Tugas praktikum SO

1. kode ASCII adalah suatu standar internasional dalam kode huruf dan simbol seperti Hex dan uni code tetapi ASCII lebih bersifat universal, Digunakan oleh komputer dan alat komunikasi lain untuk menunjukkan teks

desimal	binary	Heksa desimal	karakter
32	0100000	20	SP
33	0100001	21	!
34	0100010	22	"
35	0100011	23	#
36	0100100	24	$
37	0100101	25	%
38	0100110	26	&
39	0100111	27	'
40	0101000	28	(
41	0101001	29	)
42	0101010	2A	*
43	0101011	2B	+
44	0101100	2C	,
45	0101101	2D	-
46	0101110	2E	.
47	0101111	2F	/
48	0110000	30	0
49	0110001	31	1
50	0110010	32	2
51	0110011	33	3
52	0110100	34	4
53	0110101	35	5
54	0110110	36	6
55	0110111	37	7

56	0111000	38	8
57	0111001	39	9
58	0111010	3A	:
59	0111011	3B	;
60	0111100	3C	<
61	0111101	3D	=
62	0111110	3E	>
63	0111111	3F	?
64	1000000	40	@
65	1000001	41	A
66	1000010	42	B
67	1000011	43	C
68	1000100	44	D
69	1000101	45	E
70	1000110	46	F
71	~~1001000~~ 1000111	47	G
72	1001000	48	H
73	1001001	49	I
74	1001010	4A	J
75	1001011	4B	K
76	1001100	4C	L
77	1001101	4D	M
78	1001110	4E	N
79	1001111	4F	O
80	~~1010001~~ 1010000	50	P
81	1010001	51	Q
82	1010010	52	R
83	1010011	53	S
84	1010100	54	T
85	1010101	55	U

86	1010110	56	V
87	1010111	57	W
88	1011000	58	X
89	1011001	59	Y
90	1011010	5A	Z
91	1011011	5B	[
92	1011100	5C	\
93	1011101	5D	]
94	1011110	5E	^
95	1011111	5F	_
96	1100000	60	`
97	1100001	61	a
98	1100010	62	b
99	1100011	63	c
100	1100100	64	d
101	1100101	65	e
102	1100110	66	f
103	1100111	67	g
104	1101000	68	h
105	1101001	69	i
106	1101010	6A	j
107	1101011	6B	k
108	1101100	6C	l
109	1101101	6D	m
110	1101110	6E	n
111	1101111	6F	o
112	1110000	70	p
113	1110001	71	q
114	1110010	72	r
115	1110011	73	s

ENKO® 30 Lines, 6 mm

2. Perintah bahasa assembly lengkap untuk mesin intel keluarga x86. Digunakan sbg pedoman untuk memahami program bool.asm dan kemel.asm

Assembly Directive

- EQu => Definisian Konstanta
- DB => Definisian data dg ukuran satuan 1 byte
- Dw => Definisian data dg ukuran satuan 1 word
- DBIT => Pendefinisian data dg ukuran satuan 1 bit
- Ds => pemesanan tempat penyimpanan data di RAM
- ORG => Insialuasi alamat mulai program
- EHD => penanda akhir program
- CSEG => penempatan penanda di code segment
- X SEG => penanda penempatan di external data ''
- DSEG ^{penanda} => penempatan di Internal direct data ''
- ISEG => penanda penempatan di internal indirect data''
- BSEG => penanda penempatan di bit data ''
- CODE => penanda mulai pendefinisian program
- XDATA => pendefinisian external data
- I DATA => pendefinisian internal indirect data
- BIT => pendefinisian data bit
- *INCLUDE => mengikut sertakan file program lain
- DATA => pendefinisian internal direct data

instruksi

- ACALL (Absolut Call)
- ADD (Add)
- ADDC (Add with carry)
- AJMP (Absolute jump)
- ANL (And Logic)
- CJHE
- CLR (clear)
- CPL (complement)
- DA (Decimal Adjust)
- DEC (Decrement)
- DIV (Divide) not zero)
- DJNZ (Decrement and jump if
- INC (increment)
- JB (Jump if bit set)
- JBC (jump if bit set and clearbit)
- JC (jump if carry set)
- JMP (jump to Address)
- JNB (jump if Not bit set)
- JNC
- JN2
- JZ
- LCALL (Long call)
- LJMP (long jump)
- MOV
- MOVC
- MOVX
- MUL
- Nop
- ORL

- POP
- PUSH
- RET
- RETI
- RL
- RLC
- RR
- RRC
- SETB
- SJMP
- SUBB
- SWAP
- XCH
- XCHD
- XRL

- MOV => perintah untuk mengisi, memindahkan, memperbarui isi suatu register, variable ataupun lokasi memory

INT ≠▽

- Push => perintah untuk memasukkan isi register pada stack, dg tata penulisannya : POP

- RIP => digunakan untuk memberitahu komputer untuk memulai memproses program dari titik tertentu

- A (Assembler) => untuk tempat menulis program Assembler

- RCX (Register Cx) => untuk mengetahui dan memperbarui isi register Cx yang merupakan tempat penampungan panjang program yg sedang aktif