



DESIGN GOALS:

- ROMI SHOULD FINISH THE TRACK NO MATTER WHERE HE STARTS

NOTES:

TO GET AWAY WITH NOT USING IMU,
JUST HAVE ROMI DRIVE STRAIGHT
WHEN NO LINE DETECTED. IF NO
LINE IS DETECTED FOR X_{max} , THEN
TURN RIGHT 90° & DRIVE $y_{max}/2$.
YOU COULD "JIGGLE" AROUND UNTIL LINE
DETECTED. THEN, ~~DRIVE~~ LINE
FOLLOW UNTIL YOU HIT WALL. NAVIGATE
WALL & YOU'RE DONE.



WANT TO HAVE MODES OF OPERATION
& A "BOSS" THAT SWITCHES BTWN THEM

EX:



"WE'VE BEEN
DRIVING STRAIGHT
FOR 3,000,000
ENCODER TICKS,
LET'S TURN 90." "

"OKAY LINE DETECTED,
LET'S FOLLOW THE
LINE,"

NOTES"

WE CAN IMPLEMENT THIS WITH TASKS,
WE WILL HAVE A TASK FOR THE THINKING,
TURN 90°, FOLLOW LINE, ETC.

TASK NAMES:

MASTERMIND	{ }	DECISION-MAKING ALGO
FOLLOW LINE		SELECTED BY MASTERMIND
DRIVE STRAIGHT		
TURN 90		
ACTUATE MOTORS	{ }	MODES SHARE Vref FOR MOTORS
BUMP SENSORS		SELECTS ITSELF

SO, THERE ARE FOUR DRIVE MODES:

FOLLOW LINE
DRIVE STRAIGHT
TURN 90
BUMP SENSOR

BUMP SENSOR ASSERTS ITSELF & TERMINATES PROGRAM.

OTHER WISE, MASTERMIND DECIDES DRIVE MODE.

THE ACTIVE MODE DECIDES Vref FOR R & L MOTORS.

NOTES:

SHARED VARIABLES WILL BE mode-select.
R-v-ref, { L-v-ref, ~~code~~.

R-v-act L-v-act









