

Web & Net Design - System Analyst
Computer Engineering Teach. Advice

מצגות לימודיות להוראת מיקרו-מעבדים, אסמבלר, תקשורת, מערכות הפעלה.

שאל קובל

מבוא למערכות מחשב ואסמבלי

Computer Systems & Assembly

יום שישי 05 מאי 2006

1
saul@coval.net
http://www.coval.net

ערב טוב !

נא לשמור על
השקט!! תודה

תכינו את דפי העזר וכלי כתיבה
ורשמו הערות על פי הצורך

יום שישי 05 מאי 2006

על פי תוכנית לימודים של משרד החינוך:

מערכות מחשב ואסמבלר

חלק מהמצגות על פי בסיס ומלל של המצגות של:
גב' שרה פולק (מט"ח)

תודתי לגב' שרה פולק עבור העזרה
והנחייה בתוכנית החדשה

יום שישי 05 מאי 2006

3
saul@coval.net
http://www.coval.net

על פי תוכנית לימודים של משרד החינוך:

מערכות מחשב ואסמבלר

http://www.csit.org.il/main/default.aspx?MenuShow=DOCS&Doc_ID=47

כל המצגות, חומר לימוד וקישורים נוספים, ניתן להוריד מהאתר של המרצה:

<http://www.coval.net/teach/morim>

קשר עם המרצה:

saul @ coval.net

יום שישי 05 מאי 2006

4
saul@coval.net
http://www.coval.net

מטרות המודול

הכרת המרכיבים הפנימיים של מיקרו-מעבד (CPU)

הכרת המרכיבים של תכנות באסמבלי 8086

הקשר בין המבנה של המעבד לבין התוכנה

יום שישי 05 מאי 2006

6
saul@coval.net
http://www.coval.net

פילוסופיה של הוראת תכנות ופתרון בעיות (הבסיס של השיטות ההוראה שלי)

Philosophy of Teaching Programming and Problem Solving

יום שישי 05 מאי 2006

6
saul@coval.net
http://www.coval.net

הדוגמה הוא התיאור המתאים והחשוב ביותר

The paradigm is paramount

דוגמה: "בכיתה שלי, אתה עובד עם גולם אחד"

"In my classroom, you are working with an idiot"



7

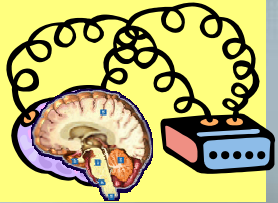
saul@covad.net
http://www.covad.net

יום שישי 05 מאי 2006

או - OR

דוגמה: "המחשב הטוב ביותר זה המוח שלך."

"The best computer is your mind"



8

saul@covad.net
http://www.covad.net

יום שישי 05 מאי 2006

זה כמו ללמוד רכיבה על סוס

It's like learning to ride a horse

לומדים מיומנות מיוחדת
גם לומדים איך הסוס חושב

- You learn specific skills
- You also have to learn how the horse thinks



9

saul@covad.net
http://www.covad.net

יום שישי 05 מאי 2006

או במילים של אלברט איינשטיין

Or in the words of Albert Einstein...

"אני רוצה להכיר את המחשבות של ה',
ההמשך הם פרטים קטנים"

"I want to know God's thoughts, the rest are details"



10

saul@covad.net
http://www.covad.net

יום שישי 05 מאי 2006

תכנית הלימודים

המטרות העיקריות של המקצוע הם:

- להקנות לתלמיד היכרות עם הארכיטקטורה של המחשב האישי – הכרת היחידות הבסיסיות במיקרו-מחשב והקשר ביניהם.
- לאפשר היכרות והתנסות בפיתוח תוכניות בשפת סף.
- להבין כיצד מתבצעת תוכנית בשפת סף במחשב.
- להכיר את ההתפתחויות שחלו במחשבים מודרניים שנועדו לשפר את זמן הריצה של תוכנות מחשב.



11

saul@covad.net
http://www.covad.net

יום שישי 05 מאי 2006

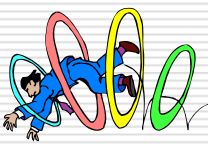
שרה פולק + שאול קובל מטרות מחשבים

חומרים לתלמיד

ספר הלימוד: (תרם יצאה למחירה)

- פרק ראשון: מבוא - המחשב הדיגיטאלי
- פרק שני: ייצוג מידע במחשב
- פרק שלישי: פעולות על ייצוג בינארי במחשב
- פרק 4: שפת סף ומודל תכנותי של מעבד 8086
- פרק 5: תכנות בסיסי בשפת סף
- פרק 6: תכנות מתקדם בשפת סף
- פרק 7: מעבדים מתקדמים

חוברת פעילויות ותרגילי מעבדה



12

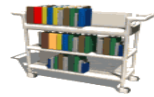
saul@covad.net
http://www.covad.net

יום שישי 05 מאי 2006

שרה פולק + שאול קובל מטרות מחשבים

ספרות מומלצת (ע"י תוכנית הלימודים)

- ארגון המחשב ושפת סף, אוניברסיטה הפתוחה (עדיין בתכנון).
- שפת סף 88/8086 הוצאת ב"ס לטכנולוגיה של האוניברסיטה הפתוחה.
- ארגון המחשב ותכנות, אריה פיקז – הוצאת האוניברסיטה הפתוחה.
- משפחת המיקרופרוססורים 8086/88, ג'ון אופנבק, תרגום לעברית של האוניברסיטה הפתוחה.
- מחשבים ומיקרומעבדים חלק א' וחלק ב' – ביה"ס לטכנולוגיה של האוניברסיטה הפתוחה, מט"ח ומשרד החינוך.
- *Computer Organization and Assembly Language Programming*, Thorne M.
- *The 80x86 Family, Design, Programming and Interfacing*, Uffenbeck J., Prentice-Hall International, Inc.
- *Structured Computer Organization*, by Andrew Tanenbaum



יום ששי 05 מאי 2006

שרה פולק + שאול קובל מטרות מחשבים

14
saul@coval.net
http://www.coval.net

המלצות פדגוגיות של מט"ח



- בעיות:
 - מושגים רבים חדשים.
 - שילוב מהיר של התנסות.
 - סדר הלימוד: הצגת חומרה קודם?
- הצעות:
 - מבט-על מופשט והעמקה בצורה ספירלית.
 - להציג פרק 3 (עיבוד על ייצוג בינארי יחד עם פרק 5).
 - שימוש בסימולאטור.
 - קישור לידע קיים במדעי המחשב.
 - התייחסות לתפישות של תלמידים על המחשב ואופן השימוש בו.

יום ששי 05 מאי 2006

שרה פולק + שאול קובל מטרות מחשבים

13
saul@coval.net
http://www.coval.net

המחשב הדיגיטאלי



- המחשב הדיגיטאלי הוא מכונה המורכבת מרכיבים רבים ובאמצעותו, ניתן לפתור בעיות שונות שפתרון מוגדר כתכנה.



יום ששי 05 מאי 2006

שרה פולק + שאול קובל מטרות מחשבים

16
saul@coval.net
http://www.coval.net

פרק ראשון - מבוא

מטרות

- הצגת מבט-על של מחשב ואופן ביצוע תכנית.
- הצגת הסיבות ללימוד המקצוע.
- הכרת המושגים הבאים: תכנה, חומרה, ארכיטקטורה, ארגון, מהדר/מפרש, שפות נמוכות ושפות עיליות, ארכיטקטורת פון ניומן.
- תכנים:
 - ארכיטקטורה וארגון של מחשב.
 - מחזור הבאה-ביצוע של הוראות ומבנה מחשב פשוט.
 - היסטוריה: התפתחות של מחשבים.



יום ששי 05 מאי 2006

שרה פולק + שאול קובל מטרות מחשבים

15
saul@coval.net
http://www.coval.net

מרכיבי חומרת המחשב

זיכרון

קלט/פלט

מעבד

דיסקים

All Rights Reserved - שאול קובל מטרות מחשבים

מרכיבי חומרת המחשב

All Rights Reserved - שאול קובל מטרות מחשבים

שפת המחשב – שפת מכונה (Machine Language)

High Level Language Program (VB, C++)

Compiler

Assembly Language Program (8086)

Assembler

Machine Language Program (8086)

Machine Interpretation

Control Signal Specification

temp = v[k];
v[k] = v[k+1];
v[k+1] = temp;

mov Al, Dh
add Dh, 2Ah
push Ax
Lea Si, Data

בשפה בינארית הוראה אוסף 0 ו-1

1100 0110 1010 1111 0101 1000 0000 1001
0101 1000 0000 1001 1100 0110 1010 1111

חסרון – תלות בחומרה, חוסר ניידות

יום שישי 05 מאי 2006

saal@caval.net
http://www.caval.net

Languages of Computers- שפות המחשב

שפת מכונה - Machine Language

- התוכניות מורכבות מפקודות מכונה.
- הפעלה מיידית ללא תחנות ביניים.
- טיפול ישיר באוגרי המכונה (CPU).
- ניצול טוב יותר במשאבי המערכת.
- קשה לתכנות

שפת אסמבלי - Assembly language

- מהדורה מתקדמת של שפת מכונה עם כיוון של הקלה למתכנת.
- שפה דרך צפנים לפעולות וכתובות.
- יש צורך במהדר לתרגום לשפת מכונה.

שפה ברמה גבוהה - High-Level Language

- תוכניות מורכבות מפקודות, ניתן לתרגם כל אחד מהם בסידרה של פקודות מכונה.
- יש צורך במהדר לתרגום לשפת מכונה.
- יחסית קל לתכנות (ביחס לקודמים).
- משאבי חומרה לא מנוצלים במקסימום.

יום שישי 05 מאי 2006

saal@caval.net
http://www.caval.net

C	Assembly Language
size = value;	mov AX, value mov size, AX
sum += x + y + z;	mov AX, sum add AX, x add AX, y add AX, z mov sum, AX

Assembly Language	Machine Language (in Hex)
inc result	FF060A00
mov class_size, 45	C7060C002D00
and mask, 128	80260E0080
add marks, 10	83060F000A

יום שישי 05 מאי 2006

שרה פולק + שאול קובל מפרסות מחשבים

saal@caval.net
http://www.caval.net

מודל השכבות - Multi-Level Machine

5 Program-Oriented language level – שפה עלית – תרגום (מהדר)

4 Assembly language level – שפת אסמבלי – תרגום (אסמבלר)

3 Operational – מערכת הפעלה – מפרש חלקי דרך מערכת הפעלה

2 Instruction set - פקודות – מפרש או ביצוע ישיר

1 Micro-architecture level – רמת המחשב – חומרה

0 Digital logic level – רמה דיגיטלית

יום שישי 05 מאי 2006

saal@caval.net
http://www.caval.net

מודל השכבות - Multi-Level Machine

רמה 0:

– שרשרת אולטימ ופסולות אולטימ

– מיאוס פכיים כאו אולטימ

– מונוס אל פכיים אלקטרוניים

רמה דיגיטלית – Digital logic level

יום שישי 05 מאי 2006

saal@caval.net
http://www.caval.net

מודל השכבות - Multi-Level Machine

רמה 1:

ארכיטקטורה של אפסון של האכונה

(HAS – hardware system architecture)

מתאר באופן עקרוני כיצד המחשב פועל

כדי לבצע את הוראות האכונה

הן היחידות העיקריות המחשב ואם הקשר

בין

איזה מידע צורם בין היחידות השונות

רמת המחשב – Micro-architecture level

יום שישי 05 מאי 2006

saal@caval.net
http://www.caval.net

מודל השכבות - Multi-Level Machine

רמה 2:
ארכיטקטורה של אוסף הוראות המכונה
(Instruction Set Architecture - ISA) וקיצור ISA)
מהן הפעולות שמחשב מסוגל לבצע למשל:
האם כפל היא אחת מפעולות של המחשב
או שלביצוע הכפל עלינו להשתמש
בפעולות חיבור בלבד?
מבנה של הוראה וסוגי נתונים בהם
המחשב מטפל ואופן ראייה אליהם.

2
פקודות - Instruction set
architecture level

יום שישי 05 מאי 2006

מודל השכבות - Multi-Level Machine

רמה 3 - מערכת הפעלה (operating system)
מערכת הפעלה היא תכנה המתווכת בין משתמש
מחשב (משתמש ביישום או מתכנת) לבין החומרה
ומטפלת בהקצאת משאבי המחשב וביצוע
פעולות הפעולות להעלית תכנית
מערכת הפעלה אחראית את ISA (Instruction Set Architecture)
(Architecture) ומספקת שירותים נוספים
ליוזמה. מערכת הפעלה מטפלת בהעברה של
תכניות מספר תכניות במקביל
מערכת הפעלה מספקת שירותים שונים וביניהם
טיפול בקצבים שמאפשרים על ביצועות חיצוניים.
זיסק קטן למשל ובאחרים.

3
מערכת הפעלה - Operational
System machine level

יום שישי 05 מאי 2006

מודל השכבות - Multi-Level Machine

רמה 4 - שפת אסמבלי
(מתורגמת לשפת מכונה)
רמה 5 - שפת עילית
(מתורגמת לשפת מכונה)
5 שפה עלית - Program-Oriented language level
4 שפת אסמבלי - Assembly language level
תרגום (מחדר)
תרגום (אסמבלר)

2 פקודות - Instruction set
architecture level
1 רמת המחשב - Micro-architecture level

יום שישי 05 מאי 2006

מודל השכבות - Multi-Level Machine

תרגום (אסמבלר)
מפרש חלקי דרך מערכת הפעלה
מפרש או ביצוע ישיר
4 שפת אסמבלי - Assembly language level
2 פקודות - Instruction set
architecture level
1 רמת המחשב - Micro-architecture level

**בקורס זה אנו מתייחסים בהרחבה
לרמות: 1, 2 ו-4**

יום שישי 05 מאי 2006

ארכיטקטורה וארגון המחשב

ארכיטקטורה (Architecture):
מאפיינים של מחשב מנקודת מבט של מתכנת:
פעולות שמחשב מסוגל לבצע, למשל, האם כפל היא אחת מפעולות של
המחשב או שלביצוע הכפל עלינו להשתמש בפעולות חיבור בלבד;
מבנה של הוראה;
סוגי נתונים בהם המחשב מטפל ואופן הגישה אליהם.

מתייחס לשאלה "מה" המחשב יודע לעשות?
רמה 2 (Instruction Set Architecture) ISA - אוסף
הוראות המכונה,
רמה 1 (Hardware System Architecture) HSA - מבנה
מינימאלי של המחשב ומבנה ה-CPU.

יום שישי 05 מאי 2006

ארכיטקטורה וארגון המחשב

ארגון (Organization):
מנקודת מבט של ארכיטקט/מהנדס, מתאר באופן עקרוני כיצד המחשב פועל
כדי לבצע את הוראות:
☐ באילו רכיבי חומרה אנו משתמשים בהוראה,
☐ מהן היחידות העיקריות במחשב ומה הקשר ביניהם,
☐ איזה מידע זורם בין היחידות השונות.
☐ הגדרה לוגית של המעגלים הדרושים לביצוע הוראות המכונה, למשל האם
פעולת כפל תמומש בעזרת חומרה (מעגל אלקטרוני) או בתכנה על ידי
סדרה של פעולות חיבור.

רמה 1 (Hardware System Architecture), HSA
רמה 0 המכונה הדיגיטאלית.

יום שישי 05 מאי 2006

ארכיטקטורה וארגון המחשב

- במחשבים אישיים נשמר עקרון תאימות (Compatibility):
- תכניות שנכתבו למחשבים ישנים (שנות ה-80) מורצים גם במחשבים מודרניים:
- ISA (Instruction Set Architecture) של מחשב מודרני כולל ISA של מחשב ישן.
- HAS (Hardware System Architecture) והמימוש של מחשב ישן שונה מזה של מחשב מודרני.

יום ששי 05 מאי 2006

שרה פולק + שאול קובל מטרות מחשבים

37
saul@coval.net
http://www.coval.net

למה ללמוד שפת סף/שפת מכונה?

- דרך טובה ללמוד את מבנה המחשב והקשר בין תכנה לחמרה וכך לשפר את ההבנה והתכניות בשפות עיליות
- כשמרצים רקורסיה למה יש בעיה של זיכרון?
- למה מתקבלת תוצאה נכונה כשמחברים שני מספרים שלמים גדולים?
- איך משפיעים מבני התכנות שבהם אנו משתמשים על זמן ביצוע התכנית?
- כתיבת יישומים המתייחסים להתקני חמרה וחלקים ממערכת ההפעלה.
- יישומים של זמן אמת (כיום משתמשים גם בשפות עיליות – שפת C)
- בכתיבת קומפילרים צריך להכיר את ISA (Instruction Set Architecture).

יום ששי 05 מאי 2006

שרה פולק + שאול קובל מטרות מחשבים

38
saul@coval.net
http://www.coval.net

השוואת דרך פעולה והבקרה על המרכיבים הפנימיים והחיצוניים של האדם ושל חומרת המחשב



39



40

השוואת דרך פעולה והבקרה על המרכיבים הפנימיים והחיצוניים של המחשב



41

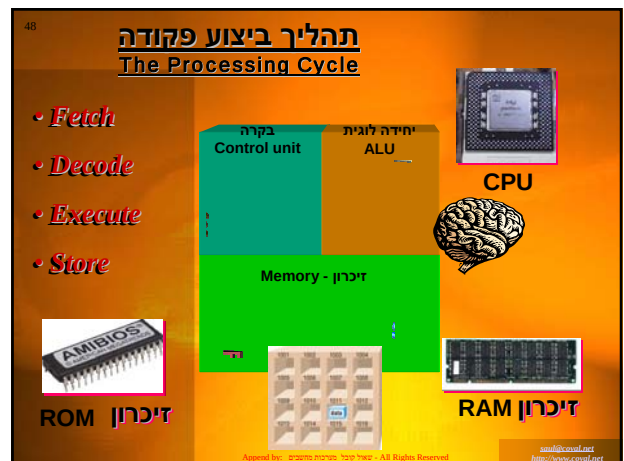
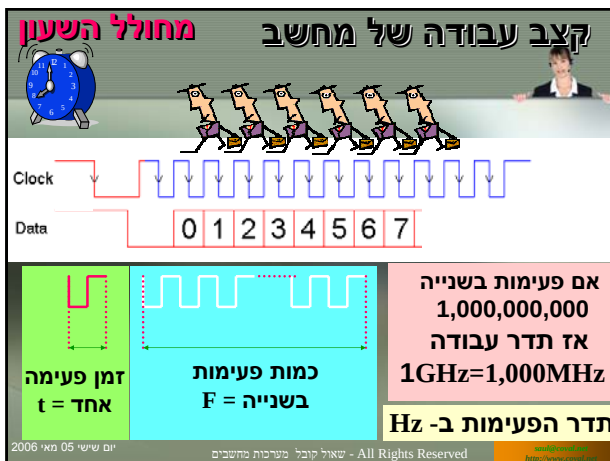
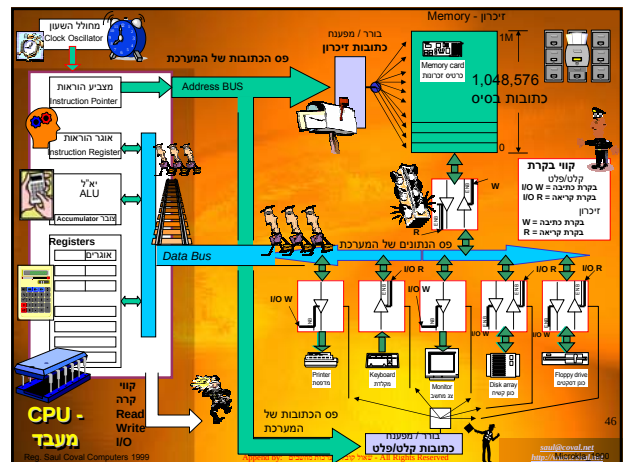
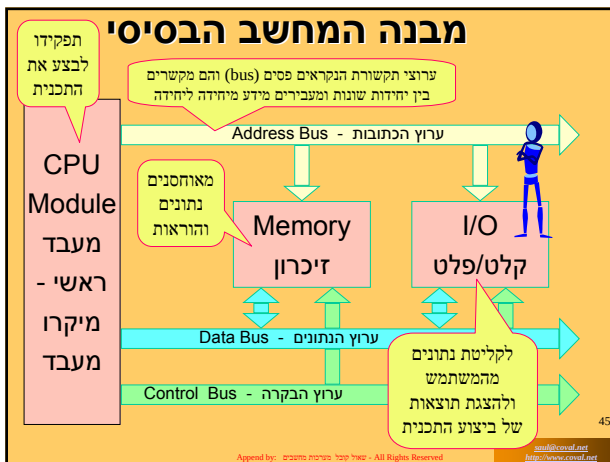
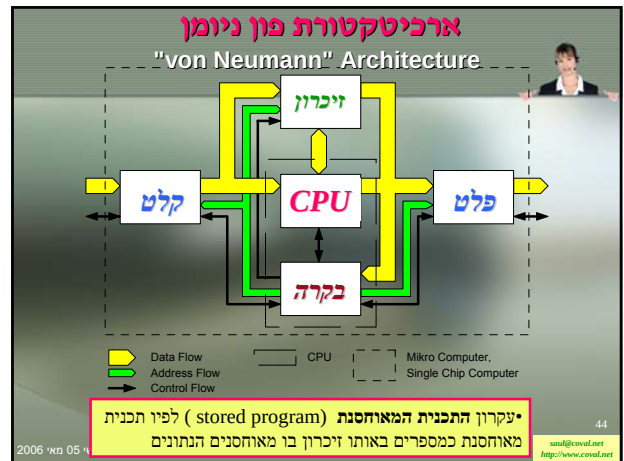
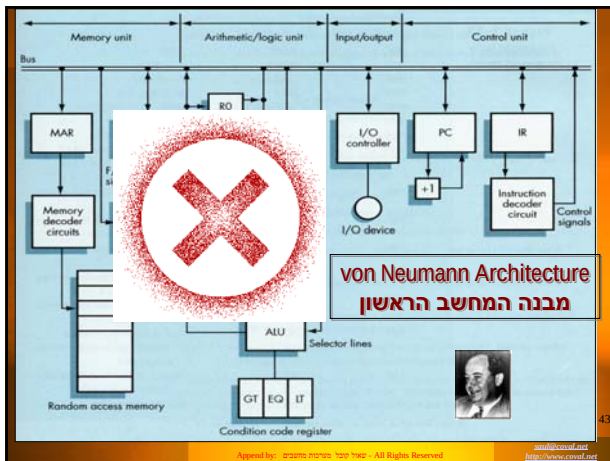
ארכיטקטורת פון ניומן von Neumann architecture

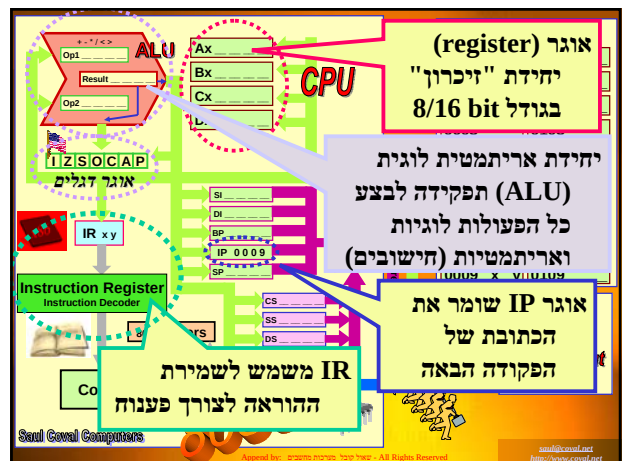
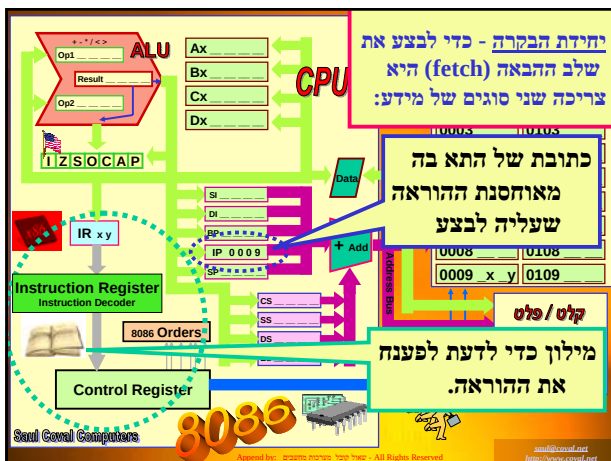
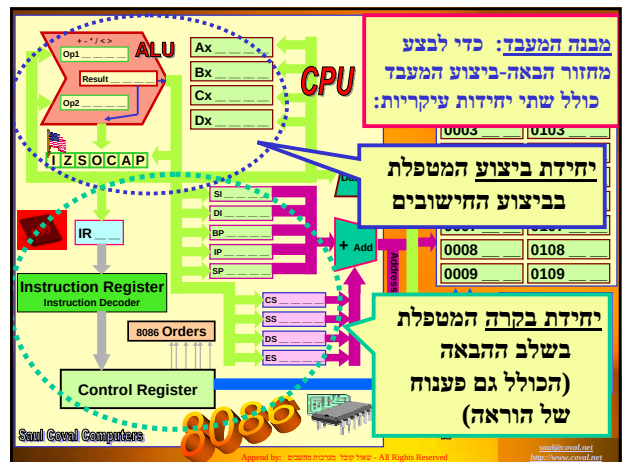
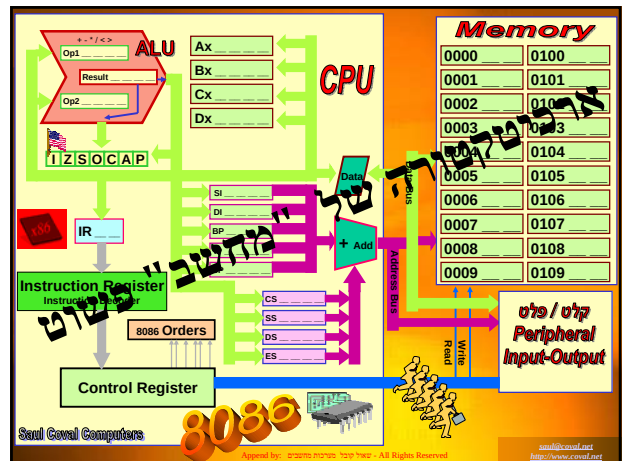
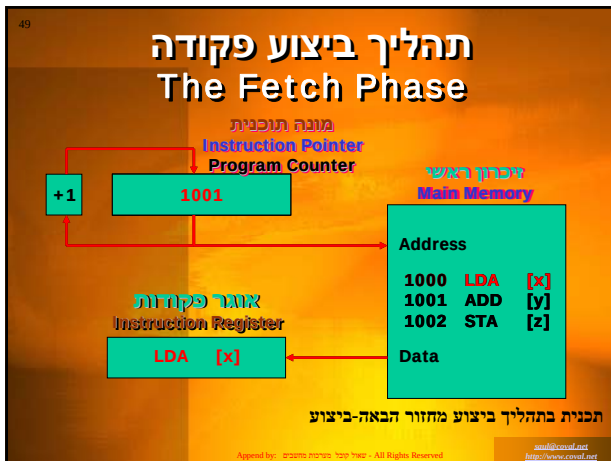
- הארכיטקטורה של מחשב מושפעת מגורמים רבים כמו: טכנולוגיה, שפות תכנות, יישומים, היסטוריה של התפתחות המחשבים וגם ההתפתחות הצפויה בתחום המחשוב.
- רוב המחשבים היום מבוססים על ארכיטקטורה פון ניומן שפותחה לפני כשישים שנה.
- פון ניומן וצוותו של מדענים פתחו בשנת 1951 מחשב שנקרא EDVAC המבוסס על רעיון מהפכני בארכיטקטורה שלו.

יום ששי 05 מאי 2006

שרה פולק + שאול קובל מטרות מחשבים

42
saul@coval.net
http://www.coval.net

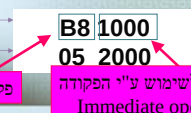




AX (16 bit)	AH (8 bit)	AL (8 bit)	Accumulator	צובר - אקומולטור
BX (16 bit)	BH (8 bit)	BL (8 bit)	Base	מצביע בסיס לאזור הנתונים וגם אוגר כללי
CX (16 bit)	CH (8 bit)	CL (8 bit)	Count	אוגר מונה וגם אוגר כללי
DX (16 bit)	DH (8 bit)	DL (8 bit)	Data	מצביע I/O גם קוד לנפיל / חילוק , אכלי
			SP (16 bit)	Stack Pointer
			BP (16 bit)	Base Pointer
			SI (16 bit)	Source Index
			DI (16 bit)	Destination Index
			IP (16 bit)	Instruction Pointer
			FR (16 bit)	Flag Register
			CS (16 bit)	Code Segment
			DS (16 bit)	Data Segment
			SS (16 bit)	Stack Segment
			ES (16 bit)	Extra Segment
			IR (16 bit)	Instruction Register
			CR (16 bit)	Control Register

High-Level Language: $C = 10 + 20;$

Assembly Language: `MOV AX, 10`
`ADD AX, 20`

Machine language: 

op code - פקודה לביצוע
Immediate operand - אופרנד לשימוש ע"י הפקודה

הגדרת מבנה הוראות המחשב

על פי ISA (Instruction Set Architecture)

יום שישי 05 מאי 2006



1. מהדר - Compiler
2. אסמבלר - Assembler
3. קובץ אובייקט - Object File
4. המקשר - Linker

תהליכים בכתיבת תוכנית ברמות שונות

יום שישי 05 מאי 2006

המהדר - The Compiler

- המהדר הופך את התוכנית משפת על (C++, VB) לתוכנית בשפת אסמבלר, צורת סמלים של שפת מכונה.
- תוכניות בשפות על נכתבות עם פחות שורות תוכנית מאסמבלר, לכן פרודוקטיביות של המתכנת הוא גבוה.
- הגידול ביכולת של הזיכרון הקטין את בעיית האופטימיזציה ומאפשר ייצור של צופן מכונה טוב כמו זה שכותב מתכנת אסמבלר.

יום שישי 05 מאי 2006

האסמבלר - The Assembler

- שפת אסמבלר היא הקשר בין רמה גבוהה - high level Programming Languages (PLs) לבין שפת המכונה.
- האסמבלר יכול להוסיף פקודות שלא מבוצעים בצידוד החומרה. אלה נקראים פקודות פסיאודו - pseudo-instructions. השימוש בהם מהקל בתרגום ובתכנות.
- פקודות פסיאודו: `MOV Dat1, Ax` מתורגם בפקודת מכונה אמיתית: קרא ערך אוגר Ax

יום שישי 05 מאי 2006

קובץ אובייקט - The Object File

- אסמבלר הופך את צופן אסמבלי לקובץ אובייקט, שמכיל צופן של מכונה, נתונים, ומידע הדרושה להכנסת הפקודות בזיכרון.
- אסמבלר חייב למפות את התוויות באסמבלי לכתובות בשפת מכונה. המידע הזה מוחזק בטבלת הסימולים - symbol table.
- לאחר הפיכת כל התוויות לכתובות, טבלת הסימולים מכילה את התוויות הלא מוגדרות, כמו נתונים חיצוניים או פרודקטורות.

יום שישי 05 מאי 2006

מבנה קובץ אובייקט - Object File Structure

- קובץ האובייקט מכיל שישה חלקים:
- Object file header** - גודל ומיקום של יתר חלקי הקובץ.
- Text segment** - צופן המכונה.
- Data segment** - נתונים סטטיים הנכללים בתוכנית.
- Relocation information** - מייצג פקודות ונתונים התלויים בכתובות מוחלטות כאשר התוכנית ניטען לזיכרון.
- Symbol table** - תוויות למצביעים חיצוניים.



יום שישי 05 מאי 2006 saul@coval.net <http://www.coval.net>

המקשר - The Linker

- A single change to one line of the program requires compiling and assembling the whole program. This is wasteful as most code won't be touched by the programmer, even code such as standard libraries which he/she didn't write, will be recompiled.
- An alternative is to compile and assemble each procedure independently. A change to a procedure will require compiling only a single procedure.
- The **link editor** or **linker** takes all the independent object files and links them together.
- The output of the linker is the **executable file** or **executable**.



יום שישי 05 מאי 2006 saul@coval.net <http://www.coval.net>

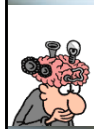
Linking Steps

- There are 3 steps for the linker:
 - Place code and data symbolically in memory.
 - Determine the addresses of data and instruction labels.
 - Patch both the internal and external references.
- The linker uses the relocation information and symbol table in each object *module* to find all undefined labels. These labels are found in branch and jump instructions and in data addresses. It finds the old addresses and replaces them with new addresses. It is faster to "patch" the code than recompile.

יום שישי 05 מאי 2006 saul@coval.net <http://www.coval.net>

Memory Locations

- If all the external references are resolved the linker determines the memory location of all procedures and data.
- Since the files were assembled separately, the assembler can't know where a module's code and data will reside in memory relative to other modules.
- When the linker places a module in memory all **absolute** references, memory addresses not relative to a register, must be relocated to their true addresses.



יום שישי 05 מאי 2006 saul@coval.net <http://www.coval.net>

מבנה כתיבת פקודות בשפת אסמבלי Assembly language instruction format

כל שורת פקודות של שפת אסמבלי משתמשת במבנה הכללי:

label operation operand(s) ; comments
תווית פקודה אופרנד/ים הערות

דוגמה:
Hibur : ADD Ax , Bx ; חיבור אוגרים



- תווית - label** מיועדת לשם למיקום הזה עבור פקודות קפיצה או לקרוא משתנים לפי שם.
- אופרטור - פקודה** **Operation** מגדיר את הוראה לביצוע.
- אופרנד/ים/משתנים** **Operand(s)** מקור ויעד לנתונים.
- הערות - Comments (Remarks)** רק בשביל לעשות הערות ותיעוד אחרי נקודה ופסיק, מוזנח על ידי אסמבלר.

יום שישי 05 מאי 2006 saul@coval.net <http://www.coval.net>

היסטוריה (1) - History

- 1937: first Digital Computer (מחשב דיגיטלי ראשון)
- 1951: first Commercial Computer (מחשב מסחרי ראשון)
- 1959: first IC's (רכיב משולב ראשון)
- 1971: 4004 4Bit Intel (~2000 Tr.)
- 1972: 8008 8Bit Intel (~5000 Tr.)
- 1974: 8080/85-8Bit Intel (~6000 Tr.)
- 1974: 6800 8Bit Motorola
- 1975: first Micro Computer (מיקרו מחשב ראשון)
- 1975: Z80 8Bit Zilog
- 1976: 9900 16Bit Texas Instruments
- 1978: 8080 8Bit Intel
- 1978: 8086 16Bit Intel (50.000 Tr.)
- ...



יום שישי 05 מאי 2006

היסטוריה (2) - History

1980:	µP 8087	16Bit Intel Coprocessor
1981:	First IBM PC	(8086 CPU)
1982:	µP 8286	24Bit Intel (100.000 Tr.) + Instr.
1984:	First Macintosh PC	(Motorola CPU)
1985:	µP 80386	32Bit Intel (300.000 Tr.) + Add.Modes
1989:	µP 80486	32Bit Intel (10 ⁶ Tr.)
1993:	µP 80586	32Bit Intel (5·10 ⁶ Tr.) Pentium I
1996:	First Hand Computer Palm	(מחשב כף יד ראשון)
1997:	µP 80686 MMX	32Bit Intel Pentium II
1999:	Celeron	32Bit Intel Pentium III
2000:		32Bit Intel Pentium IV
2001:	Itanium	32Bit Intel

יום שישי 05 מאי 2006

67

saul@coval.net
http://www.coval.net

סיכומים וחזרות לחומר הלימוד

תלמידים

צוות הוראה

הכל ברור ?

Any Questions ?

יום שישי 05 מאי 2006

68

saul@coval.net
http://www.coval.net

Saul Coval Computers

http://www.coval.net

עריכה, בנייה בסיסית והרצאות
שאול קובל מערכות מחשב

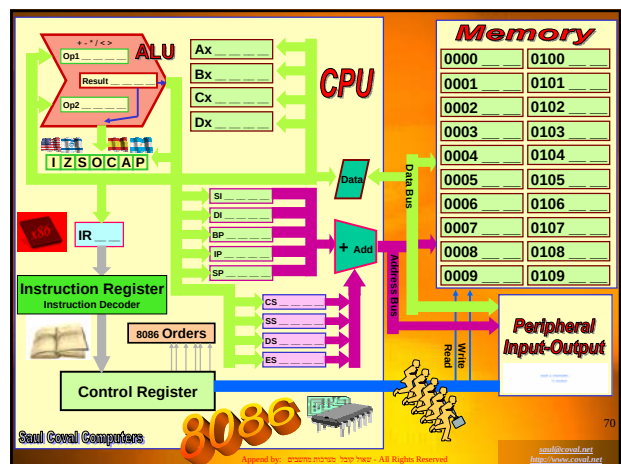
כל הזכויות שמורות

END

יום שישי 05 מאי 2006

69

saul@coval.net
http://www.coval.net



זה הכול להיום

That's all For Today!!

יום שישי 05 מאי 2006

71

saul@coval.net
http://www.coval.net