DOS and BIOS Interrupts

- Useful subroutines within BIOS and DOS that are available through the INT (interrupt) instruction.
- The INT instruction is like a FAR call. It saves CS:IP and flags on the stack and executes the subroutine with it.
- It has the format: `INT xx` ; xx can be any value from 00 to FF (256 interrupts).
- BIOS interrupts are burned into the BIOS ROM of the 80x86-based systems.

**INT 10H – Video Screen**

- The option is chosen by putting a specific value in register AH.
- The video screen is text mode is divided into 80 columns and 25 rows.
- A row and column number are associated with each location on the screen with the top left corner as 00,00 and the bottom right corner as 24,79. The center of the screen is at 12,39 or (0C,27 in hex).
- Specific registers has to be set to specific values before invoking INT 10H.

**Function 06 – clear the screen**

| AH = 06 | ; function number |
| AL = 00 | ; page number |
| BH = 07 | ; normal attribute |
| CH = 00 | ; row value of start point |
| CL = 00 | ; column value of start point |
| DH = 24 | ; row value of ending point |
| DL = 79 | ; column value of ending point |

**Function 02 – setting the cursor to a specific location**

| AH = 06 | ; function number |
| DH = row | ; cursor |
| DL = column | ; position |

**Function 03 – get the current cursor position**

| AH = 03 | ; function number |
| BH = 00 | ; currently viewed page |

The position is returned in DH = row and DL = column.

**Function 0E – output a character to the screen**

| AH = 0E | ; function number |
| AL = Character to be displayed |
| BH = 00 | ; currently viewed page |
| BL = 00 | ; default foreground color |
DOS INT 21H

Useful DOS interrupt to input information from the keyboard and display it on the screen

**Function 09 – outputting a string of data to the monitor**
- **AH = 09** ; function number
- **DX** = offset address of the ASCII data to be displayed, data segment is assumed
- The ASCII string must end with the dollar sign $

**Function 02 – outputting a single character to the monitor**
- **AH = 02** ; function number
- **DL** = ASCII code of the character to be displayed

**Function 01 – inputting a single character, with an echo**
- **AH = 01** ; function number
- After the interrupt **AL** = ASCII code of the input and is echoed to the monitor

**Function 0A – inputting a string of data from the keyboard**
- **AH = 0A** ; function number
- **DX** = offset address at which the string of data is stored (buffer area), data segment is assumed and the string must end with <RETURN>
- After execution:
  - **DS:DX** = buffer in bytes (n characters + 2)
  - **DS:DX+1** = number of entered characters excluding the return key
  - **DS:DX+2** = first character input
  - . . .
  - **DS:DX+n** = last character input

To set a buffer, use the following in the data segment:
- **Buffer DB 10, ?, 10 DUP(FF)**

**Function 07 – inputting a single character from the keyboard without an echo**
- **AH = 07** ; function number
- Waits for a single character to be entered and provides it in **AL**

INT16 – Keyboard Programming

**Function 01 – check for a key press without waiting for the user**
- **AH = 01**
- Upon execution **ZF** = 0 if there is a key pressed

**Function 00 – keyboard read**
- **AH = 00**
- Upon execution **AL** = ASCII character of the pressed key
- Note this function must follow function 01