COMPUTER ORGANIZATION (CDA – 3103)

SPRING 2005

Lab # 5: Interrupts in Intel Architecture

1 PROGRAM # 1: PRINT ASTERISK ON '*' ON CONSOLE 1
1.1 ASSEMBLY 1
1.2 MACHINE CODE 1
2 PROGRAM # 2: READ A CHARACTER AND THEN PRINT 2
2.1 ASSEMBLY 2
2.2 MACHINE CODE 2
3 MACHINE CODE TABLE 4
3.1 REGISTERS (RRR): 4
3.2 INSTRUCTIONS 4
3.3 DOS INT 21H - DOS FUNCTION CODES 5
4 ASCII TABLE 6

Note use Cygnus available on course website for writing a program in Hexadecimal form.

1 PROGRAM # 1: PRINT ASTERISK ON '*' ON CONSOLE

1.1 ASSEMBLY

```
# Start program from address 0100h
org 0100h

# Write asterisk on console
mov AH , 02             # code for printing on console
mov DL, 02Ah            # 2A is ASCII code for asterik
int o21h

# Exit from program with return code 0
mov AH , 04Ch
mov AL, 0
int o21h

END
```

1.2 MACHINE CODE

```
# Start program from address 0100h
org 0100h

# Write asterisk on console
mov AH , 02             =       B4 02
mov DL, 02Ah            =       B2 2A
int o21h                =       CD 21

# Exit from program with return code 0
mov AH , 04Ch           =       B4 4c
mov AL, 0               =       B0 00
int o21h                =       CD 21
```

2 PROGRAM # 2: READ A CHARACTER AND THEN PRINT

2.1 ASSEMBLY

```
# Start program from address 0100h
org 0100h

# Read a character from console. Character will be returned in AL Move that character to CL
mov AH, 01
int o21h
mov CL,AL

# Print pace followed by the same character read before
mov AH , 02
mov DL, 20h
int o21h

mov DL, CL
int o21h

# Exit from program with return code 0
mov AH , 04Ch
mov AL, 0
int o21h
```

2.2 MACHINE CODE

```
# Start program from address 0100h
org 0100h                                   =          No machine code.

# Read a character from console. Character will be returned in AL. Move that character to CL
mov AH, 01h          =    1011wrrr          =    1011 0100 0000 0001    =    B4 01
int o21h             =    11001101          =    1100 1101 0010 0001    =    CD 21
mov CL,AL            =    1000101woorrrmmm  =    1000 1010 1100 1000    =    8A C8

# Print pace followed by the same character read before
mov AH , 02          =    1011wrrr          =    1011 0100 0000 0010    =    B4 02
mov DL, 20h          =    1011wrrr          =    1011 0010 0010 0000    =    B2 20
int o21h             =    11001101          =    1100 1101 0010 0001    =    CD 21

mov DL, CL           =    1000101woorrrmmm  =    1000 1010 1101 0001    =    8A D1
int o21h             =    11001101          =    1100 1101 0010 0001    =    CD 21

# Exit from program with return code 0
mov AH , 04Ch        =    1011wrrr          =    1011 0100 0100 1100    =    B4 4c
mov AL, 0            =    1011wrrr          =    1011 0000 0000 0000    =    B0 00
int o21h             =    11001101          =    1100 1101 0010 0001    =    CD 21
```

Cygnus - [Program2.com]

File Edit View Tools Window Help

```
00000000  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000080  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000090  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
000000A0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
000000B0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
000000C0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
000000D0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
000000E0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
000000F0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
00000100  B4 01 CD 21 8A C8 B4 02-B2 20 CD 21 8A D1 CD 21  ...!..... .!...!
00000110  B4 4C B0 00 CD 21                                 .L...!
```

3 MACHINE CODE TABLE

3.1 REGISTERS (RRR):

000 : AL or AX
001 : CL or CX
010 : DL or DX
011 : BL or BX
100 : AH or SP
101 : CH or BP
110 : DH or SI
111 : BH or DI

3.2 INSTRUCTIONS

Name	Regs	Description
POPF		10011101
PUSHF		10011100

Name	Regs	Description
PUSH	RegWord	01010rrr
	Seg	00sss110

Name	Regs	Description
POP	RegWord	01011rrr
	Seg	00sss111

Name	Regs	Description
ADD	Reg,Reg	0000001woorrrmmm
	Acc,Imm	0000010w
	Reg,Imm	1000000woo000mmm

Name	Regs	Description
SUB	Reg,Reg	0010101woorrrmmm
	Acc,Imm	0010110w
	Reg,Imm	1000000woo101mmm

MOV	Reg,Imm	1011wrrr
	Reg,Reg	1000101woorrrmmm
	Reg16,Seg	10001100oosssmmm
	Seg,Reg16	10001110oosssmmm

INT	Imm8	11001101

JMP	Short	11101011
	Near	11101001

Jcc	Short	0111cccc

CCCC	Name	Means
0000	O	overflow
0001	NO	Not overflow
0010	C/B/NAE	Carry, below, not above nor equal
0011	NC/AE/NB	Not carry, above or equal, not below
0100	E/Z	Equal, zero
0101	NE/NZ	Not equal, not zero

LOOP	Short	11100010

3.3 DOS INT 21H - DOS FUNCTION CODES

1- AH = 01h - READ CHARACTER FROM STANDARD INPUT, WITH ECHO; Return: AL = character read
2- AH = 02h -WRITE CHARACTER TO STANDARD OUTPUT; Entry: DL = character to write; Return: AL = last character output
3- AH = 09h - WRITE STRING TO STANDARD OUTPUT; Entry: DS:DX -> '$'-terminated string; Return: AL = 24h
4- AH = 2Ah - GET SYSTEM DATE; Return: CX = year (1980-2099) DH = month DL = day AL = day of week (00h=Sunday)
5- AH = 4Ch - "EXIT" - TERMINATE WITH RETURN CODE; Entry: AL = return code; Return: never returns

4 ASCII TABLE

Dec	Hx	Oct	Char		Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL	(null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH	(start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX	(start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX	(end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT	(end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ	(enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK	(acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL	(bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS	(backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB	(horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF	(NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT	(vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF	(NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR	(carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO	(shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI	(shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE	(data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1	(device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2	(device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3	(device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4	(device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK	(negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN	(synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB	(end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN	(cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM	(end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB	(substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC	(escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS	(file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS	(group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS	(record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US	(unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL