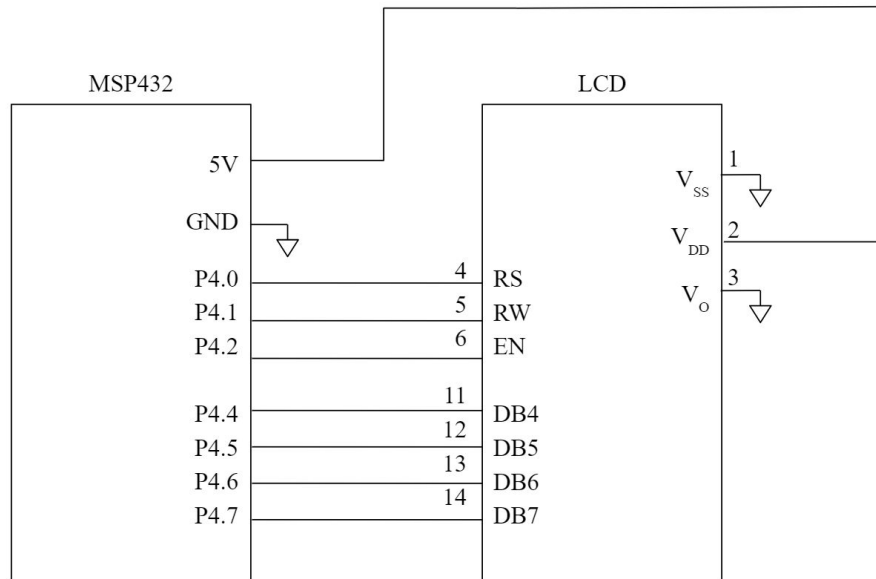
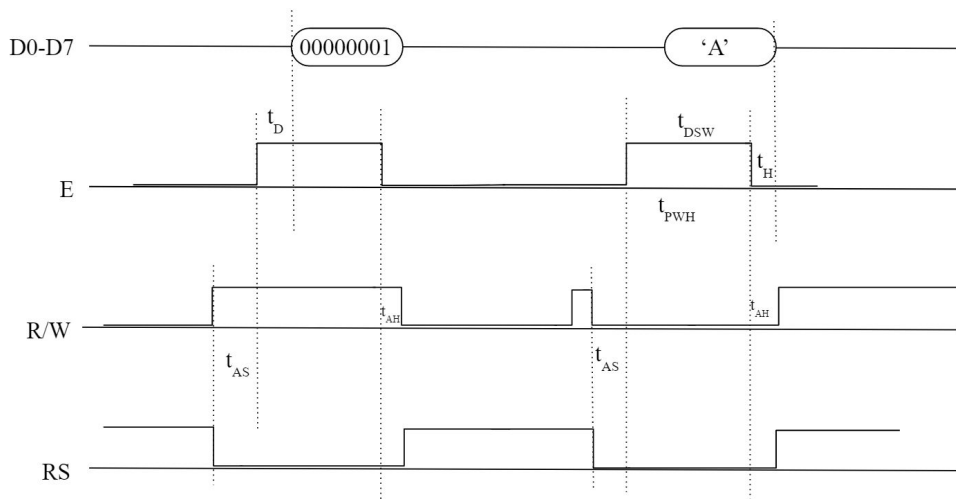


Assignment 3 LCD Display



[diagram of MSP432 to LCD]

1. Minimum amount of time necessary for the LCD to start up.
Amount of time from when device is powered up until character can be written
= 20ms Power on + 37us Set function + 37us Display Set + 1.52ms Display Clear
= calculated __



2.

Figure 2: Timing Diagram for clearing the LCD and displaying the letter A

Instruction	Instruction code										Description	Execution time (fosc= 270 KHZ)
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Write "20H" to DDRAM and set DDRAM address to "00H" from AC	1.52ms
Cursor or Display shift	0	0	0	0	0	1	S/C	R/L	-	-	Sets cursor moving and display shift control bit, and the direction without changing DDRAM data.	37μs

Clear_LCD(); // clear the display

Home_LCD(); // move the cursor to the top left of the LCD

Write_char_LCD(); // write a character on the LCD

```
void init_LCD() {
    send_command(0x00);
    delay_ms(100);
    send_command(0x30);
    send_nybble(50);
    delay_ms(50, FREQ_3_MHz);
    send_nybble(50);
    delay_ms(50, FREQ_3_MHz);
    send_nybble(50);
    delay_ms(50, FREQ_3_MHz);
    send_command(0x20);
    send_nybble(10);
    send_command(0x28);
    send_command(0x10);
    send_command(0x0F);
    send_command(0x06);
}
```

```
void Clear_LCD() {
    send_command(0x01);
    delay_ms(10, FREQ_3_MHz);
}
```

```
void Home_LCD() {
    send_command(0x02);
    delay_ms(10, FREQ_3_MHz);
}
```

```
void Write_char_LCD(Byte byte) {
```

```
P3->OUT |= BIT3; // RS high  
P4->OUT &= ~BIT1; // R/W low
```

```
output_even_pin(P6, 4, byte, 7);  
output_even_pin(P6, 5, byte, 6);  
output_even_pin(P4, 6, byte, 5);  
output_odd_pin(P1, 5, byte, 4);
```

```
send_nybble(10);
```

```
output_even_pin(P6, 4, byte, 3);  
output_even_pin(P6, 5, byte, 2);  
output_even_pin(P4, 6, byte, 1);  
output_odd_pin(P1, 5, byte, 0);
```

```
send_nybble(10);
```

```
}
```

```
void Clear_LCD() {  
    send_command(0x01);  
    delay_ms(10, FREQ_3_MHz);  
}
```

```
void Home_LCD() {  
    send_command(0x02);  
    delay_ms(10, FREQ_3_MHz);  
}
```

```
void Write_char_LCD(Byte byte) {  
    P3->OUT |= BIT3; // RS high  
    P4->OUT &= ~BIT1; // R/W low
```

```
output_even_pin(P6, 4, byte, 7);  
output_even_pin(P6, 5, byte, 6);  
output_even_pin(P4, 6, byte, 5);  
output_odd_pin(P1, 5, byte, 4);
```

```
send_nybble(10);
```

```
output_even_pin(P6, 4, byte, 3);  
output_even_pin(P6, 5, byte, 2);  
output_even_pin(P4, 6, byte, 1);  
output_odd_pin(P1, 5, byte, 0);
```

```
    send_nybble(10);  
}
```