Microprocessor and Interfacing – CSE2006

Module 3 – Advanced ALP

1. Interrupt Calls

**What are interrupts?**

Interrupts are instructions that stop the program sequence and perform the tasks given by the peripheral devices or the software. What would you do when you get a WhatsApp message when you are listening to a lecture? Which interrupt would you choose? Knock on the door or a phone call? Interrupts have priority, some interrupts are more important than the other.

They cause temporary stop in the execution of the program. When the microprocessor is executing some program, an interrupt breaks the sequence and diverts the sequence to some other program called Interrupt Service Routine (ISR). Once the ISR is tended to, the control is then returned to the main program.

Interrupts are needed to connect peripheral devices that have lower data rate than the CPU. When an external signal is applied by a peripheral device, it is called hardware interrupt. When the interrupt is received from the instructions in the running program, it is called software instruction. Another form of interrupt is interrupt due to some error conditions. When error conditions, occur, the CPU halts the program so that the error can be rectified. Figure 1 shows the hardware interrupt given by the keyboard.

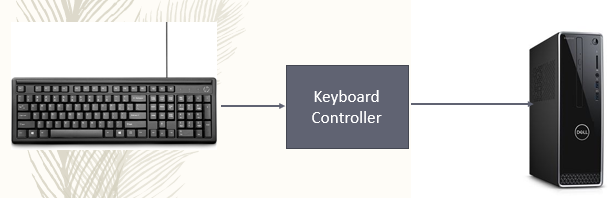


Fig. 1 Hardware interrupt from Keyboard.

**DOS and BIOS interrupts**

Figure 2 shows the interrupt process of DOS and BIOS function calls that arises from an external source such as peripheral devices to the application program running in the CPU.

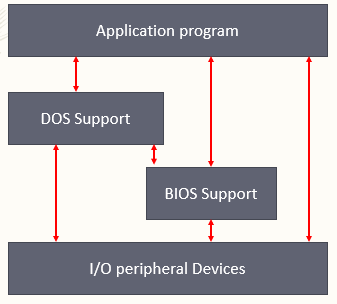


Fig. 2. DOS and BIOS function calls from peripheral device to the CPU program.

**DOS Interrupts**

When MS-DOS is loaded in the computer INT 21H is used to provide an interrupt to perform some important functions. These are called DOS INT 21H function calls. Main function calls are data input/output through keyboard

* Example:

MOV AH, 01 🡪 to input one character

INT 21H 🡪 invoke interrupt

**DOS (Disk Operating System) Interrupts**

1. Function Call 01 🡪 Read the keyboard
   * Input parameter AH = 01 🡪 Read a character from keyboard, echo it on the CRO screen and return the ASCII code of the key pressed in AL
   * Output parameter AL = ASCII code of charater
2. Function Call 02 🡪 Display on CRT screen
   * Input parameter AH = 02
   * Dl 🡪 ASCII character to be displayed in the screen
3. Function Call 03 🡪 Read character from COM 1
   * Input parameter AH = 03H 🡪 Reads data from COM port
   * Output parameter Al = ASCII code of character
4. Function Call 04 🡪 Write character to COM 1
   * Input parameter AH = 04H 🡪 Writes data to COM port
   * Output Parameter 🡪 ASCII code of character to be written to COM port.
5. Function Call 05 🡪 Write to printer attached to Lpt1
   * Input parameter AH = 05H 🡪 Writes data to printer port
   * Dl 🡪 ASCII code of character to be printed
   * Output parameter 🡪 Print the character available in DI to the printer attached to Lpt1.
6. Function Call 09 🡪 Display a character string
   * Input parameter AH = 09H
   * ds:dx Address of character string
7. Function call 0ah 🡪 Buffer keyboard input
   * Input parameter AH = 09H
   * ds:dx Address of keyboard input buffer

**BIOS Interrupts**

The INT 10H subroutines are burned into the ROM BIOS of the 8086 and the compatible hardware and instructions are used to communicate with the user through the screen.

* Example 🡪 changing colors of characters, background, changing cursor locations, clearing screen.

**BIOS (Basic Input/Output System) Interrupts**

1. Function Call 00 – Select Video mode
   * Here Al 🡪 mode Number, Ah = 00
   * Al = 00 🡪 40 x 25 black and white
   * Al = 04 🡪 320 x 200 colour
   * Al = 10H 🡪 640 x 350 x 16 colour
2. Function call 03 🡪 Read cursor position
   * Ah = 03, Bh = Page number (Input Parameters)
3. Function call 0E 🡪 Write character on CRT screen and advance cursor
4. INT 11H 🡪 Determine the type of equipment installed
5. INT 14H 🡪 To control the serial communication port attached to the computer
6. INT 16H 🡪 Keyboard interrupt

**Weblinks**

* <https://www.slideshare.net/VijayKumar486/8086-interrupts-with-dos-and-bios-by-vijay/60>
* <http://www.ece.lsu.edu/desouza/Classes/EE3751/Lab%203.pdf>
* <http://faculty.uml.edu/yluo/teaching/microprocessorii/resources/interrupts.pdf>