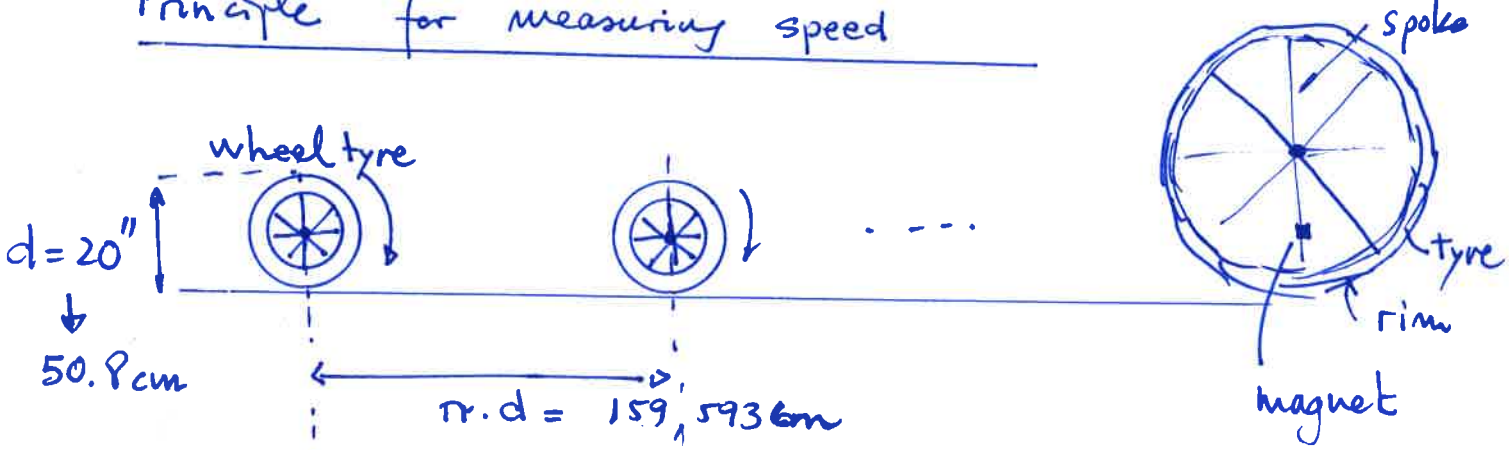
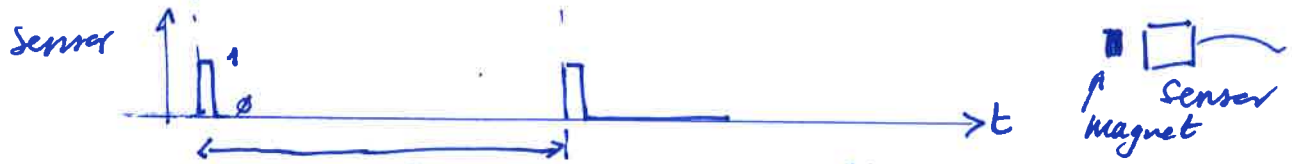
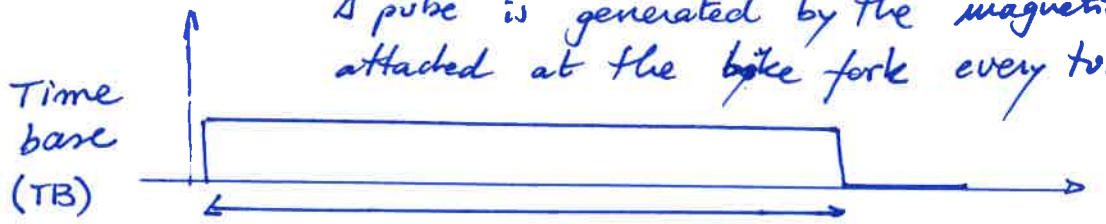
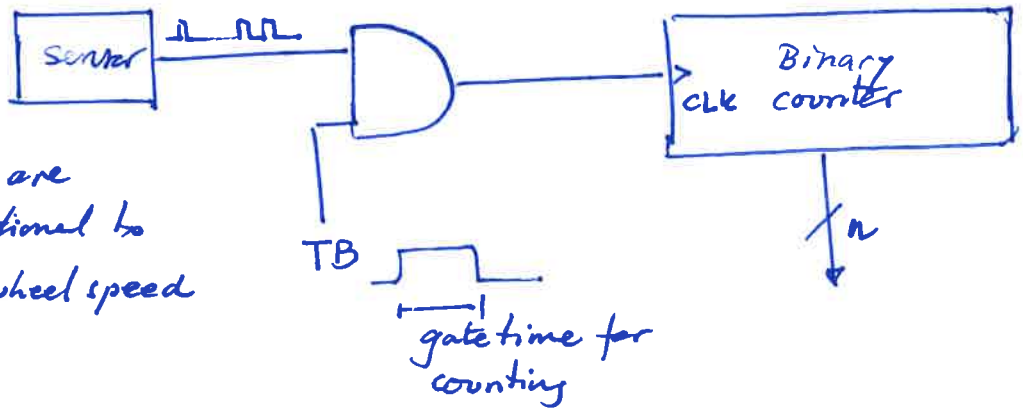
Principle for measuring speed





A pulse is generated by the magnetic sensor attached at the bike fork every turn





Pulses are proportional to the wheel speed

For instance : resolution 1 km/h = 0,277 m/s

100 km/h max → 100 pulses → n = 7 bit

resolution 0.1 km/h ⇒ 1000 pulses → n = 10 bit

100 km/h max

0.1 km/h .... 99.9 km/h

A single magnet will generate 1 pulse every turn

A tooth wheel with 8 magnets will have more resolution as will generate 8 pulses per turn.

Let's relate distance, speed and Time base for a given resolution

A single magnet $d = v \cdot t$

$$1.59593\ m = 0.27\overline{7}\ m/s \cdot TB$$

The necesary window $TB = 5,745344\ s$
to measure a single pulse ↗

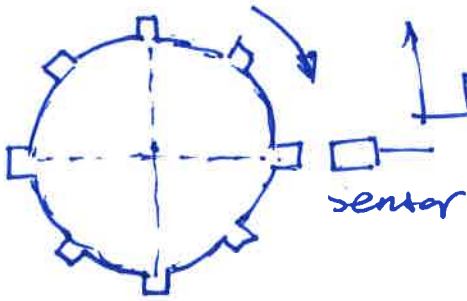
This time window (gate time) will allow at least the counting of 1 pulse when riding at minimum speed of 1km/h.

$$fp_{min} = 0.174\ Hz$$

At 100 km/h → $fp_{max} = 17.4\ Hz$

Thus, there's a new measurement every 5.75 s

If more resolution is required while keeping the TB in a few seconds ⟹ more magnets



tooth wheel or gear

8 pulses every revolution

The distance between pulses is $\frac{\pi \cdot d}{8} = \frac{1.59593\ m}{8} = 0.19949\ m$

For instance, to measure a single pulse at the minimum speed resolution of 0.5 km/h

$$0,19949\ m = 0,13\overline{8}\ m/s \cdot TB$$

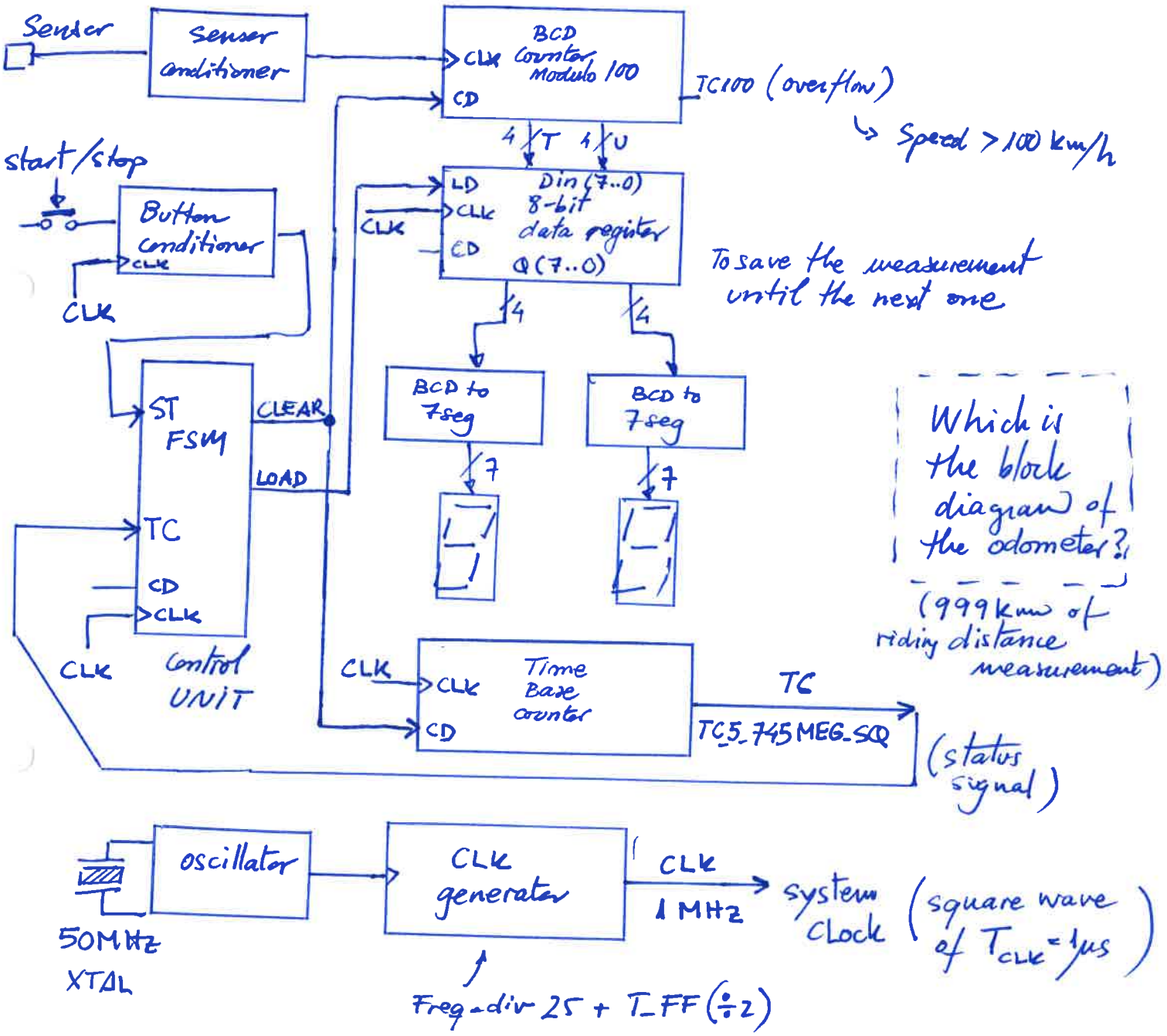
A new measurement every 1.5 s ← $TB = 1.43635\ s$ (8bit)

and so, at 100 km/h → the system will count 200 pulses.

Planning the circuit

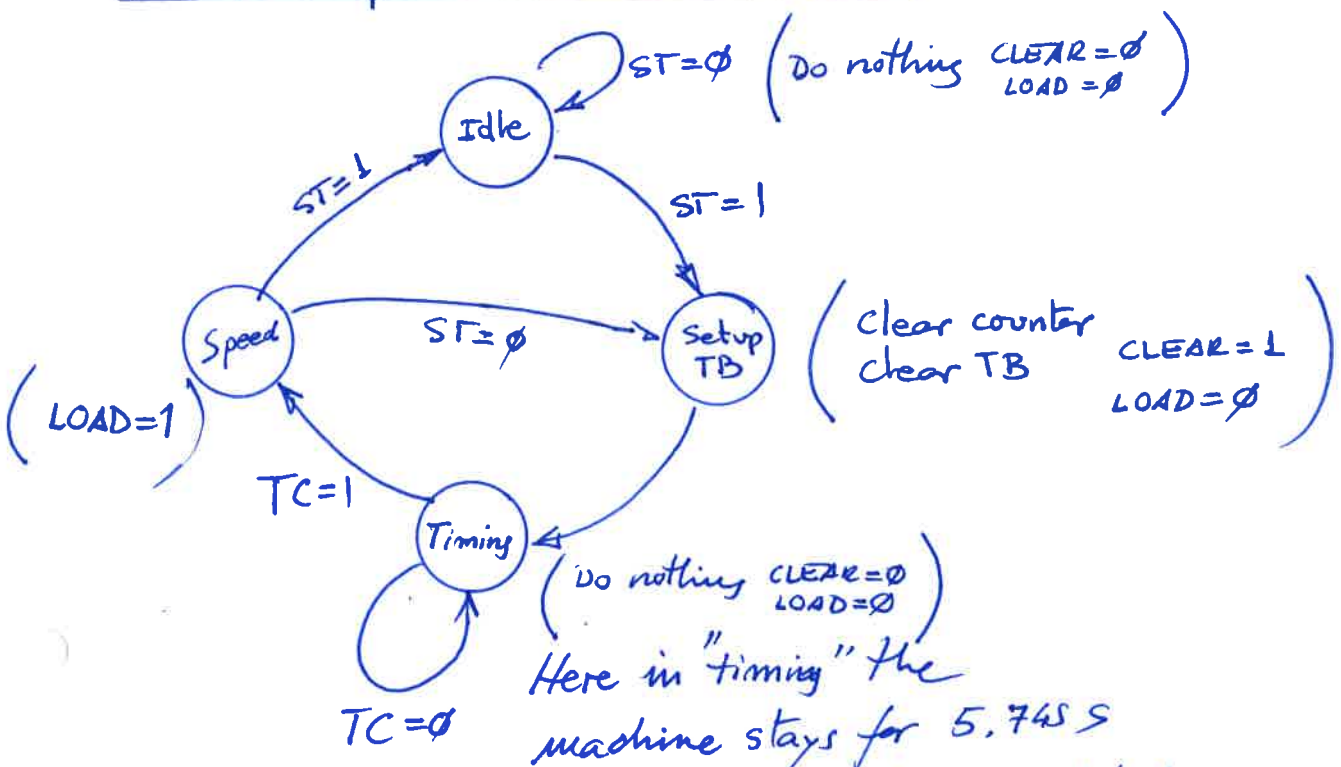
As a P8 project → dedicated processor
(Datapath + FSM)

* 1st schematic (block diagram)



- When ST/SP is pressed the button conditioner generate a 1μs synchronous pulse to start the FSM
- When running the time base "timer" counts 5.745.000 pulses to generate the TB gate window
- When running the BCD counter accumulate the pulse count from the sensor
- When TC = 1 → LOAD = 1 to save and display the speed

Example of FSM state diagram



Here in "timing" the machine stays for 5.745 s

- Generate a 2nd schematic fully annotated
- How many VHDL files are required?
- Where can you find in DIGSYS similar files to copy and adapt?
- How many 1-bit memory cells will require this application? (D_FF?)
- How to design the button conditioner? (see push-button debouncing circuits in P6)
- Start a VHDL project using EDA tools for a given FPGA/CPLD chip
- Inspect the RTL schematic
- Test the circuit using a test bench in VHDL and a simulator

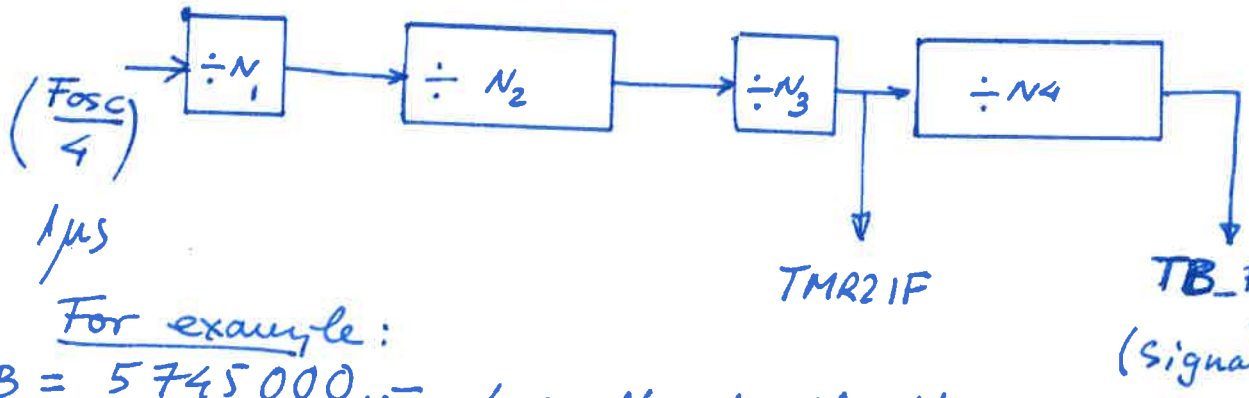
Design using a μC (PIC18F4520)

⇨ Use, for instance the Timer_18.5 s (P11) and adapt it to this speed_meter project.

⇨ So, it'll become the project P13 in CSD taking advantage of the previous ones.

⇨ ANALYSE THE CODE + PROTEUS circuit given as exaple

* Time base generation → Timer2 (TMR2)



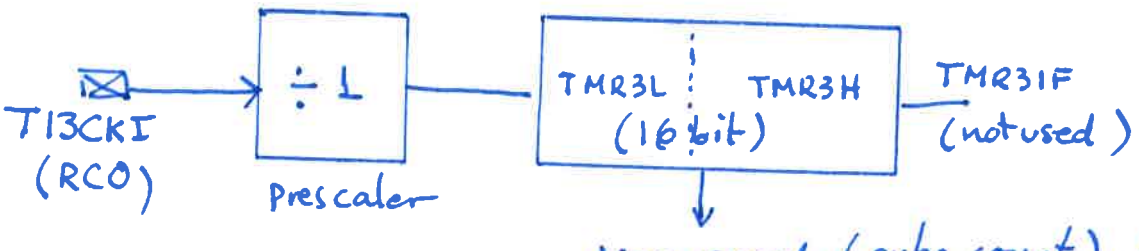
For exaple:

$TB = 5745000\,\mu s = 1\mu s \cdot N_1 \cdot N_2 \cdot N_3 \cdot N_4$

4 (pre-scaler) · 250 (TMR2 (PR2)) · 15 (post scaler) · 383 (software int variable)

TMR2IF every 15ms

* Counting external pulses from the sensor → Timer 3 TMR3



var_speed (pulse count) has to be read after the time-base gate time (state meas_speed)