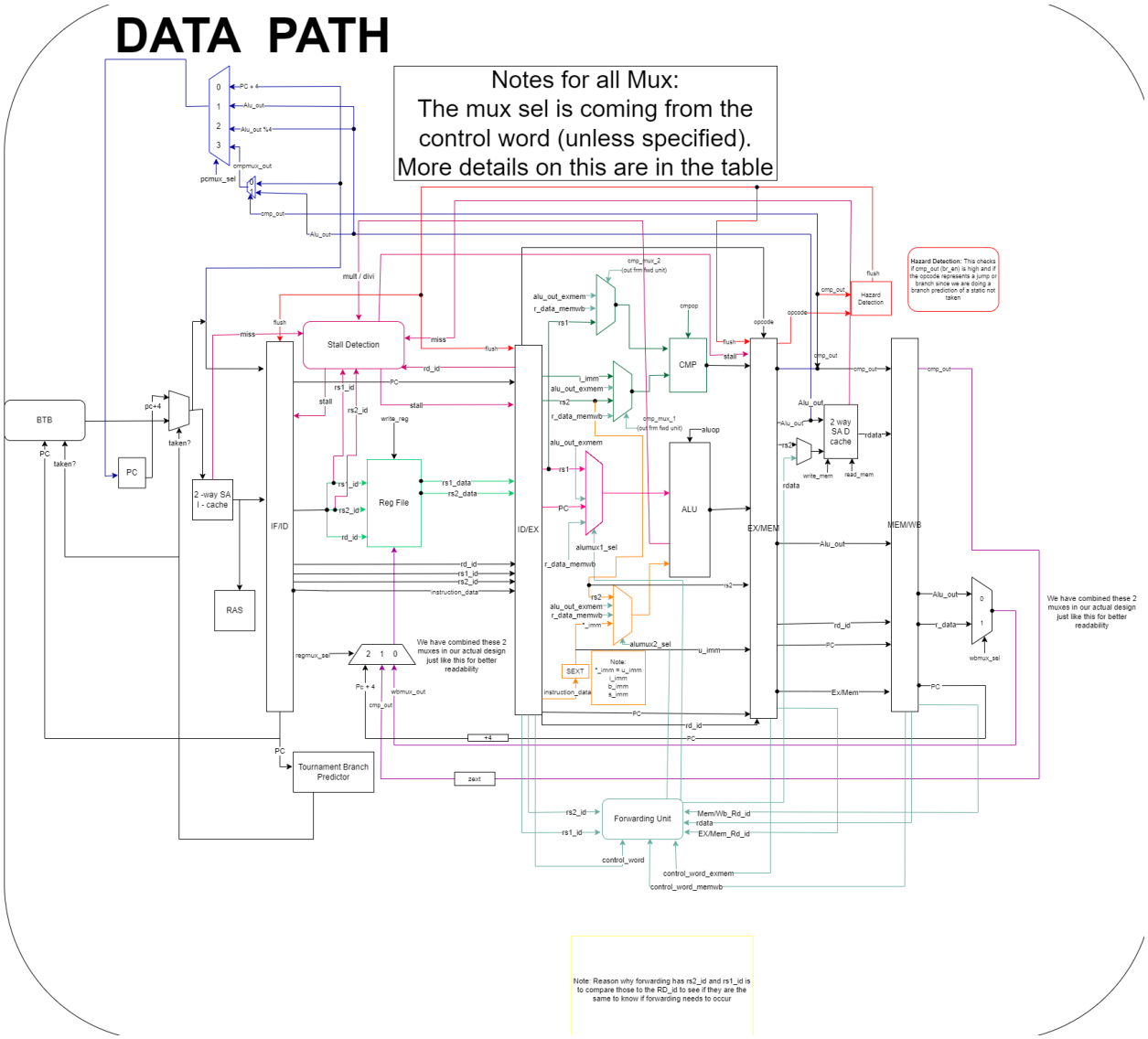
4-way Set Associative BTB (Edwin)

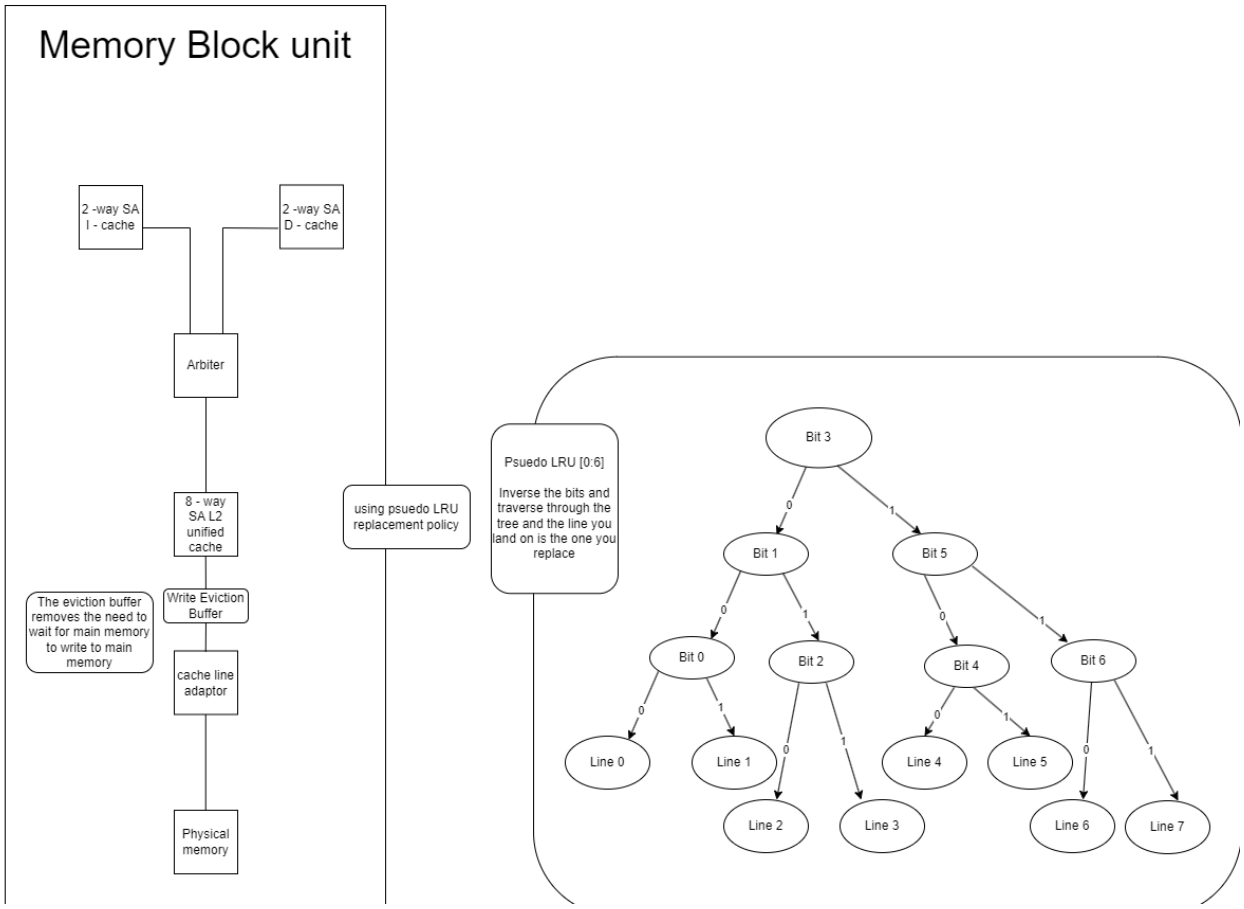
This will be 4-way set associative BTB that will aid the branch predictor. This will be like a simple 4-way SA Cache and we will store the PC as the tags and the target address as the data. (See datapath for location of implementation).

RISC-V M Extension (Edwin)

This task will involve us implementing the various MUL and DIV integer multiplication functionalities. Multiplication and division will also stall the pipeline because it will take longer than 1 cycle otherwise and slow down the entire pipeline clock rate. See multiplier topology below it is based off a booth encoding multiplier. The division will be done using either Verilog / if it is allowed or through a state machine that performs repeated subtraction and also output the remainder. template.doc (iosrjournals.org).

Updated Datapath




Memory Block unit
2 -way SA I - cache
2 -way SA D - cache
Arbiter
8 - way SA L2 unified cache
Write Eviction Buffer
The eviction buffer removes the need to wait for main memory to write to main memory
cache line adaptor
Physical memory
using psuedo LRU replacement policy
Psuedo LRU [0:6]
Inverse the bits and traverse through the tree and the line you land on is the one you replace
Bit 3
Bit 1
Bit 5
Bit 0
Bit 2
Bit 4
Bit 6
Line 0
Line 1
Line 2
Line 3
Line 4
Line 5
Line 6
Line 7

Tournament Branch Predictor