



Complete, fold and seal
(see note 2 below)

SURNAME: B

FORENAME(S): Christopher

Examination Answer Books

12 Pages

Instructions to Candidates

1. Please give the information requested in the boxes provided.
2. Enter your name in the space provided to the right, and seal down (using the label provided) to hide your name.
3. The answers must be legibly written. The Examiners can read either your handwritten 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	2322626
2	Title of Examination Paper	Micro Peripherals and interfaces
3	Module Code	ACDA4004
4	Date and Time of Examination	9:00am 08/01/25
5	Examination Venue	10310
6	Seat Number	037
7	Number of Answer Books handed in	

List the numbers of the questions you have attempted	Office Use
1	10
2	9
3	3
4	10
5	9
6	2
	43
	x2
Total	86%

10
9
3
10
9
2
43

Vn

Cu mem io

Control
address
data } System

H

1/2

memory - Cu - instruction
data mem

Q1) a)

Return - Returns from Subroutine to the
next command in the program. ✓

2

2

b) Sleep - Sets the processor into Sleep
State and turns off its clock/output. ✓

2

2

c) ADDLW - ADD Literal and W
ADDS together the contents of
Literal value, and the W register. ✓

2

2

d) BCF F, b - Bit Clear F - Clears bit
'b' to 0 in register F. ✓

2

2

e) DecF - Decrement F - decrements the
bits/value of register F by one. ✓

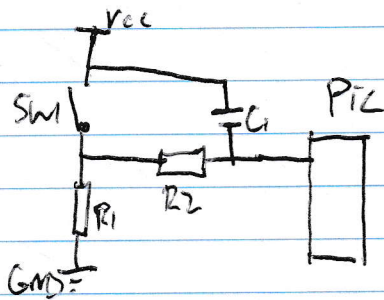
2

2

10

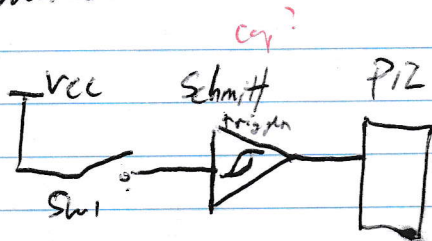
Q2) Switch bounce is an unwanted parasitic condition of a physical switch that can cause issues if a switch's operation is used in a program.

One way to fix it is to create an RC time constant in a circuit. This can be done by adding a capacitor into the switch circuit, as shown below.

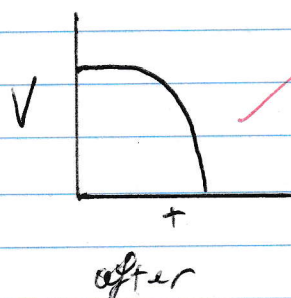
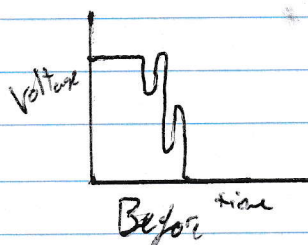


The capacitor and resistor network filters out high frequency pulses, eliminating switch bounce.

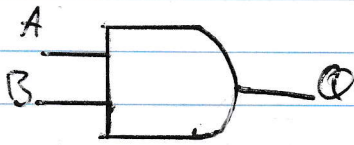
Another way to eliminate it is to use a Schmitt trigger. This device works by having a different level for its on and off state. For example, 3V may be needed for its on state, but it must decrease to 1V for it to turn off, eliminating small fluctuations in voltage, or bounce.



Below are diagrams showing an input before and after debouncing.



3)



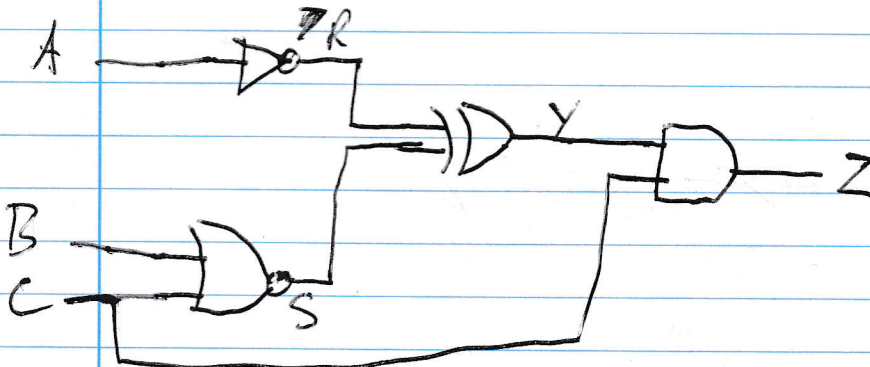
A	B	Q
0	0	0
1	0	0
0	1	0
1	1	1

3

3

4

a)



Nor

A	B	C	S
0	0	0	1
1	0	0	0
0	1	0	0
1	1	0	0

A	B	C	R	S	Y	Z
0	0	0	1	1	0	0
0	0	1	1	0	1	1
0	1	0	1	0	1	0
0	1	1	1	0	1	1
1	0	0	0	1	1	0
1	0	1	0	0	0	0
1	1	0	0	0	0	0
1	1	1	0	0	0	0

Xor

R	S	Y
0	0	0
0	1	1
1	0	1
1	1	0

AND

Y	Z
0	0
1	0
0	1
1	1

Final Truth Table

A	B	C	Z
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Q4)
b)

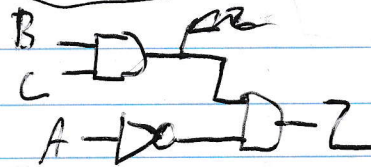
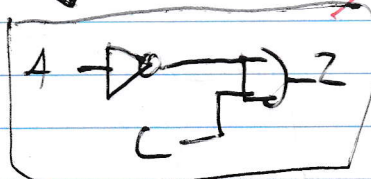
A	B	C	Z
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

$$C + B + !A = Z$$

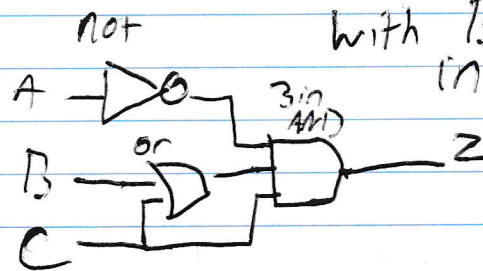
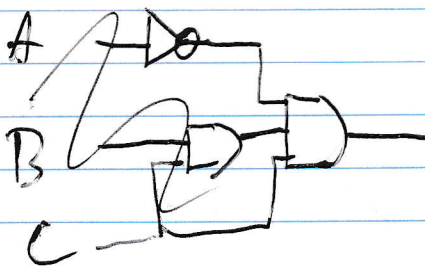
$$(OR C + !A = Z)$$

C	A	Z	A C B Z
0	0	0	0
1	0	1	
0	1	0	
1	1	0	

A	B	C
0	1	1

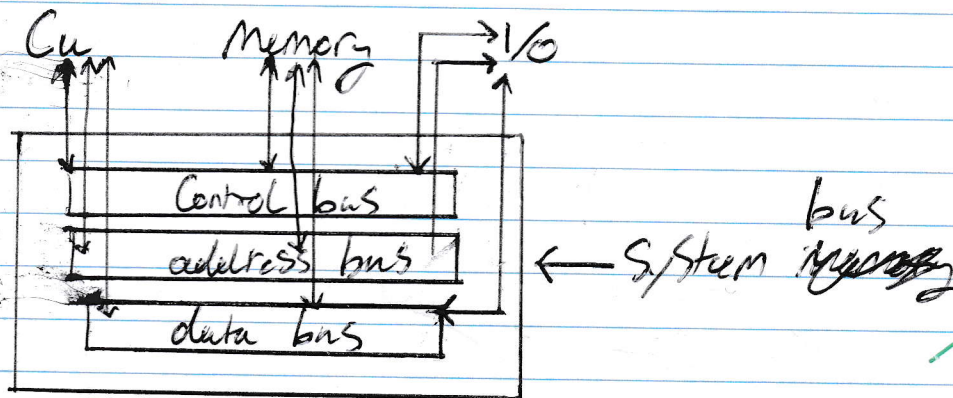


if B is removed this gives some out put, as C and A are the only inputs that change the output

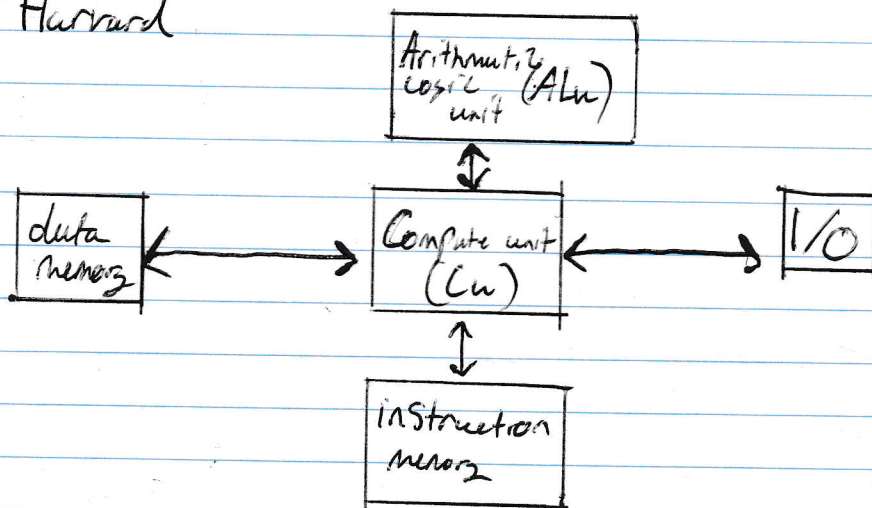


With B kept in

Q5) Von Neumann



Harvard

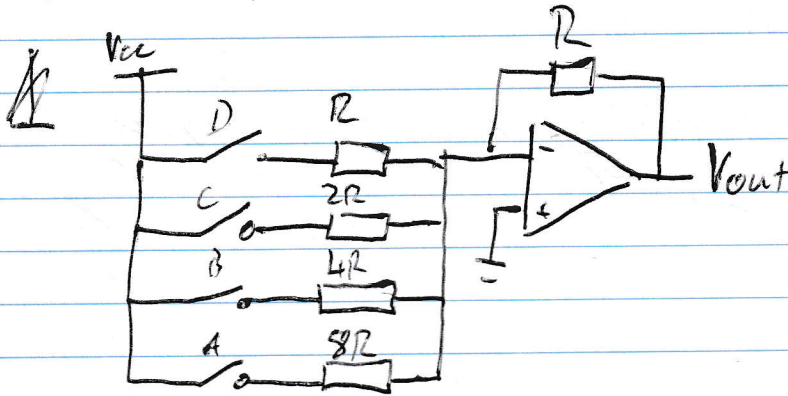


Von Neumann architecture is simpler/less complex, making it easier/cheaper to produce and adapt to different tasks.

Harvard architecture is faster, due to instruction and data memory being stored in different locations and having their own bus, meaning they can be read simultaneously, so it only takes one clock cycle to complete.

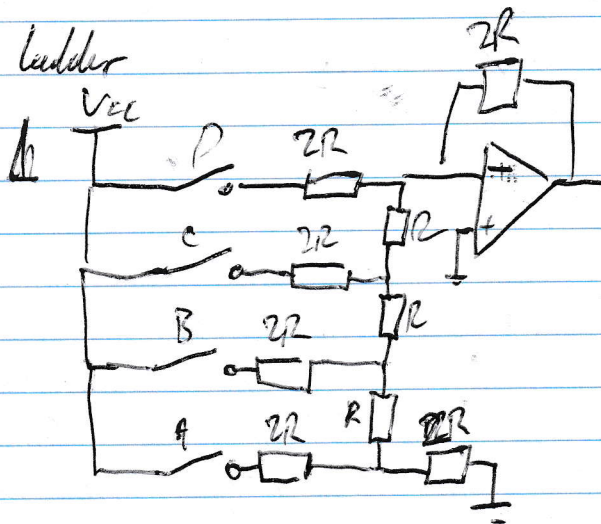
Q6) A Simple DAC can be produced two ways.
The first is a binary weighted DAC.
and the second is an R2R ladder

Binary dac example



In practice switches A-D would be replaced by Digital pins in both binary weighted and R2R ladder

R2R ladder



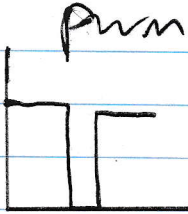
HA.

These both work by having each switch/resistor arm at a fraction of V_{cc} , for example, Switch D would be $\frac{1}{2}V_{cc}$ and A, $\frac{1}{8}V_{cc}$. So together are able to take a 4-bit Digital signal and make an analogue value by applying the sum of these voltages to the - pin of the comparator. V_{out} is calculated with:

$$V_{out} = -\left(V_D + \frac{V_C}{2} + \frac{V_B}{4} + \frac{V_A}{8}\right)$$

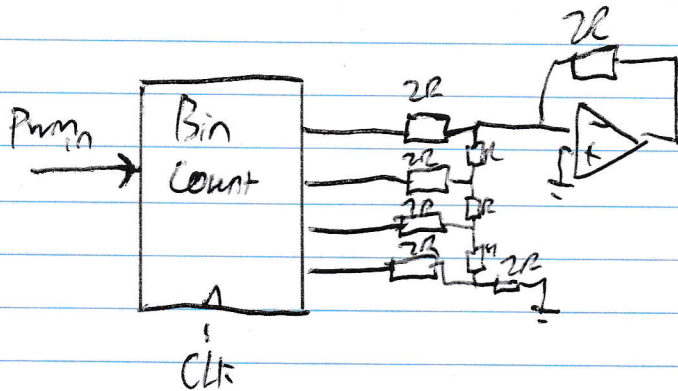
Q6)

Cent.



Pwm is Modulating the width of pulses (duty cycle) from the ~~pin~~ microcontroller's pin

To implement Pwm into a DAC you ~~would~~ could run the pin into a binary Counter with a clock, effectively counting the pulses per clock cycle, adjusting the Pwm width would give a higher or lower count from the counter, resulting in a higher or lower value sent to the DAC.



this could be done with 22R ladder or binary weighted resistors.