

Laborationsrapport Lab3.

Uppgiften var att tända samtliga segment samt samtliga lysdioder..

Kortets 8 stycken 7-segmentsdisplayer, egentligen (8-segment med decimalkommat), tänds genom att skicka kommandot 0x40, som betyder att chipet skall förbereda sig för att ta emot data och autoinkrementera adresspekaren för kommande data. Detta kallar manualen för "Datakommand Set".

Därefter ges kommandot 0xC0, vilket betyder att startadressen skall vara 0. Vill man hoppa över det första segmentet och tända lysdioden så skall man ange startadressen 0xC1.

7-segmenten ligger på jämna adresser och lysdioderna ligger på udda.

Därefter kan man skicka upp till 16 bytes data, men för att chipet ska förstå att det som följer är data, så måste Strobe ligga kvar låg.

Programmet anropar LighUpEverything(); som ser ut så här:

```
void LighUpEverything(void) {
    sendInstruction(0x8F); //Lights up
    sendInstruction(0x40); //Command, write to chip, chip autoincrements
    sendStartAdress(0xC0);
    sendMultipleData(0xFF,16);
}
```

SendInstruction anropar den tillhandahållan funktionen sendbyte()

```
void sendInstruction(uint8_t data)
{
    STB(LOW);
    CDELAY;
    sendbyte(data);
    CDELAY;
    STB(HIGH);
    CDELAY;
}
```

sendStartAdress håller kvar STB låg, därför har en särskild funktion gjorts för detta anropet. Det går alltså inte att använda sendInstruction.

```
void sendStartAdress(uint8_t data)
{
    STB(LOW);
    CDELAY;
    sendbyte(data);
    CDELAY;
}
```

Därfeter anropas funktionen sendMultipleData, vilken som inargument tar datat som skall skickas, samt antalet gånger detta skall skickas.

```
void sendMultipleData(uint8_t data, int count)
{
    int i;
```

```
    for(i = 0; i < count; i++)  
        sendbyte(data);  
  
    CDELAY;  
    STB(HIGH);  
  
}
```

Skrivningen avslutas genom att låta STB gå hög.