

אלקטרוניקה ומחשבים ג'

שתי יחידות לימוד (השלמה לחמש יחידות לימוד)

(כיתה י"ב)

הוראות לנבחן

א. משך הבחינה: שלוש שעות.

ב. מבנה השאלון ומפתח ההערכה: בשאלון שני פרקים, ובהם תשע שאלות. יש לענות על

חמש שאלות בלבד, שאלה אחת לפחות מכל פרק.

לכל שאלה – 20 נקודות. סך הכול – 100 נקודות.

ג. חומר עזר מותר לשימוש: מחשבון.

ד. הוראות מיוחדות:

1. ענה על מספר השאלות הנדרש בשאלון. המעריך יקרא ויעריך את מספר השאלות

הנדרש בלבד, לפי סדר כתיבתן במחברתך, ולא יתייחס לתשובות נוספות.

2. התחל כל תשובה לשאלה חדשה בעמוד חדש.

3. רשום את כל תשובותיך אך ורק בעט.

4. הקפד לנסח את תשובותיך כהלכה ולסרטט את תרשימיך בהירות.

5. כתוב את תשובותיך בכתב-יד ברור, כדי לאפשר הערכה נאותה שלהן.

6. אם לדעתך חסרים נתונים הדרושים לפתרון שאלה, אתה רשאי להוסיף אותם, בתנאי

שתנמק מדוע הוספת אותם.

7. בכתיבת פתרונות חישוביים, קבלת מִרְב הנקודות מותנית בהשלמת כל המהלכים

הבאים, בסדר שבו הם רשומים:

* רישום הנוסחה המתאימה.

* הצבה של כל הערכים ביחידות המתאימות.

* חישוב (אפשר באמצעות מחשבון).

* רישום התוצאה המתקבלת, ביחד עם יחידות המידה המתאימות.

* ליווי הפתרון החישובי בהסבר קצר.

בשאלון זה 9 עמודים ו-23 עמודי נספחים.

ההנחיות בשאלון זה מנוסחות בלשון זכר,

אך מכוונות לנבחנות ולנבחנים כאחד.

בהצלחה!

השאלות

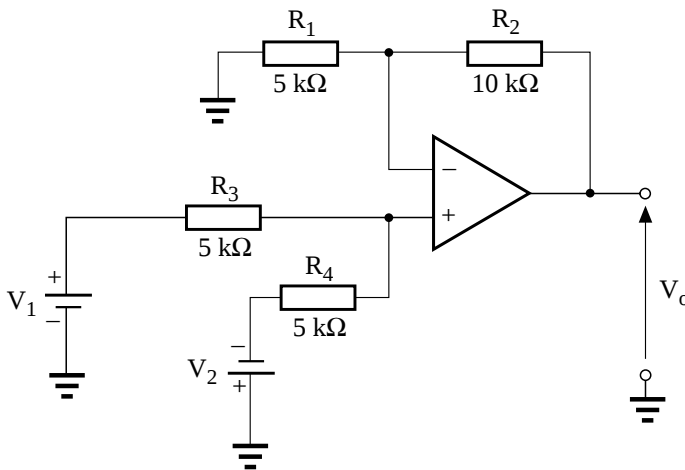
בשאלון שני פרקים ובהם תשע שאלות. יש לענות על חמש שאלות בלבד, שאלה אחת לפחות מכל פרק.

פרק ראשון: אלקטרוניקה תקבילית

ענה על שאלה אחת לפחות מבין השאלות 1-5 (לכל שאלה – 20 נקודות).

שאלה 1

המעגל שבאיור לשאלה 1 כולל מגבר שרת אידיאלי.



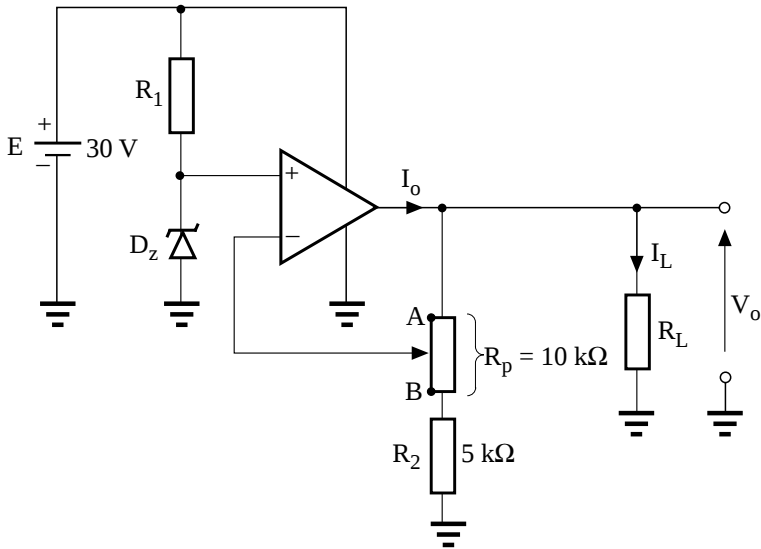
איור לשאלה 1

- א. 1. רשום ביטוי למתח המוצא V_o כתלות במתחי המבוא V_1 ו- V_2 .
2. חשב את ערכו של מתח המוצא V_o אם $V_1 = 5\text{ V}$ ו- $V_2 = 3\text{ V}$.
- ב. מחליפים את מקור המתח V_1 במקור מתח סינוסואידלי, בעל תנופה של 4 V משיא לשיא. סרטט את מתח המבוא V_1 ומתחתיו, בהתאמה, את מתח המוצא V_o בתלות בזמן.

שאלה 2

המעגל המתואר באיור לשאלה 2 כולל מגבר שרת אידיאלי. נתוני דיודת הזנר שבמעגל הם:

$$V_z = 5 \text{ V} ; I_{z_{\min}} = 5 \text{ mA} ; P_{z_{\max}} = 500 \text{ mW}$$



איור לשאלה 2

א. על-פי נתוני דיודת הזנר, חשב את תחום הערכים של התנגדות הנגד R_1 שבו תפעל הדיודה בתחום הזנר.

ב. חשב את מתח המוצא V_0 כאשר זחלן הפוטנציומטר R_p נמצא:

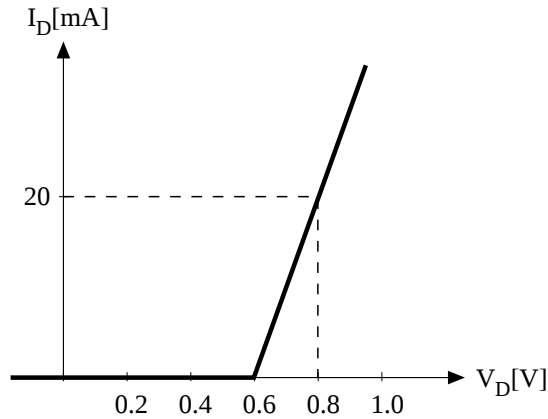
1. בנקודה A

2. בנקודה B

ג. חשב את ערכו המזערי של R_L , אם נתון שהזרם המרבי במוצא מגבר השרת הוא $I_0 = 25 \text{ mA}$, ומתח המוצא V_0 הוא 5 V .

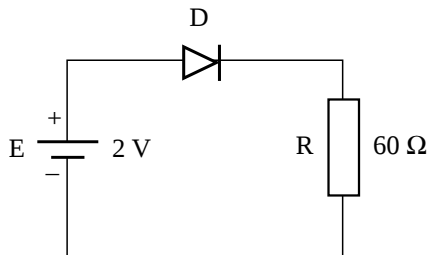
שאלה 3

א. באיור א' לשאלה 3 נתון אופיין זרם-מתח (מקורב) של דיודה.



איור א' לשאלה 3

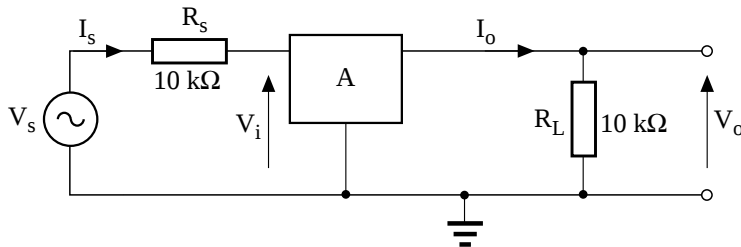
1. חשב את הפרמטרים של הדיודה (V_γ , R_f) על סמך אופיין זה.
 2. סרטט את מעגל התמורה של הדיודה כאשר היא מוליכה.
- ב. הדיודה שבסעיף א' מופיעה במעגל שבאיור ב' לשאלה. חשב את נקודת העבודה של הדיודה (V_D, I_D) .



איור ב' לשאלה 3

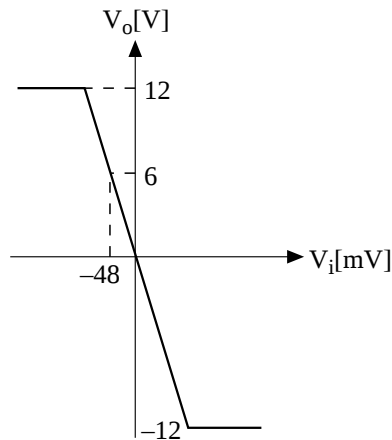
שאלה 4

המעגל שבאיור א' לשאלה 4 כולל מגבר A, שאופיין המעבר שלו בריקס נתון באיור ב' לשאלה. נתוני המגבר A: $R_o = 1 \text{ k}\Omega$, $R_i = 50 \text{ k}\Omega$.



איור א' לשאלה 4

אופיין המעבר של המגבר A



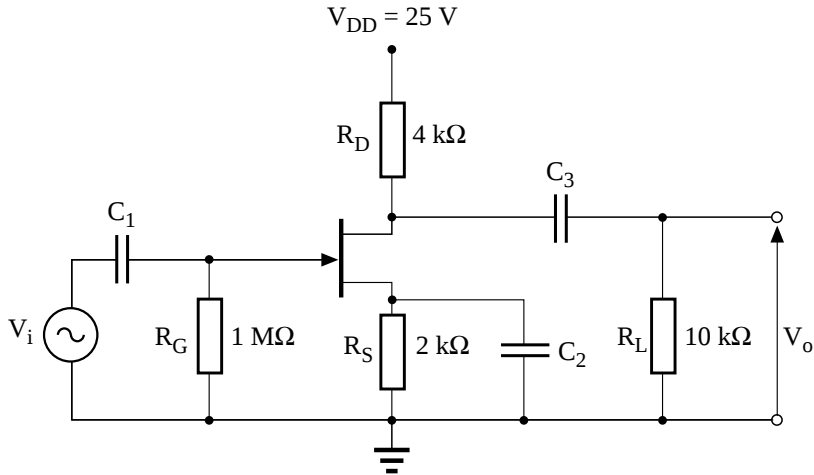
איור ב' לשאלה 4

- א. היעזר באיור ב', ומצא את הגבר המתח של המגבר A.
- ב. סרטט את מעגל-התמורה של המעגל שבאיור א', וחשב את הגבר המתח $A_{V_s} = \frac{V_o}{V_s}$.
- ג. חשב, ביחידות dB, את הגבר הזרם $\left(\frac{I_o}{I_s}\right)$ ואת הגבר ההספק $\left(\frac{P_o}{P_s}\right)$ של המעגל שבאיור א'.
- ד. חשב את התנופה המרבית של המתח V_s שניתן לספק למעגל שבאיור א', מבלי לגרום לעיוותים במתח המוצא.

שאלה 5

נתוני הטרנזיסטור במעגל המתואר באיור לשאלה 5 הם:

$$I_{DSS} = 10 \text{ mA} ; V_p = -5 \text{ V}$$



איור לשאלה 5

- א. חשב את נקודת העבודה של הטרנזיסטור (V_{DS}, I_D).
- ב. סרטט את מעגל התמורה לאות חילופין (הזנח את היגבי הקבלים).
- ג. חשב את הגבר המתח $\frac{V_o}{V_i}$ של המעגל אם $g_m = 1.56 \text{ mmho}$.

פרק שני: מחשבים ומיקרו-מעבדים

ענה על שאלה אחת לפחות מבין השאלות 6-9 (לכל שאלה – 20 נקודות).

שאלה 6

בלוק נתונים, שכתובתו ההתחלתית היא 30H וגודלו 10H בתים, מכיל נתונים מספריים. כתוב תת-שגרה בשפת הסף של המעבד 8086/88, שתמנה את כמות המספרים הזוגיים בבלוק ותציב את התוצאה המתקבלת בתא 40H.

שאלה 7

- א. הסבר את תפקידו של כל אחד מן ההדקים האלו במיקרו-מעבד 8086: $\overline{M}/\overline{IO}$, $\overline{RD}$, $\overline{ALE}$.
- ב. להלן קטע של תכנית בשפת הסף של המעבד 8086/88. ציין את ערך האוגר AL ואת ערכי דגל הנשא ודגל האפס בסיום כל הוראה בקטע הזה.

1. MOV AX, 28F9H
2. ADD AL, AH
3. AND AL, 0C3H

- ג. לרכיב קלט שכתובתו 20H מחוברים 8 מתגים. אם המתג סגור – ערכו '1', ואם הוא פתוח – ערכו '0'. כתוב תת-שגרה שתקלוט את ערכי המתגים, ותציב את סכום ערכיהם בתא שכתובתו 40H.

שאלה 8

במערכת מיקרו-מחשב, מחובר רכיב קלט/פלט בר-תכנות מסוג 82C55A למיקרו-מעבד 8086. הרכיב ממופה בכתובות $303H + 300H$, ונתוניו מפורטים בנספח לשאלה 8.

- א. רשום את מילת הבקרה שיש לשלוח לרכיב 82C55A, על-מנת שהמיפתחים יופעלו באופן

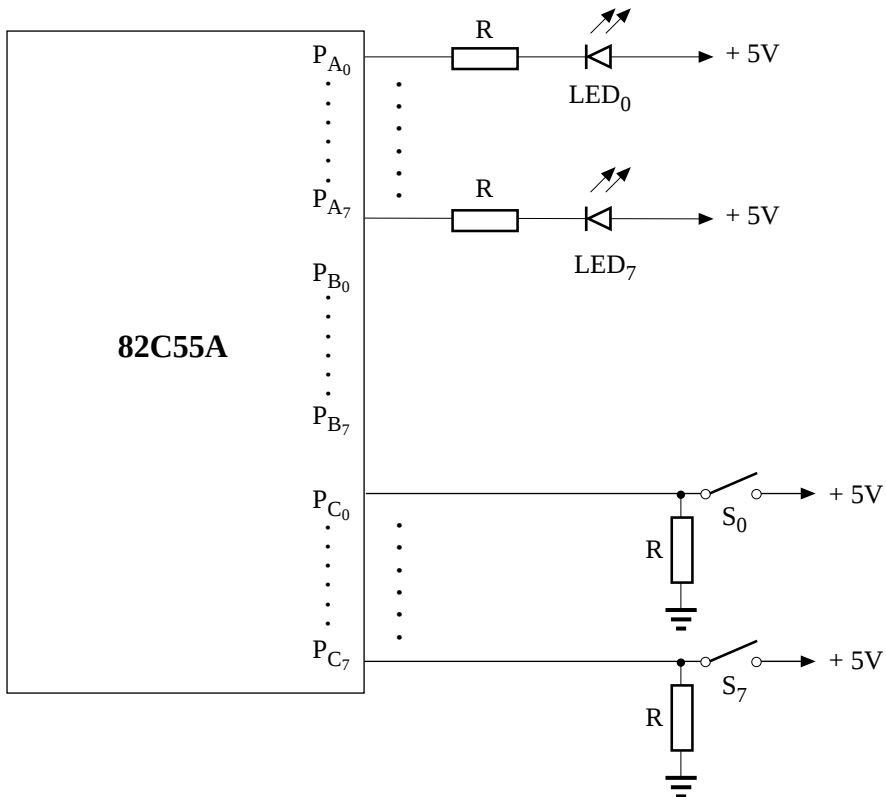
הזה: PORT A – פלט

PORT B – פלט

PORT C – קלט

ב. רשום תת־שגרה בשפת ASM-86, שתשלח את מילת הבקרה שרשמת בסעיף א' לאוגר הבקרה של הרכיב 82C55A.

ג. ל-PORT A של הרכיב 82C55A חוברו שמונה נוריות LED, ול-PORT C שלו חוברו שמונה מתגים, כמתואר באיור לשאלה 8. נוריות ה-LED מופעלות כאשר ערכו של מוצא PORT A הוא '0'.



איור לשאלה 8

כתוב תת־שגרה שתקלוט את תוכנו של כל אחד מן המתגים שב-PORT C. אם מתג נמצא במצב סגור (ON) – תגרום תת־שגרה להדלקת ה-LED המתאימה לו ב-PORT A (למתג S₀ מתאימה הנורית LED₀, למתג S₁ מתאימה הנורית LED₁, וכך הלאה).

שאלה 9

לפניך תת-שגרה הכתובה בשפת הסף של המיקרו-מעבד 8086/88 :

```
1  KK      :  MOV  SI, 10H
2              MOV  BX, 0H
3              MOV  AX, [SI]
4  AGAIN    :  ADD  BL, AH
5              JNC  AD
6              INC  BH
7  AD        :  DEC  AL
8              JNZ  AGAIN
9              MOV  [SI + 2], BX
10             RET
```

א. הסבר את ההוראות שמספריהן 3, 4, 8, 9.

ב. הסבר מה מבצעת תת-השגרה.

ג. בטבלה שלהלן נתונים תכני התאים בכתובות $10H \div 13H$ בסגמנט הנתונים שבזיכרון, לפני ביצוע תת-השגרה:

כתובת התא	תוכן התא
10H	03H
11H	80H
12H	10H
13H	20H

רשום את התכנים של תאים אלו לאחר ביצוע תת-השגרה.

בהצלחה!

82C55A CHMOS PROGRAMMABLE PERIPHERAL INTERFACE

- High Speed, "Zero Wait State"
Operation with 8 MHz 8086/88 and 80186/188
- 24 Programmable I/O Pins
- Low Power CHMOS
- Completely TTL Compatible

- Direct Bit Set/Reset Capability
- 2.5 mA DC Drive Capability on all I/O Port Outputs
- Control Word Read-Back Capability
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range

In MODE 0, each group of 12 I/O pins may be programmed in sets of 4 and 8 to be inputs or outputs. In MODE 1, each group may be programmed to have 8 lines of input or output. 3 of the remaining 4 pins are used for handshaking and interrupt control signals. MODE 2 is a strobed bi-directional bus configuration.

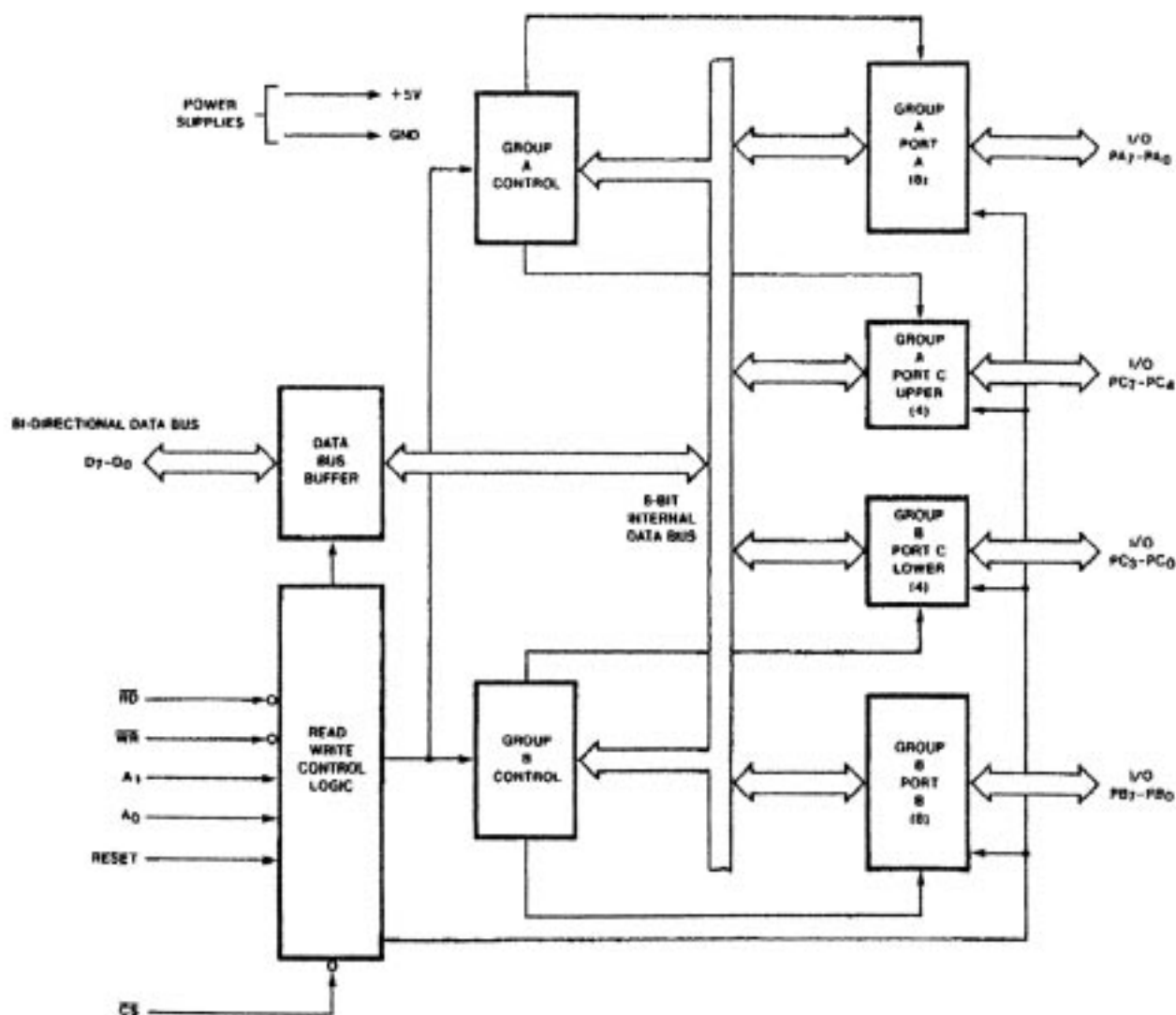


Figure 1. 82C55A Block Diagram

Table 1. Pin Description

Symbol	Pin Number Dip PLCC		Type	Name and Function					
PA ₃₋₀	1-4	2-5	I/O	PORT A, PINS 0-3: Lower nibble of an 8-bit data output latch/ buffer and an 8-bit data input latch.					
$\overline{RD}$	5	6	I	READ CONTROL: This input is low during CPU read operations.					
$\overline{CS}$	6	7	I	CHIP SELECT: A low on this input enables the 82C55A to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and $\overline{WR}$ are ignored otherwise.					
GND	7	8		System Ground					
A ₁₋₀	8-9	9-10	I	ADDRESS: These input signals, in conjunction $\overline{RD}$ and $\overline{WR}$, control the selection of one of the three ports or the control word registers.					
				A ₁	A ₀	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Input Operation (Read)
				0	0	0	1	0	Port A - Data Bus
				0	1	0	1	0	Port B - Data Bus
				1	0	0	1	0	Port C - Data Bus
				1	1	0	1	0	Control Word - Data Bus
				Output Operation (Write)					
				0	0	1	0	0	Data Bus - Port A
				0	1	1	0	0	Data Bus - Port B
				1	0	1	0	0	Data Bus - Port C
				1	1	1	0	0	Data Bus - Control
				Disable Function					
				X	X	X	X	1	Data Bus - 3 - State
				X	X	1	1	0	Data Bus - 3 - State
PC ₇₋₄	10-13	11,13-15	I/O	PORT C, PINS 4-7: Upper nibble of an 8-bit data output latch/ buffer and an 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.					
PC ₀₋₃	14-17	16-19	I/O	PORT C, PINS 0-3: Lower nibble of Port C.					
PB ₀₋₇	18-25	20-22, 24-28	I/O	PORT B, PINS 0-7: An 8-bit data output latch/buffer and an 8-bit data input buffer.					
V _{CC}	26	29		SYSTEM POWER: + 5V Power Supply.					
D ₇₋₀	27-34	30-33, 35-38	I/O	DATA BUS: Bi-directional, tri-state data bus lines, connected to system data bus.					
RESET	35	39	I	RESET: A high on this input clears the control register and all ports are set to the input mode.					
$\overline{WR}$	36	40	I	WRITE CONTROL: This input is low during CPU write operations.					
PA ₇₋₄	37-40	41-44	I/O	PORT A, PINS 4-7: Upper nibble of an 8-bit data output latch/ buffer and an 8-bit data input latch.					
NC		1, 12, 23, 34		No Connect					

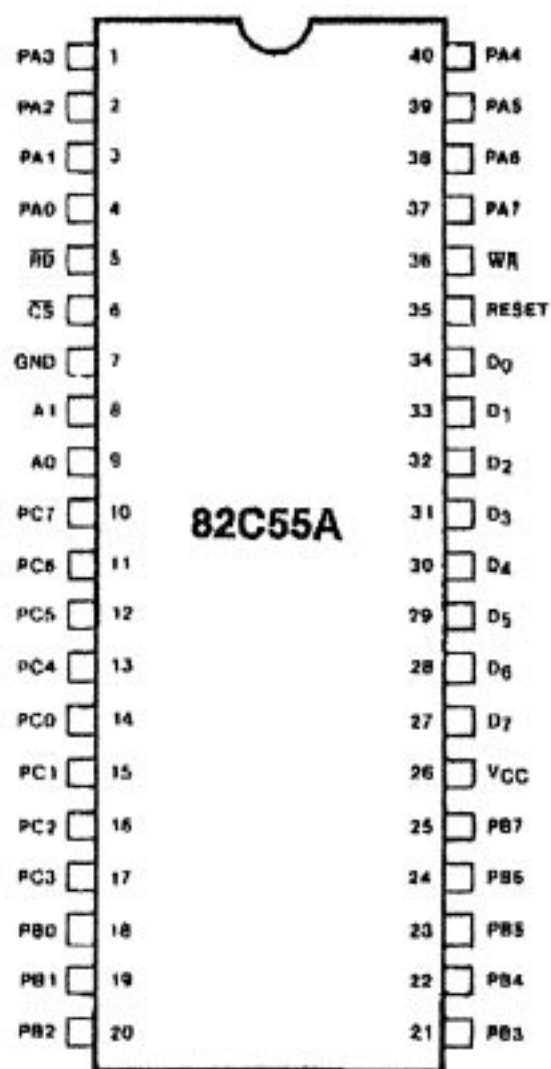
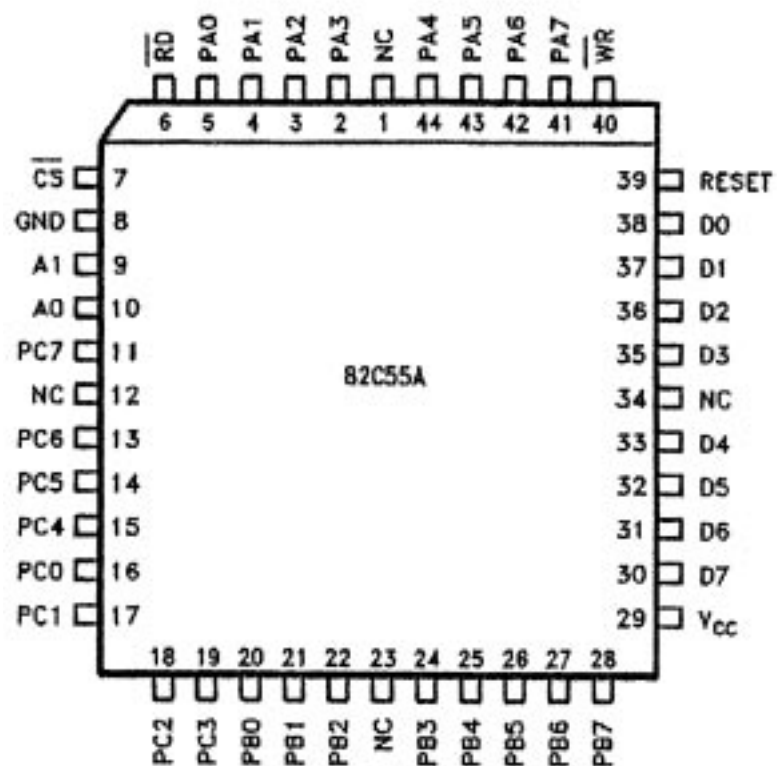


Figure 2. 82C55A Pinout

82C55A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that can be selected by the system software:

- Mode 0 — Basic input/output
- Mode 1 — Strobed input/output
- Mode 2 — Bi-directional Bus

When the reset input goes "high" all ports will be set to the input mode with all 24 port lines held at a logic "one" level by the internal bus hold devices.

After the reset is removed the 82C55A can remain in the input mode with no additional initialization required. This eliminates the need for pullup or pulldown devices in "all CMOS" designs. During the execution of the system program, any of the other modes may be selected by using a single output instruction. This allows a single 82C55A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

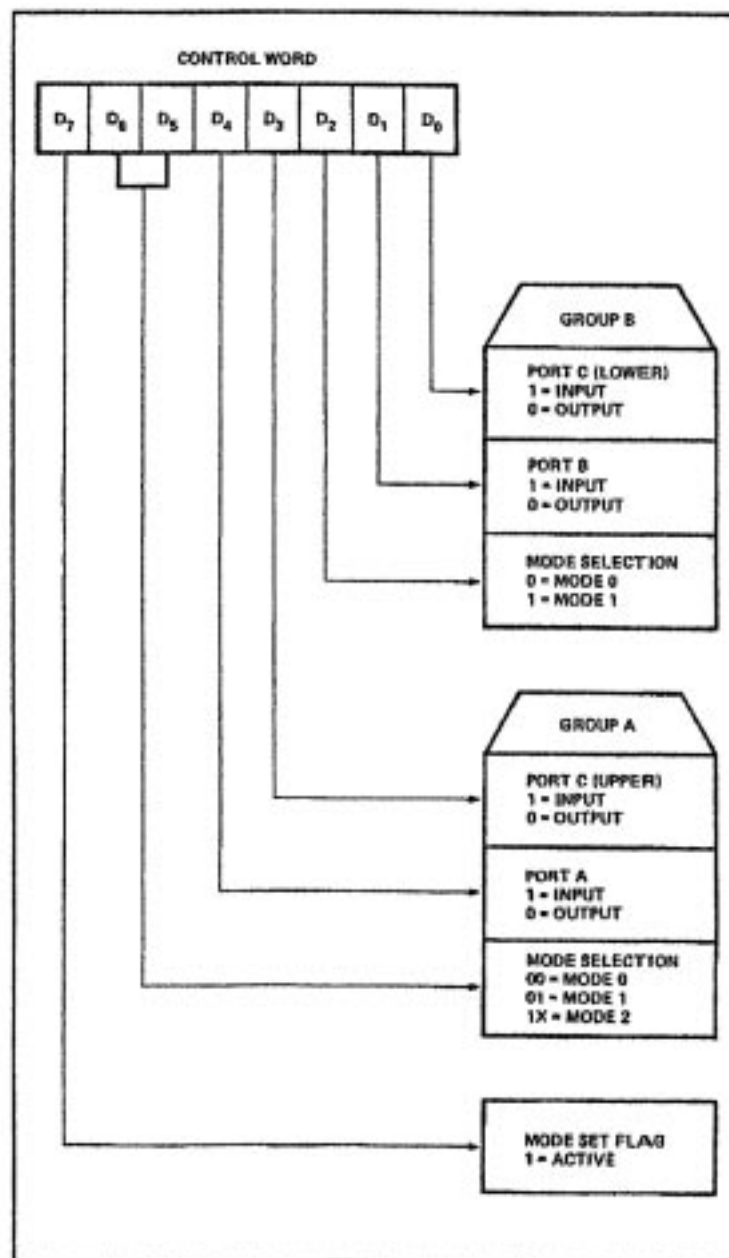


Figure 4. Mode Definition Format

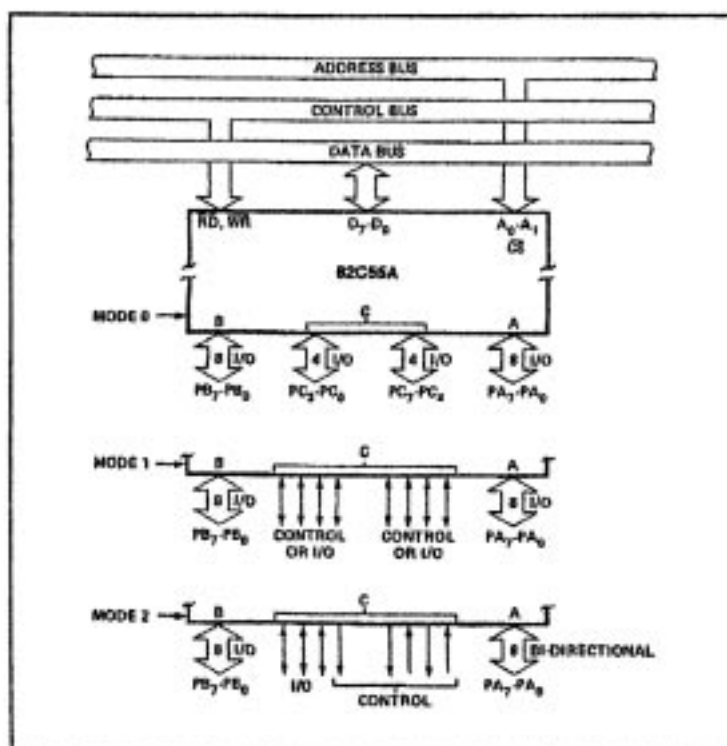


Figure 3. Basic Mode Definitions and Bus Interface

The mode definitions and possible mode combinations may seem confusing at first but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 82C55A has taken into account things such as efficient PC board layout, control signal definition vs PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

אין להעביר את הנוסחאון
לנבחן אחר

נוסחאון באלקטרוניקה ומחשבים לרמות א' – ב' (19 עמודים)

1. נוסחאות באלקטרוניקה תקבילית

חישובי הגבר

A_V — הגבר מתח
[V] V_o — מתח מוצא
[V] V_i — מתח מבוא

$$A_V = \frac{V_o}{V_i}$$

[dB] A_V — הגבר מתח בדציבלים

$$A_V = 20 \log \frac{V_o}{V_i}$$

A_I — הגבר זרם
[A] I_o — זרם מוצא
[A] I_i — זרם מבוא

$$A_I = \frac{I_o}{I_i}$$

[dB] A_I — הגבר זרם בדציבלים

$$A_I = 20 \log \frac{I_o}{I_i}$$

A_P — הגבר הספק
[W] P_o — הספק מוצא
[W] P_i — הספק מבוא
[Ω] R_L — התנגדות נגד העומס
[Ω] R_i — התנגדות מבוא

$$A_P = \frac{P_o}{P_i} = A_V \cdot A_I = A_I^2 \cdot \frac{R_L}{R_i} = A_V^2 \cdot \frac{R_i}{R_L}$$

[dB] A_P — הגבר הספק בדציבלים

$$A_P = 10 \log \frac{P_o}{P_i}$$

הגבר כולל של N דרגות
המחוברות בשרשרת (קסקדה) — A_{VT}

$$A_{VT} = A_{V1} \cdot A_{V2} \cdot A_{V3} \dots A_{VN}$$

$$A_{VT}(\text{dB}) = A_{V1}(\text{dB}) + A_{V2}(\text{dB}) + A_{V3}(\text{dB}) + \dots + A_{VN}(\text{dB})$$

הגבר כולל בדציבלים של
 N דרגות המחוברות בשרשרת
(קסקדה) — $A_{VT} [\text{dB}]$

מאזן הספקים

הספק מבוא — $P_I [W]$

$$P_I + P_{CC} = P_L + P_{diss}$$

הספק נצרך מהספקים — $P_{CC} [W]$

הספק העומס — $P_L [W]$

הספק מבזבז — $P_{diss} [W]$

משוב שלילי

הגבר עם משוב (בחוג סגור) — A_f

$$A_f = \frac{A}{1 + \beta A}$$

הגבר ללא משוב (הגבר חוג פתוח) — A

מקדם משוב — β

משוב מתח טורי

התנגדות מבוא עם משוב — $R_{if} [\Omega]$

$$R_{if} = R_i (1 + \beta A)$$

התנגדות מבוא ללא משוב — $R_i [\Omega]$

התנגדות מוצא עם משוב — $R_{of} [\Omega]$

$$R_{of} = \frac{R_o}{1 + \beta A}$$

התנגדות מוצא ללא משוב — $R_o [\Omega]$

טרנזיסטור דו-נושאי (בתחום הפעיל)

בהזנחת זרם הזליגה I_{CBO} :

$$I_C = \beta I_B, \quad I_E = (\beta + 1) I_B, \quad I_E = I_C + I_B$$

זרם הקולט — I_C [A]

זרם הפולט — I_E [A]

זרם הבסיס — I_B [A]

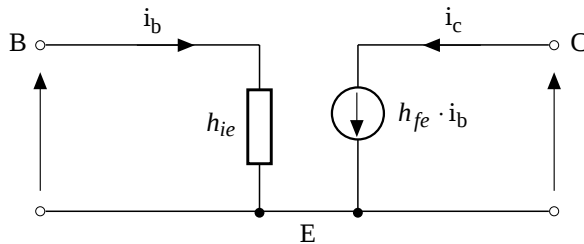
$$\alpha = \frac{I_C}{I_E} = \frac{\beta}{\beta + 1}, \quad \beta = \frac{\alpha}{1 - \alpha}$$

בהתחשב בזרם הזליגה I_{CBO} :

זרם הזליגה בין הקולט לבסיס, I_{CBO} [A] —
כאשר הפולט פתוח

$$I_C = \beta I_B + (\beta + 1) I_{CBO}$$

תרשים תמורה מקורב מסוג h של טרנזיסטור דו-נושאי



	CE	CE עם נגד R_E
A_I	h_{fe}	h_{fe}
R_i	h_{ie}	$h_{ie} + (1 + h_{fe})R_E$
A_V	$-\frac{h_{fe} \cdot R_L}{h_{ie}}$	$-\frac{h_{fe} \cdot R_L}{R_i}$
R_o	∞	∞

טרנזיסטור FET

- I_D [A] — זרם האפיק
- V_{GS} [V] — המתח בין השער למקור
- V_p [V] — מתח צביטה
- I_{DSS} [A] — זרם האפיק עבור $V_{GS} = 0$

$$I_D = I_{DSS} \left(1 - \frac{V_{GS}}{V_p} \right)^2$$

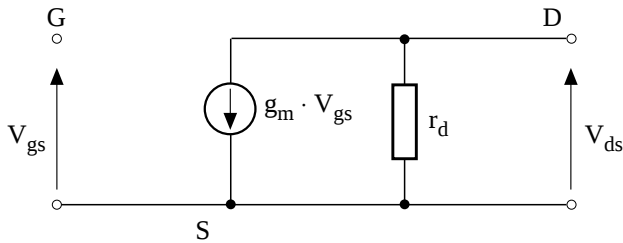
- g_m $\left[\frac{1}{\Omega} \right]$ — מוליכות הדדית

$$g_m = \frac{2I_{DSS}}{|V_p|} \left(1 - \frac{V_{GS}}{V_p} \right)$$

- g_{m0} $\left[\frac{1}{\Omega} \right]$ — מוליכות הדדית עבור $V_{GS} = 0$

$$g_{m0} = \frac{2I_{DSS}}{|V_p|}$$

תרשים תמורה מקורב של טרנזיסטור תוצא שדה – FET



- A_V — הגבר הטרנזיסטור בחיבור CS (מקור משותף)

$$A_V = -g_m \cdot (r_d \parallel R_L)$$

- A_V — הגבר הטרנזיסטור בחיבור CD (מפך משותף)

$$A_V \approx \frac{g_m \cdot R_s}{1 + (g_m + g_d) R_s}$$

- R_s $[\Omega]$ — נגד בטור למקור

$$g_d = \frac{1}{r_d}$$

דיודה

זרם דיודה — I_D [A]

מתח דיודה — V_D [V]

זרם רוויה אחורי — I_S [A]

מקדם $\eta = \begin{cases} 1 & -G_e \text{ (גרמניום)} \\ 2 & -S_i \text{ (סיליקון)} \end{cases}$

טמפרטורה — T [°K]

$$I_D = I_S (e^{V_D / \eta V_T} - 1)$$

$$V_T = \frac{T}{11,600}$$

מגבר הפרש

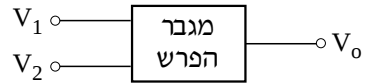
הפרש מתחי המבוא — V_d [V]

ממוצע מתחי המבוא — V_c [V]

הגבר הפרשי — A_d

הגבר האות המשותף — A_c

$$V_o = A_d \cdot V_d + A_c \cdot V_c$$



— כאשר: $V_1 = -V_2 = V_i$

$$A_d = \frac{V_o}{V_d} \Big|_{V_c=0} = \frac{V_o}{2V_i}$$

— כאשר: $V_1 = V_2 = V_i$

$$A_c = \frac{V_o}{V_c} \Big|_{V_d=0} = \frac{V_o}{V_i}$$

— $CMRR$ יחס דחיית האות המשותף

$$CMRR = \left| \frac{A_d}{A_c} \right|$$

מגברי הספק

$$P_L = V_{Leff} \cdot I_{Leff} = \frac{V_{Lmax} \cdot I_{Lmax}}{2} = \frac{V_{Leff}^2}{R_L} = I_{Leff}^2 \cdot R_L$$

הספק העומס — P_L [W]

מתח יעיל על העומס — V_{Leff} [V]

זרם יעיל בעומס — I_{Leff} [A]

מתח עומס שיאי — V_{Lmax} [V]

זרם עומס שיאי — I_{Lmax} [A]

נגד עומס — R_L [Ω]

ההספק הנצרך מהספק — P_{CC} [W]

הזרם הממוצע — I_{av} [A]

מתח המקור — V_{CC} [V]

$$P_{CC} = V_{CC} \cdot I_{av}$$

הזרם הממוצע של גל פועם — I_{av} [A]

זרם שיא — I_{max} [A]

$$I_{av} = \frac{2 \cdot I_{max}}{\pi}$$

הספק מבוזבז בטרנזיסטור — P_D [W]

$$P_D = P_{CC} - P_L$$

נצילות — η

$$\eta = \frac{P_L}{P_{CC}}$$

מתח שיא — V_{max} [V]

נצילות במגבר הספק — η

מסוג CLASS B

$$\eta = \frac{\pi}{4} \cdot \frac{V_{max}}{V_{CC}}$$

הספק פיזור מרבי — P_{Cmax} [W]

בשני הטרנזיסטורים

במגבר מסוג CLASS B

$$P_{Cmax} = \frac{2 \cdot V_{CC}^2}{\pi^2 \cdot R_L}$$

מגברי שרת

סוכם (INTEGRATOR)

$$\Delta V_o = \Delta V_C = \frac{-I_C \cdot t}{C}$$

ΔV_o [V] — השינוי במתח-המוצא

I_C [A] — הזרם הקבוע בקבל

t [sec] — זמן טעינת הקבל

מגבר מהפך

$$A_V = - \frac{R_f}{R_1}$$

R_f [Ω] — נגד המשוב

R_1 [Ω] — הנגד המחובר לכניסה המהפכת

מגבר עוקב

$$A_V = 1 + \frac{R_f}{R_1}$$

R_f [Ω] — נגד המשוב

R_1 [Ω] — הנגד היוצא מהכניסה המהפכת
לאדמה

מתנד גשר ויין

$$f_o = \frac{1}{2\pi R \cdot C}$$

f_o [Hz] — תדר התנודות

A_V — הגבר המתח

$$A_V \geq 3$$

2. אוסף פקודות למיקרו-מעבדים 8086/88

מקרא לאוגר הדגלים:

- X — מושפע מהפעולה (Modified) 0 — מתאפס
 U — לא מוגדר אחרי הפעולה (Undefined) 1 — מקבל "1"
 R — מוחזר מהמחסנית (Returned)

ADC	ADC destination, source Add with carry			Flags O D I T S Z A P C X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	ADC AX, SI
register, memory	9(10)+EA	1	2-4	ADC CX, BETA [SI]
memory, register	16(10)+EA	2	2-4	ADC ALPHA [BX] [SI], DI
register, immediate	4(4)	—	3-4	ADC BX, 256
memory, immediate	17(16)+EA	2	3-6	ADC GAMMA, 30H
accumulator, immediate	4(3-4)	—	2-3	ADC AL, 5

ADD	ADD destination, source Addition			Flags O D I T S Z A P C X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	ADD CX, DX
register, memory	9(10)+EA	1	2-4	ADD DI, [BX].ALPHA
memory, register	16(10)+EA	2	2-4	ADD TEMP, CL
register, immediate	4(4)	—	3-4	ADD CL, 2
memory, immediate	17(16)+EA	2	3-6	ADD ALPHA, 2
accumulator, immediate	4(3-4)	—	2-3	ADD AX, 200

AND	AND destination, source Logical and			Flags O D I T S Z A P C X X X U X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	AND AL, BL
register, memory	9(10)+EA	1	2-4	AND CX, FLAG_WORD
memory, register	16(10)+EA	2	2-4	AND ASCII [DI], AL
register, immediate	4(4)	—	3-4	AND CX, 0F0H
memory, immediate	17(16)+EA	2	3-6	AND BETA, 01H
accumulator, immediate	4(3-4)	—	2-3	AND AX, 01010000B

CALL	CALL target Call a procedure			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
near-proc	19(14)	1	3	CALL NEAR__PROC
far-proc	28(23)	2	5	CALL FAR__PROC
memptr 16	21(19)+EA	2	2-4	CALL PROC__TABLE [SI]
regptr 16	16(13)	1	2	CALL AX
memptr 32	37(38)+EA	4	2-4	CALL [BX], TASK [SI]

CLC	CLC (no operands) Clear carry flag			Flags O D I T S Z A P C 0
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	2(2)	—	1	CLC

CLI	CLI (no operands) Clear interrupt flag			Flags O D I T S Z A P C 0
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	2(2)	—	1	CLI

CMP	CMP destination, source Compare destination to source			Flags O D I T S Z A P C X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	CMP BX, CX
register, memory	9(10)+EA	1	2-4	CMP DH, [ALPHA]
memory, register	9(10)+EA	1	2-4	CMP [BP+2], SI
register, immediate	4(3)+EA	—	3-4	CMP BL, 02H
memory, immediate	10(10)+EA	1	3-6	CMP RADAR [BX], [DI], 3420H
accumulator, immediate	4(3-4)	—	2-3	CMP AL, 00010000B

CMPS	CMPS des-string, source-string Compare string			Flags O D I T S Z A P C X X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
dest-string, source-string (repeat)	22(22)	2	1	CMPS BUSS1, BUFF2
dest-string, source-string	9+22/rep (5+22/rep)	2/rep	1	REPE CMPS ID, KEY

DAA	DAA (no operands) Decimal adjust for addition			Flags O D I T S Z A P C U X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	4(4)	—	1	DAA

DAS	DAS (no operands) Decimal adjust for subtraction			Flags O D I T S Z A P C U X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	4(4)	—	1	DAS

DEC	DEC destination Decrement by 1			Flags O D I T S Z A P C X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
reg 16	3(3)	—	1	DEC AX
reg 8	3(3)	—	2	DEC AL
memory	15(15)+EA	2	2-4	DEC ARRAY [SI]

DIV	DIV source Division, unsigned			Flags O D I T S Z A P C U U U U U U
Operands	Clocks	Transfers*	Bytes	Coding Example
reg 8	80-90(29)	—	2	DIV CL
reg 16	144-162(38)	—	2	DIV BX
mem 8	86-96+EA (35)	1	2-4	DIV [ALPHA]
mem 16	150-168+ EA(94)	1	2-4	DIV TABLE [SI] / DIV [TABLE + SI]
IN	IN accumulator, port Input byte or word			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
accumulator, immed 8	10(10)	1	2	IN AL, 0FEH / IN AX, OFEH
accumulator, DX	8(8)	1	1	IN AL, DX / IN AX, DX
INC	INC destination Increment by 1			Flags O D I T S Z A P C X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
reg 16	3(3)	—	1	INC CX
reg 8	3(3)	—	2	INC BL
memory 8	15(15)+EA	2	2-4	INC ALPHA [DI] [BX]
INT	INT interrupt-type Interrupt			Flags O D I T S Z A P C X X
Operands	Clocks	Transfers*	Bytes	Coding Example
immed 8 (type=3)	52(45)	5	1	INT 3
immed 8 (type≠3)	52(47)	5	2	INT 67
IRET	IRET (no operands) Interrupt Return			Flags O D I T S Z A P C R R R R R R R R
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	32(28)	3	1	IRET

JC	JC short-label Jump if carry			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
short-label	16 or 4(13 or 4)	—	2	JC CARRY_SET

JE/JZ	JE/JZ short-label Jump if equal/Jump if zero			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
short-label	16 or 4(13 or 4)	—	2	JZ ZERO

JMP	JMP target Jump			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
short-label	15(13)	—	2	JMP SHORT
near-label	15(13)	—	3	JMP WITHIN_SEGMENT
far-label	15(13)	—	5	JMP FAR_LABEL
memptr 16	18(17)+EA	1	2-4	JMP [BX] TARGET
regptr 16	11(11)	—	2	JMP CX
memptr 32	24(26)+EA	2	2-4	JMP OTHER_SEG

JNC	JNC short-label Jump if not carry			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
short-label	16 or 4(13 or 4)	—	2	JNC NOT_CARRY

JNE/JNZ	JNE/JNZ short-label Jump if not equal/Jump if not zero			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
short-label	16 or 4(13 or 4)	—	2	JNE NOT_EQUAL

LEA	LEA destination, source Load effective address			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
reg 16, mem 16	2(6)+EA	—	2-4	LEA BX, [BP] [DI]

LOOP	LOOP short-label Decrement cx and jump if not zero			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
short label	17/5(15/5)	—	2	LOOP AGAIN

MOV	MOV destination, source Move			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
memory, accumulator	10(9)	1	3	MOV ARRAY [SI], AL
accumulator, memory	10(8)	1	3	MOV AX, TEMP_RESULT
register, register	2(2)	—	2	MOV AX, CX
register, memory	8(12)+EA	1	2-4	MOV BP, STACK_TOP
memory, register	9(9)+EA	1	2-4	MOV COUNT [DI], CX
register, immediate	4(3-4)	—	2-3	MOV CL, 2
memory, immediate	10(12-13) +EA	1	3-6	MOV MASK [BX] [SI], 2CH
seg-reg, reg 16	2(2)	—	2	MOV ES, CX
seg-reg, mem 16	8(9)+EA	1	2-4	MOV DS, SEGMENT_BASE
reg 16, seg-reg	2(2)	—	2	MOV BP, SS
memory, seg-reg	9(11)+EA	1	2-4	MOV [BX] SEG_SAVE, CS

MOVS/MOVSW	MOVS/MOVSW (no operands) Move string (byte/word)			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	18(9)	2	1	MOVS
(repeat) (no operands)	9+17/rep (8+8/rep)	2/rep	1	REP MOVSW

MUL	MUL source Multiplication, unsigned			Flags O D I T S Z A P C X U U U X
Operands	Clocks	Transfers*	Bytes	Coding Example
reg 8	70-77 (26-28)	—	2	MUL BL
reg 16	118-133 (35-37)	—	2	MUL CX
mem 8	76-83+ EA(32-34)	1	2-4	MUL MONTH [SI]
mem 16	124-139+ EA(41-43)	1	2-4	MUL [BAUD_RATE]

NOP	NOP (no operands) No Operation			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	3(3)	—	1	NOP

NOT	NOT destination Logical not			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
register	3(3)	—	2	NOT AX
memory	16(3)+EA	2	2-4	NOT [CHARACTER]

OR	OR destination, source Logical inclusive or			Flags O D I T S Z A P C X X X U X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	OR AL, BL
register, memory	9(10)+EA	1	2-4	OR DX, PORT_ID [DI]
memory, register	16(10)+EA	2	2-4	OR FLAG_BYTE, CL
accumulator, immediate	4(3-4)	—	2-3	OR AL, 01101100B
register, immediate	4(4)	—	3-4	OR CX, 01H
memory, immediate	17(16)+EA	2	3-6	OR [BX], CMD_WORD

OUT	OUT port, accumulator Output byte or word			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
immed 8, accumulator	10(9)	1	2	OUT 44, AX
DX, accumulator	8(7)	1	1	OUT DX, AL

POP	POP destination Pop word off stack			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
register	8(10)	1	1	POP DX
seg-reg (CS illegal)	8(8)	1	1	POP DS
memory	17(20)+EA	2	2-4	POP [PARAMETER]

PUSH	PUSH source Push word onto stack			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
register	11(10)	1	1	PUSH SI
seg-reg (CS legal)	10(9)	1	1	PUSH ES
memory	16(16)+EA	2	2-4	PUSH RETURN_CODE [SI]

RCL	RCL destination, count Rotate left through carry			Flags O D I T S Z A P C X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, 1	2(2)	—	2	RCL CX, 1
register, CL	8+4/ bit(5+1/bit)	—	2	RCL AL, CL
memory, 1	15(15)+EA	2	2-4	RCL ALPHA, 1
memory, CL	20+4/ bit(17+ 1/bit)+EA	2	2-4	RCL [BP].PARAM, CL

RCR	RCR destination, count Rotate right through carry			Flags O D I T S Z A P C X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, 1	2(2)	—	2	RCR BX, 1
register, CL	8+4/ bit(5+1/bit)	—	2	RCR BL, CL
memory, 1	15(15)+EA	2	2-4	RCR [BX].STATUS, 1
memory, CL	20+4/ bit(17+ 1/bit)+EA	2	2-4	RCR ARRAY [DI], CL

REP	REP (no operands) Repeat string operation			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	2(2)	—	1	REP MOVS DEST, SRCE

RET	RET optional-pop-value Return from procedure			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
(intra-segment, no pop)	16(16)	1	1	RET
(intra-segment, pop)	20(18)	1	3	RET 4
(inter-segment, no pop)	26(22)	2	1	RET
(inter-segment, pop)	25(25)	2	3	RET 2

ROL	ROL destination, count Rotate left			Flags O D I T S Z A P C X X X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, 1	2(2)	—	2	ROL BX, 1
register, CL	8+4/ bit(5+1/bit)	—	2	ROL DI, CL
memory, 1	15(15)+EA	2	2-4	ROL FLAG_BYTE [DI], 1
memory, CL	20+4/ bit(17+ 1/bit)+EA	2	2-4	ROL ALPHA, CL

ROR	ROR destination, count Rotate right			Flags O D I T S Z A P C X X X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, 1	2(2)	—	2	ROR BX, 1
register, CL	8+4/ bit(5+1/bit)	—	2	ROR BX, CL
memory, 1	15(15)+EA	2	2-4	ROR PORT_STATUS, 1
memory, CL	20+4/ bit(17+ 1/bit)+EA	2	2-4	ROR CMD_WORD, CL

SBB	SBB destination, source Subtract with borrow			Flags O D I T S Z A P C X X X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	SBB BX, CX
register, memory	9(10)+EA	1	2-4	SBB DI, [BX].PAYMENT
memory, register	16(10)+EA	2	2-4	SBB BALANCE, AX
accumulator, immediate	4(3-4)	—	2-3	SBB AX, 2
register, immediate	4(4)	—	3-4	SBB CL, 1
memory, immediate	17(16)+EA	2	3-6	SBB COUNT [SI], 10

STC	STC (no operands) Set carry flag			Flags O D I T S Z A P C 1
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	2(2)	—	1	STC

STI	STI (no operands) Set interrupt enable flag			Flags O D I T S Z A P C 1
Operands	Clocks	Transfers*	Bytes	Coding Example
(no operands)	2(2)	—	1	STI

SUB	SUB destination,source Subtraction			Flags O D I T S Z A P C X X X X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	SUB CX, BX
register, memory	9(10)+EA	1	2-4	SUB DX, MATH_TOTAL [SI]
memory, register	16(10)+EA	2	2-4	SUB [BP+2], CL
accumulator, immediate	4(3-4)	—	2-3	SUB AL, 10
register, immediate	4(4)	—	3-4	SUB SI, 5280

TEST	TEST destination, source Non-destructive logical and			Flags O D I T S Z A P C X X X U X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	TEST SI, DI
register, memory	9(10)+EA	1	2-4	TEST SI, END_COUNT
accumulator, immediate	4(3-4)	—	2-3	TEST AL, 00100000B
register, immediate	5(4)	—	3-4	TEST BX, 0CC4H
memory, immediate	11(10)+EA	—	3-6	TEST [RETURN_COUNT], 01H

XCHG	XCHG destination, source Exchange registers			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
accumulator, reg 16	3(3)	—	1	XCHG AX, BX
memory, register	17(17)+EA	2	2-4	XCHG SEMAPHORE, AX
register, register	4(4)	—	2	XCHG AL, BL

XLAT	XLAT source-table Translate			Flags O D I T S Z A P C
Operands	Clocks	Transfers*	Bytes	Coding Example
source-table	11(11)	1	1	XLAT ASCII_TAB

XOR	XOR destination, source Logical exclusive or			Flags O D I T S Z A P C 0 X X U X X
Operands	Clocks	Transfers*	Bytes	Coding Example
register, register	3(3)	—	2	XOR CX, BX
register, memory	9(10)+EA	1	2-4	XOR CL, MASK_BYTE
memory, register	16(10)+EA	2	2-4	XOR ALPHA [SI], DX
accumulator, immediate	4(3-4)	—	2-3	XOR AL, 01000010B
register, immediate	4(4)	—	3-4	XOR SI, 00C2H
memory, immediate	17(16)+EA	2	3-6	XOR RETURN_CODE, 0D2H