Hands-on Session

- Communicate with FPGA peripheral within Linux

Exercise 8: Using FPGA Peripherals within Linux

- We will use the red LEDs and the slider switches
- The program copies the value of the switches to the LEDs

MMAP

```
// Open /dev/mem
if( ( fd = open( "/dev/mem", ( O_RDWR | O_SYNC ) ) ) == -1 ) {
    printf( "ERROR: could not open \"/dev/mem\"...\n" );
    return( 1 );
}

// get virtual addr that maps to physical
virtual_base = mmap( NULL, HW_REGS_SPAN, ( PROT_READ | PROT_WRITE ),
   MAP_SHARED, fd, HW_REGS_BASE );

if( virtual_base == MAP_FAILED ) {
    printf( "ERROR: mmap() failed...\n" );
    close( fd );
    return(1);
}
```

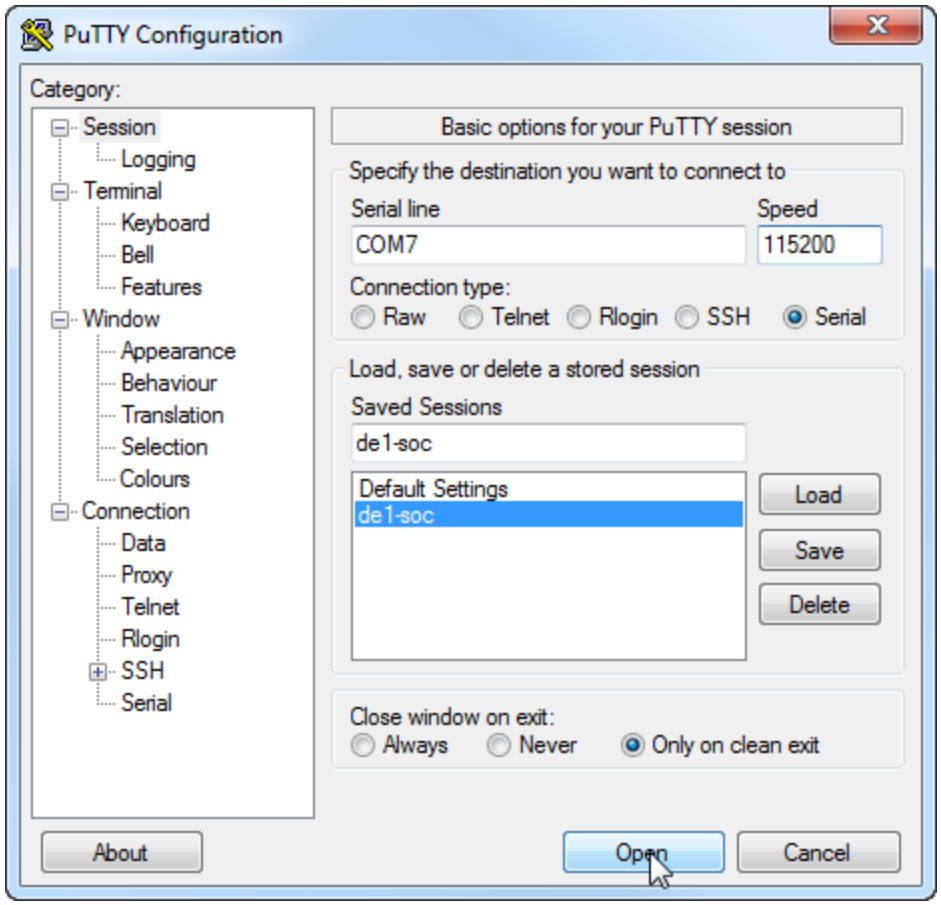
Using the Virtual Address

```
// Get the address that maps to the LEDs
h2p_lw_led_addr=(unsigned int *)(virtual_base + (( LED_PIO_BASE ) & (
   HW_REGS_MASK ) ));
h2p_lw_sw_addr=(unsigned int *)(virtual_base + (( SW_PIO_BASE ) & (
   HW_REGS_MASK ) ));

while(!stop){
    *h2p_lw_led_addr = *h2p_lw_sw_addr;
}
```

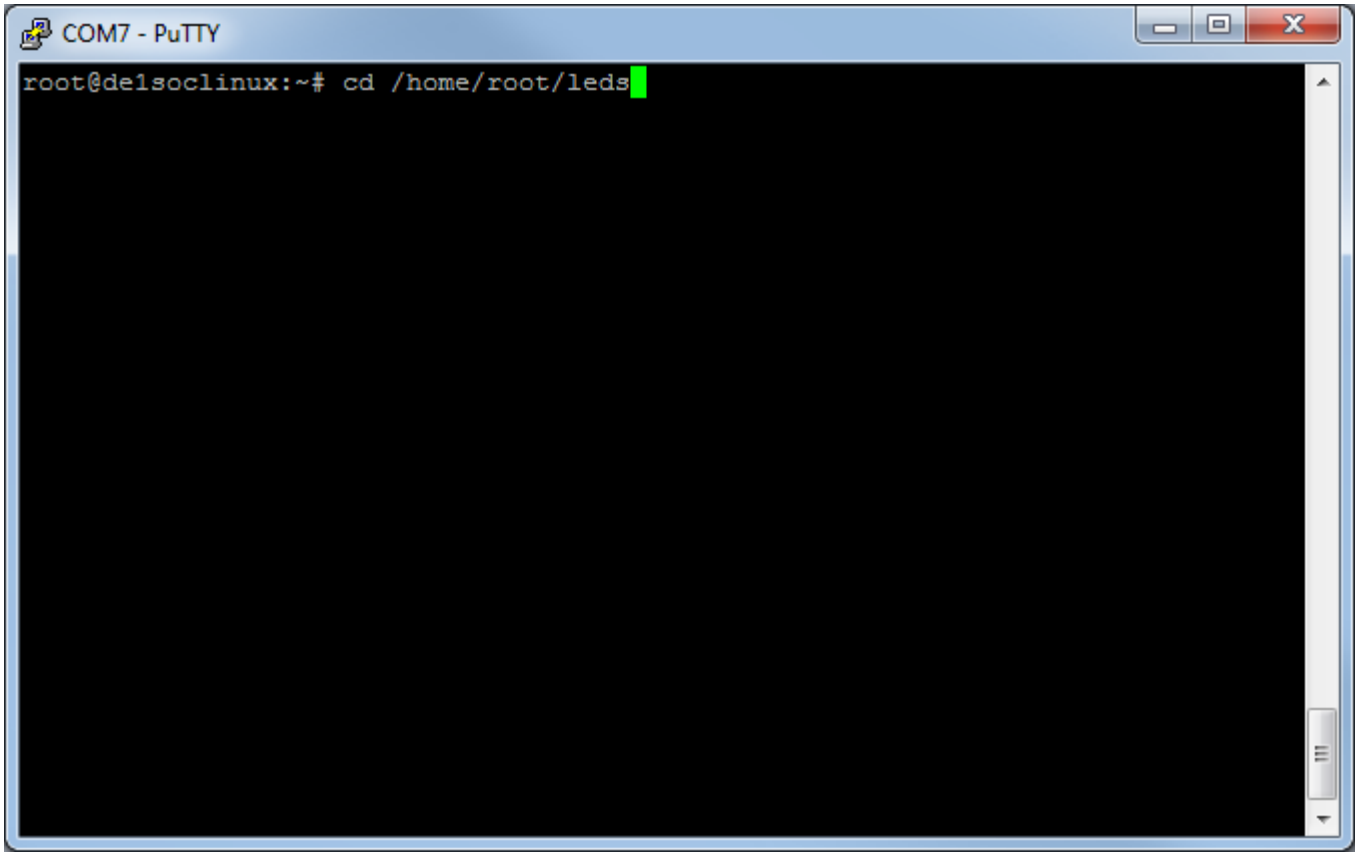
MUNMAP

```
if( munmap( virtual_base, HW_REGS_SPAN ) != 0 ) {
    printf( "ERROR: munmap() failed...\n" );
    close( fd );
    return( 1 );
}

close( fd );
```

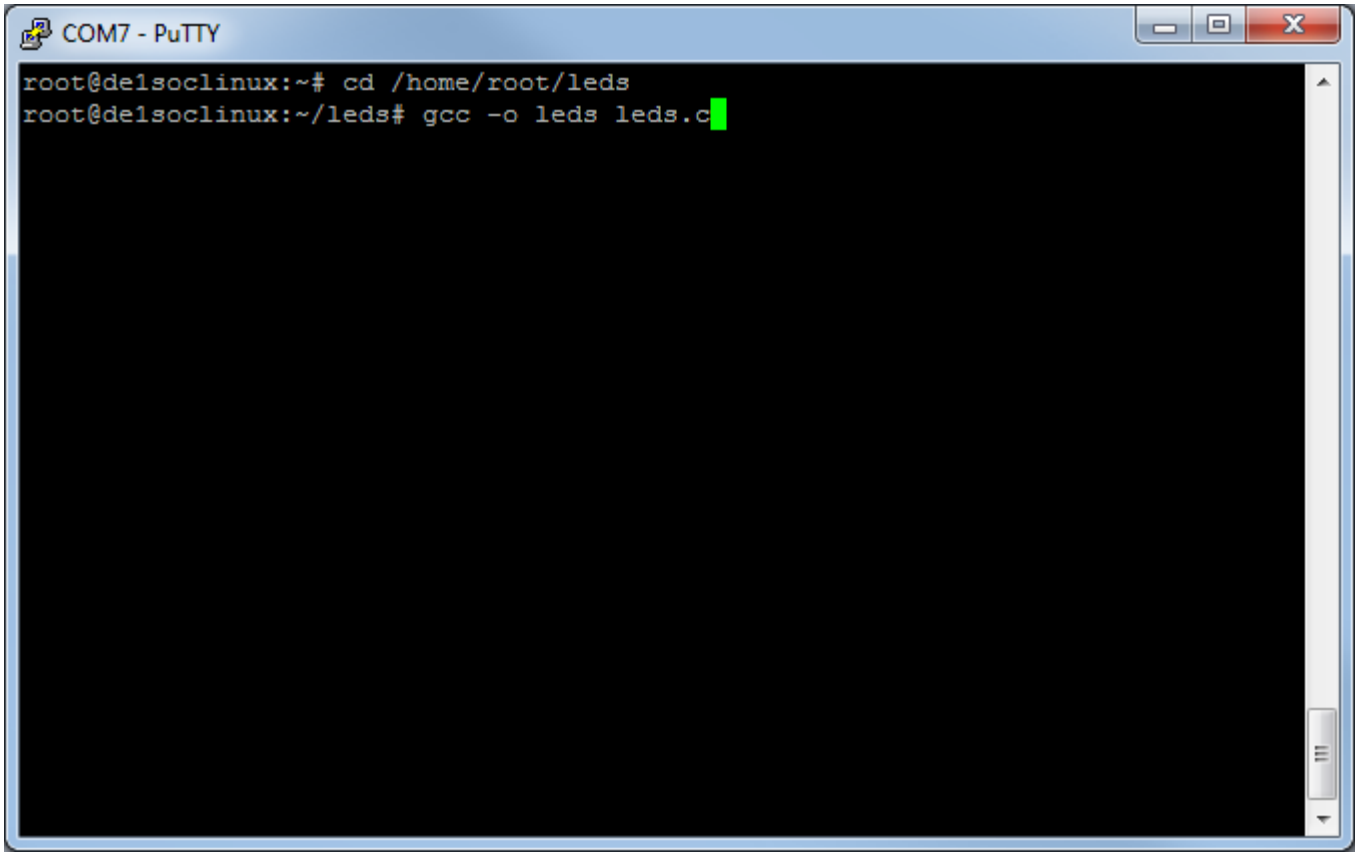
Step 1: Open the Connection to the DE1-SoC using Putty



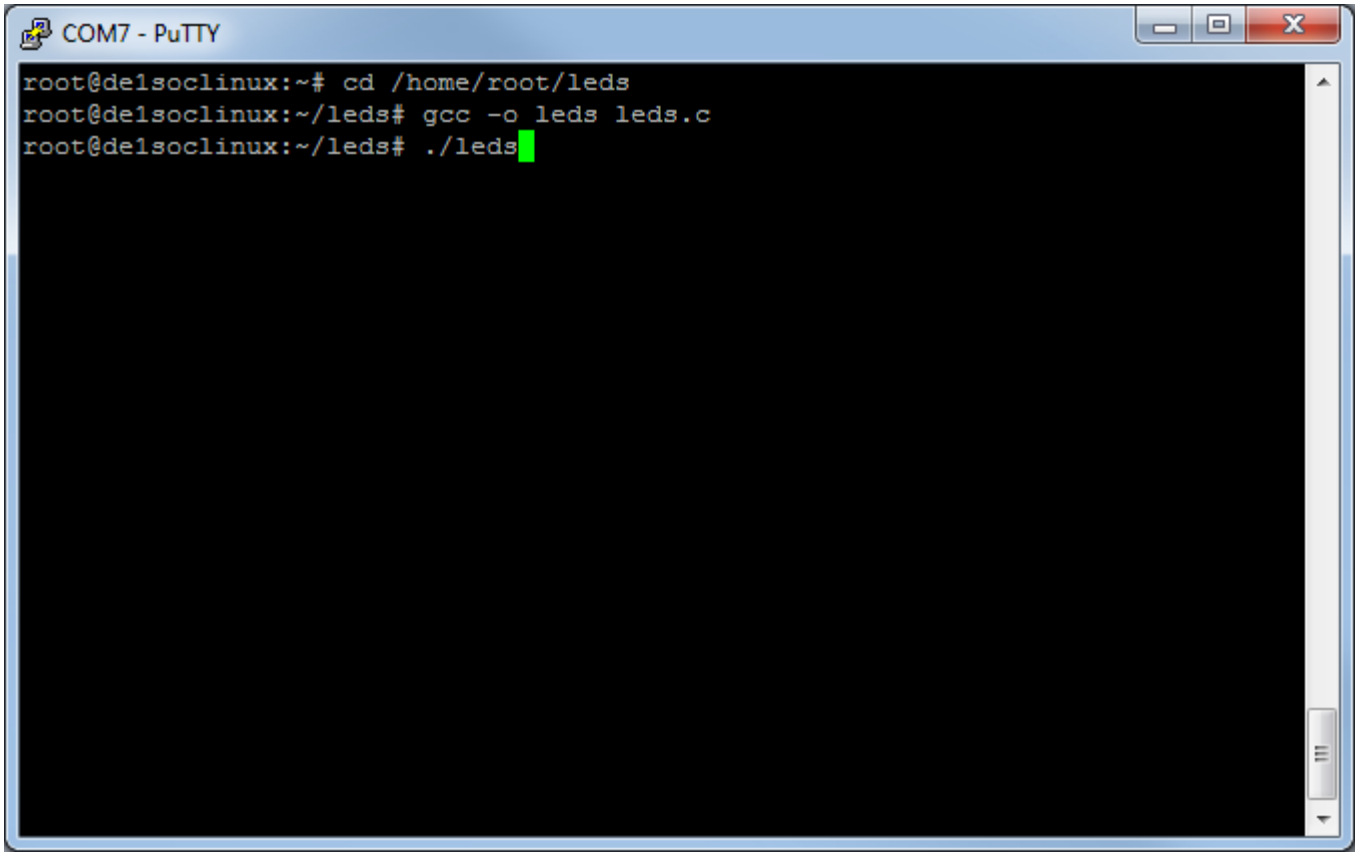
Step 2: Change to the Example Directory



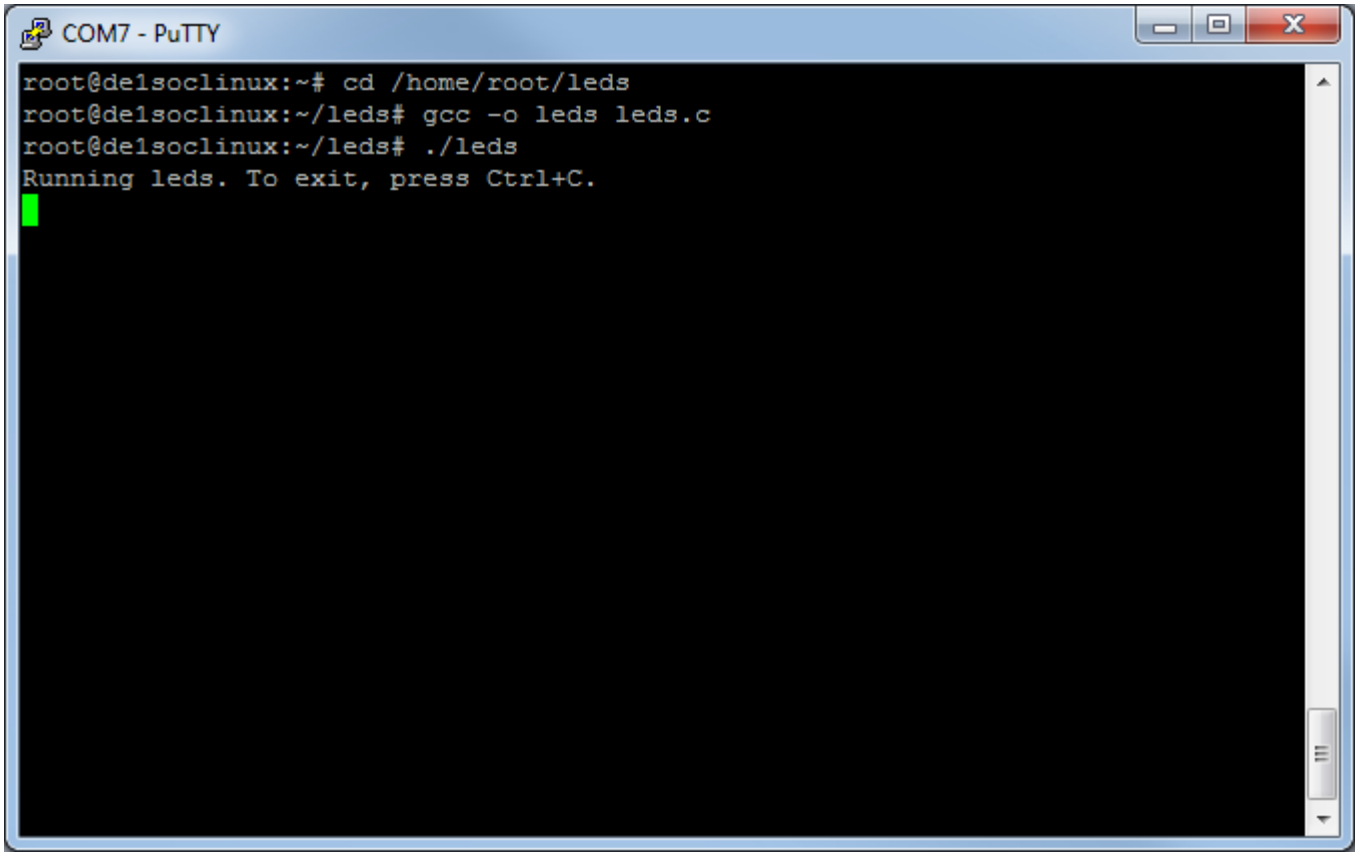
Step 3: Compile the Sample Program



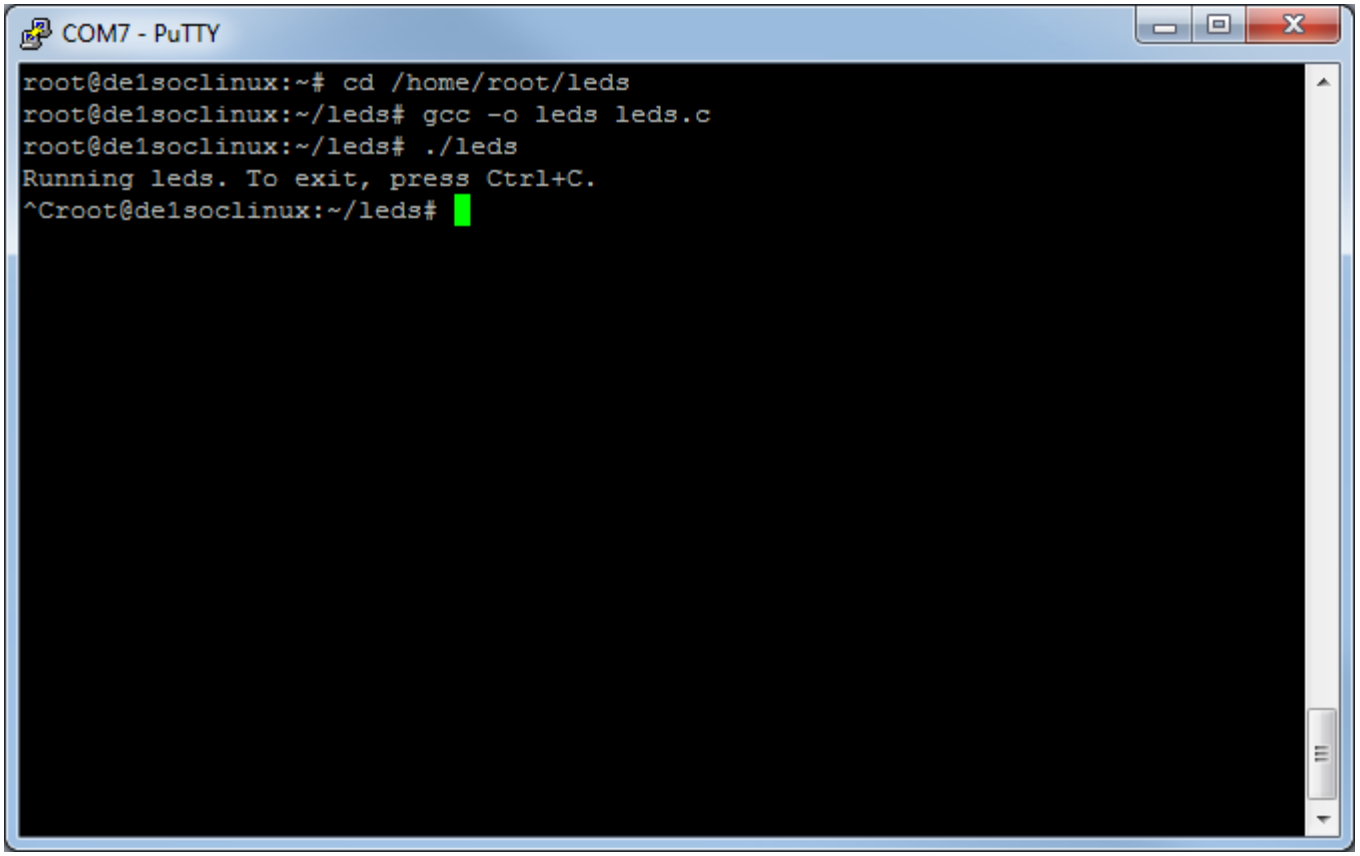
Step 4: Execute the Sample Program



Step 5: The Program is Running. Try Toggling the Switches.



Step 6: Press Control-C to Exit the Program



Step 7: Open the leds.c file. See the mmap usage.