SOFTWARE DIAGRAM:

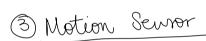
(conveyor belt) runs for X reconds

Notes: X deuter an amount of time that will be specified at a later stage (through experience). * denotes the different clarrer (see following pages).

Potential additions to the code are written in gray.

(from motion source*) (wer input) & Motion=0 Motion = 1 Servo 1 Jurus back 90° Servo* 1 turns _ Wait for. X records 900 Wait for button* (marble input) < Button=1 Button = 0 2) Potentially: ask user "Are you ready to play _ and give some number outhuctions. Ideally LED dimlay* turns on no user right needed; game starts after X time > Toysticle* allow susceed (ment) > LED game Stants if lost: restant up to 3 times wait for game to finish , Finish=1 if wow (or if lost 1 stimes) Servo 2 turus 90° Wait X seconds Continuous servo

CLASSES:	
D Servo For: Servo 1, Servo 2 & Continuous Servo D Button	evill have to undify the parameters for the project, but the code was given in class.
3) Nation Seuror 4) Layrtick 5) LED display	



Wait for \leq Mation = 0

Wation = 1

Call Serva 1

to start

Q: does the MS ignore further movement if The just sustancted Servo 1 to start? (eg: set X secs)

9 Jaystick

Noit for I Button = 1 Button = 0 Wait X (while LED turns on / potentially seconds Wait for wait joi user movement (up, down, left, right) \finish=0 Instruct to Move on LED accordingly

Finish = 1

(5) LED display	
Note: will le the game of	ook more ruto the specifics of uplemented
\ -b	lait for 5 Button=1 utton press Button = 0
ik	splay on potentially what sw desorbed in main SW diagram
if lost: restaut up to 3	Stants LED display, sparts bank when tuput (Tougstick)
times \ w	Joint for Jenish and to fivish = 0 if won (or if lost atimes)
	Turn LED (potential final display off menage)
	Call Servo 2 to start