Lab 8 : Banma :

① Lab7 - part B ✓
②

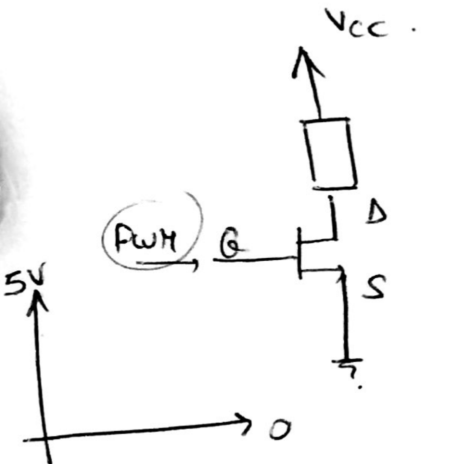
PWM .

① DC Motor : 2 opposite magn. fields..

→ to control of DC motor :
- ↗ speed : depends on pwm
- ↘ direction : depends on Hbridge .

① Controlig Speed of DC Motor → vary input voltage .

→ usually from pic,
we don't have enough current to drive the motor ⟹ we need to amplifie using PWM :



for PWM from pic → 5V used just to open / off transistor

∴ PWM input to transistor : connect motor between Vcc .

↳ when varying duty cycle :

$$V_{average} = 20\% \times V_{cc}$$

N.B : average voltage is independent of the frequency.

∴ Vaverage delivered to the motor $V_{cc} \times$ this duty cycle.

↳ PWM signal must have frequency : (certain specs for motor) .

- for frequency has 2 limits :
  - max frequency
  - min frequency ..

/2016

- max frequency:

→ higher switching frequency for transistor
  - ↳ more heat
  - ↳ transistor switching super fast => Bye Snab.

- min frequency:

→ motor won't see DC voltage: jerking on and off:

because only seeing DC on and off.

$$f_{chosen} = 10KHz$$ (*)

↳ whenever using DC motor: **

- use Catch diode, snubber, diode, fly back diode, freewheeling suppressor → what's it used for?

- model for the motor → 
  (since it has coil)

very big change of current for a very short time:

$\frac{di}{dt}$ super high → $L\frac{di}{dt}$ is super high → might damage all comp in circuit.

- take good care for direction.

flyback = spike.

inductive load.

/2016

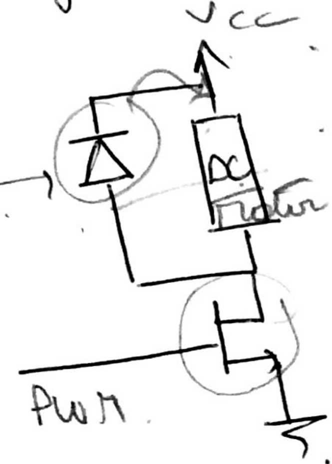
° we add "Diode" // to the motor:

→ useful to dissipate the energy of internal resistance.

in parallel

→ fly back diode

N.B: direction of diode is super important →



parallel flyback diode.

° Opto isolator: → isolation between 2: good interface between pic and transistor.

↓

→ Lab 9

Lab 9, 10

5V

① power pic → Gd 5

② power motor → 12V, gd = 12.

③ separate Grounds

· PWM

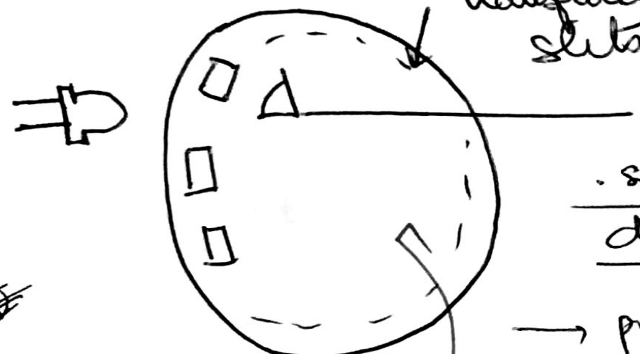
2016

How to measure speed of motor:

→ using shaft encoder that detects speed of DC motor.

↳ Rotatory Optical incremental Encoder ≡ Shaft Encoder.

• **How does this work?** → small dip to get resolution •



signal shape of light detector:

→ pulses: output of photosensor

1024 : # of slits

resolution of encoder.

↓ to get rpm:

$$\frac{\text{Resolut. of encoder}}{}$$

↳ measure pulses / sec, then ÷ by # of slits (or) ÷ Resolution of Encoder Shaft

↳ $\frac{\text{frequency}}{\text{Resol. of Shaft Encoder}}$

Encoder Connections

→ **N.B:** A and B are 2 channels that are in 90° phase shift.
↳ to be able to get direction of rotation using tables [given usually].

Resolution = 1024

NB: Ex.1: ∘ measure voltage across DC motor.
∘ connect channel A to oscilloscope to read frequency = pulses/sec (a square signal)

↳ Speed = $\frac{\text{freq}}{\text{Resol (given)}}$ ✓

/2016

Part 2: ① PWM from pic:

② Read speed from motor

↳ we need to count # of ~~cycles~~ pulses/sec.

Code: → to read # pulses/sec then divide resolution

↳ use exactly same: Timer 0 to generate interrupt every one second.

↳ Timer 1: external source: to count # of pulses from external source.

8 bits	8 bits
High	Low

↳ Timer 1: overflows $2^{16} \simeq$ after 65 KHZ, since the motor can reach much higher frequencies ⇒ we need to add another variable of 8 bits

⇓

postscale

Timer 1 Control:

0 ⟶ 7. (bes we write 0x7 ⟶ 0).

0 1 2

1 1 1

↓ ↓ ↓

Enable Timer 1. | external | not sync. with internal

NB: post scalar is just like loop used in previous labs. ⟶ int = 8 bits

---

Issue:

external interrupt TMR1

internal - TMR0

8	8	8
post scalar	High	low

⟶ **In what order it reads:**

for low i/p bits:

High: shift 8 bits

Postscalar: shift 16 bit ⟵ (16)

Pulses_per_sec = ((long) TMR1H << ?) + TMR1L).

↳ what type? int: not enough. (8)

we need 24 bits

/2016

NB: to test code on proteus, give pwm as input, give ~~[illegible]~~ input

[set finput → we need to reach fout)

measured.

↳ Timer 1 only needs f

RCO → T1clk

Pulses

∘ 1sec → 15sec

↪ TMR0: interrupt every

→ internal interrupt.

256 x 256 x 15

for one sec