

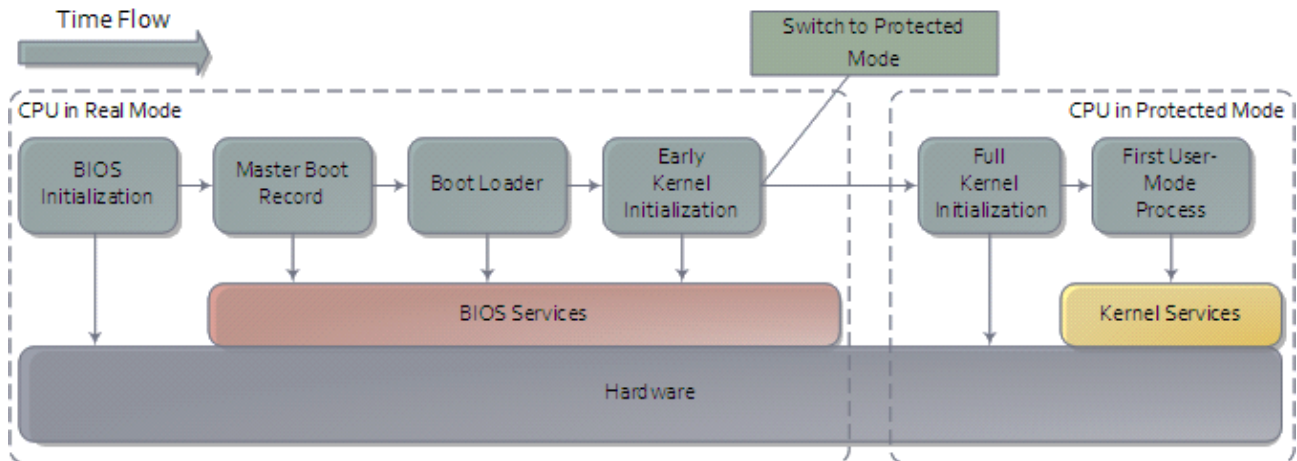
## BOOT SEQUENCE

### LAB 2

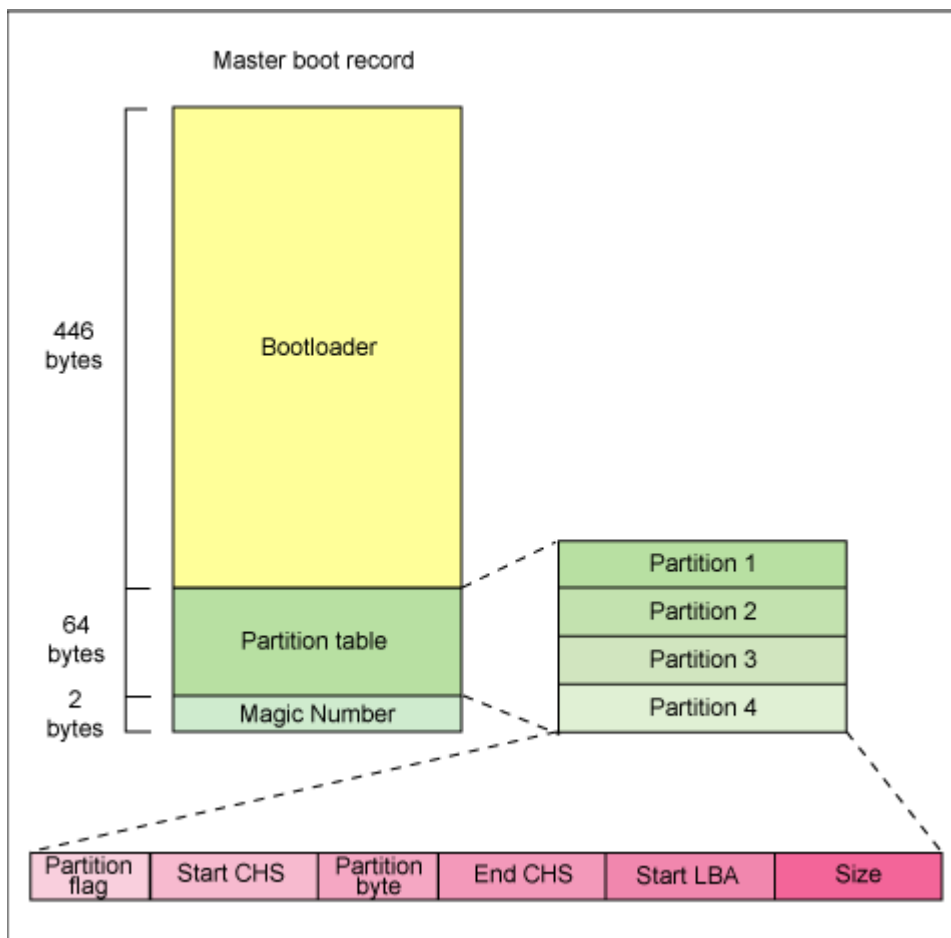
**Aim:** To understand the functioning of the boot loader and to implement the same.

#### THEORY:

##### Boot sequence



##### Boot loader



**BOOT LOADER 1: Blank black screen****CODE**

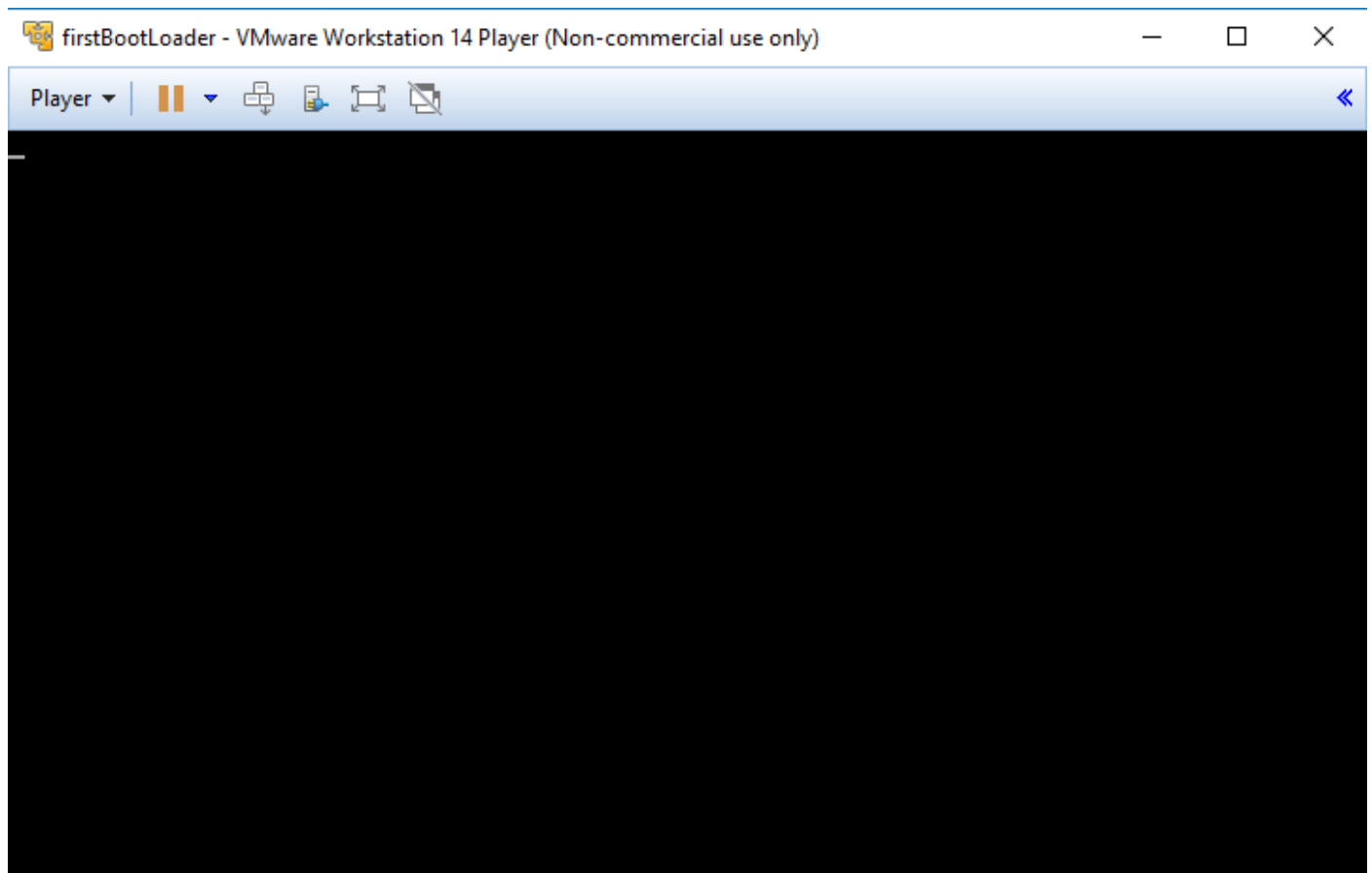
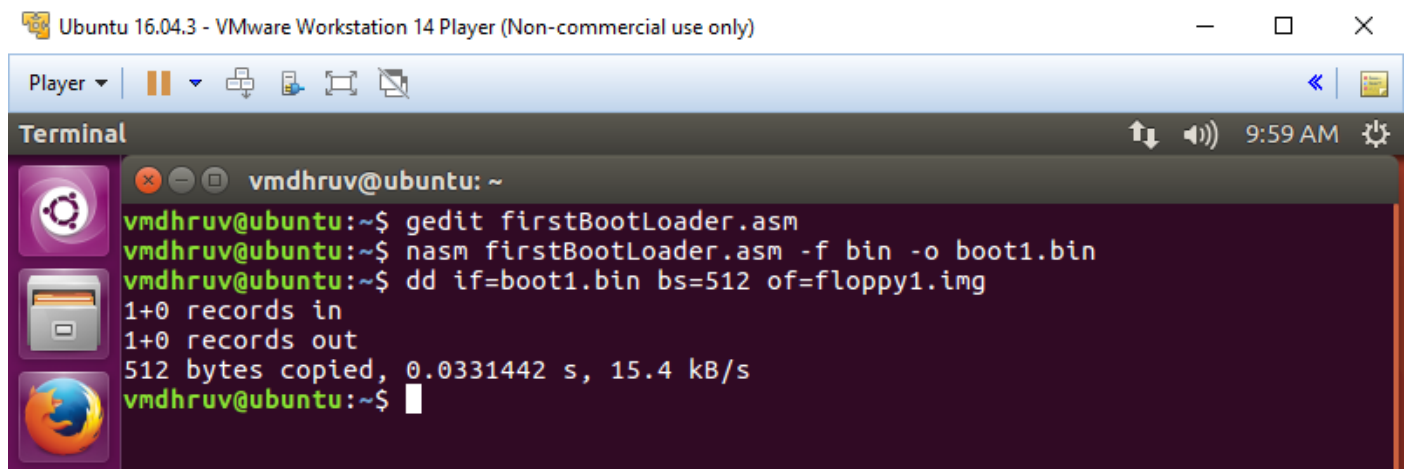
```
[BITS 16]          ;tell the assembler that its a 16 bit code

[ORG 0x7C00]       ;Origin, tell the assembler that where the code will be in memory after it is been loaded

JMP $              ;infinite loop

TIMES 510 - ($ - $$) db 0      ;fill the rest of sector with 0

DW 0xAA55           ; add boot signature at the end of bootloader
```

**SCREENSHOTS**

**BOOT LOADER 2: Prints character 'A' on the screen****CODE**

```

[BITS 16]          ;Tells the assembler that its a 16 bit code

[ORG 0x7C00]       ;Origin, tell the assembler that where the code will be in memory after it is been loaded

MOV AL, 65

CALL PrintCharacter

JMP $              ;Infinite loop, hang it here.

PrintCharacter:    ;Procedure to print character on screen
                  ;Assume that ASCII value is in register AL

MOV AH, 0x0E       ;Tell BIOS that we need to print one charater on screen.

MOV BH, 0x00       ;Page no.

MOV BL, 0x07       ;Text attribute 0x07 is lightgrey font on black background

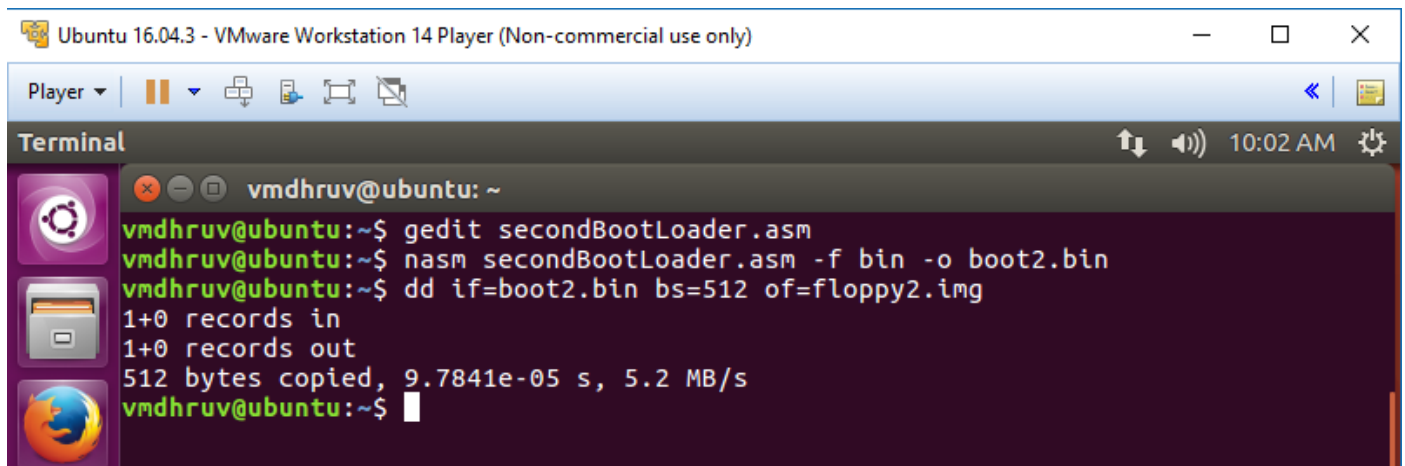
INT 0x10           ;Call video interrupt

RET                ;Return to calling procedure

TIMES 510 - ($ - $$) db 0          ;Fill the rest of sector with 0

DW 0xAA55                ;Add boot signature at the end of bootloader

```

**SCREENSHOTS**


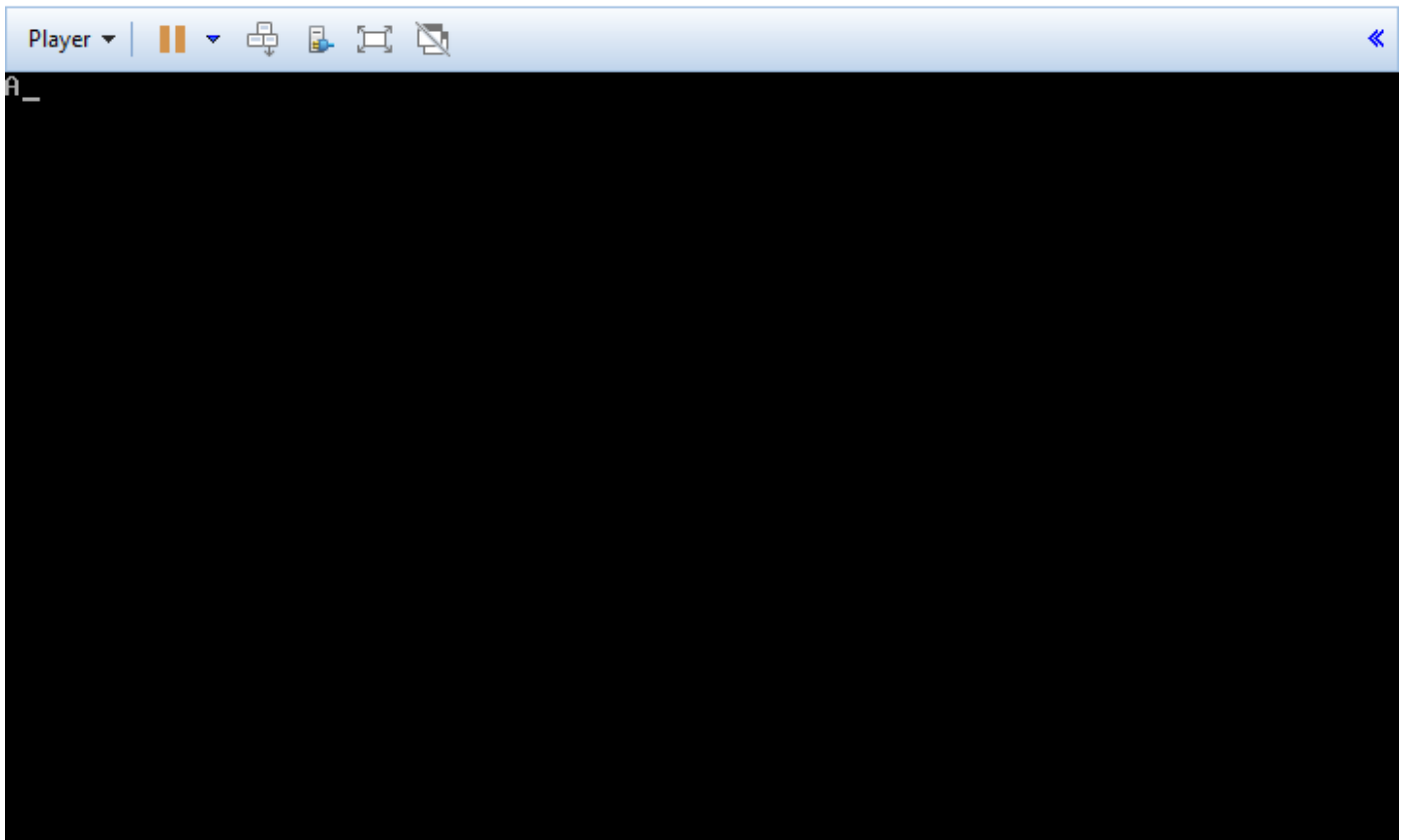
```

Ubuntu 16.04.3 - VMware Workstation 14 Player (Non-commercial use only)

Terminal

vmdhruv@ubuntu: ~
vmdhruv@ubuntu:~$ gedit secondBootLoader.asm
vmdhruv@ubuntu:~$ nasm secondBootLoader.asm -f bin -o boot2.bin
vmdhruv@ubuntu:~$ dd if=boot2.bin bs=512 of=floppy2.img
1+0 records in
1+0 records out
512 bytes copied, 9.7841e-05 s, 5.2 MB/s
vmdhruv@ubuntu:~$

```



### BOOT LOADER 3: Prints 'Hello world'

#### CODE:

```
[BITS 16]
jmp main
nop
main:
    mov ax, 07C0h
    add ax, 288
    mov ss, ax
    mov sp, 4096
    mov ax, 07C0h
    mov ds, ax
    call GetPressedKey
    jmp .InfiniteLoop
.InfiniteLoop:
    jmp .InfiniteLoop
HelloWorld    db    "Hello world",0x0d,0x0a,0x00
```

GetPressedKey:

```
    mov ah, 0
    int 0x16
    CALL PrintHelloWorld
    ret
```

PrintHelloWorld:

```
    mov si, HelloWorld
    call PrintStr
    ret
```

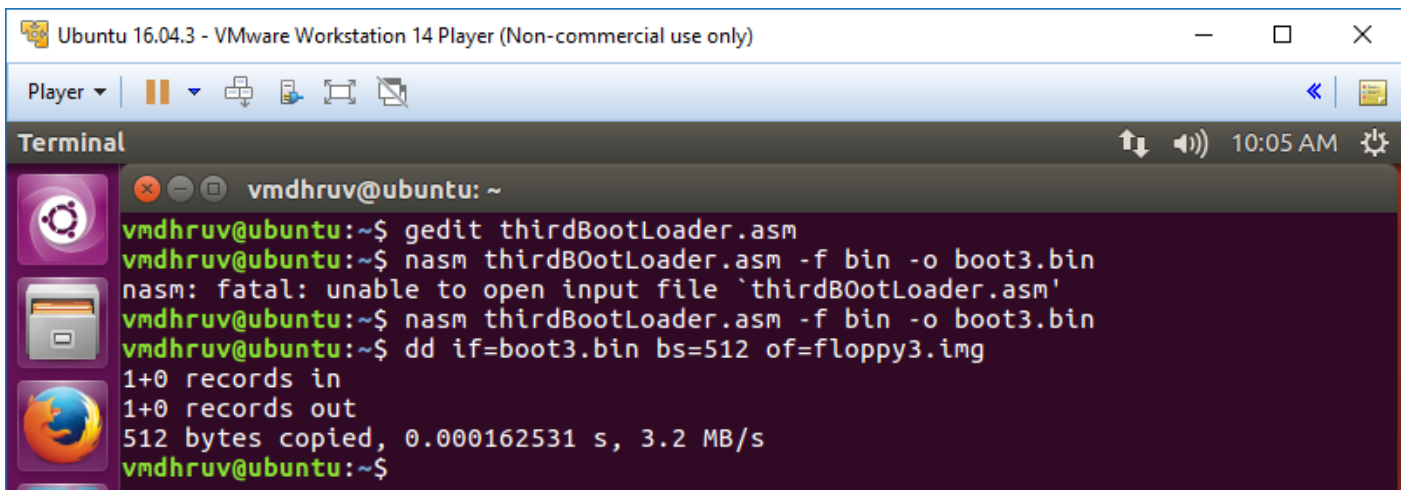
PrintStr:

```
    push ax
    mov ah, 0Eh
.loop:
    lodsb
    cmp al, 0x00
    je .done
    int 10h
    jmp .loop
.done:
    pop ax
    ret
```

times 510-(\$-\$\$) db 0

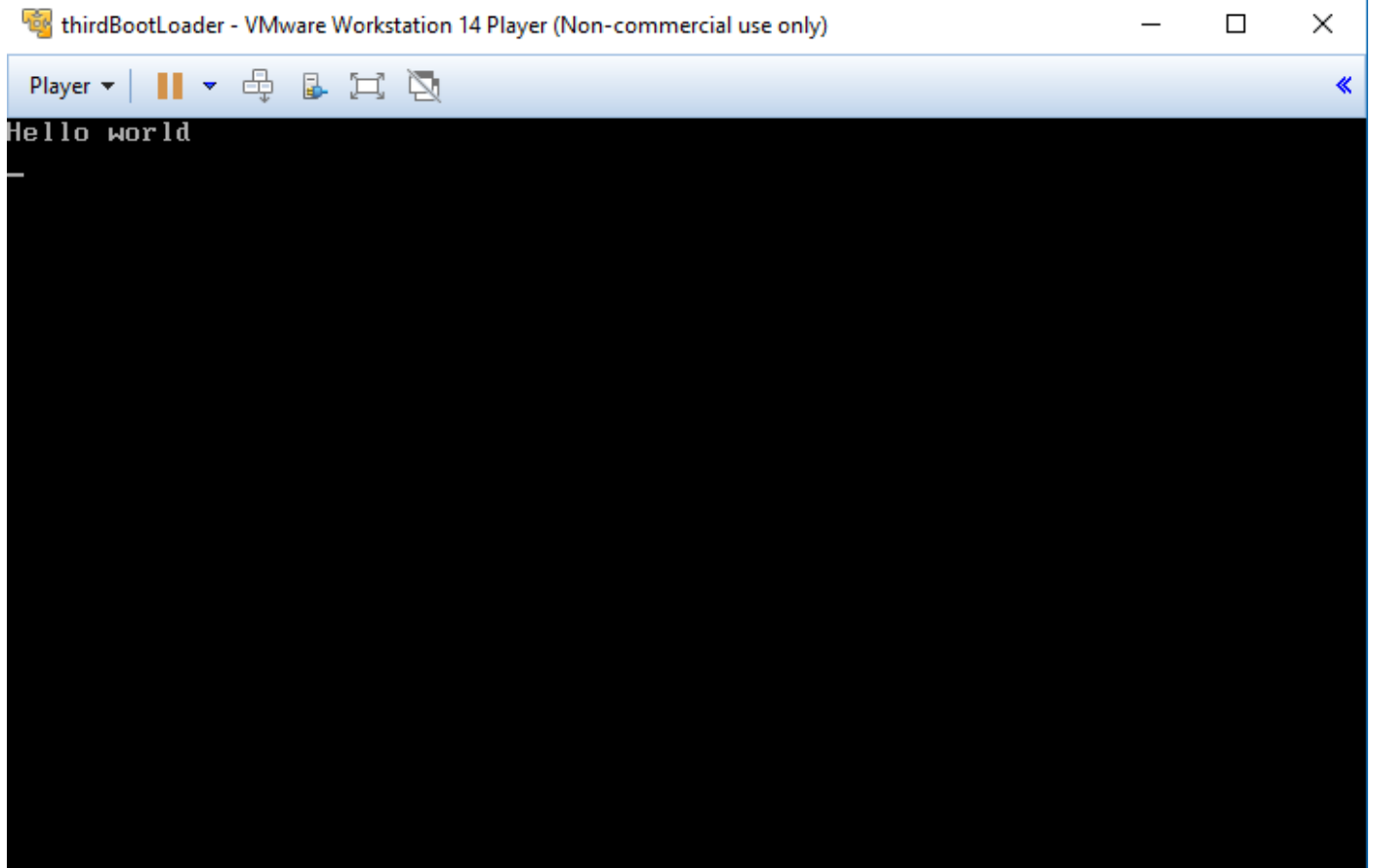
dw 0xAA55

## SCREENSHOTS



```

Ubuntu 16.04.3 - VMware Workstation 14 Player (Non-commercial use only)
Player ▾ | [Icons] | 10:05 AM
Terminal
vmdhruv@ubuntu: ~
vmdhruv@ubuntu:~$ gedit thirdBootLoader.asm
vmdhruv@ubuntu:~$ nasm thirdB0otLoader.asm -f bin -o boot3.bin
nasm: fatal: unable to open input file `thirdB0otLoader.asm'
vmdhruv@ubuntu:~$ nasm thirdBootLoader.asm -f bin -o boot3.bin
vmdhruv@ubuntu:~$ dd if=boot3.bin bs=512 of=floppy3.img
1+0 records in
1+0 records out
512 bytes copied, 0.000162531 s, 3.2 MB/s
vmdhruv@ubuntu:~$
  
```



#### BOOT LOADER 4: Prints 'Hello, Dhruv Garg'

##### CODE

```
[BITS 16]

jmp main

nop

main:

    mov ax, 07C0h
    add ax, 288
    mov ss, ax
    mov sp, 4096
    mov ax, 07C0h
    mov ds, ax
    call GetPressedKey
    jmp .InfiniteLoop

.InfiniteLoop:
    jmp .InfiniteLoop

HelloUser    db    "Hello, Dhruv Garg!",0x0d,0x0a,0x00
```

GetPressedKey:

```
    mov ah, 0
    int 0x16
    CALL PrintHelloUser
    ret
```

PrintHelloUser:

```
    mov si, HelloUser
    call PrintStr
    ret
```

PrintStr:

```
    push ax
    mov ah, 0Eh
.loop:
    lodsb
    cmp al, 0x00
    je .done
    int 10h
    jmp .loop
.done:
    pop ax
    ret
```

times 510-(\$-\$\$) db 0

dw 0xAA55

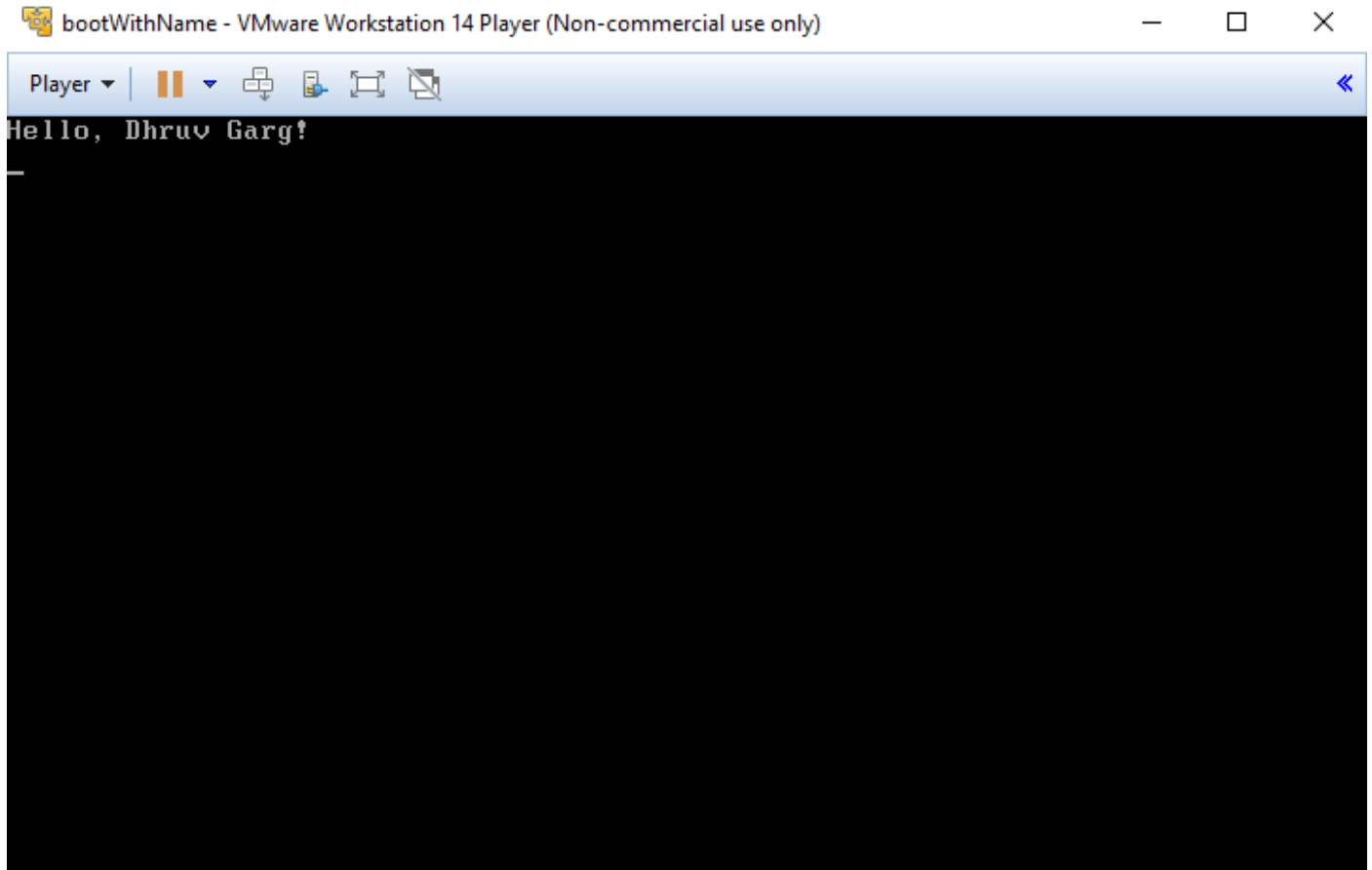
## SCREENSHOTS

Ubuntu 16.04.3 - VMware Workstation 14 Player (Non-commercial use only)

Player ▾ | [Icons] | 10:09 AM [Settings]

**Terminal**

```
vmdhruv@ubuntu: ~
vmdhruv@ubuntu:~$ gedit bootWithName.asm
vmdhruv@ubuntu:~$ nasm bootWithName.asm -f bin -o boot4.bin
vmdhruv@ubuntu:~$ dd if=boot4.bin bs=512 of=floppy4.img
1+0 records in
1+0 records out
512 bytes copied, 0.000493961 s, 1.0 MB/s
vmdhruv@ubuntu:~$
```



**SCREENSHOT:** Different bootloaders created

