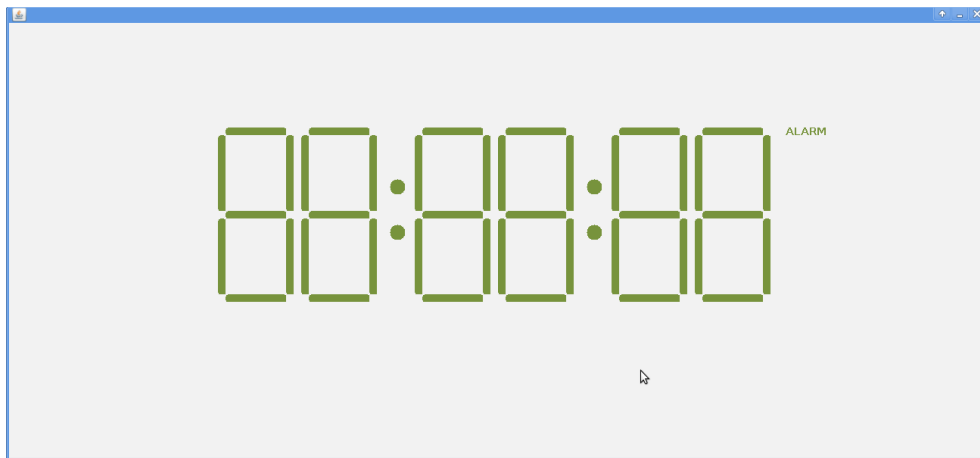


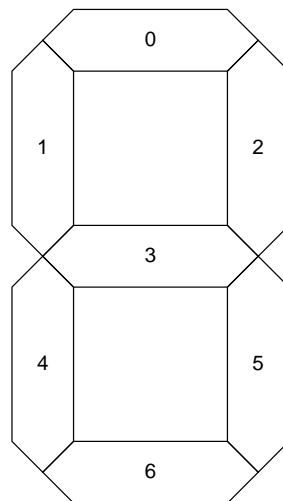
MULTISEGMENT DISPLAY EMULATOR



In het project gebruik je een emulator die de display van een wekker emuleert. Het programma op de ATmega32 stuurt dit display aan via RS232 via het volgende protocol.

Een zeven segment display is een veel gebruikt elektronisch device dat op een eenvoudige manier de cijfers 0..9 kan weergeven. Het wordt veel toegepast in allerlei apparaten als digitale klokken, radio's, magnetron ovens, meetapparatuur enz. De meest gebruikte devices bestaan uit zeven LED's die als een cijfer "8" zijn geplaatst.

De volgorde van de segmenten is als volgt:



Voorbeeld : Om het getal 4 op het beeldscherm te tonen sturen we de byte 0x2E want,

0 0 1 0 1 1 1 0

7 6 5 4 3 2 1 0

Er zijn zes cijfers op het display. Er zijn hiervoor 6 bytes om de cijfers te bepalen. Van elke byte worden de eerste 7 bits (bit 0 t/m 6) gebruikt om de zeven segmenten van een cijfer aan te sturen. Een '1' betekent dat een segment AAN is.

Op deze manier stuur je zes bytes voor de tijd:

- bytes 0..1: uren links en rechts;
- bytes 2..3: minuten links en rechts;
- byte 4..5: seconden links en rechts.

Echter, in totaal worden zeven bytes verstuurd. In de zevende byte zitten de tekens voor de dubbele punten en de alarm indicatie. Het zevende byte is als volgt opgebouwd:

- bit 0: ALARM indicator aan;
- bit 1: linker dubbele punt ":" aan;
- bit 2: rechter dubbele punt ":" aan;
- bit 3: alarm buzzer aan;
- bits 4..7 : niet gebruikt.

De display emulator stuurt een 0x01 terug om aan te geven dat er zeven bytes zijn ontvangen. Pas dan verwerkt de emulator de zeven bytes door deze te tonen. De emulator stuurt tenslotte een 0x02 terug om aan te geven dat het verzoek correct is afgehandeld.

Het is ook mogelijk om speciale commando's naar de display emulator te sturen.

Als de emulator 0x80 of 0x81 ontvangt worden de dan toe ontvangen bytes gewist en wordt een 0x03 teruggestuurd. Het verschil tussen 0x80 en 0x81 is dat is het laatste geval ook nog het segment compleet wordt gewist.

Opmerking : byte 0xff wordt dus niet gebruikt in dit protocol.