A Trip to the Laundry Room

To illustrate the principle of pipelining, we will take a look at the dreaded chore: laundry.

Buddy, Holly, Michael and Jackson live in the same apartment building and because of work schedules and busy social calendars must do their laundry on Tuesday evenings. The apartment has one washer, one dryer and one folding station. The washer takes 30 minutes, the dryer takes another 30 minutes and it takes 15 minutes to fold the laundry and 15 minutes to put the laundry in a basket and take it back upstairs. If each has one load of laundry to wash, how long will it take...

1. If they do laundry sequentially (one person in the laundry room at a time).

Person	**6:30p**	**6:45p**	**7:00p**	**7:15p**	**7:30p**	**7:45p**	**8:00p**	**8:15p**	**8:30p**	**8:45p**	**9:00p**	**9:15p**	**9:30p**	**9:45p**	**10:00p**	**10:15p**	**10:30p**	**10:45p**	**11:00p**	**11:15p**	**11:30p**	**11:45p**	**12:00a**	**12:15a**	**12:30a**
Buddy	WASH		DRY		FOLD	STASH																			
Holly							WASH		DRY		FOLD	STASH													
Michael													WASH		DRY		FOLD	STASH							
Jackson																			WASH		DRY		FOLD	STASH	

6 HOURS!!!

2. If the start their clothes washing as soon as the washer becomes available.

Person	**6:30p**	**6:45p**	**7:00p**	**7:15p**	**7:30p**	**7:45p**	**8:00p**	**8:15p**	**8:30p**	**8:45p**	**9:00p**	**9:15p**	**9:30p**	**9:45p**	**10:00p**	**10:15p**	**10:30p**	**10:45p**	**11:00p**	**11:15p**	**11:30p**	**11:45p**	**12:00a**	**12:15a**	**12:30a**
Buddy	WASH		DRY		FOLD	STASH																			
Holly			WASH		DRY		FOLD	STASH																	
Michael					WASH		DRY		FOLD	STASH															
Jackson							WASH		DRY		FOLD	STASH													

Only 3 Hours!

3. If the heating element is broken on the dryer and it takes twice as long to dry clothes

Person	**6:30p**	**6:45p**	**7:00p**	**7:15p**	**7:30p**	**7:45p**	**8:00p**	**8:15p**	**8:30p**	**8:45p**	**9:00p**	**9:15p**	**9:30p**	**9:45p**	**10:00p**	**10:15p**	**10:30p**	**10:45p**	**11:00p**	**11:15p**	**11:30p**	**11:45p**	**12:00a**	**12:15a**	**12:30a**
Buddy	WASH		DRY				FOLD	STASH																	
Holly			WASH				DRY				FOLD	STASH													
Michael							WASH				DRY				FOLD	STASH									
Jackson											WASH				DRY				FOLD	STASH					

5 Hours!

4. If the next person does the unthinkable and takes laundry out of the washer and PUTS IT ON TOP OF THE DRYER

Person	**6:30p**	**6:45p**	**7:00p**	**7:15p**	**7:30p**	**7:45p**	**8:00p**	**8:15p**	**8:30p**	**8:45p**	**9:00p**	**9:15p**	**9:30p**	**9:45p**	**10:00p**	**10:15p**	**10:30p**	**10:45p**	**11:00p**	**11:15p**	**11:30p**	**11:45p**	**12:00a**	**12:15a**	**12:30a**
Buddy	WASH		DRY				FOLD	STASH																	
Holly			WASH		STACK		DRY				FOLD	STASH													
Michael					WASH		STACK				DRY				FOLD	STASH									
Jackson							WASH				STACK				DRY				FOLD	STASH					

Still takes five hours, but the washer is available sooner

Latency vs. Throughput

latency is the time it takes to accomplish one task. In the case of our laundry room, it is the time it takes to wash,dry,fold and stash laundry.

throughput is the number of tasks completed per unit of time.

The latency + throughput in our examples above:

1. latency: 1.5 hours average throughput: (4 / 6) = .67 loads/hour
2. latency: 1.5 hours average throughput: (4 / 3) = 1.33 loads/hour
3. latency: 2 hours average throughput: (4 / 5) = .8 loads/hour
4. latency: 2 hours average throughput: (4 / 5) = .8 loads/hour

Example: Consider an unpipelined system that takes 320ps (10^-12s) to complete an instruction. Compute the throughput:

```
                Throughput = (1 instruction/320ps) * (1000ps/1ns)
                              = .003125 i/ps * 1000ps/ns
                                  = 3.12 i/ns
                           =3.12 billion instructions/second=GIPS
```

In this examples, the *clock* would cycle every 320ps.

Key observations

- Pipelining does not change latency, but can improve throughput
- The potential for speedup is limited by the number of unique tasks. We call each unique task a ***pipeline stage***
- Overall improvement is limited by the slowest pipeline stage
- Unbalanced stage lengths, reduces speedeed
- When there is resouce competition, we must stall in order to maintain the pipeline

Computational Pipelining

Lets say that our throughput example from above comes from a system that has three unique computational tasks that each take 100ps to complete and then takes 20ps to update the register file. It might look something like this:

```
                       300 ps              20ps
     [    COMBINATIONAL LOGIC     ]  -> [   reg ]
     [  A  ]-> [ B   ] -> [   C   ]
      100ps      100ps       100ps
```

By introducing *pipeline registers* for maintaining state between each unique computational task, we can build a pipelined system with stages A, B and C. We call this a *3-stage pipeline*:

```
             100ps        20ps            100ps            20ps            100ps             20
      [ COMB. LOGIC A] -> [ P.Reg1 ] -> [COMB. LOGIC B]-> [P.Reg2] - > [COMB. LOGIC C]->[P.Reg 3]
```

Question:How often does the clocl cycle? What is the latency and throughput for this system?

```
The clock cycles in a wave pattern from high to low:

------       ------
      |      |      |
      ------       ------


Timing must be synchronized so that registers are fully updated and ready to become input for the next instruction when the clock cycles from low-to-high.  Therefore, the clock must cycle every 120ps.


Latency is the time for one instruction:

   (100)+(20)+(100)+(20)+(100)+(20) = 360ps

Throughput:

        1 op                         1000ns
        -----------------      *  --------------   = 8.33 GOPS
```

```
            120ps                     1ps
```

When each stage of the pipeline has the same latency, we call it a *uniform* pipeline or *uniform partitioned* pipeline. However, this is rarely the case in modern computer systems. It is still possible to build a pipelined system when different stages have different latency, but the overall performance of the pipeline will be bounded by the slowest stage.

Example We have a three stage pipeline. Stage A takes 50ps, Stage B 150ps and Stage C 100ps. Register loading takes 20ps. What is the clock cycle, latency and throughput of such a system?

```
Our clock must cycle based on the longest delay which occurs in Stage B.

  Clock = (150ps) + (20ps) = 170ps

latency will increase because each stage must be synchronized with the clock which is timed off of the slowest
   pipeline stage:

 Latency = nClock  where n is the number of stages:
               (3)(170ps) =  510ps

 Throughput =          1 operation               1000ps
                      -----------------   *     -------------   =  5.88 GOPS
                          170ps                    1ns
```

Example: Draw a pipeline diagram for this system as it executes 3 instructions.

Instruction	**CC 1**	**CC 2**	**CC 3**	**CC 4**	**CC 5**
I1	A	B	C		
I2		A	B	C	
I3			A	B	C

Example: Suppose we have 6 unique computational units that execute sequentially, named A-F and having delays of 80ps, 30ps, 60ps, 50ps, 70ps and 10ps, respectively. Register delay in this system is 20ps.

1. Where should we insert a register to create a two-stage pipeline? What would be the throughput and the latency?

```
       if we place the register between C and D, the delays are:

        stage 1: A+B+C+r = 80+30+60+20 = 190ps
        stage 2: D+E+F+r = 50+70+10+20 = 150ps

       latency = n*Clock
         our clock will have to cycle at 190ps, so latency = 2*(190) = 380ps

       Throughput =      1 op                   1000ps
                       -----------      *       ----------- = 5.26 GOPS
                             190ps                 1ns
```

2. Where should we insert two registers to create a three-stage pipeline? What would be the throughput and the latency?

```
     if we place our pipeline registers between B&C and between D&E, the delays are:

        stage 1: A+B+r = 80+30+20 = 130ps
        stage 2: C+D+r = 60+50+20 = 130ps
        stage 3: E+F = 70+10+20 = 100ps

        latency = n*Clock
          clock will cycle at 130ps, so latency = 3*130 = 390ps

       Throughput =   1 op            1000ps
                     ---------  *     ----------    = 7.69 GOPS
                       130ps             1ns
```

3. What is the throughput and latency of a 5-stage pipeline implementation?

```
     stage 1: A+r = 100ps
     stage 2: B+r =  50ps
     stage 3: C+r = 80ps
     stage 4: D+r = 70ps
     stage 5: E+F+r =100ps

     latency = n*Clock
       clock will cycle at 100ps, so latency = 5*100ps = 500ps

     Throughput =   1 op        1000ps
                 ---------  * ------------   =  10.00 GOPS
                   100ps           ns
```