

Project Management

3/23

- WBS (Work Breakdown structure):
 - list and details of every part in the project
 - e.g. (LED com.) identify intensity of overall project
 - " " " individual LEDs

- Rule by 6:
 - have outsiders help give an estimate
 - set short, estimate, and long times
 - $\therefore Rb6 = \frac{1(S) + 4(E) + 1(L)}{6}$

- Color mapping:
 - Green - could do
 - Yellow - possible
 - Red - no idea how

- Label steps in WBS
 - 1
 - 1.1
 - 1.2
 - :
 - 2
 - :
 - etc.

- Pert chart:
 - leave blank for what you can finish today
 - Dependencies & non-dependencies

overall example:

- 1.1 identify intensity of overall project
- 1.2 " " " individual "
- 1.3 determine the # of LEDs
- 1.4 determine sensitivity of LEDs

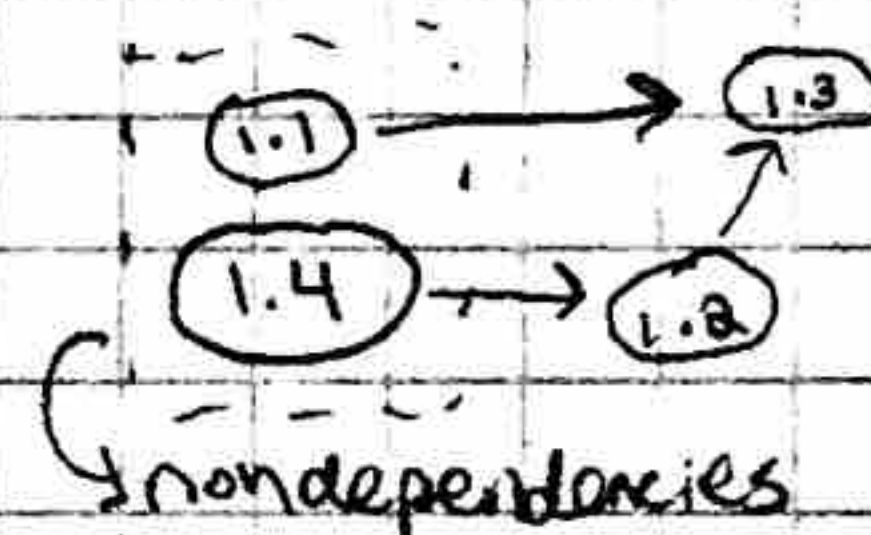
Rb6

Dependencies

Pert:

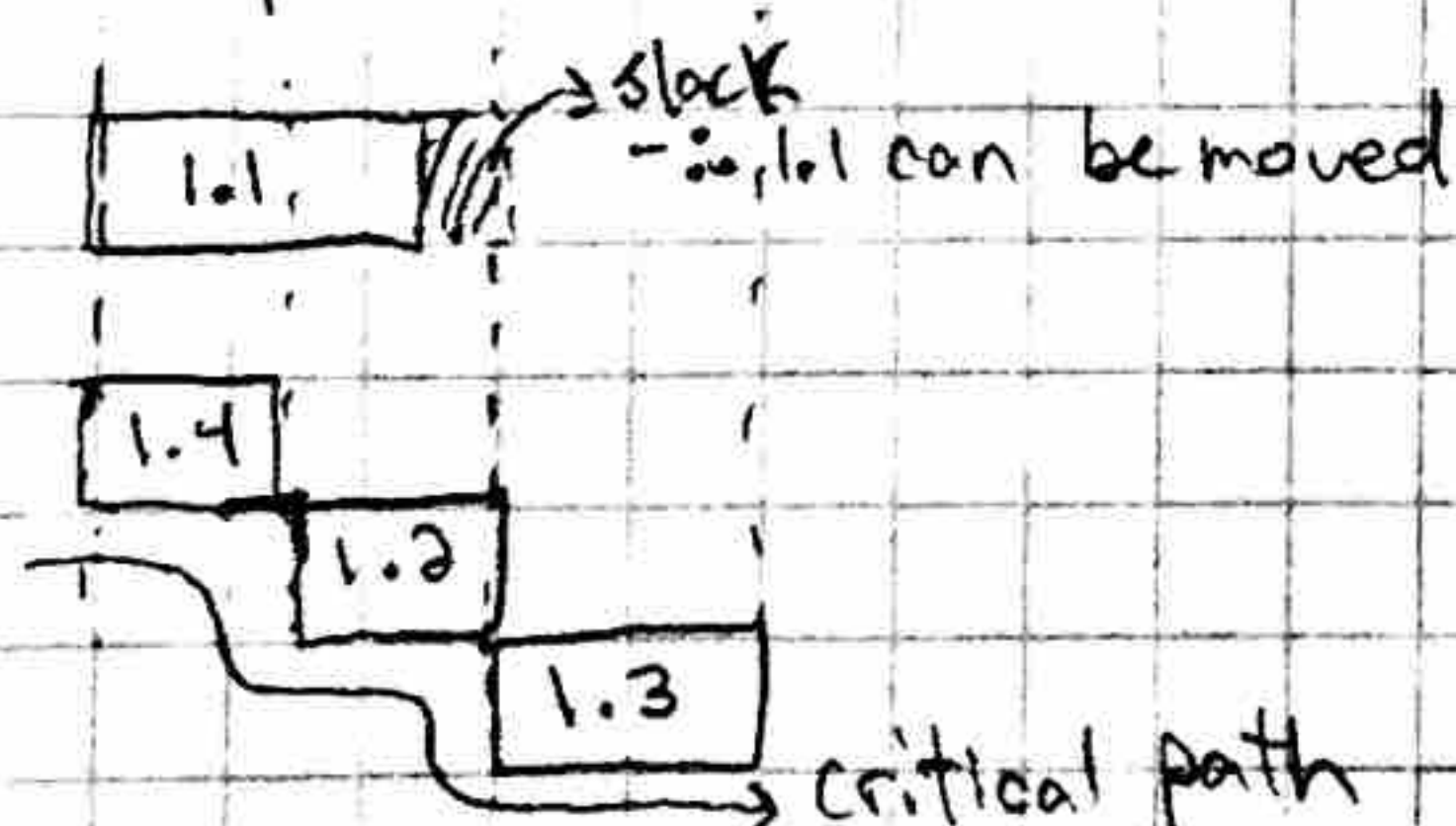
G
Y
R
R

1.4
1.1, 1.2



- Gantt chart:
 - adds time to project/WBS (comes from Pert)

example cont.



- Critical Path:
 - Finds largest path through Gantt chart (see above)
 - allows for other WBS to move (slack)
 - of person can get trained for a task later down the road
 - person on 1.2 can get trained during 1.4

- Crash:
 - shorten 1.2 for example to shorten critical path (found through slack)

- Take home message for engineers \rightarrow organize/time mang. WBS, Color map, Rb6, Pert (1)

8/25

Pert Chart/Gantt example:

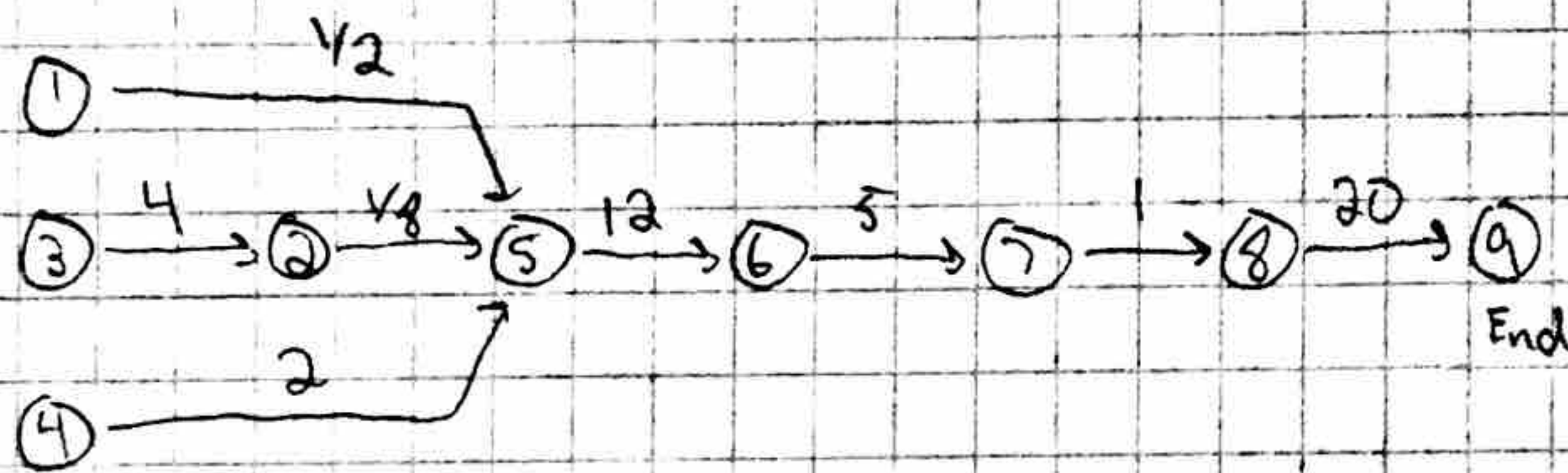
LED communication:

key points:

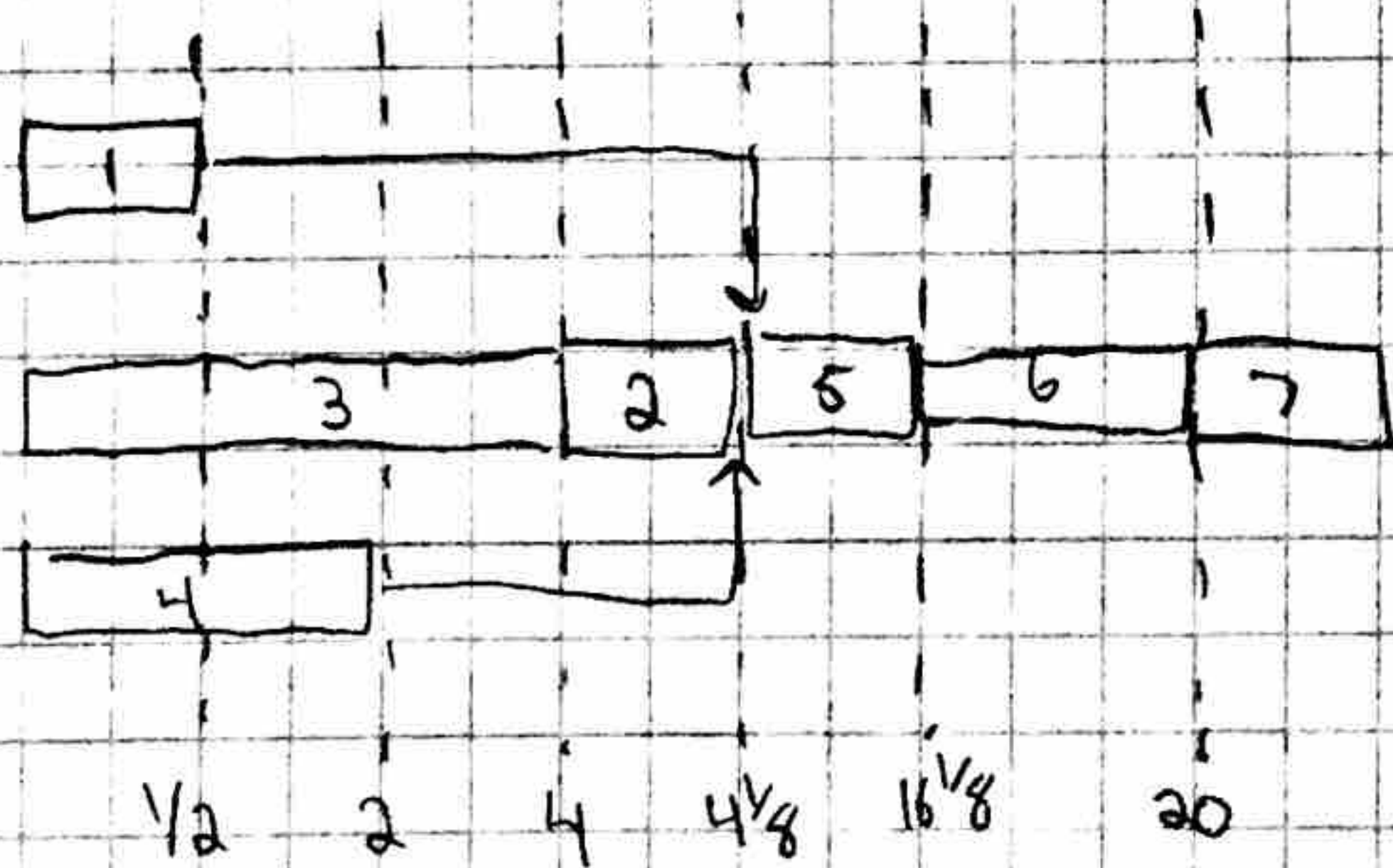
- Light Room
- Transmit info.
- 4 hubs

	R66 (time):	Dependencies:
1 Light intensity	$\frac{1}{2}$	
2 Brightness of source	$\frac{1}{8}$	3
3 LED as receiver	4	
4 Physio. impact	2	
5 Transmission char.	12	1, 2, 4
6 message char.	5	5
7 Location of hubs	1	6
8 Distribution mechanism	20	7
9 Close out & Documentation (User guide		

Pert:



Gantt:



Sumo Bot (WBS)

		Rb G				Dependencies
		S	E	L	G	
1	Research SAC Rules	1/8	1/2	1	1/4	
2	Research uController/sensors	1	2	3	2	
3	Gather/Assemble Chassis (base)	4	7	7	6.5	1
4	Gather specifications/dimensions	1	2	3	2	3
5	Design in AutoCAD/3D print	6	8	10	8	4
6	Create movement logic/alg.	1	2	3	2	
7	Get u controller running / Gain knowledge.	1	4	6	4	
8	- pins	1	2	3	2	7
9	- sensors	1	2	3	2	8
10	Implement software with bot	10	19	20	17.5	4, 6, 9
11	Fix alg. and buss					11

Implemented in Microsoft Project.

4/30 Topics for today:

- Standards
- Notebook & Portfolio
- Schedule
- Cooperation

• Cooperation:

- out-source help to get tasks done
- help others - make trades to get projects done - teamwork

★ Notebook/Portfolio:

- 4 sections

1.) Projects

- Description
- Deliverables
- Milestones
- Final Project Report
- Project Management concepts
- Extensions/changes
- Links to notebook, SLOs

2.) Notebook

- Weekly summary (few sentences)
- Scanned images of notebook
- ↳ Expect at least page for every class day.

3.) SLOs

- Page for each program
- description
- courses list to complete SLO
- Add statement to how each course contributed
- This course specifically addresses
 - include Artifact - link to project

4.) Video Site

1.) Project Proposal

2.) Project Report/Progress

- ↳ supervisor

3.) ↳ Team

4.) Project Sales pitch to customer

5.) Final Report supervisor

6.) Project Closeout Report

- documentation, success/failure, what went right/wrong, etc.

★ Portfolio - Grades come from here

↳ Post everything here

- Schedule:

- follows Gantt chart

- check due dates

- alter schedule - sit down with Dr. Spalletta

- First Project → create list of supplies (Sumo Bot)

- uController (Arduino)

- Sumo Bot chassis (DC Motors)

- servos w/ pins → look into price/weight

- Sumo Ring (I believe there is one constructed)

- Filament (3D print body)

- Sensors (IR) (pressure)

- Test bot ↳ start here

Majid
↳ Test bot

- Consider Hardware vs. Software

Software - flexible

Hardware - less programming (build and forget)

Project Computing:

Real Time: (interrupts)

Events

↳ interrupt

- tell processor to stop what it is doing and reconfigure
↳ e.g. keyboard press → interrupts system → prints → goes back to previous task

- Each interrupt has:

- Priority - two interrupts at once, which one first (hardware)
- Recursion - can interrupt be interrupted? (software)

- Types of interrupts:

- mask - software can tell to ignore
- non-mask - cop - cannot be ignored
 - high priority
 - power level drops (example)
- Recursion - occurs whether or not an interrupt can be ignored
- Priority structure (Fixed in hardware)

Processors:

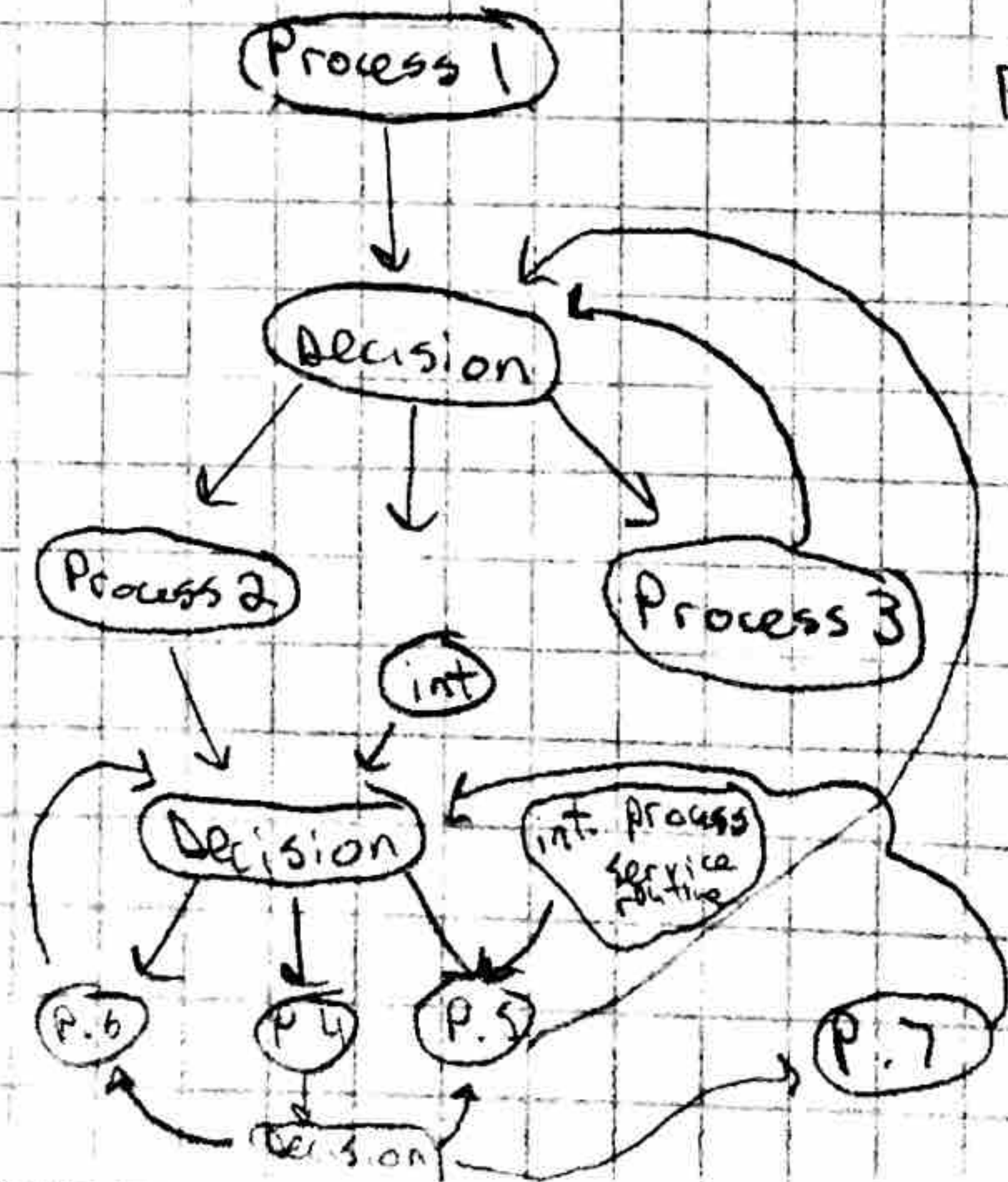
Pi - threaded system

Dragon 8.10 - Hybrid system

- Android - threaded
- uController - Real time

Arduino - run both ways

- software → not real time
- sequential system not real or threaded (Not recommended)



Basically software state machine

AVE

Sumo Bot

Milestone:

Comments:

- | | | |
|-------------------------------|---------------------------------|-------------------------------|
| 1) Choice of all components | 1-2 days to decide → + shipping | (week most likely)
Sept 13 |
| 2) Sensors Working | 5-7 days | Sept 20 |
| 3) Movement obtained | 2-3 days | Sept 23 |
| 4) Trial strategy implemented | | Oct 5 |
| 5) Strategy Tested | | Oct 14 |

Notebook sites:

- Log in
 - ID#
 - Password

- Posts
 - Blog like
 - daily updates

- Pages!
 - detailed projects

- Videos
 - Link to embed video in page