



Complete, fold and seal
(see note 2 below)

SURNAME: H

FORENAME(S): SAFFINA

Examination Answer Books

12 Pages

Instructions to Candidates

1. Please give the information requested in the boxes below.
2. Enter your name in the space provided to the right, fold over and seal down (using the label provided) to hide your name.
3. The answers must be legibly written. The Examiners can either ignore illegible script or have it typed and the expense of typing charged to the candidate.
4. Write your answers in ink.
5. Begin each answer on a fresh page. Do all rough work in the answer book and cross it through.
6. No part of this book is to be torn out; it must be handed in complete.
7. If more books than one are used, fasten them together using the tags available from the invigilator(s).
8. **Every book must be handed in. In no circumstances may answer books, used or unused, be taken out of the examination venue.**

1	Student Number	2322601
2	Title of Examination Paper	MICROS PERIPHERALS & INTERFACES
3	Module Code	ACDA4004
4	Date and Time of Examination	08/01/25 09:00
5	Examination Venue	1Q310
6	Seat Number	018
7	Number of Answer Books handed in	

List the numbers of the questions you have attempted	Office Use
1	10
2	10
3	3
4	10
5	10
6	5
	48
	22
Total	96%

10
10
3
10
10
5
48

- ① a) RETURN - means to return from a subroutine, this will jump back to the line of code in the program that initially called the subroutine and ~~then~~ then proceed sequentially through the code.
→ stands for return
- b) SLEEP - means the device will be put to sleep, and go into a stand-by mode $\checkmark$ not executing any instructions until 'woken-up'. (pressing and)
→ stands for sleep
- c) ADD LW - the ADD stands for the add operation, the L ~~stands for~~ ~~stands for~~ a literal value, and the W stands for the working register. The ADDLW instruction will sum the numbers stored in the working register with the literal number provided. The output of the calculation is then stored in the working register
- d) BCF f, b - BCF stands for 'bit clear f', which means that it will set a specified bit ('b') in the 'f' register to 0, clearing it.
- e) DECF - stands for decrement f. This reduces the value stored in the 'f' register by 1, ~~then~~

2

2

2

2

2

2

2

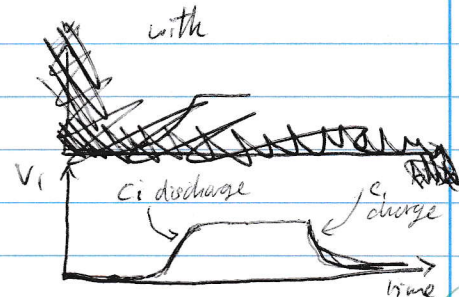
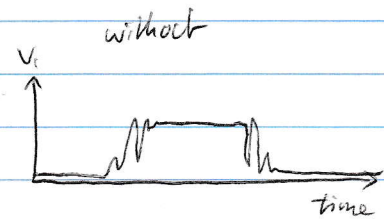
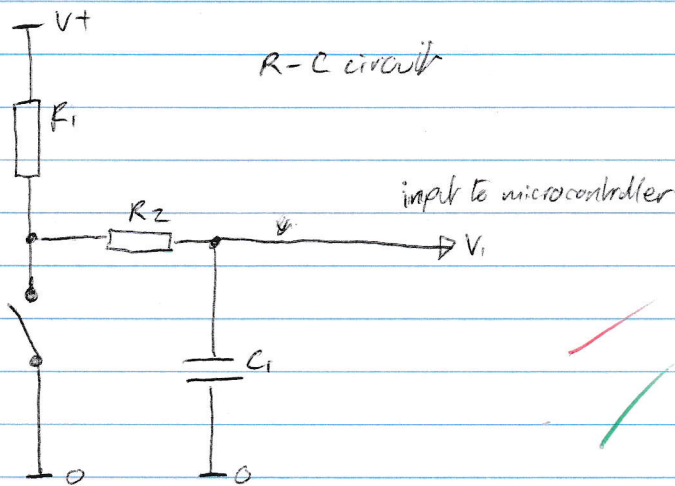
2

2

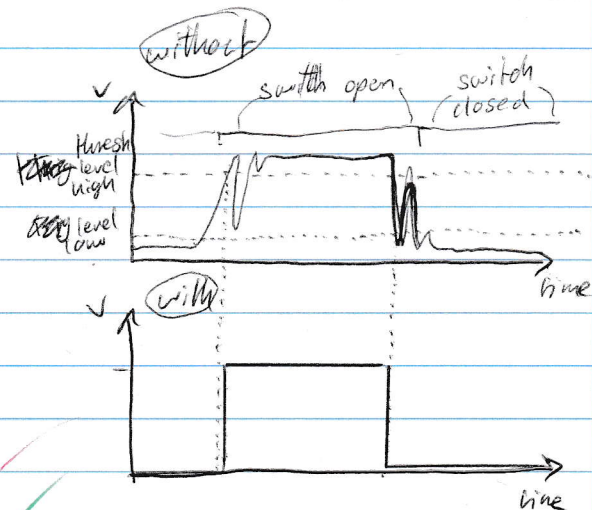
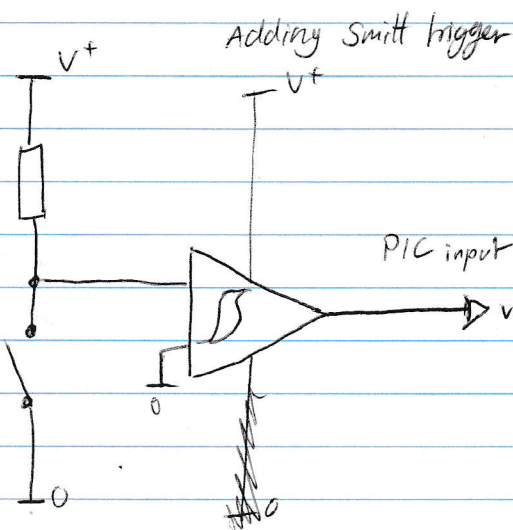
2

10

(2)

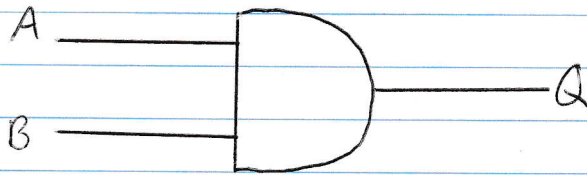


while the switch is open, the capacitor can charge through R_1 and R_2 . When the switch is changed, the capacitor will discharge, providing ~~extra voltage to smooth~~ smoothing to reduce switch bounce. Similarly, when the switch opens again, the capacitor will begin charging again, also smoothing the switch bounce by "topping" up the voltage when it drops sharply.



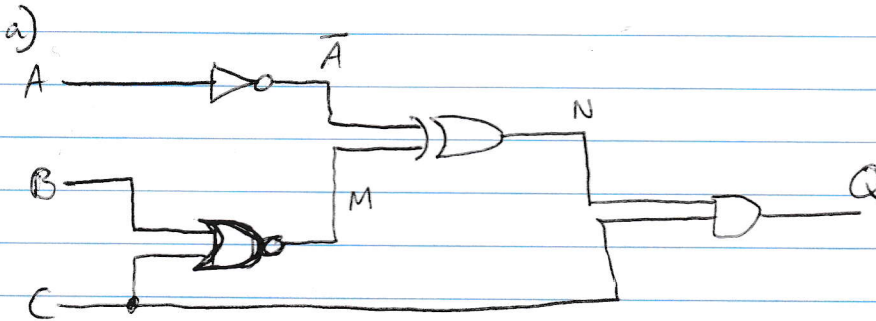
By adding a Schmitt trigger before the PIC, the trigger will only output high when a certain level of voltage has been reached, and will not switch to outputting low until a low threshold voltage has been reached. This means that any noise or switch bounce between these threshold voltages will not affect the output of the Schmitt. ~~This creates a clean signal input, reducing switch bounce effect.~~ This creates a clean signal input, reducing switch bounce effect.

③



A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

④



A	B	C	$\bar{A}$	$B+C$	$\bar{A} \oplus M$	N	Q	No. C output
0	0	0	1	0	1	0	0	
0	0	1	1	1	0	1	1	*
0	1	0	1	1	0	1	0	
0	1	1	1	1	0	1	1	*
1	0	0	0	0	1	1	0	
1	0	1	0	1	0	0	0	
1	1	0	0	1	0	0	0	
1	1	1	0	1	0	0	0	

b)

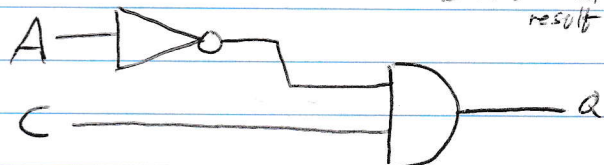
$$\bar{A} \cdot C$$

$$\text{positive results: } (\bar{A} \cdot \bar{B} \cdot C) + (\bar{A} \cdot B \cdot C) = \bar{A} \cdot C \cdot (\bar{B} + B) = \bar{A} \cdot C \cdot 1 = \bar{A} \cdot C$$

$$= \bar{A} \cdot C \cdot (\bar{B} + B) = \bar{A} \cdot C$$



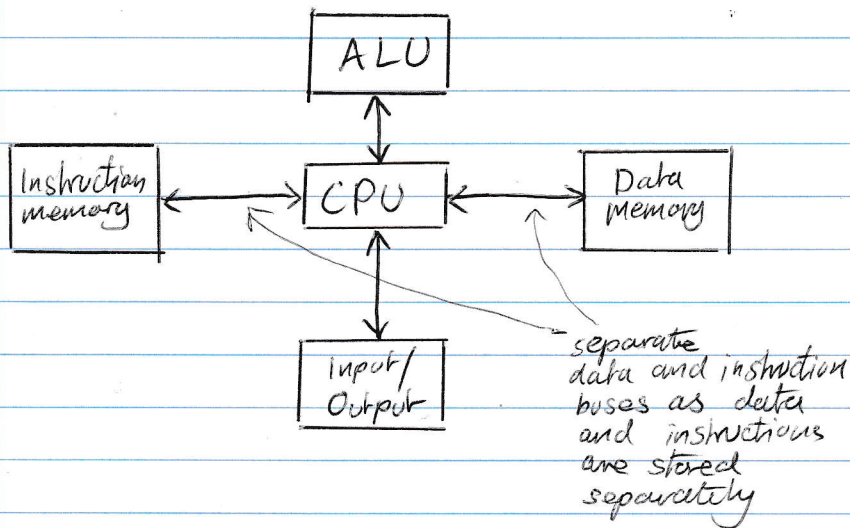
⇒



→ B doesn't affect result

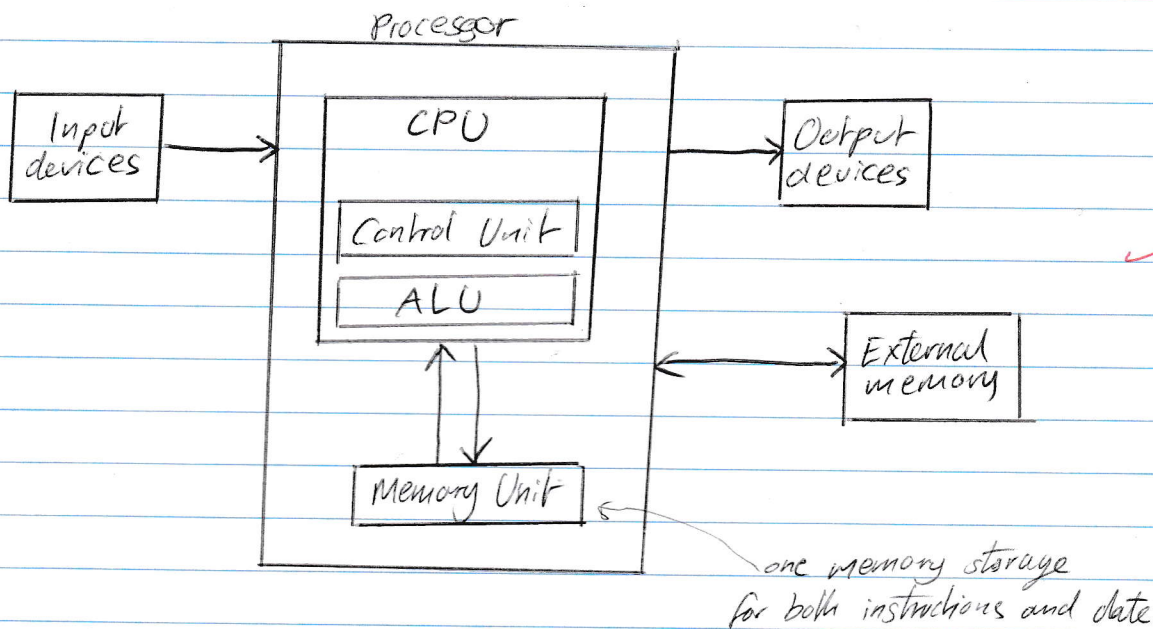
(5)

Harvard :

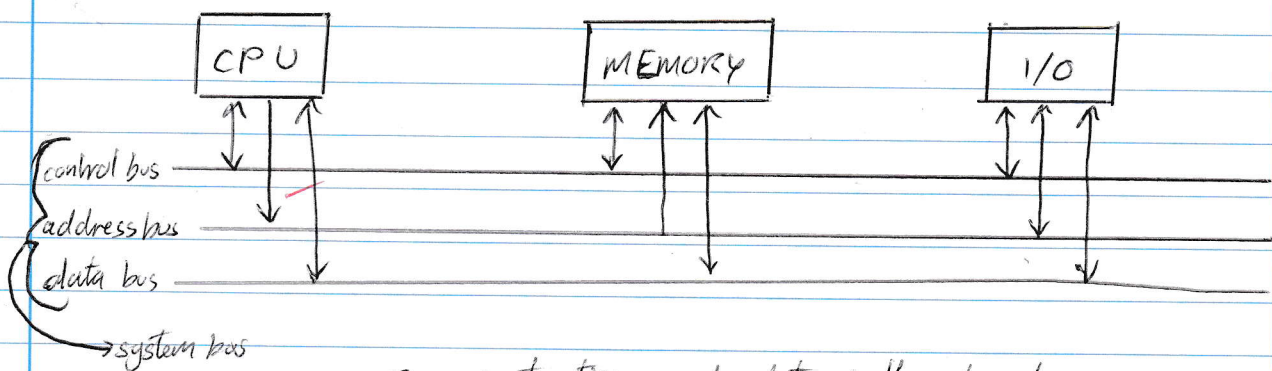


Advantage of Harvard: As the data and instructions are stored separately, with different addresses, and ~~can~~ are also accessed by separate buses; this means that the device can fetch instructions and read/write data simultaneously, making some processes more efficient.

Von Neumann



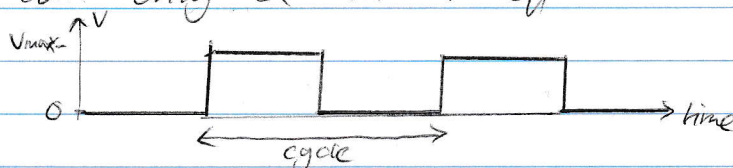
bus structure



As instructions and data both stored in same location, must be accessed by the same bus, so cannot fetch instructions and read/write data simultaneously.

Advantage of Von Neumann: as there is only one central memory block for both instructions and data, which must both be accessed with the same bus, this makes Von Neuman processors a lot cheaper and easier to program.

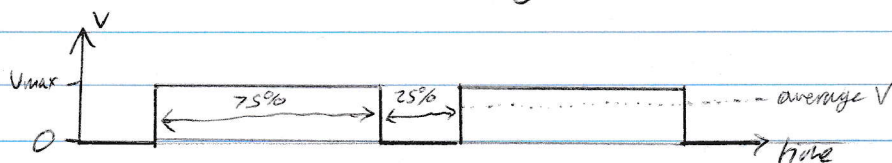
- ⑥ a digital output of a PWM signal can be used to ~~replicate~~ mimic an analog output.
~~The~~ The digital output will be as a square wave, as it can only be on or off:



This can replicate an analog signal however by changing the duty cycle of the wave.

The duty cycle is the ~~of~~ percentage of the cycle (indicated above - a wavelength of the square wave) where the signal is high / ~~at~~ V_{max} .

For example, a 75% duty cycle looks like:



The frequency will not affect the duty cycle of the PWM signal. When read ~~as~~ an analog ~~signal~~, the average value of the signal will be read, ~~as~~

This is worked out by: $\rightarrow$ meaning a large range of values can be represented

$$V = V_{max} \times \left(\frac{\text{duty cycle}}{100\%} \right)$$

for example, the average value of a PWM wave with a 60% duty cycle, and V_{max} of 10v, will be read as 6v

$$V = 10 \times \left(\frac{60\%}{100\%} \right)$$

$$V = 10 \times 0.6$$

$$V = 6V$$

Therefore, from a digital output ~~as~~ a square wave, an analog ~~signal~~ can be read (making a DAC). As the digital value a microcontroller ~~has~~ can be programmed to affect the duty cycle of the PWM signal by the digital value being a proportion of the maximum digital output, the same proportion will relate to the duty cycle percentage, converting digital to analog.

smoothing: