G x 895

OLLSCOIL NA hÉIREANN
GAILLIMH

NATIONAL UNIVERSITY OF IRELAND
GALWAY

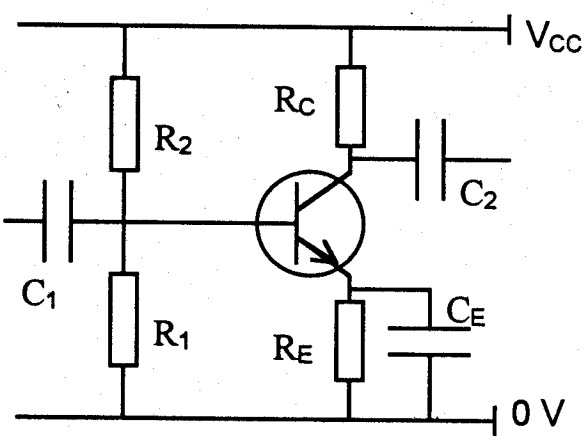
SEMESTER 1 EXAMINATIONS 2002 - 2003

3rd year B.Sc.Unit EP312: Electronics and Devices

Prof. E. Kennedy
Prof. S.G. Jennings
Dr. J. Martin

Time allowed TWO hours	Answer THREE questions

Q.1 The diagram shows a Common Emitter (CE) amplifier stage where the known values are: V_{CC} = 10 V, R_C = 3 kΩ, R_E = 1 kΩ, and where the transistor has a current gain β = 200 (nominal) and f_T = 400 MHz (nominal). The circuit Q point is to be set at (I_C = 1 mA, V_{CE} = 5 V).



Calculate suitable values for R1 and R2, and find the circuit stability factor, S. State briefly why S is important in the CE amplifier circuit, and explain how its numerical value can be reduced. What are the drawbacks of reducing S?

[5 marks]

State the purposes of C_1 and C_2, and choose suitable values for them. Calculate the value required for C_E, if the amplifier low frequency cut-off is to be 50 Hz. Find also the mid-band amplifier input impedance and its small signal voltage gain. Estimate, approximately, the high frequency cut-off, and the bandwidth of the amplifier. *[5 marks]*

Q.2 Compare the internal structure of an 8 bit *microcontroller* (μC) device, such as the INTEL 8051, with that of a general purpose 8 bit *microprocessor* (μP). State the main areas of application of μC's like the 8051, indicating where and why they are preferred over g.p. μP's. *[5 marks]*

An 8051 chip is to be used to implement the Truth Table shown opposite, where P, Q and R are the system inputs and X is the single output. P, Q and R are wired as lines 0, 1 and 2 on the 8051 I/O port 2, and X is output to line 7 on I/O port 1.

P	Q	R	X
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Write the necessary 8051 Assembly Language code, which will implement this Truth Table.

NB: a short 8051 instruction set is appended to this paper.

[5 marks]

Q.3 Briefly compare the main parameters of an ideal operational amplifier (op amp) with those of real devices, such as the 741 or 351 op amps. Explain why the input and output impedances of an ideal op amp should have the particular values which you give. ***[4 marks]***

State and justify the *Barkhausen Criteria* for sustained oscillation in a closed loop amplifier configuration. Sketch the circuit diagram and explain the operation of either (a) a Phase Shift Oscillator, or (b) a Wien Bridge Oscillator, based on a single op amp amplifying stage. ***[6 marks]***

Q.4 State briefly, without explanation, what the following terms stand for in the field of digital electronics: *TTL, ECL, CMOS, SRAM, DRAM, PROM, OTP, MUX.* ***[4 marks]***

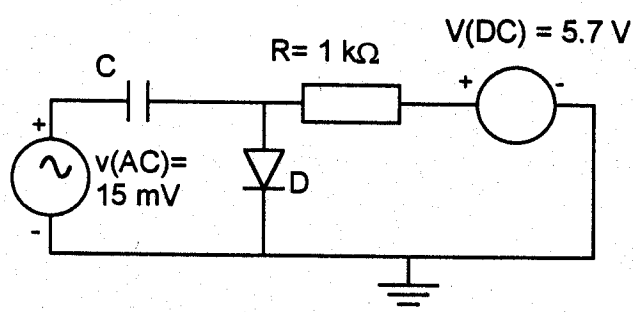
Describe the pin-out, operation and facilities of a typical TTL *Data Selector / MUX*, such as the 74xx151 device. Draw up circuit diagrams to show how the 151 is used (a) as a conventional Data Selector, and (b) as a 3-input Truth Table implementer. Show how to implement the Truth Table given in Q.2 above, using a 151, to illustrate your answer to part (b). ***[6 marks]***

Q.5 Answer any *TWO* of the following. ***[5 marks each]***

(a) Use a diagram to illustrate the physiçal structure (fabrication) of a npn BJT (Bipolar Junction Transistor). Explain briefly how and why the BJT acts as a CCCS (Current Controlled Current Source). Give its circuit symbol, and sketch its typical output characteristics. Define the quantities α, β and f_T, as applied to BJT's, and give typical values for these parameters.

(b) Sketch the internal circuit diagram of a typical 74 series TTL NAND gate, and use it to help in giving a full description of the gate operation.

(c) Estimate the approximate values of the DC and AC current components in the silicon p-n diode, D, for the circuit shown opposite. The capacitor C can be assumed to have negligible impedance at the frequency of the AC generator.



Explicitly state any assumptions made about the diode in your calculations.

Short 8051 Assembly Language Instruction Set

A = Accumulator ACC
B = Additional B register
Pn = 8 bit I/O Port 0-3
Rn = Register 0-7 of selected register bank
#data = immediate 8 bit data
rel = 2's compl 8 bit relative jump (-128 to +127)
@Ri = 8 bit address held in R0 or R1
bit = Directly addressed bit (e.g. P0.5, B.7, 7Fh)
<byte> = generalised 8 bit quantity (e.g. one of A, B, Rn, Pn, @Ri, etc)

DATA TRANSFER INSTRUCTIONS	
MOV A, Rn	Move Register Rn to A (Move Register)
MOV A, Pn	Move I/O port to A (Move Direct)
MOV A, @Ri	Move contents of RAM address held in **Ri** (i = 0,1) to A (Move Indirect)
MOV A, #data	Move immediate data to A (Move Immediate)
MOV Rn, A	Move A to Rn
MOV Pn, A	Move A to I/O Port Pn
MOV Pn, #data	Move immediate data to I/O Port Pn
MOV Pm, Pn	Move I/O Port Pn to I/O Port Pm
XCH A, Rn	Exchange A and Rn

ARITHMETICAL AND LOGICAL INSTRUCTIONS	
ADD A, <byte>	Add <byte> = B, Rn, Pn, or @Ri to A
ANL A, <byte>	Logical AND <byte> = B, Rn, Pn, or @Ri with A
CLR A	Clear A (all 8 bits) to 0
CPL A	Complement A
DEC <byte>	Decrement <byte> = A, B, Rn, Pn, or @Ri
INC <byte>	Increment <byte> = A, B, Rn, Pn, or @Ri
ORL A, <byte>	Logical OR <byte> = B, Rn, Pn, #data, or @Ri with A
XRL A, <byte>	Logical XOR <byte> = B, Rn, Pn, #data, or @Ri with A

BOOLEAN (BIT) MANIPULATION	
ANL C, bit	Logical AND direct bit with C
ANL C, /bit	Logical AND the complement of direct bit with C
CLR bit	Clear direct bit to 0. E.g. CLR ACC.5 resets bit #5 in A.
CPL bit	Complement direct bit. E.g. CPL 47h, CPL PSW.6, etc
CLR C	Clear Carry C to 0
CPL C	Complement Carry C
MOV bit, C	Move C to direct bit. E.g. MOV P2.1, C copies C to bit #1 on P2
MOV C, bit	Move direct bit to C. E.g. MOV C, B.3 copies bit #3 of B to C
ORL C, bit	Logical OR direct bit with C
ORL C, /bit	Logical OR complement of direct bit with C
SETB bit	Set direct bit to 1. E.g. SETB P0.0 sets bit 0 of Port 0 to 1
SETB C	Set Carry to 1

PROGRAM BRANCHING	
ACALL adr11	CALL subroutine at 11 bit address unconditionally (Absolute Call)
CJNE A,Pn,rel	Compare I/O port Pn with A and Jump (relative) if Not Equal
CJNE A,#data,rel	Compare immediate data with A and Jump (relative) if Not Equal
DJNZ Rn,rel	Decrement Rn and Jump (relative) if Rn is Not Zero
JB bit,rel	Jump (relative) if direct bit is set
JC rel	Jump (relative) if Carry is set (= 1)
JNC rel	Jump (relative) if Carry is Not set (= 0)
JZ rel	Jump (relative) if A is Zero
JNZ rel	Jump (relative) if A is Not Zero
LJMP adr16	Long Jump to 16 bit absolute address
RET	Return from subroutine : pop PC = Program Counter
SJMP rel	Short Jump to relative address (-128 to +127 bytes from current PC)