

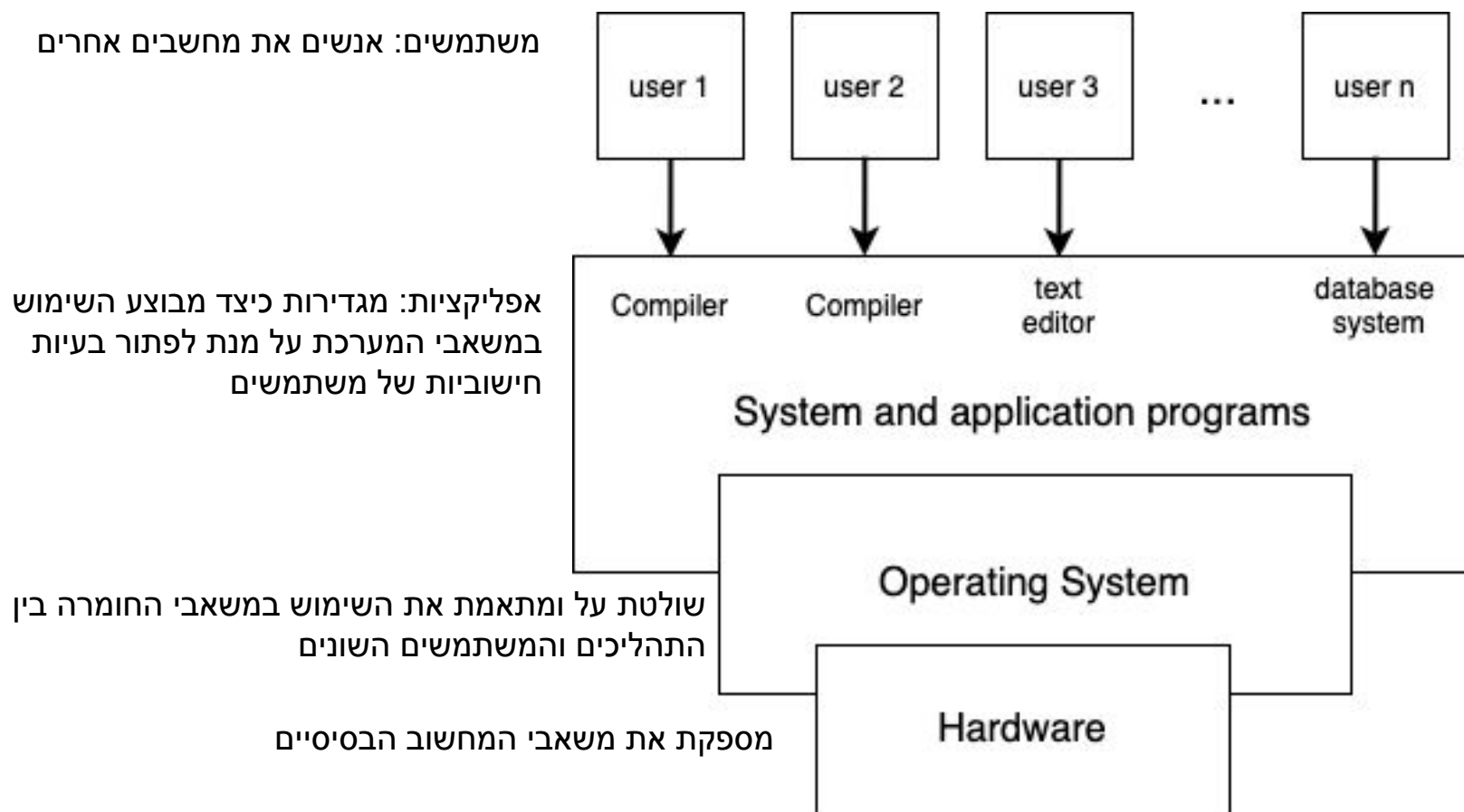
מערכות הפעלה

שיעור 2 - מבוא

מרצה: אליאב מנשה

מבנה מערכת מחשב

ניתן לחלק את מערכת המחשב ל- 4 מרכיבים עיקריים



מ"ה – הגדרה ומטרות

הגדרה:

תכנית אשר משמשת **כחוצץ** (intermediary) בין **המשתמש** של מערכת מחשב **והחומרה** של מערכת המחשב.

מטרות מ"ה:

- הרצת תכניות המשתמש (User Programs) בצורה קלה
- הפיכת השימוש במערכת מחשב לנוח יותר
- שימוש יעיל יותר (efficient) בחומרת מחשב

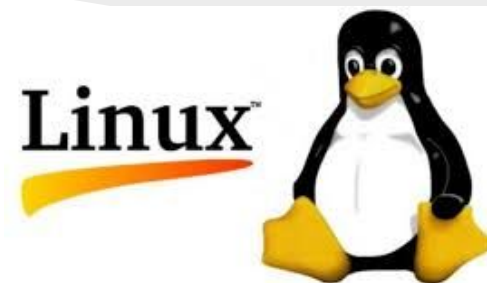
דוגמה: השימוש במ"ה עבור MS-Paint

נניח שאנו משתמשים ב- MS-Paint של Windows, מתי

נצטרך שירותים של מ"ה?

- Loading the application / terminating the application
- Interrupt management
- Memory allocation / managing
- Access to I/O devices – keyboard, mouse, printer, monitor...
- CPU allocation
- Copy / Paste (IPC)

מערכות הפעלה פופולריות



מערכות הפעלה פופולריות



Google



דיון

האם יכולה להיות מערכת מחשב ללא מערכת הפעלה?

עקרונית כן אבל ייצור המון בעיות

- נצטרך לכתוב תכנית בקוד מכונה
- לא נוכל להריץ יותר מתכנית אחת בו"ז
- נצטרך לטפל בכל המקרים והתגובות (שגיאות, הזזת עכבר, מקלדת...) (מקלדת...)
- נצטרך לנהל את החומרה בעצמנו
- מערכת קבצים...

מתאים בעיקר למערכות סגורות

OS – Design & Goals

לכל מערכת הפעלה יש מטרות ועיצוב שונים:

Mainframe – ניצול מקסימלי של משאבי החומרה

PC – תמיכה מקסימלית בהרצת תכניות משתמש

Handheld – ממשק נח להרצת אפליקציות, ביצועים טובים, ניצול

הסוללה



קטגוריות מחשבים

- **Supercomputer** – frontline of current processing capacity, particularly speed of execution
- **Mainframe** – powerful computers used mainly by large organizations for critical application
- **Personal** computers - any general purpose computer whose size, capabilities and price makes it useful for individuals
- **Handheld** – pocket size computing device, typically having a touch screen or mini keyboard

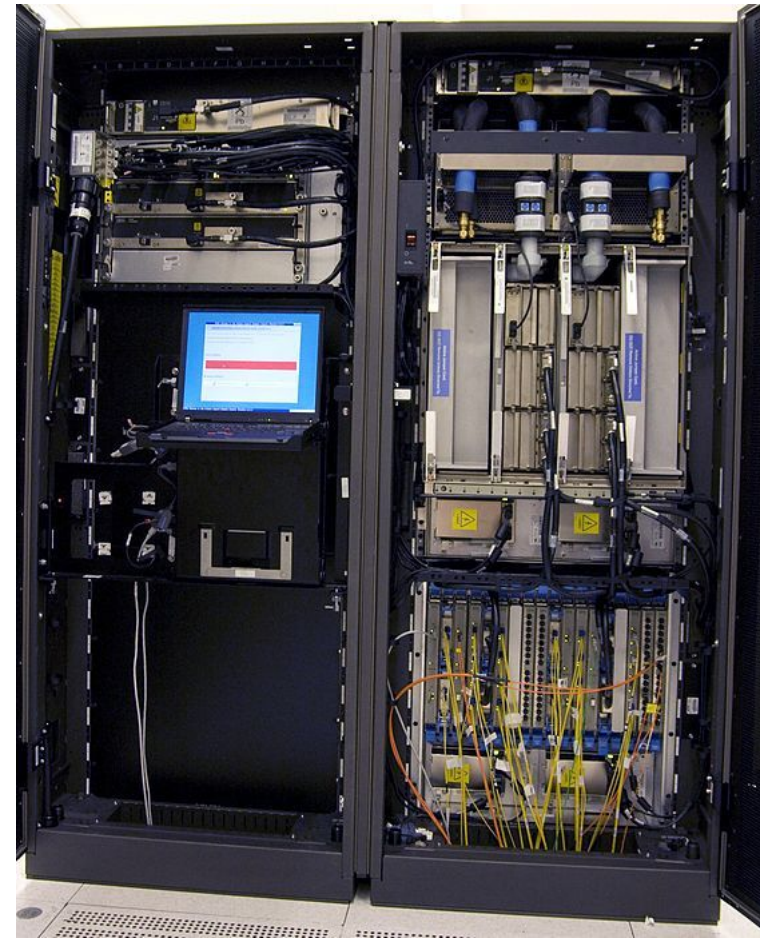
Supercomputer

The IBM Blue Gene/P supercomputer at Argonne National Laboratory runs 164,000 processor cores



IBM Mainframe

One mainframe can execute up to 1 trillion secure web transactions per day, delivers high performance with up to 190 configurable cores and up to 40 TB memory.



מה טבעה של מערכת ההפעלה?

מערכת ההפעלה היא resource allocator:

- מנהלת את המשאבים (resources) בדומה לממשלה

- מחליטה במצבים של קונפליקט על מנת שהשימוש במשאבים יהיה יעיל (efficient) והוגן (and fair)

מערכת ההפעלה היא control program:

- שולטת על ההרצה (execution) של התכניות השונות על מנת למנוע שגיאות ושימוש מוטעה / חורג במערכת המחשב

חשוב במיוחד
כאשר ישנם
מספר
משתמשים
המחזיקים
לאותו מחשב
או מספר
תכניות
שרצות
במקביל

תכולת מערכת ההפעלה

- האם סוליטייר, internet explorer, אנטי וירוס וצייר הם חלק ממערכת ההפעלה?
- אין הגדרה אחת אוניברסלית
- יש שסוברים:
"everything a vendor ships when you order an"
"operating system"
- התכנית/התהליך שרץ/ה כל הזמן במערכת המחשב נקראת **kernel**. כל שאר התהליכים שרצים מקוטלגים כ-system programs (תכניות שהגיעו יחד עם מערכת ההפעלה) או Application program

The matter of what constitutes an operating system is important! In 1998 this was the essence of a suit filed by the US department of justice against Microsoft

הפעלת מחשב

Firmware = computer program that provides the low-level control for the device's specific hardware

BIOS (Basic Input/output system) = non-volatile firmware used to perform hardware initialization during the booting process (power-on) = bootstrap program for PC

Bootstrapping - usually refers to a self-starting process that is supposed to proceed without external input.

בהדלקת המחשב או reboot – עולה bootstrap program

- בד"כ מאוחסנת ב-ROM או EPROM (לרוב מתייחסים אליה בתור firmware)

- מאתחלת את כל מרכיבי המערכת (CPU registers, device controllers, memory content, etc)

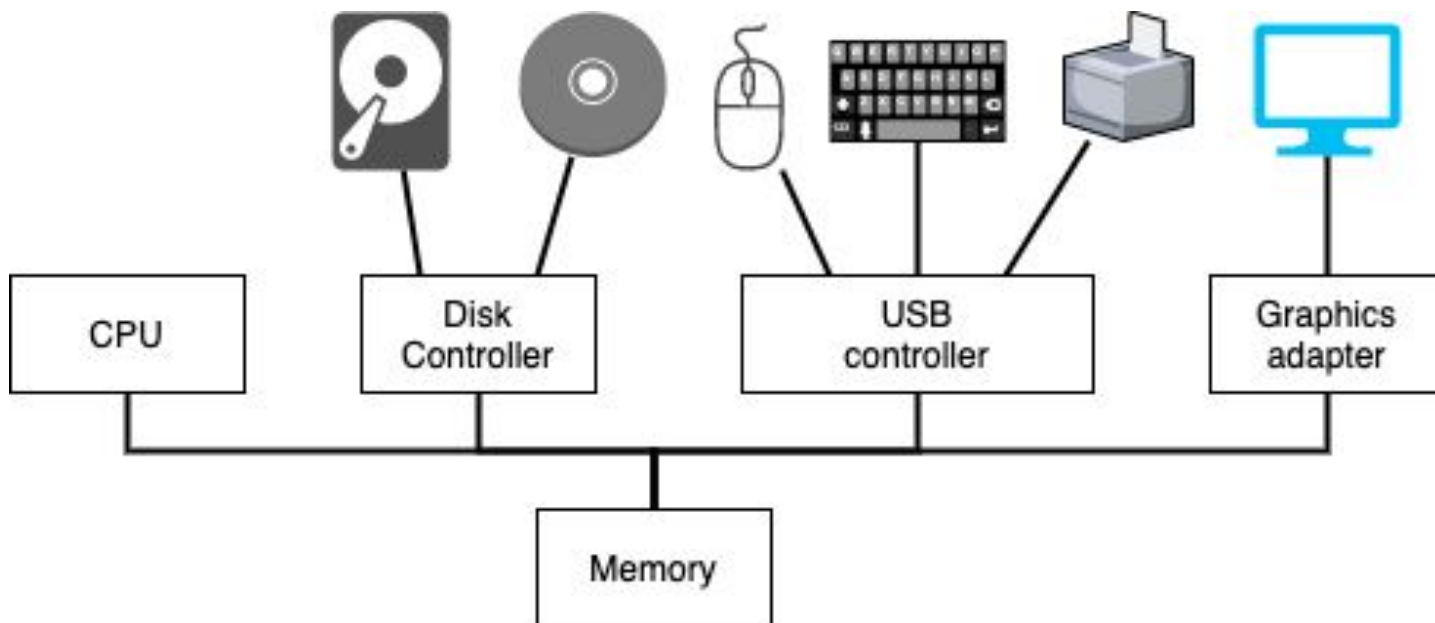
- מעלה את ה-kernel של מערכת ההפעלה ומעבירה אליה שליטה

OS initializes, starts its first process and waits for an event...

מערכת המחשב

מערכת המחשב מאופיינת ע"י:

- CPUs (אחד או יותר) ו- Data controllers שמחוברים באמצעות common bus להם גישה לזיכרון המשותף.



How Modern Computer Works

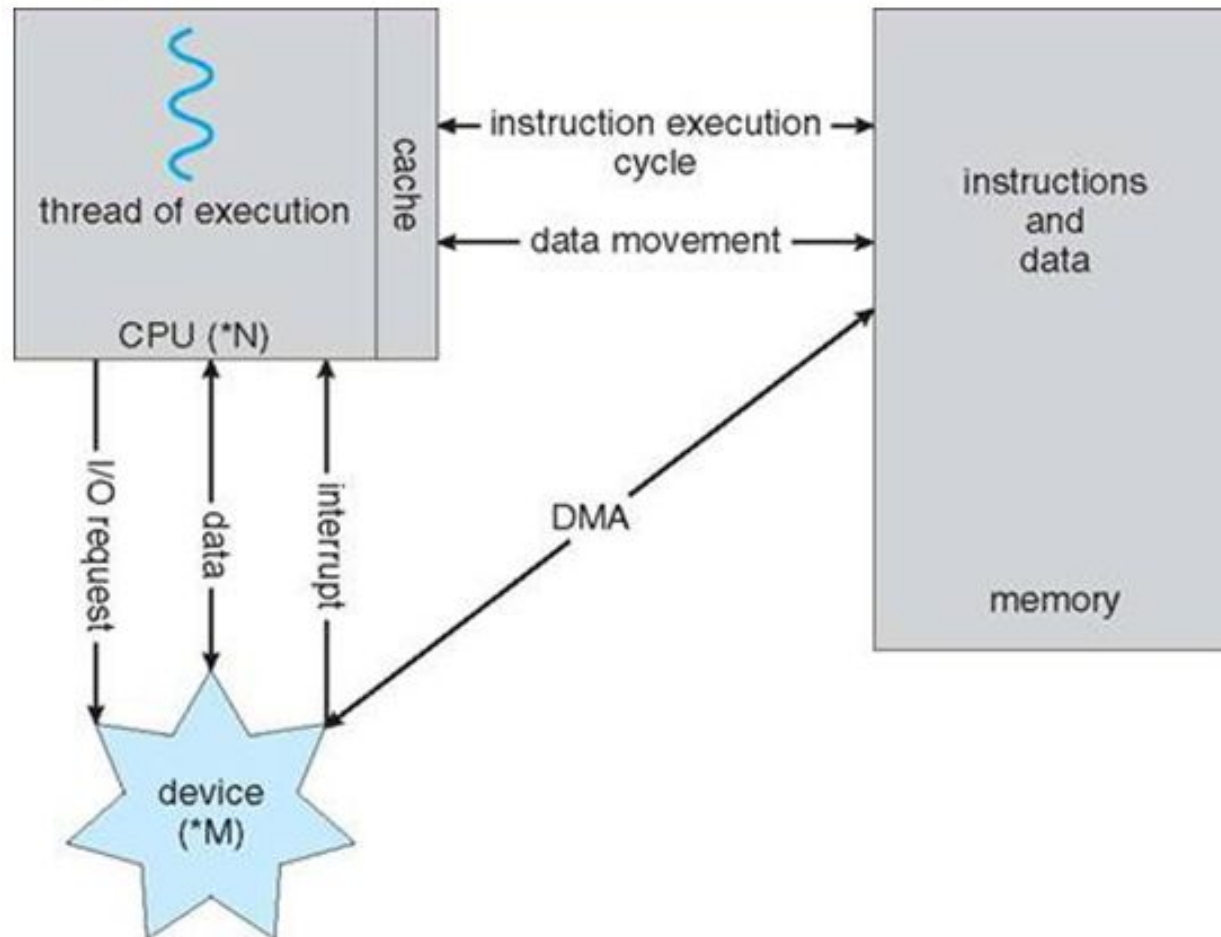
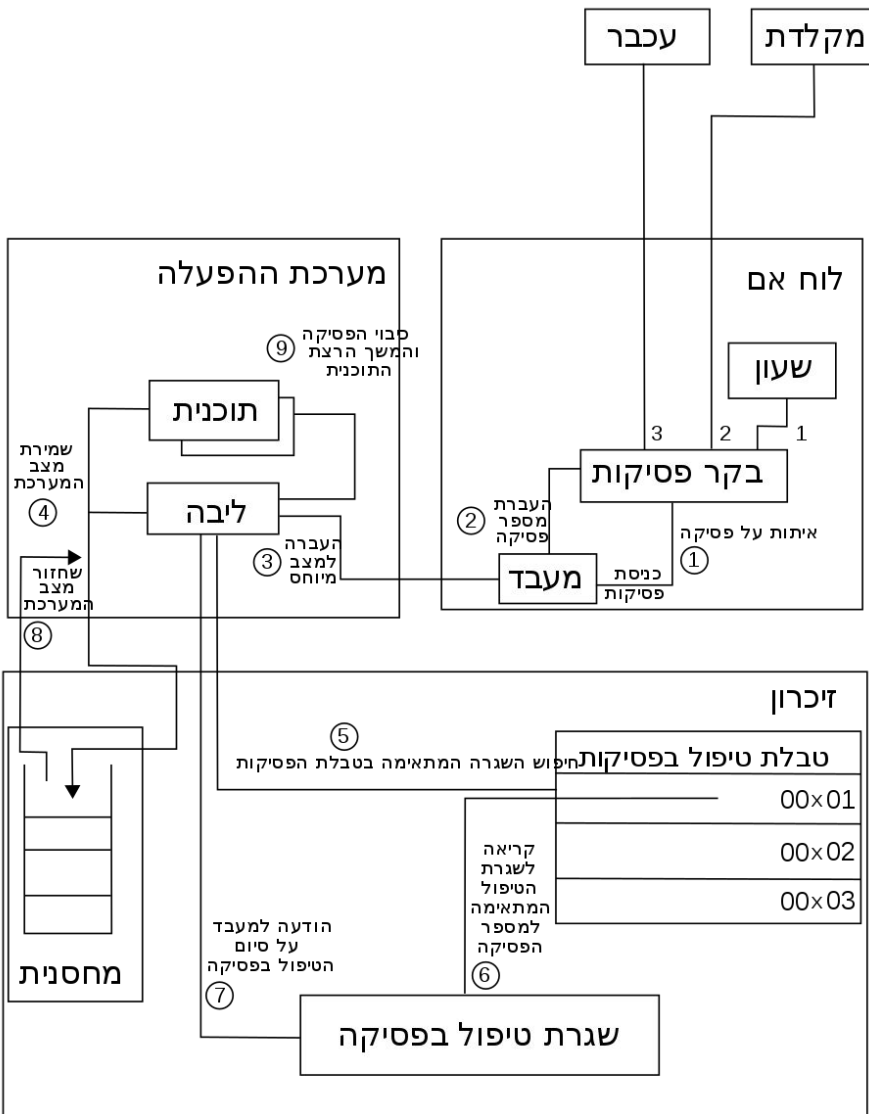


Figure 1.5 How a modern computer system works.

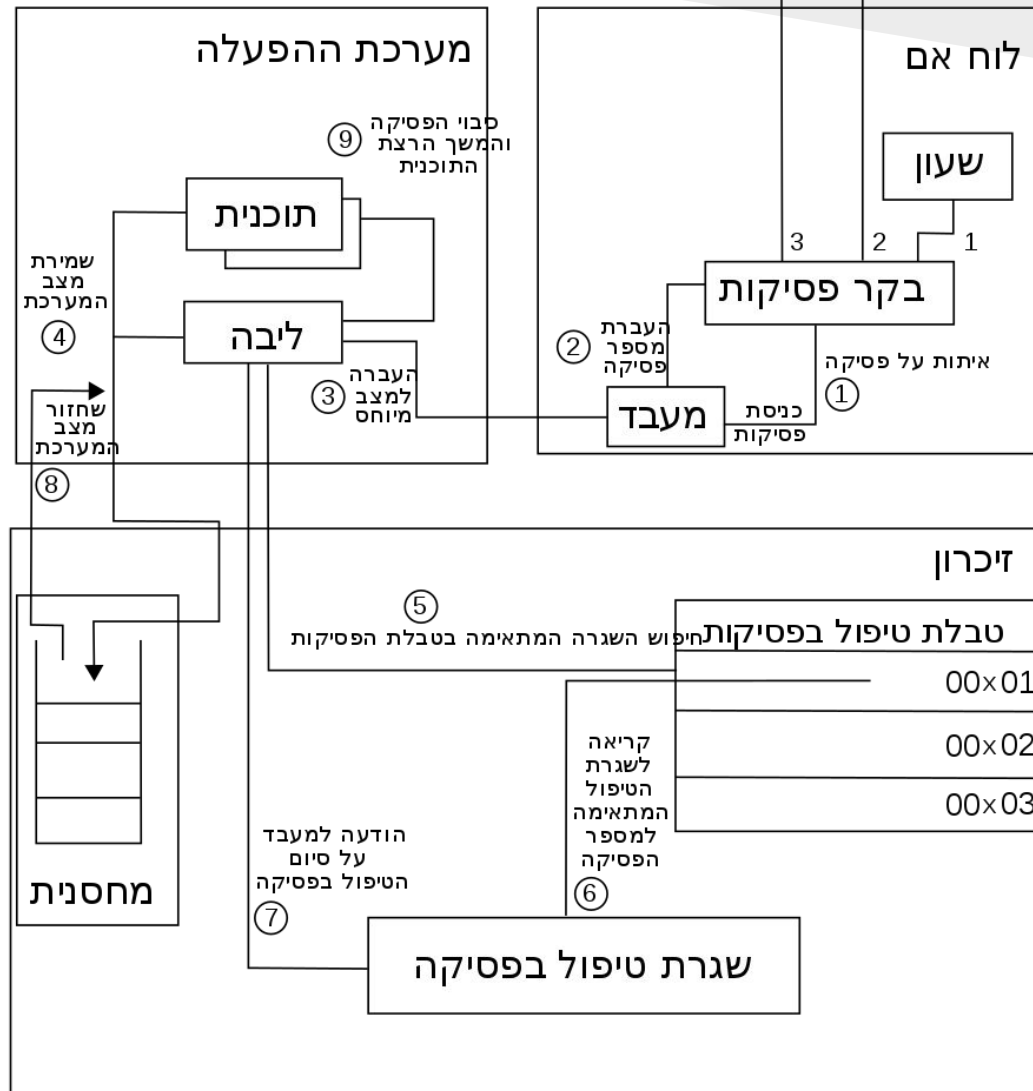
Interrupt

- פסיקה (באנגלית Interrupt)
- היא אות המתקבל במעבד מרכיב חומרה
- נוצר על ידי device
 - מעביר את השליטה (במעבד) ל- interrupt service routine
 - מחייב שמירה של הכתובת של ה- interrupted instruction ושל תוכן הרגיסטרים
 - לרוב בעת הטיפול ב- interrupt המערכת תמנע שליחת interrupts נוספים על מנת לא לאבד interrupt



עכבר

מקלדת



Interrupts and Traps

ה-trap הוא software generated interrupt אשר נגרם ע"י שגיאה
או בעקבות בקשה של תכנית של משתמש

- An OS is interrupt driven

Interrupts Handling

תחילה יש לקבוע איזה סוג של interrupt קרה (למשל תזוזה של עכבר):

- Polling

לאחר קבלת פסיקה מתחקרים את הרכיבים השונים כדי לפענח מאיפה הגיע הפסיקה

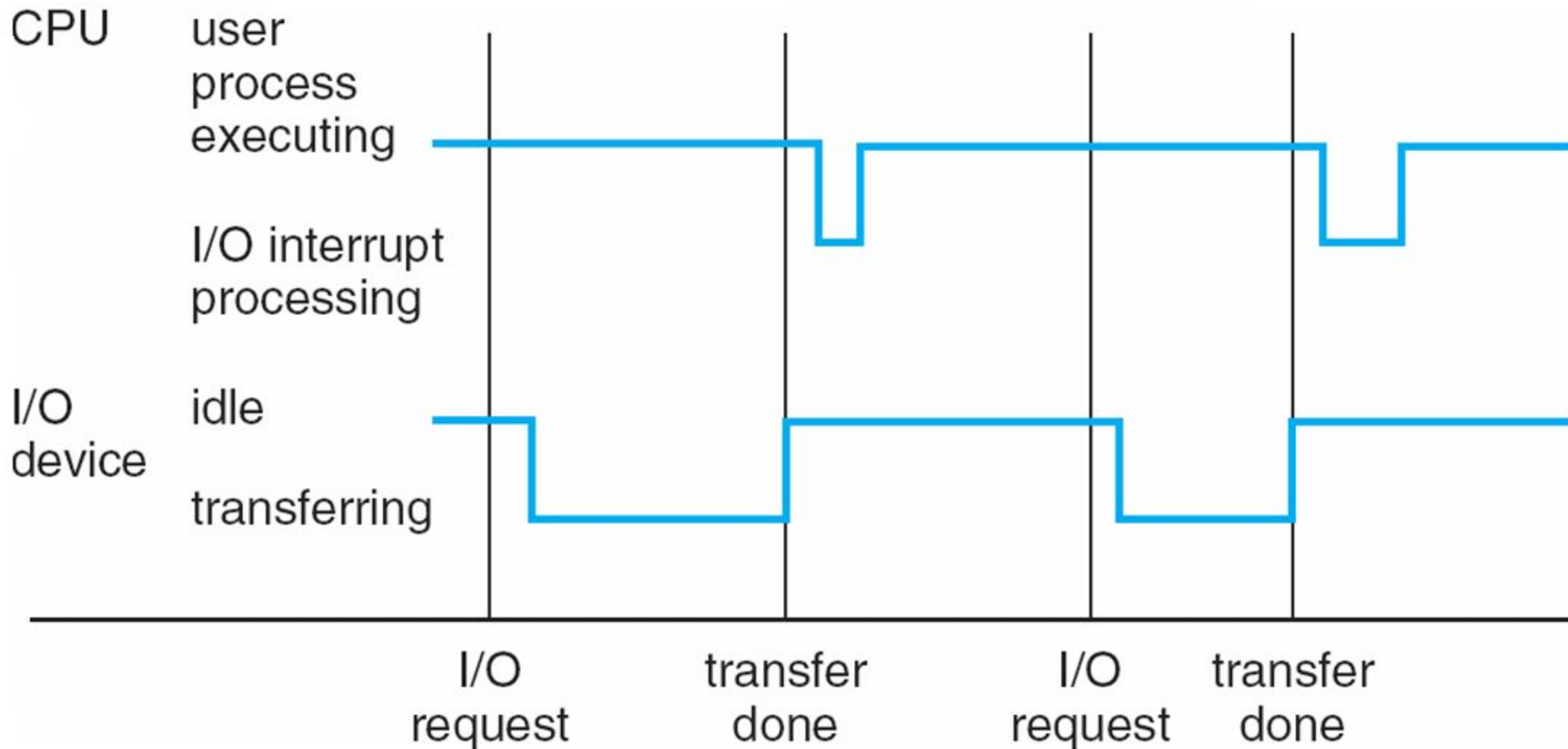
- Vectored interrupt system

קבלת קוד לכל פסיקה בפסיקה עצמה

בהתאם לסוג הפסיקה יופעל קטע הקוד הרלוונטי השמור בזיכרון

Ideally, we would have used a generic code for analyzing the interrupt information and deciding what code to run, however speed is critical here

Interrupts Timeline

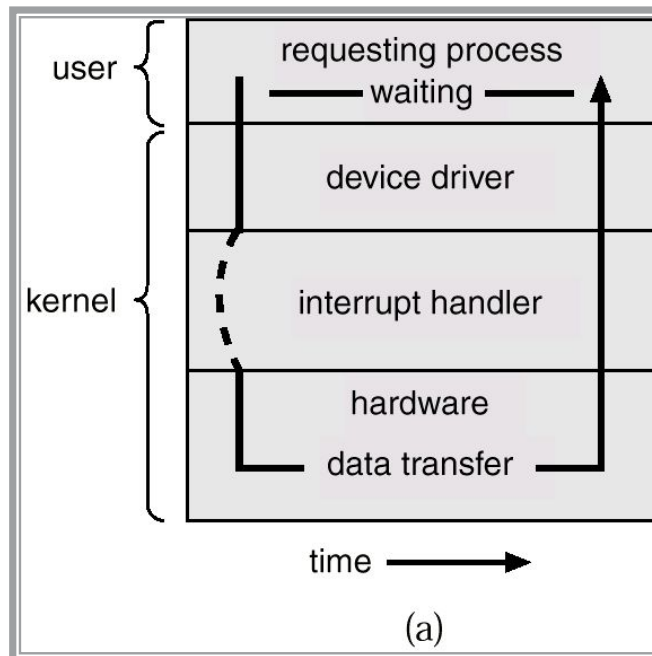


טיפול ב-I/O

קיימות שתי צורות לטיפול בבקשות ה-I/O – סינכרונית וא-סינכרונית

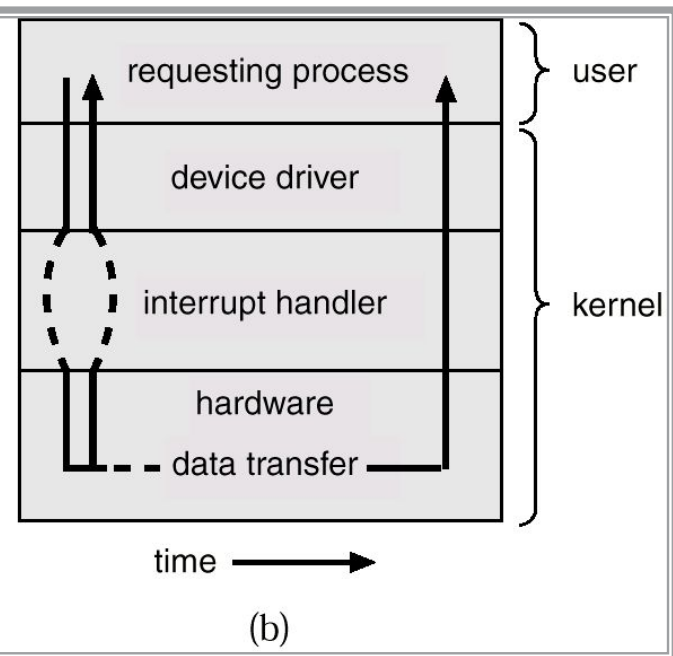
Synchronous

After I/O starts, control returns to user program only upon I/O completion

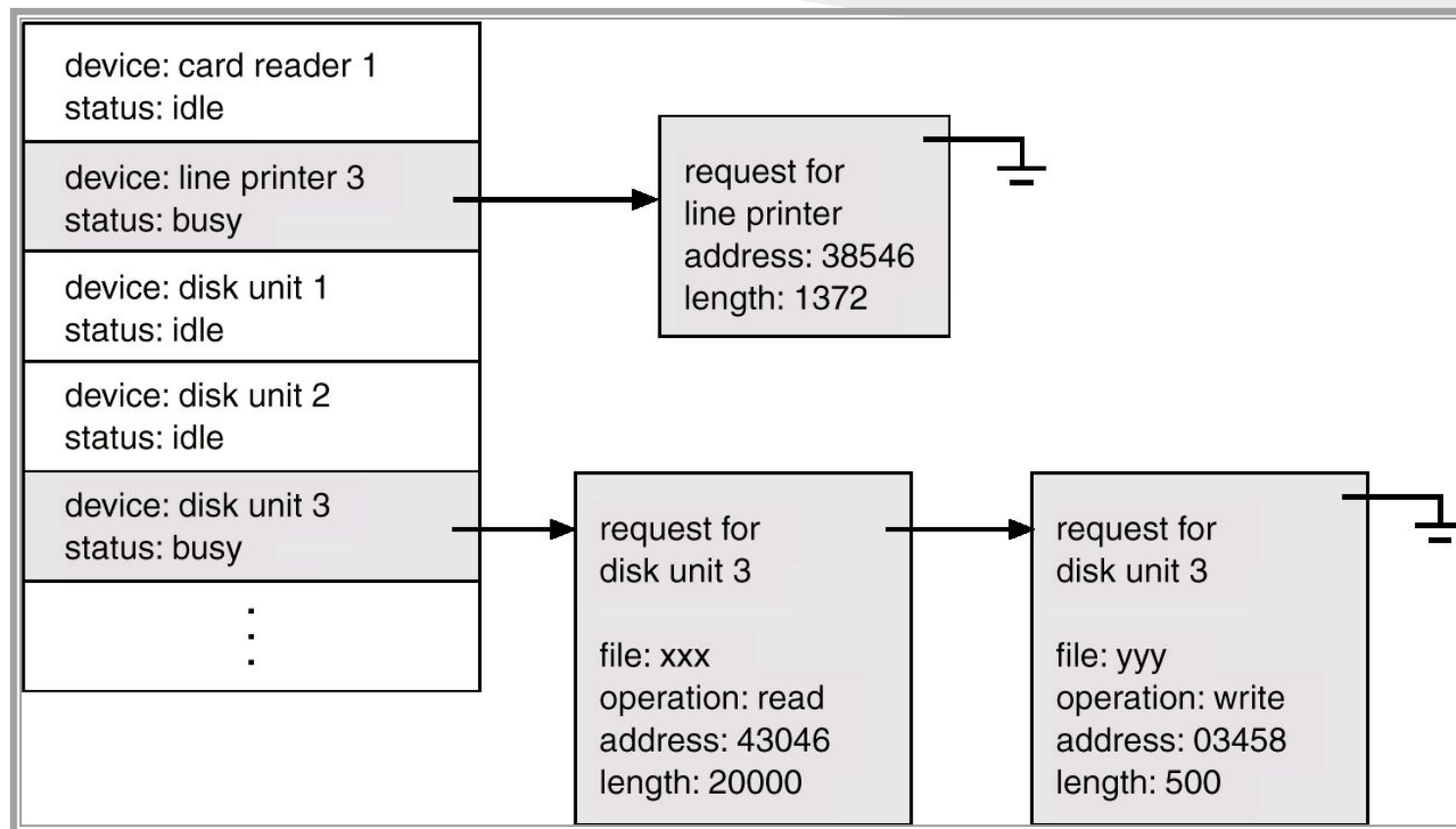


Asynchronous

After I/O starts, control returns to user program without waiting for I/O completion



Device-State Table



נדרש במוד הא-סינכרוני וגם בסינכרוני כאשר יש יותר ממעבד אחד

ניהול הזיכרון

חשיבות הזיכרון:

- ה- instruction חייבות להיות בזיכרון על-מנת שנוכל לבצע אותן
 - ה- data חייב להיות בזיכרון על-מנת שנוכל לעבד אותו
 - לאחר עיבודו חוזר שוב לזיכרון
- ניהול הזיכרון עוסק בהחלטה של מה יהיה/ישהה בזיכרון בכל רגע ורגע:
- מרכיב קריטי לניצולת (utilization) של המעבד ובעל השפעה מכרעת על זמן התגובה למשתמש
 - תפקיד מערכת ההפעלה בכל הקשור לניהול זיכרון:
 - מעקב ורישום החלקים בזיכרון שנמצאים כרגע בשימוש
 - החלטה על העברת תוכן זיכרון המשוויך לתהליכים מ/אל הזיכרון (swapping)
 - הקצאת זיכרון לתהליכים (Allocation and deallocating)

Mass-Storage Management

- Main memory – only large storage media that the CPU can access directly
- Why using disks?
 - Store data that does not fit in main memory
 - Store data that must be kept for a “long” period of time
- Proper management is of central importance
- Entire speed of computer operation hinges on disk subsystem and its algorithms
- OS activities:
 - Free-space management
 - Storage allocation
 - Disk scheduling

Mass-Storage Management

Some storage need **not** to be fast

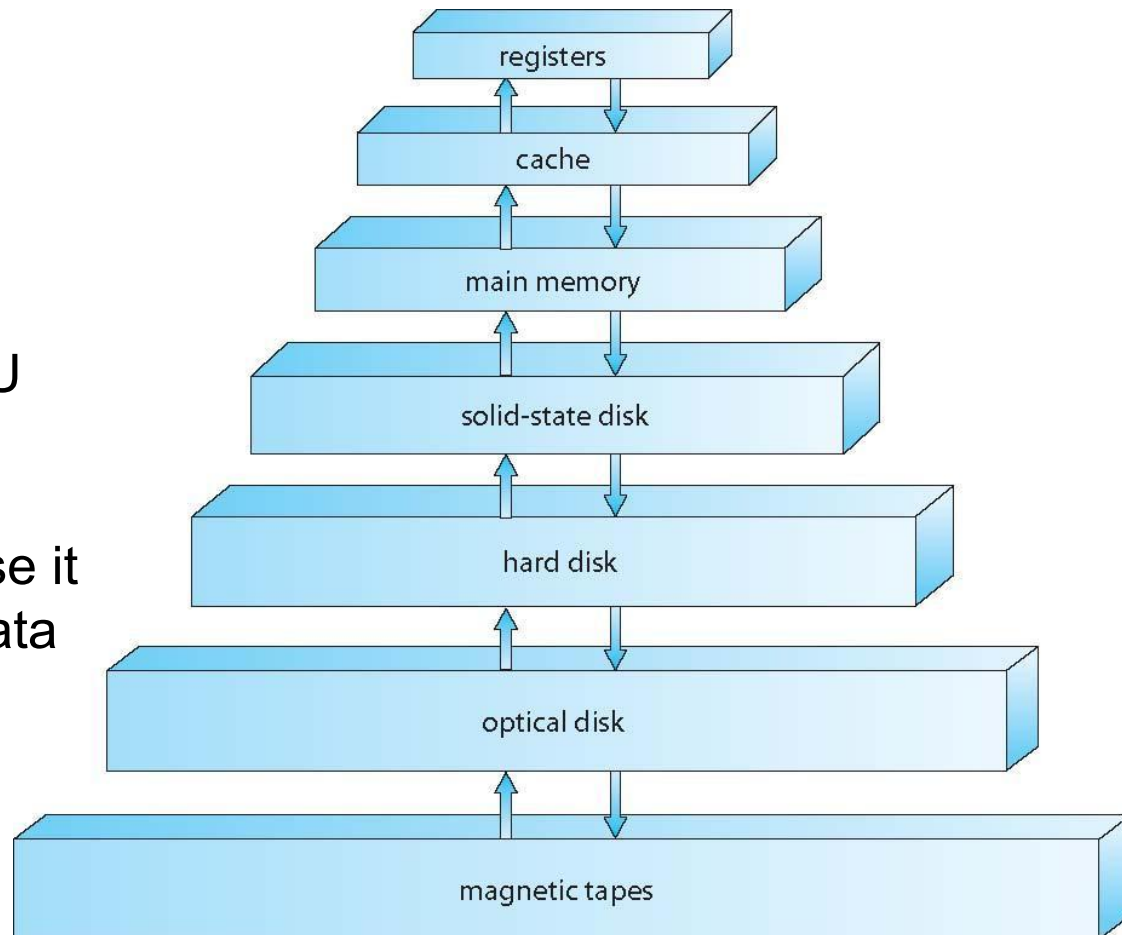
- Includes optical storage, magnetic tape
- Not critical to the computer performance but still must be managed
- Varies between WORM (write once, read many) and RW (Read / Write)

Storage Hierarchy

Storage systems organized in hierarchy:

- Speed
- Cost
- Volatility

It takes some time (several CPU cycles) to read/write to main memory – in the meantime the processor needs to stall because it does not have the necessary data



ניהול Storage

בנוסף לניהול הקצאת מקום בהתקני ה- Storage מספקת מ"ה גם ממשקים לישות הלוגית file ובכלל זאת:

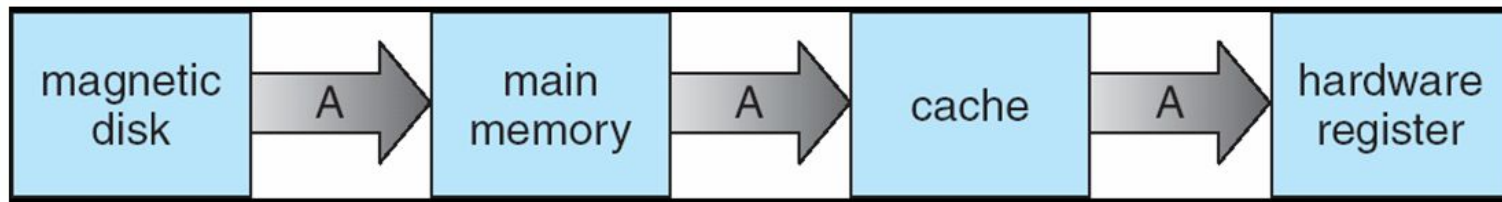
- יצירת ותחזוקת קבצים
- כלים לעריכת שינוי תוכן של קבצים
- איגוד קבצים במחיצות ומעבר בין רמות שונות ב-

directory

- ניהול הרשאות הגישה לקבצים
- גיבוי קבצים

Caching

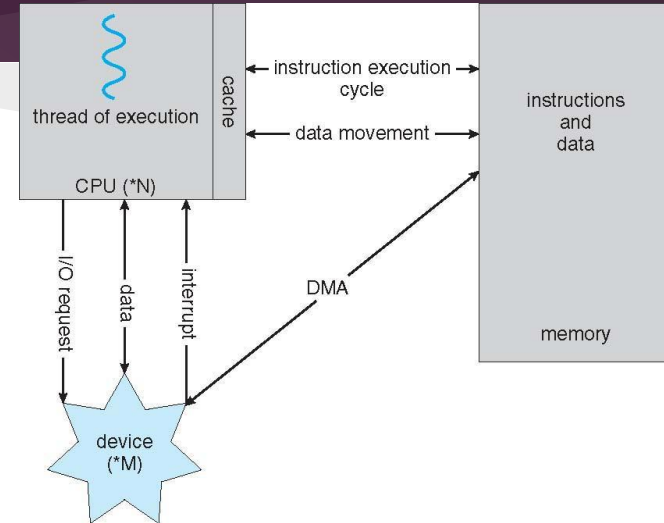
Caching – copying information into faster storage system; main memory can be viewed as a last *cache* for secondary storage



- Important principle, performed at many levels in a computer (in hardware, operating system, software)
- Faster storage (cache) checked first to determine if information is there
 - If it is, information used directly from the cache (fast)
 - If not, data copied to cache and used there
- Multiprocessors environment must provide cache coherency in hardware such that all CPUs have the most recent value on their cache

Direct Memory Access

- Used for high-speed I/O devices able to transmit information at close to memory speeds
 - Like: tape, disk...
 - Not like: keyboard...
- Device controller transfers blocks of data from buffer storage directly to main memory without CPU intervention
- Only one interrupt is generated per block, rather than the one interrupt per byte



DMA

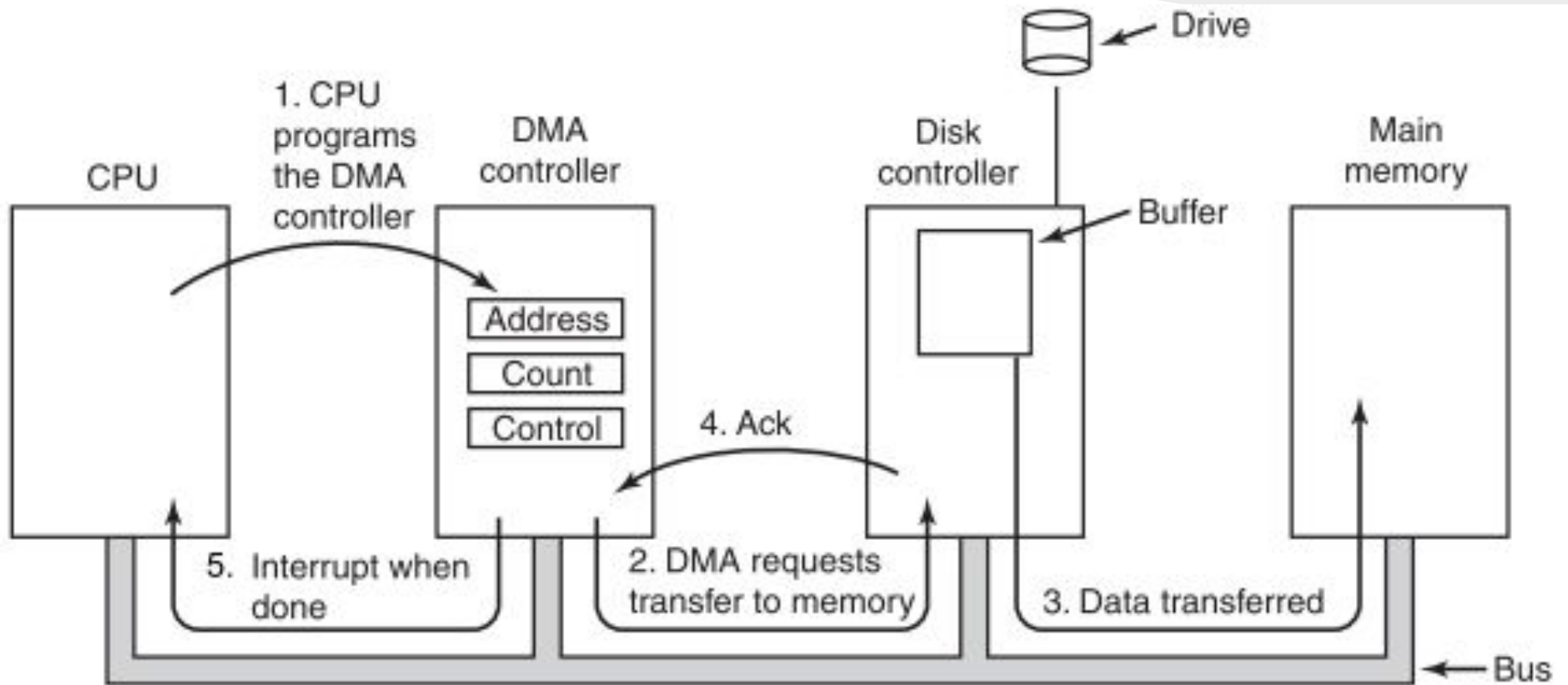


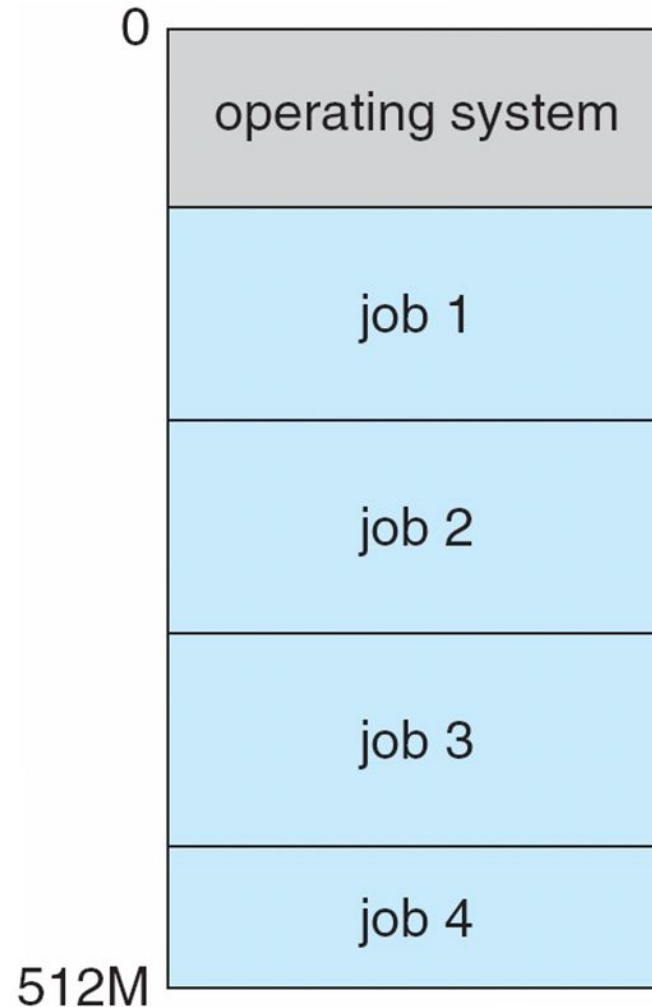
Figure 5-4. Operation of a DMA transfer.

Multiprogramming

הבעיה

- כאשר יש לנו משתמש בודד, הוא לא יכול לשמור את המעבד וההתקנים השונים בניצולת מלאה (למה?)
- אידיאלית היינו רוצים שבכל רגע נתון תהיה לנו תכנית שתרוץ על המעבד
- השימוש ב- multiprogramming בא לפתור את הבעיה:
- נאפשר הרצה של מספר תכניות במקביל (יותר מכמות המעבדים)
- בצורה זו יש סיכוי גבוה שאם מעבד מתפנה (בעקבות יציאה ל- I/O או סיום הרצה) תהיה תכנית שיכולה לתפוס את המעבד
- מחייב לשמור מספר תכניות בזיכרון ולנהל תהליך של

.Mem layout for Multiprog



Timesharing

Timesharing (multitasking) is logical extension in which CPU switches jobs so frequently that users can interact with each job while it is running, creating interactive computing

- **Response time** should be < 1 second
- Each user has at least one program executing in memory
 - process
- If several jobs ready to run at the same time □ **CPU scheduling**
- If processes don't fit in memory, **swapping** moves them in and out to run
- **Virtual memory** allows execution of processes not completely in memory

User / Kernel Mode

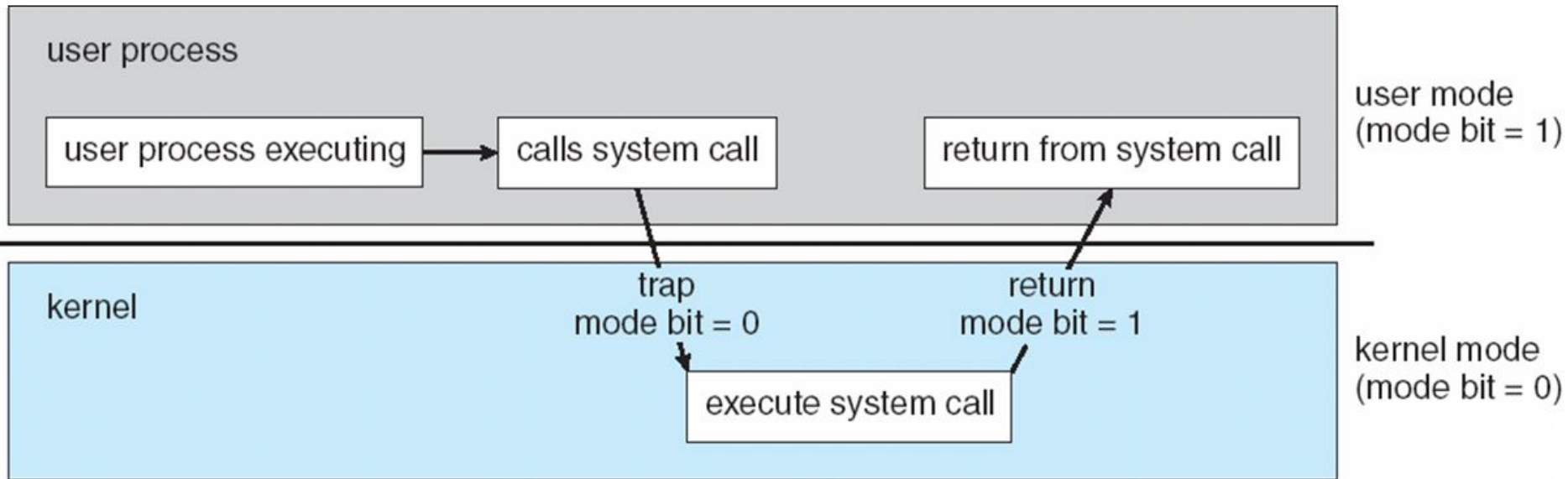
הבעיה – מ"ה נדרשת להגן על עצמה ומרכיבי מערכת נוספים (למשל התקנים) מפני שימוש לא נכון או זדוני
הפתרון – פעולה בתצורת Dual Mode:

User mode and **kernel mode**

Mode bit provided by hardware

- Provides ability to distinguish when system is running user code or kernel code
- Some instructions designated as **privileged**, only executable in kernel mode
- System call changes mode to kernel, return from call resets it to user

User / Kernel Mode



Example

Which of the following instructions should be privileged?

- Read the clock.
- Clear memory.
- Issue a trap instruction.
- Turn off interrupts.
- Modify entries in device-status table.
- Switch from user to kernel mode.
- Access I/O device.

Example

Which of the following instructions should be privileged?

- Read the clock.
- **Clear memory.**
- Issue a trap instruction.
- **Turn off interrupts.**
- **Modify entries in device-status table.**
- Switch from user to kernel mode.
- **Access I/O device.**

שאלה?

למה משמש DMA? עבור אילו התקנים לא כדאי להשתמש ב-DMA?

?שאלה

למה משמש DMA?
להעברה ישירה של נתונים מהבפר הלוקאלי של התקנים חיצוניים לזיכרון ללא התערבות המעבד.

עבור אילו התקנים לא כדאי להשתמש ב-DMA?
לא כדאי להשתמש עבוד התקנים בעלי קצב נמוך כגון מקלדת

שאלה?

ידוע כי העיצוב והמטרות התכנוניות של מערכת ההפעלה נקבעים על-פי מערכת המחשב עליה היא מתוכננת לעבוד. בחר 3 מערכות מחשב השונות מבחינת מטרות מערכת ההפעלה המותקנות עליהן. כיצד שונות המטרות התכנוניות של מ"ה למערכות השונות שבחרת ומדוע?

שאלה?

3 מערכות מחשב שונות מבחינת מטרות מערכת ההפעלה:

Mainframe – ניצולת מרבית של משאבי המערכת

PC - הרצת תכניות משתמש, נוחות הפעלה

Handheld – ניצול סוללה, נוחות הפעלה בסביבה מוגבלת (מסך, מקלדת...)