Memory Hierarchy and I/O

Andre Lancour
Victor Soto
Paul Spillane

Group 2
Instruction Set & Initialization

How much memory do we need to address?

Based how much data we need to store and how many bits each instruction uses

Do we need special load and store instructions?

No.
We are using memory-mapped I/O.

What type of memory hierarchy are we going to design?

Registers
Cache
ROM
RAM

What will we be storing?

Current time
Stopwatch time
Alarm time
Current date

Group 3
Processor Pipeline, Bus, Registers

Define a memory size for us to be able to address.

How much data do we need to store?
How many bits for each instruction?
Circular dependency!

Define memory addresses for the I/O.

We first need to determine how many I/O pins (bits) we need.

Define memory addresses for the I/O.

Considering the various buttons, segmented displays, alarm sound, and backlight...

We need over 100 I/O pins.

Will your memory be fast enough where we don't need registers?

Probably not.

Group 4
Power Consumption & Testing/Performance

How many modules do we plan on implementing (if not in one giant piece)?

Registers
Cache
ROM
RAM
I/O buses/pins

How are we going to receive and output values?

Different for each component

What unit tests we did? (To compare and possibly redo them).

We haven't done any yet.

What unit tests we did? (To compare and possibly redo them).

Some ideas for unit tests:

Change watch time
Save new time to memory
Input alarm time
Save alarm time to memory

Group 5
Integration/Project Management

Components that need to be defined.

Registers
Cache
ROM
RAM
I/O buses/pins

Estimated time of component task.

To be determined.

Estimated time to create/perform something.

To be determined.

Many circular dependencies due to questions about requirements

Questions?