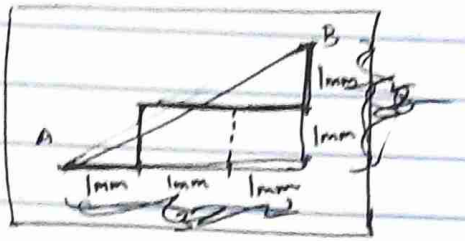
PICT, PUNE



(penX, penY);
(usrX, usrY);

$\bar{v} = (usrX - penX)\,\hat{i} + (usrY - penY)\,\hat{j}$

define $S_x = |usrX - penX|$
& $S_y = |usrY - penY|$.
i.e. S_x & S_y are the no. of steps to cover to reach from (penX, penY) to (usrX, usrY).

To find: the delays b/w 2 consecutive pulses applied to the X & Y motors so that the pen traces a straight line from A to B; i.e. depart from A & reach B in the same duration 't' following a visibually straight line path.

$t_{dx} \propto \frac{1}{x}$ & $t_{dy} \propto \frac{1}{y}$ (∵ as Δ increases, if a single move instruction is to take constant time, the distance should reduce delay b/w consecutive pulses accordingly.)

also, the move instruction starts & ends with a pulse. Delays are interleaved b/w consecutive pulses;

<move starts> - P - D - P - D - ..... - D - P - <move ends>

This implies the number of delays produced are 1 less than the number of pulses applied.

$\Rightarrow D_* = S_* - 1$ $\left[\begin{matrix} D_x = S_x - 1 \\ D_y = S_y - 1 \end{matrix}\right]$

PICT, PUNE

- Let T be the time to complete 1 move operation. Assume it to be configurable by the user programmer.

- The total delay = (delay b/w consecutive pulses) * (No of times the delay occurred)
  = T
  $= t_{dx} \cdot (S_x - 1) = t_{dx} D_x$
  $= t_{dy} \cdot (S_y - 1) = t_{dy} D_y$.

$\therefore \boxed{t_{dx} D_x = t_{dy} D_y}$ $\quad (D_x = S_x - 1 \; ; \; D_y = S_y - 1)$.

$\Rightarrow \boxed{\frac{t_{dx}}{t_{dy}} = \frac{D_y}{D_x} = \frac{S_y - 1}{S_x - 1}}$ & $t_{dx} = \frac{T}{D_x} \Rightarrow \boxed{t_{dx} = \frac{T}{S_x - 1}}$

& $t_{dy} = \frac{T}{D_y} \Rightarrow \boxed{t_{dy} = \frac{T}{S_y - 1}}$

- Let t_p be the time required to step either motor in either directions.

- Thus, we can say: $T = S_x t_p + \underbrace{(S_x - 1)}_{D_x} t_{dx} = S_y t_p + \underbrace{(S_y - 1)}_{D_y} t_{dy}$
  where t_{dx} & t_{dy} are delays b/w consecutive X & Y pulses respectively.

PICT, PUNE

- Thus, $t_{dx} = \frac{T - S_x t_p}{S_x - 1}$ & $t_{dy} = \frac{T - S_y t_p}{S_y - 1}$

- S_x & S_y are the total steps taken to reach from A_x to B_x & A_y to B_y respectively.

i.e. $S_x = \frac{B_x - A_x}{(\text{linear step distance})}$; $S_y = \frac{B_y - A_y}{(\text{linear step distance})}$

- 'linear step distance' can be calculated by:
  (i) assemble the plotter completely;
  (ii) attach the pen to pen-end;
  (iii) (a) write a program that rotates the motor only, fully (360°), whilst keeping the pen down based on the stepping mode selected. Measure the marked line's length.
  (b) Next, rotate the motor by 180° & measure the new line's length.
  (c) Finally, rotate the motor by 90° & measure the last line's length.

declare as a macro or a global const in the config file

Find Linear step Distance (a) = $\frac{\text{measured distance in (a)}}{(\text{steps for 1 complete rotation})} 200$

Linear step Distance (b) = $\frac{\text{measured distance in (b)}}{(\text{steps for } \frac{1}{2} \text{ rot.})} 100$

Linear step Distance (c) = $\frac{\text{measured distance (c)}}{(\text{steps for } \frac{1}{4} \text{ rot.})} 50$

In the end, $\text{Linear step Distance} = \frac{\sum_{i=a}^{c} \text{Linear step Distance}(i)}{3}$

(one can take >3 readings for greater accuracy).

PICT, PUNE

penX, penY, usrX, usrY

↓

Calculate Δx, Δy as usrX - penX, usrY - penY

↓

Set direction accordingly

↓

Linear Step Distance → Calculate S_x & S_y from Δx, Δy & linear step distance

↓

T, t_p → Calculate t_{dx} & t_{dy} from S_x, S_y, T, t_p

↓

Set 2 timers to generate interrupt after t_{dx} & t_{dy} respectively. Enable Timer interrupts

↓

→ Wait for interrupt to arrive

↓

If t_p is small enough, pulse the respective motor in the ISR itself

↓

if target is reached

- no → reset respective timer to generate interrupt after appropriate delay. → (back to Wait for interrupt to arrive)
- Reset the respective timer to generate interrupt → (back to Wait for interrupt to arrive)
- yes → Disable timer interrupts → return