ביה"ס למקצועות המחשב

קורס מנהל **DC**

חוברת כניסה לקורס

## תוכן עניינים

[תוכן עניינים 2](#_Toc492495999)

[חומרה 4](#_Toc492496000)

[מבוא לחומרה 5](#_Toc492496001)

[הצורך במחשב 5](#_Toc492496002)

[עקרון פעולת המחשב 6](#_Toc492496003)

[היסטורית המחשוב בצבא 7](#_Toc492496004)

[סוגי מחשבים 8](#_Toc492496005)

[יחידות מידה 9](#_Toc492496006)

[רכיבים פנימיים 10](#_Toc492496007)

[תפקידי המעבד 10](#_Toc492496008)

[הזיכרון 11](#_Toc492496009)

[סוגי זיכרון 12](#_Toc492496010)

[תצורת עבודת המעבד מול הזיכרון 13](#_Toc492496011)

[לוח האם 14](#_Toc492496012)

[ספק כוח(Power Supply) 15](#_Toc492496013)

[מושגים בעבודה עם רכיבים פנימיים 19](#_Toc492496014)

[אמצעי אחסון חיצוניים 21](#_Toc492496016)

[Disk On Key 23](#_Toc492496017)

[הדיסק הקשיח 23](#_Toc492496018)

[SSD 26](#_Toc492496019)

[כונן 27](#_Toc492496020)

[צורות אחסון בצבא 26](#_Toc492496021)

[אמצעי קלט פלט 29](#_Toc492496023)

[תקשורת בין רכיבים 31](#_Toc492496024)

[חומרת שרת 33](#_Toc492496024)

[סוגי שרתים 34](#_Toc492496024)

[חוות שרתים 37](#_Toc492496024)

[CMD 38](#_Toc492496024)

## חומרה

בחלק זה, תפגשו את כל עולם התוכן של מבנה המחשב והשרת ואופן עבודתם הפיזית.

ממבנה המחשב והשרת ומה הם מכילים, עד תצורת העברת המידע וסוגי חיבורים.

שימו לב, בחוברת זו אתם עלולים לפגוש מילים רבות שלא תכירו לעומק. כדי שתוכלו להעמיק בחומר, גוגל תמיד נמצא לרשותכם בזמן קריאת החוברת. אל דאגה, בקורס עצמו תראו כי הרבה מהמונחים מתבהרים ויש העמקה בנושאים רבים.

המטרה של החוברת היא להציג מידע בסיסי ביותר שישמש אתכם בהמשך הקורס. אל דאגה, התכנים שתעברו יהיו מגוונים, אך כדי להתחיל ללמוד מעבר, צריך להכיר את הבסיס.

# מבוא לחומרה

# הצורך במחשב

**מדוע נוצרו מחשבים?**

מכשירים עתיקים לעריכת חישובים היו קיימים עוד מהמאות לפני הספירה והלכו והתפתחו עם השנים. ישנם סוגים שונים של מחשבים, עליהם נרחיב בהמשך, אך חשוב להבין שהצורך במחשב נובע מהצורך שלנו במכונה שתאפשר לנו לבצע המון חישובים במעט זמן.

מחשבים הם חלק בלתי נפרד מחיי היום יום שלנו, והשימוש בהם נעשה כמעט בכל תחום ובכל מקצוע. כמעט כל מכונה או מכשיר חשמלי הנמצאים בשימוש כיום, מכילים מחשב המשובץ בהם ואחראי על תפקודיהם השונים. בהמשך תבינו כיצד במיקרוגל שלנו, משובץ מחשב.

**הצמיחה בכמות המידע**

בשנת 1965 נכתב חוק מור אשר מסביר באופן תיאורטי את הצורך ההולך וגדל במחשבים, וכמו כן את צמיחת יכולות המחשבים במקביל לצורך בהם.

חוק מור - נכתב ב- 1965 ע"י גורדון מור. החוק אומר כי מידי שנה וחצי-שנתיים, מספר הטרנזיסטורים[[1]](#footnote-2) בשבב מוכפל, ללא עלות במחיר.

מספר הטרנזיסטורים בשבב משפיע על כוח העיבוד, שהוא החשוב והמרכזי במרכיבי המחשב. ככל שכוח העיבוד גדל, ניתן לבצע פעולות הרבה יותר מסובכות, בזמן קצר בהרבה.

משמעות חוק זה היא שיכולות השבב מוכפלות אך המחיר נשאר זהה. ממשפט זה נוכל להסיק כמה התפתחות הטכנולוגיה מהירה.

יש הסבורים כי החוק תקף לזמן מוגבל (לפי הערכת אינטל – עד סוף העשור הנוכחי) אך צוותי הפיתוח ממשיכים לשבור את גבולותיו של חוק מור באמצעות ננוטכנולוגיה ואמצעים חדישים אחרים.

**מהו מחשב**

מחשב: מכונה לעיבוד נתונים.

נתון Data)): ייצוג פורמאלי של מידע, המתאים לתקשורת, פירוש או עיבוד ע"י מחשב.

מידע (Information): נתון בעל משמעות מיוחדת בהקשר מסוים.

# עקרון פעולת המחשב

**קלט עיבוד פלט**

המחשב מבצע עבודה קבועה בעבודתו של קלט-עיבוד-פלט.

קלט (Input): התהליך בו נתונים מגיעים למחשב לשם עיבודם. נתוני הקלט הם נתונים המיועדים לעיבוד במחשב.

עיבוד (Processing): פעולת המעבד על נתוני הקלט על-פי פקודות התוכנית.

פלט (Output): התהליך בו המחשב מעביר נתונים החוצה לאחר עיבודם, הנתונים מועברים או למשתמש או לתהליך[[2]](#footnote-3) חדש הצורך קלט.

בתהליך זה המחשב מקבל מידע כלשהו, מהמשתמש או מתוכנית מסוימת, מבצע עיבוד על אותו מידע שהתקבל ובסופו של דבר מוציא את הפלט הסופי )למסך לדוגמא).

עיבוד

פלט

קלט

# היסטורית המחשוב בצבא

**המחשב הראשון בצבא**

את ההחלטה על רכישת המחשב הראשון והקמת יחידת ממר"ם (מרכז מחשבים ורישום ממוכן) קיבל המטה הכללי ב-1958. תוך זמן שיא של 3 שנים הגיע לארץ המחשב הראשון של ממר"ם והיחידה הוצבה בבסיס השלישות ברמת גן.

המחשב הגיע לצבא בתחילת שנות ה-60, והיה מדגם "פילקו 211" ובעל זיכרון בגודל K16. הקניה לא התקבלה יפה בעיני הציבור מכיוון שהייתה קניה יקרה מאד ("*מוח יש, שכל אין*" – הייתה הכותרת בעיתונים לאחר קניית המחשב), אך הצבא הבין את חשיבותו של המחשב לתפקוד טוב יותר.

כאשר ניגשו אל המלאכה של הקמת מתקן המחשב ולאחר בחירת מיקומו, יצאו נציגי צה"ל לארה"ב ואירופה כדי ללמוד כיצד נכון לבנות את המתקן. הוחלט על בניית המחשב מתחת לאדמה כדי למנוע מאבק לחדור למערכות הרגישות וכדי לשמור על לחות קבועה.

# סוגי מחשבים

**מחשב אישי**

מחשב אישי (Personal Computer): מחשב המריץ עליו מערכת הפעלה אישית ויכול לתמוך במשתמש אחד בזמן נתון.

המצאת ה-PC בתחילת שנות ה-80 חוללה מהפכה אמיתית בעולם המחשבים. בגלל מחירו הנמוך יחסית ובשל מערכות ההפעלה הידידותיות למשתמש שהותקנו בו, המחשב האישי חדר לרבים מהבתים ולמרבית בתי העסק והפך לחלק בלתי נפרד מחיי היום יום.

**מחשב מרכזי**

מחשב מרכזי (Mainframe): מחשב המריץ עליו מערכת הפעלה מרובת משתמשים ויכול לתמוך באלפי משתמשים.

המחשבים הראשונים שהומצאו בשנות ה-50 היו מחשבים מרכזיים. בהתחלה היו אלו מחשבים גדולים ומסורבלים ובעלי יכולות חישוב נמוכות. עם הזמן הם הלכו והשתכללו, והיום הם מסוגלים לתמוך באלפי משתמשים ובעלי יכולות חישוב וזיכרון עצומות. בשל מחירם היקר היו מחשבים אלו בתחילה נחלתם הבלעדית של צבאות. אח"כ קנו אותן גם אוניברסיטאות וגופים עסקיים גדולים כגון בנקים.

# יחידות מידה

**ייצוג מידע המחשב**

השפה הבינארית: השפה בה "מדבר" המחשב. מורכבת מהספרות 0 ו-1.

0 ו-1 נקראים גם סיבית, או Bit. המספרים האלו מייצגים זרמים חשמליים בשני מצבים של מעגל חשמלי – הספרה 1 מציינת שיש זרם חזק, והספרה 0 שיש זרם חשמלי נמוך. רצף של הספרות 0 ו-1, כלומר רצף של ביטים, יוצר מידע שהוא בעל משמעות עבור המחשב – בדיוק כמו מילים המורכבות מאותיות.

**יחידות מידה**

המחשב מיצג את המידע שלו ב - Bit-ים (סיביות). את כמות ה- Bit-ים אנחנו יכולים לייצג לפי יחידות מידה שונות, למטרת נוחות וכדי שלא נצטרך להגיע למספרי ענק כאשר נרצה לתאר כמות גדולה מאוד של Bit-ים.

**8 Bit = Byte**

**KB = 1024Byte1**

**MB = 1024KB1**

**GB = 1024MB1**

**TB = 1024GB1**

על מנת להמיר בין יחידות המידה האלו, נשתמש בפעולות הכפל והחילוק.

אם נרצה לעבור מיחידה קטנה יותר ליחידה גדולה יותר – נשתמש בחילוק.

אם נרצה לעבור מיחידה גדולה יותר ליחידה קטנה יותר – נשתמש בכפל.

הפעולות יבוצעו בכפולות של 1024.

למשל, אם נרצה לדעת כמה הם 20MB ביחידות מידה של KB, נכפיל: 20x1024=20,480KB

\* **מאחר וחישוב כזה הוא לא פשוט, במידה ונצטרך לחשב המרה בין יחידות מידה בעל פה נשתמש במספר 1000 במקום ב-1024.**

**כלומר במקום לכפול ב-1024, נכפול ב-1000, ונגיע לתוצאה 20,000. מכאן נוכל להסיק ש-20MB, ביחידות של KB, שווים בקירוב 20,000KB. הסימון המתמטי לאומדן הוא ~, ולכן נוכל לכתוב ש: 20x1024 ~ 20,000.**

# רכיבים פנימיים

# תפקידי המעבד

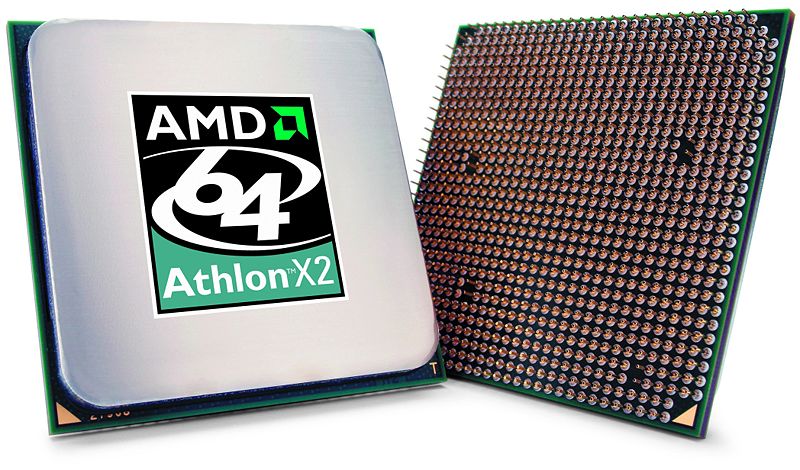
**מהו מעבד?**

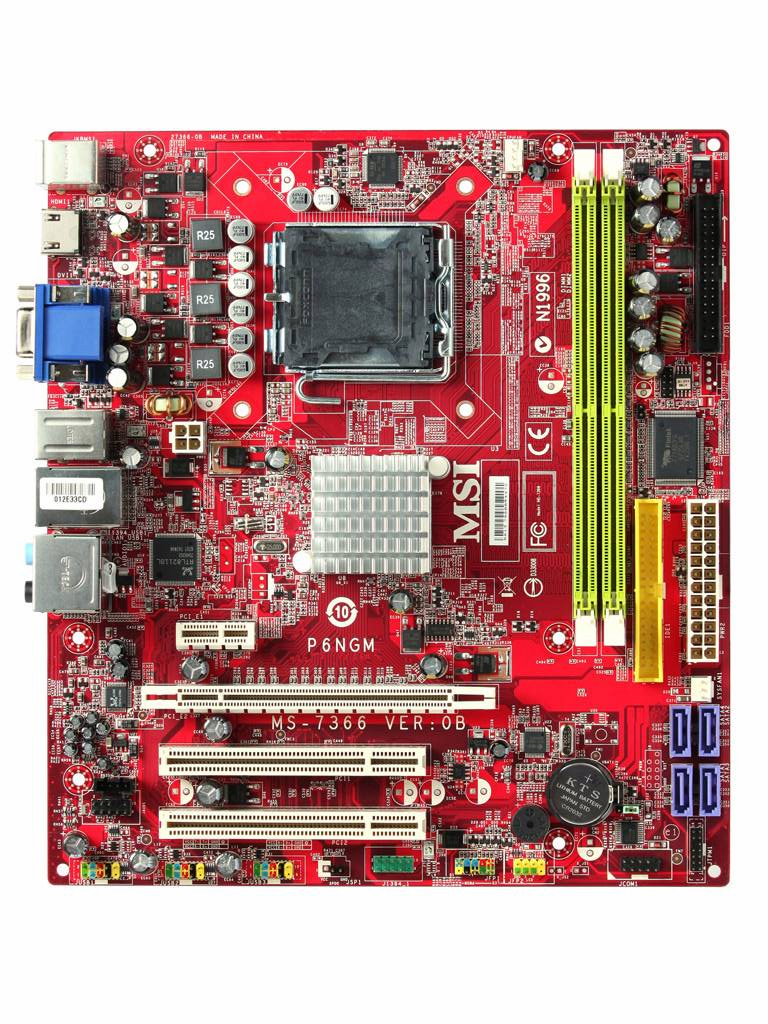
CPU ((Central Processing Unit: רכיב החומרה המבצע את פעולות עיבוד הנתונים במחשב.

יחידת העיבוד המרכזית, המוח של המחשב אשר מבצע את פעולות החישוב. המעבד בעבודתו מבצע חישובים לטובת התוכניות שרצות במערכת ומעביר מידע ממקום למקום בזיכרון.

ברגע נתון ישנה הוראה אחת בלבד בזיכרון, כלומר המעבד מסוגל לבצע בכל פעם הוראה אחת בלבד.

מהירות המעבד נמדדת בג'יגה-הרץ (:GHz – Giga Hertz גיגה = מליארד, הרץ = יחידת מידה בפיזיקה של "מחזורים בשנייה"). כאשר 3GHz אומר כי המעבד יכול לבצע עד 3 מיליארד פעולות בשנייה.





**חלקי המעבד**

ALU (Arithmetic Logic Unit): יחידה אריתמטית-לוגית, האחראית לביצוע חישובים אריתמטיים-לוגיים במעבד (חיבור חיסור כפל חילוק).

אוגרים: חלק בזיכרון השייך אך ורק למעבד, כל החישובים של המעבד נעשים בעזרת יחידות זיכרון אלו והוא משמש את המעבד לפעילותו השוטפת.

העבודה מול האוגרים מהירה מהעבודה מול זיכרון המחשב בזכות הקרבה הפיזית שלהם אל המעבד.

פעולות אריתמטיות (חשבוניות): פעולות חיבור, חיסור וחילוק שהמעבד מבצע על נתונים. פעולות לוגיות הן היכולת של המעבד להשוות נתונים, להעריך תנאים מסוימים אם הם שקריים או נכונים ולקפוץ מפעולה לפעולה. היחידה משתמשת באוגרים לצורך ביצוע חישובים אלו.

במעבד יש מספר אוגרים מהירים מאוד שמשמשים לשתי פונקציות עיקריות:

User-Visible Registers: אוגרים המשומשים על ידי תוכנות משתמש או תוכנות מ"ה. אלו האוגרים שמותר לתכניתן לפנות אליהם בעת כתיבת הקוד.

כלומר, כאשר התכניתן יבנה תוכנית אשר תיגש למעבד ותרצה להכניס\לשנות בו נתונים היא תעבוד עם יחידה מסוג זה.

Control and status Registers: אוגרים המשומשים על ידי המעבד לשליטה על פעולת המעבד. בשימוש על ידי מערכת ההפעלה לשליטה על הרצה של תוכנות.

שעון מעבד (CPU clock): מכתיב את קצב פעולת המעבד בסיוע שעון המערכת שמספק זרם חשמלי למחשב. השעון קובע את קצב העבודה, בכל פעימה עוברים אותות חשמליים וכך ככל שתדירות הפעימות גבוהה יותר העבודה תהיה מהירה יותר.

# הזיכרון

**מהו זיכרון**

אמצעי אחסון: רכיב חומרה שנוכל לאחסן עליו מידע.

זיכרון: אמצעי אחסון אשר משמש את מערכת ההפעלה לשמירת מידע ונתונים בזמן עבודת המערכת.

הזיכרון הוא רכיב הכרחי בעבודת המערכת וזאת מפני שהמעבד עובד מולו בעבודה שוטפת.

\* כאשר נדבר על אמצעי אחסון נדבר על כל רכיב חומרה המשמש לשמירת מידע. שכן, יש לשים לב להבדל בין הזיכרון לבין הדיסק הקשיח.

**זיכרון פנימי\חיצוני**

נוכל לאפיין שני סוגי אמצעי אחסון - פנימי וחיצוני.

אמצעי אחסון פנימי: רכיב זיכרון היושב פיזית על גבי לוח האם. לדוגמא, זיכרון RAM.

אמצעי אחסון חיצוני: רכיב זיכרון שאינו יושב פיזית על לוח האם אלא מחובר אליו. משמעות אמצעי אחסון כזה הוא שהוא מחובר מבחוץ ומעביר את המידע בעזרת חיבור שונה מאמצעי אחסון פנימי. לדוגמא, דיסק קשיח.

# סוגי זיכרון

**RAM**

RAM (Random Access Memory): סוג זיכרון נדיף שמתאפיין במהירות ואפשרות לקריאה ולכתיבה.

משמעות היותו נדיף הוא שישנה תלות במתח חשמלי, כלומר שהמידע נכתב בצורה חשמלית. מצד אחד זה מה שמאפשר את המהירות שלו (הזמן מקבלת המידע, ועד החזרת תגובה, יכול להגיע לפחות מ- 10-9 שניות), אך גם את העובדה שהמידע בו נמחק כאשר הוא מנותק מזרם חשמלי.

כרטיס RAM: כרטיס זיכרון מסוג RAM הנמצא בחריצים על גבי לו ח האם, עליו נשמרים כלל התהליכים הרצים על המחשב ברגע נתון.

כל תוכנית או יישום שנפתח (בין אם זה דפדפן אינטרנט, משחק מחשב או אפליקציה כלשהי) תרוץ על ה-RAM. משמע, ככל שזיכרון ה-RAM גדול יותר, נוכל לפתוח ולהריץ יותר תוכניות כבדות במקביל.

**ROM**

ROM (Read Only Memory): סוג זיכרון לא נדיף שמתאפיין באיטיות יחסית ומאפשר קריאה בלבד.

ה-ROM הינו רכיב זיכרון פנימי איטי מה-RAM (הזמן עד שמציג מידע הוא פי כמה סדרי גודל יותר) אשר מאפשר קריאה בלבד והמידע בו הוא בלתי ניתן לשינוי. בנוסף, בניגוד ל-RAM המידע שעל זיכרון זה נשמר גם בהעדר מתח חשמלי.

**Flash**

Flash: סוג זיכרון לא נדיף המאפשר קריאה וכתיבה.

כיום זיכרון ה-FLASH מאוד פופולארי מכיוון שפותר את הבעיות שנובעות מהשימושים ב-RAM ו-ROM ונותן את היתרונות של שני הסוגים, אך כמובן יקר יותר. כמו כן, השימוש ב Flash הוא איטי יותר מהשימוש בRAM או בROM, אך היתרונות גוברים, ולכן כיום הוא משומש בהרבה רכיבים שונים (עליהם תלמדו בהמשך).

**Cache**

זיכרון Cache: רכיב זיכרון נדיף מסוג RAM המשמש את המעבד לפעולות שוטפות.

כיום, הזיכרון מחולק ל2 חלקים, חלק פנימי במעבד וחלק חיצוני שיושב צמוד למעבד. תפקידו ליעל את פעולת המעבד. כל פעם שהמעבד זקוק למידע הוא מנסה לגשת קודם כל ל-Cache הפנימי, אליו הגישה הכי מהירה, אחריו לCache החיצוני שהוא הבא בתור (הקרבה הפיסית למעבד מאפשרת לגישה להיות מהירה יותר). אם המעבד מוצא את המידע המבוקש פעולת המעבד תתבצע בצורה מהירה יותר, ואם לא, המעבד יחפש את המידע ב-RAM, תוך כדי עדכון ה-Cache במידע שהמעבד כנראה יצטרך שוב בקרוב, ולכן כדאי לו לייבא את המידע קרוב יותר אליו, כך שיהיה קל יותר ומהיר יותר לפנות אליו בפעם הבאה.

חשוב להבין, לזיכרון הcache ולאוגרים אין את אותו התפקיד. האוגרים אחראיים על ניהול תפקוד המעבד ומשומשים לביצוע חישובים, ואילו הCache מיועד לשמור עליו מידע חיוני שעליו יבצעו את החישובים, כדי שהמעבד יוכל לגשת אליו בצורה הכי מהירה.

# תצורת עבודת המעבד מול הזיכרון

ללוח עבודת המעבד הינה צמודה לרכיבי הזיכרון שכן מהם הוא לוקח את הנתונים לעבודתו, תצורת העבודה היא כזו:

**CPU**

**(מעבד)**

**Cache**

**זיכרון מטמון**

**כרטיס RAM**

**אמצעי אחסנה חיצוני**

**1 - פנייה**

**2**

**3**

**4**

**5**

**5 - עדכון**

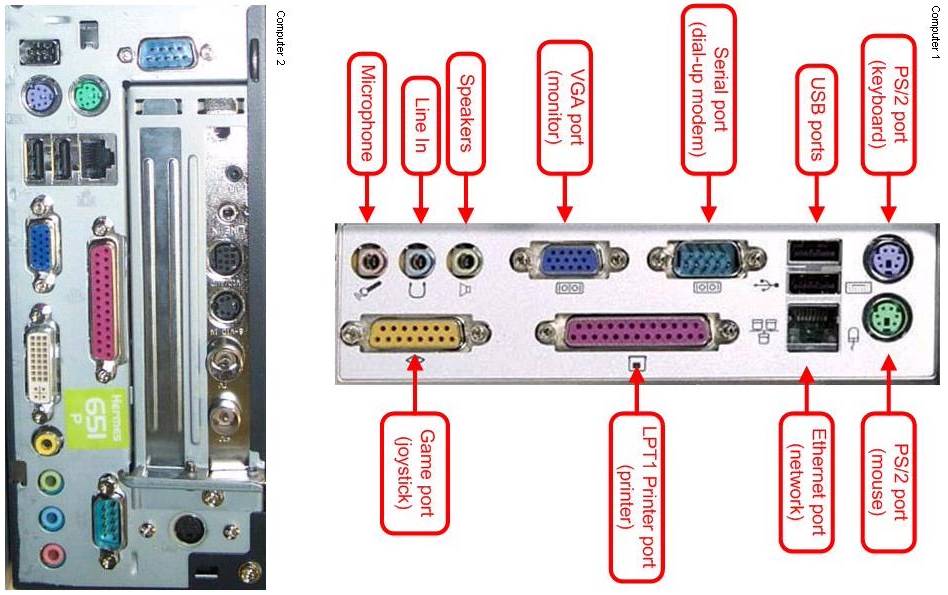
שלבי הגישה של המעבד לזיכרון הם:

1. המעבד מגיע לשלב שבו הוא ממתין למידע כלשהו שייכנס אליו. הוא בודק האם המידע שהוא צריך קיים ב-Cache, הקרוב מאוד אל המעבד מבחינה פיזית, עבודתו מהירה מאוד ובו נשמרים נתונים המשמשים לפעולות שוטפות כמו אלה.
2. אם המידע לא נמצא, הוא בודק בכרטיס ה-RAM, גם הוא מהיר יחסית, אך פיזית ממוקם רחוק יותר מהמעבד. נשמרים בו כל התהליכים שרצים כרגע על גבי המחשב.
3. אם המידע לא נמצא, הוא פונה לאמצעי האחסון החיצוני – לרוב הדיסק הקשיח או כל מדיה אחרת. ביחס לאמצעי האחסון האחרים, אמצעי אחסון חיצוני הוא איטי בהרבה. שם סביר להניח שהמידע יימצא.
4. כאשר נמצא המידע, המידע עובר לכרטיס ה-RAM (מאחר ונפתח תהליך כלשהו).
5. המידע יגיע אל המעבד לפעולות העיבוד הדרושות, והמעבד יעדכן את ה-Cache בהתאם.

# DSCN1162לוח האם

**מהו לוח האם**

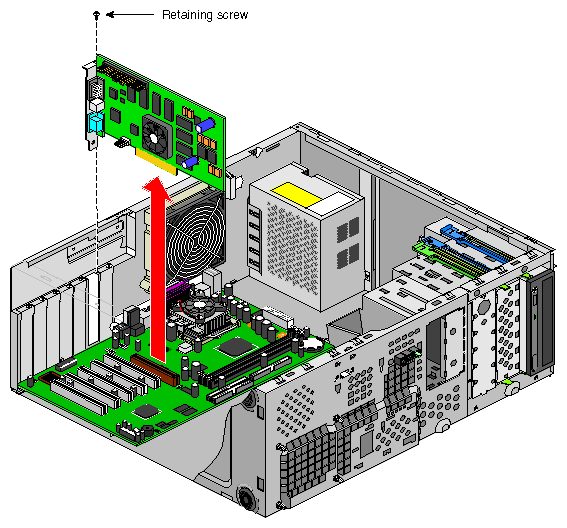
לוח האם (Mother Board): הלוח האלקטרוני הראשי של המחשב.

איכותו של לוח האם משפיעה ישירות על איכותו של המחשב מכיוון שהיא קובעת את טיב וסוג הרכיבים שניתן לחבר אליו.

**תפקידי לוח האם**

1. קישור פיסי בין רכיבי המחשב הפנימי.

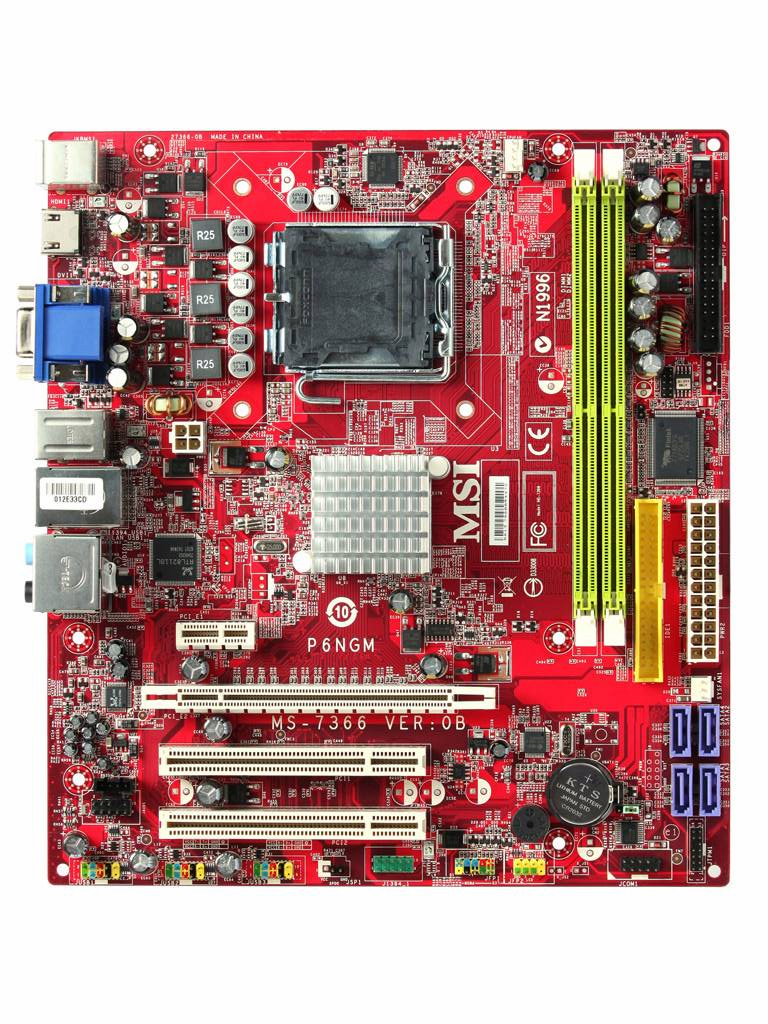
2. קישור חשמלי בין רכיבי המחשב הפנימיים.

****חיבורי Onboard: חיבורים המובנים מראש על גבי לוח האם ומהווים ממשק חיבור בין לוח האם לאמצעי קלט ופלט שונים.

**Onboard Card**

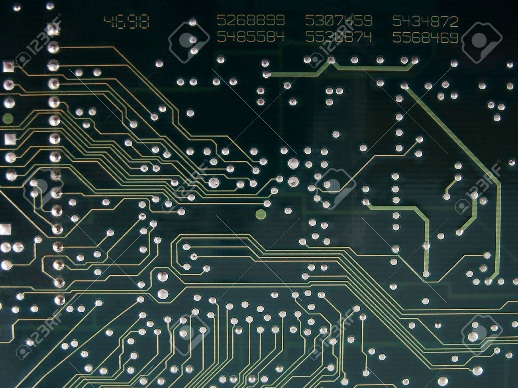
כרטיס הרחבה (Expansion Cards): כרטיס אלקטרוני המורכב בחריץ של לוח האם במחשב.

כרטיס ההחרבה נועד להוות ממשק בין המחשב לבין התקנים חיצוניים או להשלים תכונות ופונקציות שונות של המחשב. נשתמש בו כאשר חיבורOnboard הפסיק לעבוד או כאשר נרצה לשדרג או להוסיף פונקציות למחשב.



**Expansion Card**

**Onboard Connectors**

**מהו אפיק**

אפיק (Bus): ערוץ העברת אותות בין רכיבים במחשב. תפקיד האפיק הוא להעביר נתונים, כתובות ואותות בקרה בין רכיבי המחשב.

האפיק הוא למעשה מעין חוט חשמלי ממוזער המאפשר העברת נתונים מהירה בין כל רכיבי המחשב. ישנם סוגים שונים של אפיקים ולמעשה מדובר בארכיטקטורה שלמה הכוללת את האפיקים עצמם, בקרים ורכיבים אשר מסוגלים לדבר עם האפיק. עם הזמן ניסו להגדיל את קצב העברת המידע ולכן הגדילו את כמות האפיקים להעברת מידע במקביל (גם לפתרון זה יש בעיות, העברת מידע רב בו זמנית יוצר יותר בעיות כמו Crosstalk- ערבוב מידע אחד בשני).

יש להבין שגם במעבד פועלים לנו אפיקים, אשר תפקידם להזרים מידע בין חלק אחד לאחר בתוכו.

# ספק כוח(Power Supply)

ספק כוח (Power Supply): התקן המורכב במחשב הממיר את המתח של רשת החשמל הארצית למתח הנדרש על-ידי המחשב.

המחשב שלנו פועל על חשמל, ועל מנת לקבל אותו הוא מתחבר, באמצעות כבל, לשקע החשמל בקיר.

הבעיה היא שמתח החשמל (וולט) כפי שהוא מועבר מהשקע בקיר, לא תואם את מתח החשמל שהמחשב זקוק לו על מנת לעבוד. בנוסף, לא נוכל לחבר ישירות כל חלק בפנים המארז החוצה, באמצעות שקע או כבל.

לכן, קיים ספק הכוח. ספק הכוח אחראי להמיר את הזרם החשמלי שבשקע לזרם החשמלי הנדרש למחשב. מבחינה חיצונית, אל ספק הכוח מתחבר כבל המתח.

בצדו הפנימי, במארז, יוצאים מספק הכוח מספר חיבורי חשמל – המספקים את המתח החשמלי לרכיבים בפנים המארז, שאינם מתחברים ישירות לחשמל (כמו מסך, למשל).קיימים שני חיבורים מרכזיים:

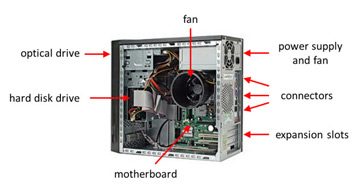
* חיבור ספק הכוח ללוח האם: החיבור הרחב ביותר שיוצא מספק הכוח. מתחבר ללוח האם ישירות. חיבור חיוני מאוד, בלעדיו המחשב לא יוכל להידלק כלל. באמצעות חיבור זה, מגיע חשמל לכלל הרכיבים על הלוח.
* חיבור ספק הכוח לכוננים: ישנם מספר חיבורים זהים כאלו, היוצאים מספק הכוח. הם מתחברים לכוננים השונים – בין אם מדובר בכונן דיסקים, דיסקטים, וכמובן דיסקים קשיחים.

מאוורר: התקן שתפקידו לקרר את רכיבי המארז השונים (ספק כוח, מעבד, וכו').  
החל ממעבדי 486 נאלצו היצרנים לצרף מאוורר לכל מעבד על מנת למנוע התחממות ייתר שתתקע את המחשב. פעולתו של המאוורר הכרחית לפעולת המעבד. בלעדיו המחשב יתקע כל זמן קצר. תקלות במאוורר מאופיינות במחשב שקט במיוחד או מרעיש במיוחד. יש מעבדים אליהם מחובר גוף קירור אחר כמו מוטות אלומיניום.

\*לספק הכוח יש מאוורר נוסף בנפרד מהמעבד.



**תרשים מסכם:**



ארכיטקטורות 32bit ו- 64bit

**ארכיטקטורת 64\32 ביט**

כיום קיימים 2 סוגי מעבדים עיקריים-

מעבד 32 ביט: מעבד אשר גודל האוגר שלו תומך במקסימום 32 סיביות (bits).

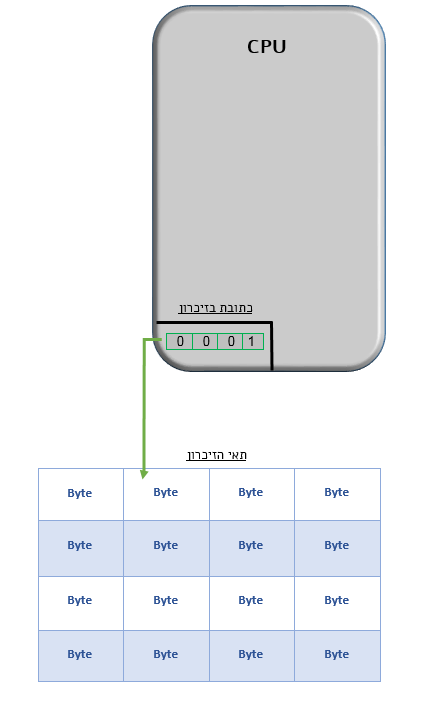
מבחינת המעבד, המשמעות היא שניתן לעבוד עם קטעי מידע שאורכם עד 32 סיביות.

עבודת המעבד מול הזיכרון:

כשהמעבד עובד מול הRAM, הוא עובד מול תאי זיכרון. המעבד יכול לפנות בכל רגע נתון אל תא זיכרון אחד על ידי השמת כתובת התא באוגר, ופנייה לכתובת שבאותו האוגר. הכתובת היא מספר בינארי כלשהו (מספר המורכב מאחדות ואפסים).  
המעבד יכול לגשת רק אל כתובות בזיכרון המורכבות מ-32 ביטים (ולא יותר), כיוון שכמו שאמרנו - הגודל המקסימלי של אוגר במעבד מסוג 32bit הוא 32 ביטים. הכתובת הכי קטנה היא 00000000000000000000000000000000 והכתובת הכי גדולה היא 11111111111111111111111111111111.  
אם ננסה לגשת לכתובת הבאה, נגיע לכתובת 100000000000000000000000000000000 (33 ביטים), ולמעבד אין יכולת לגשת לכתובת זיכרון זו – היא יותר מ-32 ביטים.

בארכיטקטורת מעבד 32Bit, המעבד יכול לגשת אל 232 = 4294967296 כתובות זיכרון. מדוע 232?  
32 מסמל את מספר הסיביות שהאוגר יכול לשמור.  
2 מסמל שתי ספרות אפשרויות – או הספרה 0 או הספרה 1. לכן יש לנו 232 אפשרויות של כתובות.

נקח את הדוגמא הבאה, בה יש לנו מעבד אשר יכול לגשת ל16 כתובות זיכרון - 24 (0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111):

\* לפניך רק 2 דוגמאות לכתובות בזיכרון של המעבד מתוך 16 אפשריות.



המעבד הנ"ל יכול לגשת רק ל16 כתובות זיכרון המפנות ל16 תאי זיכרון שונים ולכן הוא לא יכול לפנות לתא ה-17 בזיכרון.

זה בדיוק המצב של מעבד 32Bit, רק שמעבד 32Bit יכול לפנות ל 232 = 4294967296 כתובות זיכרון.

כלומר, אם יש לי מידע שנמצא בתא מספר 232+1, המעבד שלי לא יוכל לגשת אליו והמידע יהיה אבוד. לכן, בשימוש בסביבות של 32Bit, ניתן להשתמש רק עד GB4 של זיכרון RAM.

החישוב:

ה232 מתייחס למספר כתובות הזיכרון במעבד.

אז 232 = 4294967296 כתובות זיכרון.

כל כתובת מפנה לByte שונה בזיכרון (ראה בשרטוט למעלה).

לכן 4294967296 Bytes.

כעת נמיר את המספר מBytes לGB:

4294967296 \ 1024 = KB 4194304

\ 10244194304 = 4096 MB

4096 \ 1024 = **4 GB**

מעבד 64 ביט: מעבד אשר רוחב האוגר שלו תומך במקסימום 64 סיביות (bits).

משמעות מעבד זה זהה לזו של ה-32 ביט, רק שרוחב האוגר שלו תומך ב64 סיביות. משמעות דבר זה היא שהמעבד יכול להריץ פקודות יותר גדולות ולשמור בזיכרון פעולות יותר גדולות.

מבחינת השימוש בRAM, במקרה זה ניתן להשתמש ב GB 17179869184 של RAM.

# מושגים בעבודה עם רכיבים פנימיים

**צוואר בקבוק**

צוואר בקבוק: מצב בו נוצר עומס נתונים על פס תקשורת.

כל רכיב במחשב עובד בצורה שונה (מבחינת קצב, סדר פעולות וכדומה) דבר היוצר בעיות. תופעת הלוואי של מצב היא שישנם רכיבים איטיים במחשב המאטים את התפוקה הכוללת. רכיבים איטיים במחשב יוצרים "פקק" של מידע והיכולת של המעבד לא מנוצלת במיטבה.

כאמור, הרכיב המהיר ביותר במחשב הינו המעבד. אמצעי האחסון ואמצעי הקלט/פלט איטיים ממנו בכמה סדרי גודל (הביטוי סדר גודל שקול לפי 10). חוזקה של השרשרת נמדד לפי חוזק החוליה החלשה ביותר בה. מסיבה זו המחשב לעולם לא ינצל את המעבד שלו בצורה האופטימאלית.

לבעיה זאת אין פתרון אמיתי. אפשר לצמצם את הבעיה ע"י שיפור הרכיבים (גם לאפיקים יש חלק בתופעה, שהרי הם אלו שמעבירים את המידע, ולכן הם צריכים להיות מסוגלים להעביר הרבה מאוד מידע בזמן מאוד קצר).

**בקרים**

בקר (Controller): הינו רכיב חומרה המאפשר עבודה מסונכרנת בין רכיבי המחשב.

הבקר משמש לגישור על הבדלי המהירויות שבין ה-CPU לרכיבי המחשב השונים. כידוע המעבד הינו הרכיב המהיר ביותר במחשב ועל-כן יש צורך ביחידות בקרה היודעות להעביר נתונים ל-CPU במהירות גבוהה על מנת לנצל את כל יכולתו של המעבד, ומצד שני יודעות להעביר מידע לאמצעי האחסון ואמצעי הקלט והפלט במהירות התואמת את ביצועיהם. לדוג' בעוד ה-CPU מסוגל לשלוח בקשות I/O בקצב של MB ה-HD מסוגל לקצבי עבודה נמוכים בהרבה של KB ועל-כן עבודה ביניהם עלולה לגרות למצב של איבוד מידע קריטי או עיוותו. פעולת הבקר במקרה זה לתווך ולהתאים את קצבי העבודה בין הרכיבים.

פעולה נוספת של הבקר היא להקל על ה-CPU בעבודתו. לדוגמא בקר ה-DMA (Direct Memory Access) אשר מנהל בקשות מידע מה-RAM במקום המעבד. בנוסף הם מגבירים את יעילותה ויציבותה של המערכת. לדוגמא: בקשת מידע מהדיסקט שאינה מתאפשרת מסיבה מסוימת (אולי כי הוא תקול) אינה תוקעת את המעבד אשר יכול להמשיך בפעולתו השוטפת אלא נתקעת בבקר.

# אמצעי אחסון חיצוניים

# אמצעי אחסון חיצוניים

**הצורך ברכיבי אחסון חיצוניים**

אמצעי אחסון חיצוני: אוסף רכיבים המבוססים על טכנולוגיות אחסון שונות שאינם יושבים ישירות על לוח האם.

עבודה עם אמצעי אחסון פנימיים, כגון RAM או ROM תלויה במתח חשמלי. המידע ששמרתי נשמר ב-RAM ועל כן נמחק כאשר ננתק את המתח. על מנת לאפשר שימור מידע גם כאשר נכבה את המחשב או בעת הפסקת חשמל פתאומית, כדי לאפשר לנו להגדיל את מאגר המידע, לשאת אותו איתנו לכל מקום בכיס, לנתק אותו ממחשב אחד ולחבר למחשב אחר ולדאוג שכל המידע נשמר, או אפילו לשמור עותק של המידע במקום מרוחק מהמחשב לצורך גיבוי, נוסיף למחשב אמצעי אחסון חיצוניים.

אמצעי האחסון החיצוניים מאפשרים לנו לאחסן כמויות גדולות של מידע, ללא צורך במתח ונוכל לנייד את המידע שלנו בקלות יחסית.

במהלך השנים אמצעי האחסון החיצוניים עברו מגמה של מזעור הרכיבים בד בבד עם הגדלת יכולת האחסון, כלומר כמות האינפורמציה שהם מסוגלים להכיל.  
על אמצעי אחסון אלו, לרוב נשמור דברים כמו: תמונות, מוסיקה וכדומה. על הHD, נשמור גם קבצים חיוניים למערכת ההפעלה בעלייתה.

**אמצעי אחסון חיצוניים לאורך ההיסטוריה**

כרטיסיית ניקוב: אלו כרטיסיות נייר עליהן נוקבו חורים באופן סידרתי, כל חור שווה לסיבית שהיא יחידת המידע הבסיסית.

במקום בו יש חור הסיבית שווה ל-1, במקום בו אין חור הסיבית שווה ל-0 (בדומה לשיטת ה"אין זרם - יש זרם"). בכדי לקרוא את המידע, המחשב היה מעביר קרני אור דרך כרטיס הנייר תוך גלילתו על גבי עינית סורקת.

במקום בו עברה קרן אור, כלומר קיים חור, המחשב הבין כי הסיבית היא 1. במקום בו לא עברה קרן אור, כלומר אין בו חור הבין המחשב כי הסיבית 0. בכל כרטיסיית ניקוב יכולות להיות עד 32 סיביות + סיבית בקרה.

קלטת מגנטית: בשנות ה-50 הומצאה הקלטת המגנטית המהווה שכלול של הסליל המגנטי. בקלטת נעשה שימוש בטכנולוגיה מגנטית לאחסון מידע (חשמלי), הסליל המגנטי בקלטת מכוסה שרוול קשיח האטום בפני אבק. התפתחות טכנולוגית של חומר הסליל וכונן הקלטות עצמו, הביאו למזעור משמעותי בגודלה של הקלטת ועלייה עצומה ביכולת האחסון שלה.

יתרון נוסף של השרוול השקוף בתוכו נמצאת הקלטת הוא פיחות משמעותי בבלאי שלה אשר מאפשר לאחסן את המידע לזמן ארוך יותר. לקלטת המגנטית יש Write protect בצורת כפתור שמצבו מאפשר או מונע כתיבה.

ב2014, הצליחו סוני וIBM ליצור קלטת מגנטית המסוגלת לאחסן TB185 של מידע.

כיום, ישנם מקומות המשתמשים בקלטות לגיבוי המידע שלהם, וזאת מכיוון שהיא אמצעי בעלת מחיר נמוך, היא מסוגלת לאחסן הרבה מידע, יש אפשרות ההקלטה החוזרת, היא אטומה ולכן יש לה בלאי נמוך.

דיסקט: בשנות ה-70 במקביל להמצאת ה-PC הומצא הדיסקט. בדומה לקלטת המגנטית, טכנולוגית האחסון הינה מגנטית כאשר ראש קורא כותב ממגנט את החומר הפרומגנטי ממנו עשוי הדיסקט.

קיים Write Protect בצורת כפתור שכאשר הוא חסום הדיסקט מוגן בפני כתיבה.

יתרונותיו של הדיסקט הם במחירו המזערי, ניידותו והשימוש בגישה ישירה בפניה אליו (זמן אחזור הנתונים אינו תלוי במיקומם) ואולם הבלאי שלו ומהירות הגישה שלו נמוכה יחסית. כן, עדיין יש שימוש ב2016, למרות שכמות המידע שניתן לאחסן בו הוא מזערי (MB1.44).

\* **קיימות שתי גישות שבעזרתן נוכל לגשת למידע השמור על אמצעי האחסון- הגישה הישירה והגישה הסדרתית. משמעות הגישה הסדרתית היא שבמטרה לגשת למידע השמור, נצטרך לעבור על כל המידע מההתחלה (כמו קלטת וידיאו ישנה) עד שנגיע למקום המבוקש. מנגד, הגישה הישירה, לפיה נוכל לגשת ישירות למידע המבוקש מבלי לעבור על המידע השמור לפניו.**

CD-ROM (Compact Disk Read Only Memory): בשנות ה-90 הומצא ה-CD-ROM שהיווה שיטה חדשה לאחסון מידע.

כונן ה-CD מכיל ראש קורא כותב אופטי המורכב מקרן לייזר הנשברת מפניה הלא חלקים של הדיסק.

ישנם כמה סוגים של CD- CD-ROM לקריאה בלבד, WORM כתיבה (צריבה) חד פעמית וקריאה במס' בלתי מוגבל של פעמים ו-CDRW כתיבה וקריאה במס' בלתי מוגבל של פעמים.

DVD-ROM (Digital Versatile Disk): ב-1995 יצאה לראשונה מדיית ה-DVD אשר התבססה על ארכיטקטורת ה-Compact Disk. בדומה לכונן ה-CD, כונן ה-DVD מכיל ראש קורא/כותב אופטי המורכב מקרן לייזר הנשברת מפניה הלא חלקים של הדיסק.

יתרונו המשמעותי של ה-DVD הוא שהמידע מקודד על המדיה בצורה יעילה יותר, המאפשרת ניצולת טובה יותר של שטח הדיסק.

קיימים כוננים מיוחדים המיועדים לקריאת וצריבת מדיות DVD. כוננים אלו מסוגלים לקרוא CD-ים רגילים ללא כל בעיה.

# Disk On Key

רכיב זיכרון מסוג Flash שהומצא בישראל ע"י חברת M-Systems. כיום רכיב פופולארי מאוד המשומש להעברת מידע. בעקבות חיבור הUSB והתמיכה הגלובלית לחיבור זה, ניתן להשתמש בו להעברת מידע בין מחשבים וגם בין רכיבים אחרים (כמו קונסולות משחקים).

# ללא שםהדיסק הקשיח

**מהו Hard Disk**

דיסק קשיח: רכיב במחשב המשמש כהתקן זיכרון לא נדיף, מאפשר אחסון אמין של נתונים דיגיטליים בנפח גדול ובזמן גישה קצר יחסית.

ה-HD (נקרא לעיתים גם "כונן קשיח") מופיע בשתי תצורות: האחת כרכיב פנימי המוטמע בתוך המחשב, והשנייה כרכיב חיצוני עצמאי המתחבר למחשב באמצעות כבלים מסוגים שונים.

כיום משמש כמדיה המגנטית הנפוצה ביותר לשימוש במחשבים.

כאשר המעבד פונה להתקן אחסון כל שהוא במערכת בעבודתו למטרת לקיחת מידע, הוא יכול לקחת אותם או מהזיכרון שלו (RAM) או מהדיסק הקשיח. חשוב להבין כי העבודה מול הזיכרון היא מהירה ויעילה יותר. אך על הזיכרון לא נשמרים נתונים קבועים, שכן הוא זיכרון נדיף. לכן, אם המעבד יחפש נתונים לעבודתו, הוא יחפש אותם קודם כל הזיכרון ורק אח"כ יפנה ל-HD.

הדיסק הקשיח מהווה חלק עיקרי ברמת הביצועים הכללית של המחשב. קצב הקריאה והכתיבה של הדיסק הם חלק חשוב מאוד מתהליך עיבוד הנתונים, מהירות הסיבוב שלו והקריאה/כתיבה הם חיוניים לתפקוד שוטף.

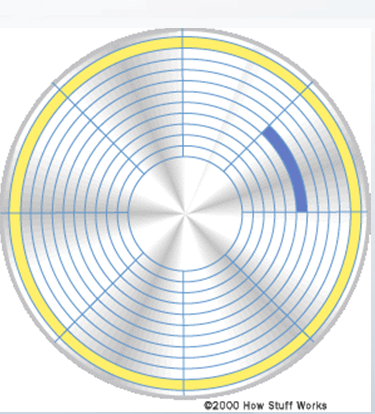
**מבנה ה-Hard Disk**

הדיסק הקשיח מורכב ממספר פלטות אלומיניום, המצופות בחומר מגנטי שבאמצעותו נשמרים הנתונים על הפלטה.

הפלטות מחולקות ל-Trackים (מסלולי ריצה) כאשר כל Track מחולק ל-Sectorים בעלי גודל זהה ((512B. מטרת החלוקה הזו היא לעזור לסדר את המידע בצורה שתהיה נוחה לקרוא אותו ולסדרו ע"ג הפלטה.

המידע ב HD נמצא בשני צידי הפלטה, כלומר יש יותר מקום לאחסון מידע ונשמר בצורה של נקודות מגנטיות. המידע נקרא בעזרת ראש קורא כותב שהוא חלק מה HD, כאשר הראש קורא כותב נוגע ב HD המידע שעליו עלול להיהרס ולכן יש לנהוג עם הHD ביתר זהירות.

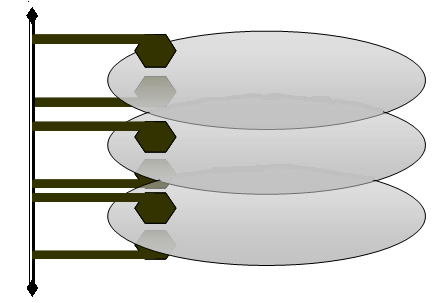
**Sector**



**Sector**

**מסלול**

**מסלול**

**כתיבה וקריאת המידע ב-HD**

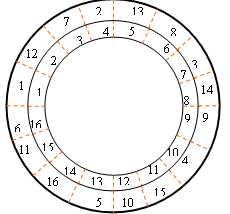
את תהליכי הקריאה והכתיבה מבצעים הראשים הקוראים/כותבים. במצב של אי פעולה כל הראשים נמצאים תמיד באותו מקום בפלטה, מונחים על Track 0, שעל Track זה אף פעם לא נשמר מידע.

במצב של פעולה הפלטות מסתובבות במהירות עצומה, בעקבות הסיבוב המהיר של הפלטה (כ-7,200 סיבובים לדקה) נוצרת כרית אויר אשר מרימה את הראש קורא\כותב למרחק מזערי מן הפלטה, וכך נקרא ונכתב המידע. פעולה זו נעשית משני צידי הפלטה, מה שמאפשר אחסון מידע רב יותר.

מכוון שהראש נמצא במרחק כה זעיר מן הפלטה, יש סכנה שבמקרה של ירידת מתח פתאומי או טלטול יגע הראש בפלטה וכך יגרום לפגיעה במידע השמור, מצב זה של פגיעה במידע נקרא Crash. מנגנון ה- Voice Coil הינו מנגנון האחראי להזזת הראשים הקוראים כותבים. בעת ירידת מתח חשמלי הוא "מקפיץ" את הראשים לעברTrack 0 (שם כאמור לא נשמר מידע). בצורה כזו נמנע מצב של Crash ב-HD.

המידע שנשמר על הדיסק נכתב בצורה מבוקרת. יש מספר תהליכים ופרוצדורות שבאים ליעל את דרך הקריאה/כתיבה, וזאת בעקבות בעיות אשר אותן היה צריך לפתור על מנת לבצע ולייעל את העבודה.

Zoned Bit Recording**:** בעיה הקיימת, לפיה המסלולים החיצוניים בפלטה גדולים יותר מהפנימיים. חלוקת ה-Trackים וה-Sectorים ביניהם שווה, אך הפנימיים הם ברווחים קטנים יותר מהחיצוניים. משמע, בזבוז של רב מקום אחסון במסלולים החיצוניים מכיוון שכמות המידע בסקטור הוא 512B בכל מקרה, אך במסלולים החיצוניים יש פיזית יותר מקום.

הפתרון הוא פשוט. מחלקים את כל המסלולים לאזורים (zones). את המסלולים החיצוניים מחלקים ליותר אזורים, וכך ניתן היה להכניס בהם יותר סקטורים, בהתאם לגודלם.

Interleave**:** בעיה נוספת, לפיה הפלטות של הדיסק הקשיח מסתובבות מהר מאד והראש קורא כותב אינו עומד בקצב הסיבוב בפעולת הקריאה, הוא קורא מסקטור אחד ולפני שהוא מצליח להגיע לשני הפלטה כבר הסתובבה והוא צריך לחכות עד שיגמר הסיבוב.

מה הכוונה? כאשר ראש הקריאה עובר על סקטור 1 הוא מתחיל לקרוא, הבקר ימשיך לעבד את הנתונים ובינתיים בעקבות הסיבוב המהיר ההתחלה של סקטור 2 כבר חלפה מתחת לראש הקורא. משמע, כדי לקרוא את המידע שהפסדנו אנו נאלצים להמתין סיבוב שלם של הדיסק והבעיה תחזור על עצמה.

הפתרון הוא ביצוע Interleave של הסקטורים. המשמעות היא כביכול "לפזר" אותם כדי ליצור מרחק קריאה אשר יקל על מהירות הקריאה.

איך ייעשה? במסלול הפנימי ישנו Interleave זה 1:1 - וסדר הסקטורים הוא 1,2,3,4 וכו'. במסלול החיצוני יש Interleave של 3:1 כשהמרחק בין הסקטורים הסדורים הוא 3- 1,12,7,2,13,8,3.

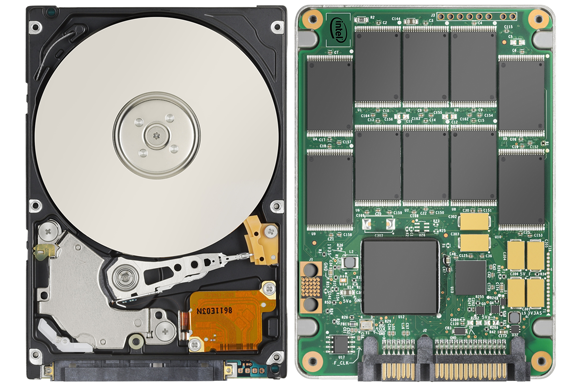
עכשיו בזמן עיבוד נתוני סקטור 1 סקטורים 12 ו- 7 חולפים ו- 2 מוכן לקריאה אחרי שהבקר כבר סיים. במקום שייקח 16 סיבובים של הדיסק כדי לקרוא מסלול שלם, ייקח רק 4 סיבובים. בידיעה שבמסלול של דיסק אמיתי יש כמה מאות סקטורים ולא 17 כמו בדיסקים מאוד ישנים, החיסכון הוא של ענק – שיפור בביצועים בכ- 90%.

ב Interleave השתמשו בדיסקים ישנים. כיום הבקרים מספיק מהירים והבעיה לא קיימת.

# SSD

HD

SSD

Solid-State Drive - זיכרון מסוג Flash, יעיל יותר מהHD מכיוון שאינו מכני, ולכן, זמן הגישה למידע מהיר יותר, פרגמנטציה היא כבר לא גורם בעייתי, אין ראש קורא/כותב העלול לפגוע בזיכרון, אין פליטת חום כמו בHD ואין רעש מכיוון שאינו מכני.

עם זאת, קיימים גם שתי חסרונות- מחיר הזיכרון גבוה, נכון ל2016, תוכלו למצוא TB1 במחיר של כ-300$ (לעומת HD שעולה כ50$ לTB1). הכתיבה לתוך תא הזיכרון היא סופית (כלומר, אחרי הרבה כתיבות ומחיקות לתוך תא זיכרון, המידע יישאר באופן קבוע, ולא ניתן יהיה לשנות אותו).

# כונן

כונן: התקן קלט/ פלט אשר קורא וכותב את המידע שעל אמצעי אחסון החיצוניים.

בתוך הכונן נמצא הראש קורא כותב, ובתפקידו שייך לאמצעי אחסון חיצוניים כמו CD-ROM או דיסקט אשר הוא קורא מהם את המידע (הכונן קורא את המידע השמור על אמצעי האחסון, לא שומר עליו מידע).

הכונן אינו קשור ל-HD, לו יש ראש קורא כותב משלו.

הקשחה פיזית: מדיניות ההקשחה הפיזית בצבא אומרת כי אין כוננים יותר במחשבים. חוץ מעמדות שער ספציפיות (במתקני המחשב).

כיום בצה"ל לא משתמשים בכוננים.  
הסיבה לכך הוא נושא אבטחת המידע, שכן הצבא לא מאפשר הכנסת מדיה חיצונית לרשת מחשש לפגיעות סייבר.

\*תזכורת:  
על מנת שהתקן כלשהו יפעל, הוא חייב להיות מחובר בשתי דרכים:

לספק הכוח-שיספק לו חשמל.

ללוח האם-מה שיאפשר העברת נתונים בינו לבין המעבד ושאר הרכיבים במחשב.

לפעמים החיבור ללוח האם יספיק, מכיוון שלוח האם יודע גם לספק כוח.

# צורות אחסון בצבא

**קונסולידציה**

קונסולידציה: ריכוז של משאבי המחשוב במקום אחד (כינוס).

בעבר, מערכות המחשוב היו לא אמינות ולכן העדיפו לפזר את המידע על מספר מקורות שונים של אחסון (שרתים, קלטות וכדומה) במקומות שונים.

כיום, מערכות המחשוב אמינות הרבה יותר והמגמה היא לעבור ליחידה מרכזית אחת בה יישמר כל המידע.

יתרונות מדיניות זו- חסכון במשאבים, תחזוקה פשוטה יותר, פחות כוח אדם נדרש לתפעול, זמינות ושרידות – רק מערכת אחת לתפעל ולנטר, גיבויים – צורך בגיבוי אחד.

החיסרון המובהק זוהי נקודת כשל מרכזית, משמע במקרה וישנה פגיעה באמצעי האחסון המרכזי כל המידע נהרס.

# אמצעי קלט פלט

# אמצעי קלט פלט

צורת התקשורת של המשתמש עם המחשב היא בעזרת מערכת הפעלה, שהיא תוכנית המנהלת את כל הפעולות המתבצעות במחשב. מערכת ההפעלה מהווה ממשק בין המשתמש לבין חומרת המחשב, אך הצורה שבה המשתמש יתקשר איתה ו"יכניס" נתונים אליה אלו אמצעי הקלט והפלט.

**מהו אמצעי קלט\פלט**

אמצעי קלט((Input device: התקן או ציוד היקפי במחשב המשמש לתרגום נתונים לשפה המובנת על ידי המחשב (השפה הבינארית). נתונים אלו ישמשו כנתוני קלט בתהליך העיבוד במחשב. לדוגמא- מקלדת, עכבר, סורק וכו'.

אמצעי פלט((Output device: התקן במחשב המשמש לתרגום תוצאת העיבוד לדרך שתהיה ברורה למשתמש. לדוגמא: רמקולים, מסך, מדפסת.

תפקידם של רכיבים אלו הוא למעשה לקשר בינינו לבין המחשב – באמצעותם אנחנו יכולים לתקשר עם המחשב, להזין לו נתונים ולקבל את תוצאת העיבוד בצורה שנוכל להבין. בסופו של דבר, המחשב "מדבר" בשפה הבינארית. אנחנו לא יכולים להזין לו נתונים ישירות בשפה הזו – ובאותה מידה, גם אם נקבל את התוצאות בשפה הזו, לא נוכל להבין אותן.

אמצעי הקלט והפלט עומדים כמתווך בינינו לבין המחשב, מתרגמים את השפה שלנו (לחיצה על אייקון על גבי המסך) לנתונים שהמחשב יכול להבין, הנתונים האלו מגיעים למעבד, הוא נותן תוצאה שמגיעה לאמצעי הפלט בצורה שנוכל להבין (תמונה על המסך, פלט מהרמקולים).

**אמצעי קלט פלט נפוצים**

עכבר (Pointing Device\(Mouse: רכיב שהוא אמצעי קלט, המשמש לתנועה על גבי המסך ולסימון ובחירת אובייקטים באמצעות הקשה על אחד ממקשיו. קיימים סוגים שונים של עכברים כגון: עכבר כדורי, עכבר עט, עכבר אופטי וכו'.



סוגי חיבורים- PS\2 או USB (בעיקר).

מקלדת (Keyboard): רכיב הקלט העיקרי של המחשב. ניתן לתפעל את המחשב ללא העכבר כלל (המונח המקצועי נקרא Power User), אך לא ניתן לתפעל את המחשב ללא המקלדת. המקלדת משתמשת להזנת נתונים, הוראות, מתן פקודות לביצוע פעולות שונות במחשב, ולניווט בין האפשרויות השונות בתוכנות הרצות על המחשב.

המקשים של המקלדת מסודרים בשורות כאשר מתחת למקשים פרושות שתי רשתות בעלות נקודות מגע נפרדות לכל מקש. כאשר לוחצים על מקש הוא נוגע בשתי הרשתות והבקר מזהה את הלחיצה. לכל מקש יש "קוד סריקה" שמצביע על מיקום המקש הספציפי כך שהמחשב ידע לזהות אותו.

סוגי חיבורים- PS\2 או USB.

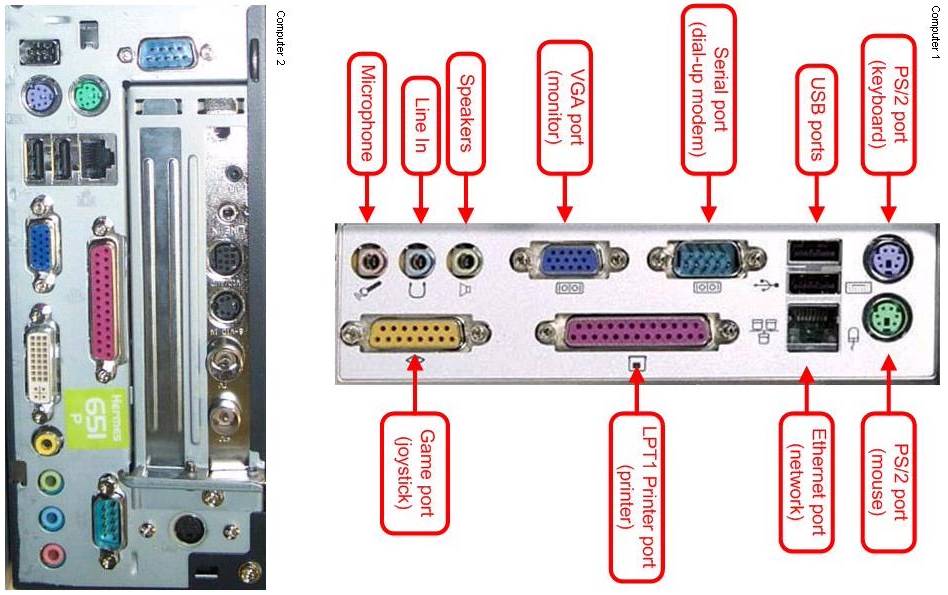
צג (Screen): אמצעי הפלט העיקרי של המחשב. לרוב, גם פעולות הקלט שיבצע המשתמש יוצגו על הצג, כך נוצר דו שיח בין המשתמש למחשב. קיימים סוגים שונים של מסכים כגון: מסך CRT, מסך LCD וכו'. מסך מגע הינו אמצעי קלט ופלט ביחד.

סוגי חיבורים- VGA (בעיקר) או DVI.

מדפסת (Printer): המדפסת היא אמצעי פלט, שמפיק מסמכים או תמונות על דפים.

סוגי חיבורים-LPT או USB.

סורק (Scanner): התקן קלט המאפשר לסרוק מסמך או שרטוט, ולהעבירו כקובץ למחשב.

בסורק קיים רכיב הקולט את האור שמוחזר מהדף הנסרק וממיר אותו לאות חשמלי. מדובר בשורה ארוכה של קולטי אור שכל אחד מהם אחראי על פיקסל יחיד בסריקה.

סוגי חיבורים- USB.

רמקולים: אמצעי פלט שמפיק מוסיקה, מתחברים לחלקו האחורי של המארז.

סוגי חיבורים- Audio. לרוב, צבע החיבור הוא ירוק בהיר.

**PS\2**

**מפת סוגי חיבורים**

**USB**

**VGA**

**LPT**

**DVI**

# תקשורת בין רכיבים

**תקשורת טורית**

תקשורת סריאלית (טורית): תקשורת בה עוברים הנתונים Bit אחר Bit. כבלי התקשורת שעובדים בצורה טורית מאופיינים בכבל ארוך, צר וגמיש. לדוגמא: עכבר, מקלדת, כבל רשת, כבל USB.

חיבור סריאלי: חיבור בין רכיבים ה"מדברים" בתקשורת טורית. חיבור זה מתאפיין בכבל דק וארוך. רכיבים כמו המקלדת והעכבר יתחברו ללוח האם בצורה זו.

בעשור האחרון חלו התפתחויות בתחום התקשורת הסריאלית, בהם מחברי PS/2 שהפכו נפוצים החל ממחשבי הפנטיום לחיבור עכבר ומקלדת במקום יציאות ה-COM הישנות, וחיבורי USB המאפשרים חיבור סריאלי מהיר ויעיל לרכיבי I/O רבים (גם כאלו שבמקור יש להם חיבורים פרללים).

כיום כבר מחברים את רוב העכברים והמקלדות לחיבורי USB ולא לחיבורי 2/PS.

השימוש ביציאת USB נפוץ כיום יותר ויותר ומשמשים בהם לחיבור מצלמה דיגיטלית, כונן HD חיצוני, מקלדת, עכבר ועוד.



Universal Serial Bus) USB): חיבור טורי חדיש, אוניברסלי ונפוץ. משמש לחיבור התקני קלט ופלט ורכיבים שונים למחשב.

### USB 2.0: התקן הנפוץ ל-USB כיום נקרא USB 2.0. תקן זה הוא שדרוג של USB 1.1 ונקרא גם High Speed USB.

### הסיבה לכך היא שהוא מהיר יותר, עד פי 40 מ-USB 1.1. USB 2.0 תומך אחורנית בכל התקני ה-USB הישנים, ומסוגל לעבוד ב-3 מהירויות, כשהגבוהה ביותר היא 480 מגה ביט לשנייה (כ-60 מגה בייט לשנייה). מתאימה לעבודה עם רכיבים הדורשים ביצועים גבוהים, כמו- מצלמות דיגיטליות, HD-ים, מדפסות, פלאפונים ועוד.

### כיום כבר יצאו USB 3.0, USB 3.1 שמהירים כל אחד פי כמה מהקודם. (3 מאפשר מהירות של 5 גיגה ביט לשנייה ואילו 3.1 מאפשר מהירות של 10 גיגה ביט לשנייה.)

Serial ATA) SATA): ממשק נפוץ לחיבור התקנים ללוח האם (כמו HD, SSD, כוננים).

כבל טורי בעל 4 גידים אליו ניתן לחבר התקן אחד בלבד.

המהירות המקסימלית היום לכבל מסוג זה היא GB/S1.9 וניתן להוסיף ולהסיר התקנים בלי צורך בכיבוי המחשב (Hot Swappable), למרות שהדבר לא מומלץ.

**תקשורת מקבילית**

תקשורת פרללית (מקבילית): המידע עובר כמה -Bitים בכל פעם. התקשורת מאופיינת בכבלים קצרים ועבים. לדוגמא: מסך, מדפסת.

המידע בסוג תקשורת זה עובר בדרך כלל Byte אחרי Byte, או כמה Byte-ים במקביל.

חיבור פרללי: חיבור פרללי הינו חיבור בין רכיבים ה"מדברים" בתקשורת מקבילית. חיבור זה מתאפיין בכבל עבה מאוד ולא גמיש בשל הצורך בגידים רבים (כל סיבית זקוקה לגיד) וכבלים קצרים יחסית בשל בעיות השראה (תגובה אלקטרומגנטית הנגרמת מצמידות הסיבים אחד לשני וגורמת להרס המידע). המדפסת, והמסך יתחברו ללוח האם בצורה זו.

דוגמאות לחיבורים פרלליים- LPTבמדפסות, VGA, DVI (חיבורי מסך).

רוב החיבורים הסריאליים כיום מהירים יותר (וכך גם משומשים יותר) מהפרלליים בשל הפרעות אלקטרומגנטיות בעת העברת כמויות גדולות של מידע בתקשורת פרללית, בעקבות כך התעשייה החליטה לפתח יותר את הטכנולוגיה הטורית.

בעבר, כשכמויות המידע שהיו יכולות לעבור על סיב אחד היו קטנות יחסית, אז לא הייתה השפעה במידה וחיברנו כמה רכיבים כאלה ממש קרוב אחד לשני. כיום, כשכמות המידע היא גדולה, לא ניתן להניח מספר סיבים אחד ליד השני מכיוון שתהיה השראה אלקטרומגנטית שתשפיע ותהרוס את המידע. בנוסף, כבלים של תקשורת פרללית הם קצרים בהרבה, פחות גמישים ובאופן כללי לא נוחים.

היום גם ניתן באמצעות סיבים אופטיים להעביר כמות עצומה של מידע בסיב אחד במהירות עצומה, ולכן אין צורך בחיבורים שהם פרלליים.

לסיכום:

|  |  |  |
| --- | --- | --- |
| **קריטריון** | **תקשורת טורית** | **תקשורת מקבילית** |
| אופן מעבר המידע | Bit אחר Bit | כמה Bitים במקביל |
| צורת הכבל | צר, ארוך וגמיש | עבה וקצר |
| מהירות | מהיר יחסית | איטי יחסית |
| דוגמאות לרכיבים | עכבר, מקלדת | מסך, מדפסת |

## חומרת שרת

הצורך בשרתים

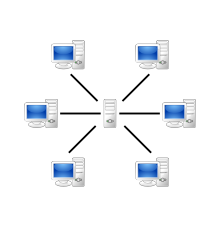
**מדוע נוצרו שרתים?**

שרתים, בשונה מהמחשבים נדרשים להיות זמינים, שרידים ומגובים כל הזמן.

שרת נועד לספק שירותים אל צד הלקוח (Client), זאת אומרת להאזין תמיד לבקשה

(Request) ולהחזיר תגובה (Response) לבקשה. השרתים יכולים לקבל בקשה בכל רגע, ולכן השרת תמיד פועל, וממתין לבקשה. לעומת מחשב אישי שנועד לספק שירותים אישיים. לרוב אנו נראה שרתים בארגונים ותעשיות גדולות, לדוגמה Google, Amazon, צה"ל וכו'.

כיום כל אפליקציה, שירות ואפילו הרשת שאנו מכירים יושבים על שרתים, הם מכילים מאגרי מידע עצומים ולכן הם מאוד קריטיים לתפקוד היום יומי שלנו.



**מהו שרת?**

מבנה השרת לא שונה ממבנה המחשב אך שרת נועד להקצות משאבים או שירותים למחשבים אחרים ברשת, זאת אומרת שהוא מכיל רכיבים אשר דומים לרכיבי המחשב ההבדל הוא שלצורך שרידות ,כוח עיבוד וזמינות, השרתים מכילים חומרה הרבה יותר חזקה, ובכמות יותר גדולה ממחשב אישי. זאת אומרת שלשרת יהיה יותר ממעבד אחד, כמות גדולה של דיסקים קשיחים במהירות ונפח יותר גבוהים וכו'..

השרתים לרוב מאוחסנים בארונות אשר נקראים Rack. כל ארון מחולק ליחידת מידה הנקראת - Rack Unit (U) שהיא יחידת מידה סטנדרטית לכלל הרכיבים והשרתים הקשורים לארונות שרתים .

## סוגי שרתים

**ישנם שלושה סוגי שרתים:**

שרת Tower: שרת ישן, אשר נראה כמו מחשב אישי. וכל הרכיבים שמחוברים אליו שייכים לו בלבד. שרתים אלה פחות נמצאים בשימוש בגלל גודלם וצורתם הלא נוחה לעבודה ולהחלפת רכיבים.  
לדוגמה:



שרת פיצה: השרת מכונה כך בגלל צורתו החומרתית אשר מזכירה מגש פיצה, שרתים אלה מצויים בארונות (Racks) ויכולים לתפוס מקומות של 1U ומעלה.  
 לדוגמה:



שרת 1U



שרת 2U

שרתBlade : השרתים הכי חדישים ונפוצים כיום בשוק. השרתים נמצאים במארז הנקרא Blade Center אשר יוצר **מערך שמסוגל לנהל עשרות שרתים**.  
מארז ה Blade Center **מרכז** את התשתיות (חשמל ותקשורת) ששרתי הBlade זקוקים להם.

לדוגמה:



מארז Blade System של חברת HP , מקדימה.

מארז Blade System של חברת HP , מאחורה.



שרת Blade בודד.

## חוות שרתים – Data Center / Server Farm

**מהי חוות שרתים?**

לרוב אנו לא נראה שרת בודד מבצע פעולה כלשהי או נותן שירות מסוים לבדו מכיוון שהוא יכול לקרוס מסיבות כגון עומס יתר, תקלות או לצורכי תחזוקה.

לכן לרוב תמיד לכל שירות, פעולה או אפליקציה יהיו הרבה שרתים אשר תפקידם יהיה זהה על מנת למנוע נפילה. את כל השרתים האלה יהיו מוחזקים במקום אשר נקרא חוות שרתים, חוות שרתים היא אוסף של שרתים אשר מיועדים לבצע משימות ופעולות אשר מעבר ליכולת של שרת אחד. חוות שרתים מאובזרת במערכות קירור מתקדמות, מתגים ונתבים. לעיתים קרובות בחוות שרתים יהיו פתרונות להפסקות חשמל, קירור ותמיד יוקצו מספר שרתים לאותה משימה כך במקרה של תקלה באחד מהם, השירות לא יופסק.

לדוגמה – חוות שרתים, ניתן לראות את הארונות (Racks) שבתוכם שרתים ומעל מסילות לכבלי העברת נתונים וחשמל.



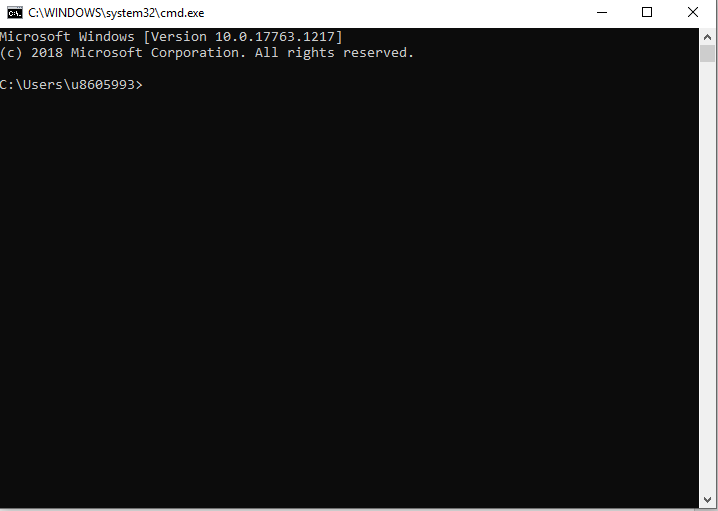
## CMD

במחשב שלנו, אנחנו יכולים לבצע כל מיני פעולות בעזרת העכבר והמקלדת כמו לפתוח תיקייה חדשה (קליק ימני בdesktop -> חדש -> תיקייה) או למחוק קבצים – באמצעות לחיצה על קובץ ולחיצה על delete במקלדת. הצורה הזו נקראת הצורה הגרפית:

*ממשק גראפי (חלונאי) – ממשק עבודה המשלב בתוכו גראפיקה.*   
*העבודה עם מערכת ההפעלה מלווה בשימוש בעכבר, אייקונים וצבעים.*

קיימת צורת עבודה נוספת בה אנו יכולים לבצע פעולות במחשב, והיא נקראת הצורה הקריקטריאלית. בצורה זו אנו ניתן פקודות למערכת, כמו פתיחת תיקייה או מחיקת קובץ, דרך הקלדתן בטקסט. ההגדרה:

*Console (קריקטריאלי) – ממשק המאופיין בטקסט בלבד.*  
*כל פעולות המשתמש מתבצעות על ידי הקלדת טקסט, פקודות בלבד.*

נשתמש בצורת העבודה הזו בתוך ממשק הנקרא (**CMD (Command Prompt**. הוא נראה כך: 

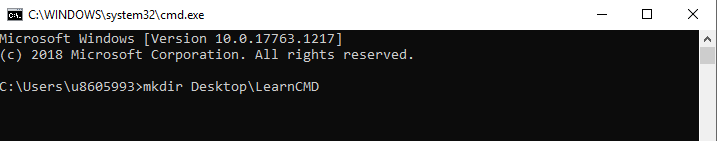
נגיע אליו דרך התחל -> CMD.exe.

הגדרה – **CMD – (Command Prompt)**

ממשק קריקטריאלי בו ניתן להקיש פקודות על מנת לבצע פעולות במחשב.

אנו נקליד את הפקודות שברצוננו לבצע במערכת לפי תחביר מסוים (syntax) – כלומר, לכל פקודה יש דרך מסוימת שבה היא צריכה להיכתב על מנת שה-CMD יבין מה עליו לבצע. אם נכתוב בעברית לדוגמה, הפעולה לא תיקלט ולכן לא תתבצע, וגם אם נכתוב בצורה חופשית לפי תחביר השפה האנגלית – הCMD גם כן לא יבין מה הפעולה, כי יש תחביר מסוים לכל פעולה ופקודה ולא נוכל לכתוב בCMD ללא חוקים מסוימים.

לדוגמה, אנו רוצים לפתוח תיקייה חדשה על שולחן העבודה בשם LearnCMD. נפתח את חלון הCMD ונקליד mkdir Desktop\LearnCMD.



התחביר של הפקודה ליצירת תיקייה חדשה הוא קודם כל שם הפקודה (במקרה שלנו - mkdir), אחר כך יגיע הנתיב בו אנו רוצים למקם את התיקייה (desktop), ולבסוף שם התיקייה החדשה שניצור (LearnCMD). התו '\' מפריד לנו בין תיקיות. בהמשך נראה עוד דוגמאות של פקודות בCMD.

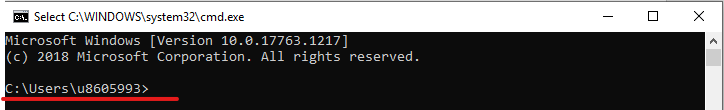
\*\*mkdir – קיצור של make directory. ייצא לכם לראות הרבה קיצורים של פקודות כאלו בCMD.

**נתיב נוכחי:**

בכל רגע נתון בממשק ה- CMD אנו נימצא בתוך תיקיה כלשהי.

**כל פקודה שנבצע ולא נציין ספציפית על מה נבצע אותה, תתבצע בנתיב הנוכחי.**

* הנתיב הנוכחי מופיע בשורת הפקודה.



לדוגמה, אם נבצע mkdir LearnCMD בלי להגדיר איפה אנחנו רוצים לפתוח את התיקייה (כמו שהגדרנו desktop בדוגמה הקודמת), התיקייה תפתח בנתיב הנוכחי שהוא–   
C:\Users\u8605993>

אם נמחק קובץ בלי לפרט את הנתיב שהוא נמצא בו, הCMD יניח שהקובץ נמצא בנתיב הנוכחי ושם יחפש אותו.

**מאפייני הCMD:**

יתרונות: פשטות, יעילות, גמישות בביצוע תהליכים, אוטומציה של תהליכים.

חסרונות: אין גרפיקה, אין שימוש בעכבר, אין עברית, מקום לטעויות בעקבות ריבוי הקלדה, זכירה של פקודות.

**פקודות בסיסיות:**

* **Dir** - הצגת התוכן הקיים תחת הנתיב הנוכחי.

שימוש: dir – בשימוש לבד, מציג את תוכן התיקייה בה אנחנו נמצאים.

dir *path* – מציג את תוכן התיקייה path.

דוגמה: dir c:\windows\myFolder – מציג את תוכן התיקייה myFolder.

\*\* פקודת dir מציגה את התיקיות והקבצים שנמצאים בנתיב שנכתב בפקודה. נבדיל בין קבצים לתיקיות כך: התיקיות מסומנות על ידי <DIR> ליד שמן, וליד הקבצים מופיע מספר – גודל הקובץ בבייטים.

* **Echo** - הצגת פלט למסך.

שימוש: echo *text* – מדפיס את text למסך.

Echo *text* > *File.txt –*  מדפיס את text לתוך קובץ File.txt. אם file לא קיים, הוא יווצר.

דוגמה: echo today is a good day > myFile.txt – מדפיס טקסט כרצוננו לתוך קובץ טקסט בשם myFile.

* **Type –** קריאת טקסט מתוך קובץ (הדפסתו למסך).

שימוש: type *path* – מדפיס את התוכן של קובץ path.

דוגמה: type c:\users\guest\desktop\story.doc – ידפיס למסך את מה שכתוב בתוך story.doc (doc = קובץ word).

* **CD** - שינוי נתיב (change directory)

שימוש: cd *path-* משנה את הנתיב בו אנחנו נמצאים לנתיב path.

cd /d path – משנה גם את הכונן בו אנחנו נמצאים (נכניס בpath כונן שונה מזה שבו אנחנו נמצאים).

cd .. – חוזר לתיקיית האב של התיקייה בה אנו נמצאים.

דוגמה: אם אנחנו נמצאים ב c:\windows\temp ונקליד cd .., אנו נחזור

לתיקיית האב – c:\windows.

אם אנחנו נמצאים בc:\windows\temp ונרצה להחליף נתיב לנתיב שנמצא בכונן שונה, נקליד cd /d z:\drivers.

* **Mkdir** - יצירת תיקייה.

שימוש: mkdir *path –* יוצר תיקייה תחת הנתיב path.

דוגמה:mkdir hello\hi – ייצור לנו תיקיה בשם hi תחת תיקיית hello.

אם תיקיית hello לא קיימת, היא גם תיווצר.

* **Del** - מחיקת קובץ.

שימוש: del *path* – מוחק את הקובץ path.

דוגמה: del file1 file2 file3 – מוחק את כל הקבצים שפירטנו.

* **Rmdir –** מחיקת תיקייה (remove directory).

שימוש: rmdir *directory* – מוחק את התייקיה directory אם היא ריקה.

דוגמה: rmdir folder1\folder2 – מוחק את התיקייה folder2 אם היא ריקה.

* **Copy –** העתקת קובץ.

שימוש: copy file1 file2 – מעתיק את תוכן קובץ file1 לקובץ file2.

* **Find** - מציאת מחרוזת כלשהי.

שימוש: find string file – מציאת המחרוזת string בתוך הקובץ file.

דוגמה: find hello stories\morning.txt – מחפש את המחרוזת hello בתוך הקובץmorning .

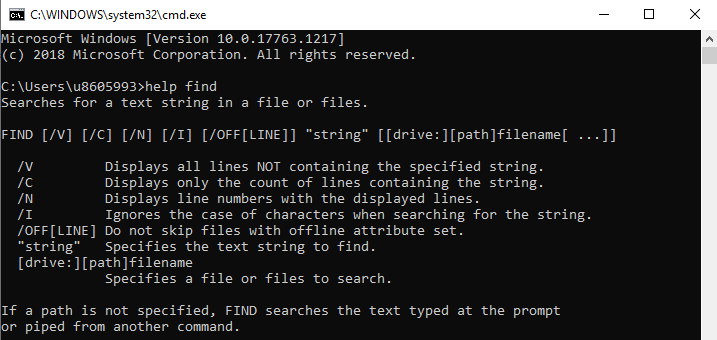
* **Start** - התחל, בדר"כ סקריפט או תכנית.

שימוש: start *path –* מריץ\פותח את path.

דוגמה: start c:\windows\goodFile – הקובץ goodFile ייפתח.

* **Shutdown** - כיבוי.
* **Help** – עזרה.

**HELP:**

נשתמש בפקודת הhelp כאשר נרצה לקבל מידע על מה פקודה מסוימת עושה, איך כותבים אותה (סינטקס) ואיזה אפשרויות יש לה. לדוגמה אם נבצע help על הפקודה find, נקבל את הפלט הבא:  


**מבנה הפקודה**

**תיאור הפקודה**

**מאפיינים**

בפלט של פקודת "help find" מוסבר לנו תפקיד הפקודה (שהוא למצוא טקסט מסוים בתוך קובץ/ים) את המבנה שלה (איך היא כתובה, באיזה סדר לכתוב אותה) והמאפיינים שלה – לדוגמה /U שעם שימוש בו, נקבל בפלט את כל השורות **שלא** מכילות את הטקסט שברצוננו למצוא.

\*\* ניתן להשתמש בפקודה זו גם בעזרת /? בסוף הפקודה. בחלק מהפקודות שנרצה למצוא עליהן מידע, help לא נתמך ולכן נשתמש במקבילה אליו - /?.

Help del = del /?

**מבנה פקודה:**

כל פקודה בCMD מורכבת משלושה חלקים: command, qualifier ו- parameter.

הפקודה עצמה (Command) **-** מה לבצע, פקודה תופיע אך ורק פעם אחת בשורת הפקודה (חוץ ממקרים נדירים של פקודה שמריצה פקודה).

פרמטר/ים (PARAMETER) **-** על מה או עם מה לבצע את הפעולה, פרמטר יכול להופיע פעם אחת או יותר, או בכלל לא, ואז ערך הפרמטר הוא ברירת מחדל כלשהי.

מאפיין/ים (QUALIFIER) **-** איך לבצע את הפעולה, מאפיין יכול להופיע פעם אחת או יותר, אם אינו מופיע ערכו נקבע כברירת מחדל.

לדוגמא: **Dir /O /Q C:\**   
  
**Dir** – הפקודה (מציגה תוכן של תיקייה)  
 **/O /Q** – המאפיינים. לפי המאפיין O, תוכן התיקייה יוצג בסדר מסוים – קודם תיקיות ואח"כ קבצים לפי גודלם. לפי המאפיין Q, יוצגו בפלט גם בעלי הקבצים שבתוך התיקייה.  
**C:\** - הפרמטר (התיקייה שעליה מורצת פקודת dir)

**דגשים:**

* הפעולה תבוא תמיד בהתחלה.
* לפני פרמטר ובין פרמטרים יבוא רווח.
* לפני מאפיין תמיד יבוא "/" (SLASH).
* פרמטרים ומאפיינים לא תמיד חובה לציין (קיים שימוש בברירת המחדל).
* לכל פעולה פרמטרים ומאפיינים משלה – הכל אפשר למצוא בhelp.

**שילוב פקודות:**

Pipline (|) – אופרטור לשילוב פקודות. נשתמש בו כאשר נרצה להעביר פלט של פקודה אחת כקלט של פקודה שנייה.

דוגמה: command1 | command2

Output Redirections– תווי ניתוב קלט ופלט. ניתן לנתב (להכניס, לשלוח) פלט של פקודות לתוך קבצים, במקום למסך.

* < - שליחת פלט פקודה לתוך קובץ מסוים. במידה והקובץ קיים וכבר יש בו תוכן, הוא יידרס (התוכן הקודם יימחק ובמקומו ייכתב התוכן החדש).

דוגמה: dir > showInsideDir – יכניס את הפלט של הפקודה dir לתוך הקובץ showInsideDir.

* << - כתיבה ועדכון. במידה ואנחנו רוצים שהתוכן הקיים בקובץ לא יידרס, נכתוב << וכך התוכן החדש יתווסף אל התוכן הקיים.
* > - קבלת קלט מתוך פקודה או קובץ. במקום להפנות פלט, כאשר נרצה להכניס קלט לפקודה מסוימת נעשה זו בצורה הבאה: command < text. Text ייכנס לפקודה command בתור קלט.

**תרגול CMD**

בכל שאלה בתרגול תתבקש לבצע פעולה מסוימת.  
רשום את הפקודה**/פקודות אשר ביצעת בשאלה**, וכן את הפלט שלה, אם התבקשת לכתוב אותו.

טיפים לעבודה נוחה עם הCMD:

* בלחיצה על החיצים למעלה ולמטה, תוכלו להגיע להיסטוריית הפקודות שביצעתם ולא תצטרכו לכתוב כל פקודה מחדש.
* מקש הtab: המקש מעל הcaps lock, משלים אוטומטית פקודות/שמות של קבצים ותיקיות. לדוגמה אם יהיה לנו קובץ בשם כל האותיות בשפה האנגלית שנרצה למחוק, נתחיל להקליד del abc ונלחץ tab ושם הקובץ יושלם אוטומטית.
* CMD הוא לא case sensitive - כלומר לא רגיש לאותיות קטנות וגדולות, אז אין צורך להקפיד על כך כאשר מקלידים פקודות או שמות של תיקיות וקבצים.

היכנס אל ממשק ה-CMD.

* + - * 1. רשום כאן את הנתיב הנוכחי שבו אתה נמצא.

2. השתמש בפקודה שתציג את תוכן התיקייה שבה אתה נמצא.

3. הדפס למסך את שמך.

4.עבור לתיקיית Desktop, ולאחר מכן צפה בתוכן שלה.

5. תחת תיקיית Desktop, צור תיקיה חדשה בשם כלשהו. בדוק בשולחן העבודה שלך שהתיקייה באמת נוצרה.

6. צפה שוב בתוכן תיקיית Desktop ובדוק כי התיקייה שיצרת נמצאת. כתוב את תוכן הפלט אל קובץ חדש בשם log.txt.

7. פתח את Log.txt דרך הCMD.

8. מחק את Log.txt.

9. צור תחת תיקיית Desktop קובץ טקסט בשם כרצונך, שבתוכו כתוב השם שלך.

10. העתק את הקובץ שיצרת לקובץ שנקרא animal.doc.

11. הדפס למסך את הטקסט מתוך animal.doc.

12. רשום מה עושה הפקודה Move, ובאיזה פקודה השתמשת כדי למצוא זאת.

13. רשום מה עושה הפקודה Sort, ובאיזה פקודה השתמשת כדי למצוא זאת.

14. הצג **בשורת קוד אחת** את תוכן התיקייה בה אתה נמצא כאשר היא מסודרת לפי הפקודה שמצאת בסעיף הקודם.

15. שנה את שם התיקייה שיצרת בסעיף 5 ל- bsmh.

16. צור תיקייה בשם happy כרצונך תחת תיקיית bsmh.

17. מחק את התיקייה bsmh. עשה זאת בפקודה אחת בלי למחוק ידנית את התיקייה happy שבתוך Bsmh (היעזר בhelp).

18. הסבר מה הפקודה systeminfo עושה.

19. מצא בעזרת הפלט של הפקודה systeminfo מהי ארכיטקטורת המעבד במחשב שלך (X64 bit / X32 bit) וכמה זיכרון פיזי יש לך במחשב (ההסבר לאלו מופיע בחוברת החומרה שקיבלתם).

בונוס:

שאלות אלו לא מבוססות על חומר שלמדתם, אלא דורשות מעט חקירה בגוגל/שימוש בHELP וכוללות "טיפים" לשימוש נוח וכיף בCMD.

אין חובה למלא אותן.

1. מצא והשתמש בפקודה שתנקה את המסך של חלון הCMD.
2. שנה את הרקע של חלון הCMD לכחול ואת הכיתוב לוורוד.
3. צא מחלון ה-CMD באמצעות שימוש בפקודה.
4. Easter egg – מצא איך לצפות במלחמת הכוכבים בCMD (היעזרו באינטרנט).
5. מצא איך להציג את תוכן הנתיב הנוכחי בצורה היררכית – כך שמתחת לכל תיקיה יוצגו גם הקבצים והתיקיות שבתוכה.

**תוכן החוברת הינו חיוני כבסיס לקורס, קראו בעיון.**

1. (טרנס) – מוליך (רזיסט) – מתנגד, רכיב אלקטרוני הבנוי מחומר מוליך למחצה (החומר מאפשר העברת זרם חשמלי דרכו, בתנאים מסוימים). משתמשים בו בעיקר לביצוע חישובים – יש חשמל, אין חשמל. [↑](#footnote-ref-2)
2. קטע קוד אותו המחשב מעבד. [↑](#footnote-ref-3)