

Drivning av LED-matrisen

DAMatrix

Michael Josefsson

Version 0.1

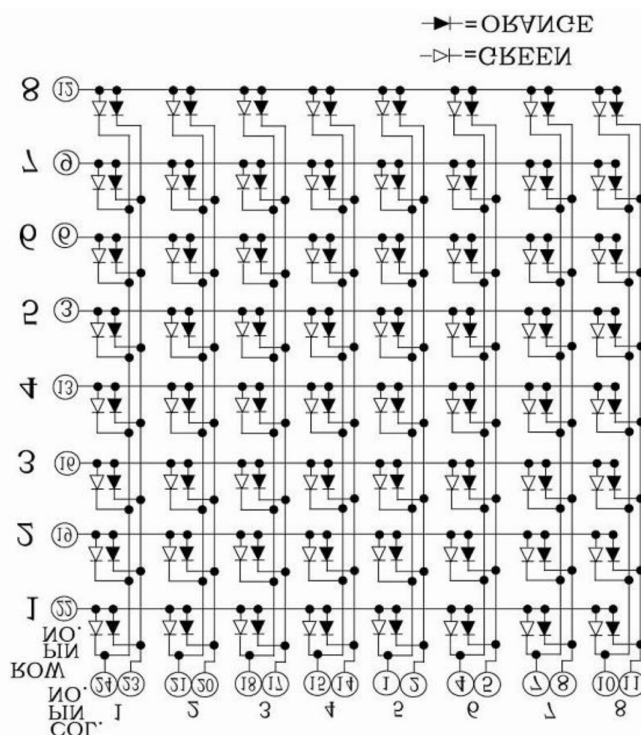
Inledning

Denna text är en beskrivning av den seriella 8x8 RGB-lysdiodsmatrisen DAMATRIX. Texten inleds med en teknisk beskrivning om hur man rent allmänt kan styra en tvåfärgad lysdiodsmatris. Den beskrivningen är nödvändig för att senare förstå alla delar av schemat till DAMATRIX. Med schemat som ledning beskrivs slutligen hur indata skall utformas för korrekt funktion.

Allmänt

Det finns några olika sätt att driva en LED-matrisdisplay. Beroende på hur många pin-
nar som finns tillgängliga, eller displayens storlek kan lite olika strategier tillgripas. Här beskrivs några av dem.

Direkt ur databladet kan man hämta tillverkarens kopplingsschema över en matris' alla lysdioder. För enkelhets skull med bara två färger (orangea och gröna lysdioder) just nu. Tyvärr var databladstillverkaren ingen ingenjör och ritade hela schemat upp och ner, så vi vänder det och flippar det sedan sidledes för att få normal signalflödesriktning och kunna beskriva funktionen bättre:



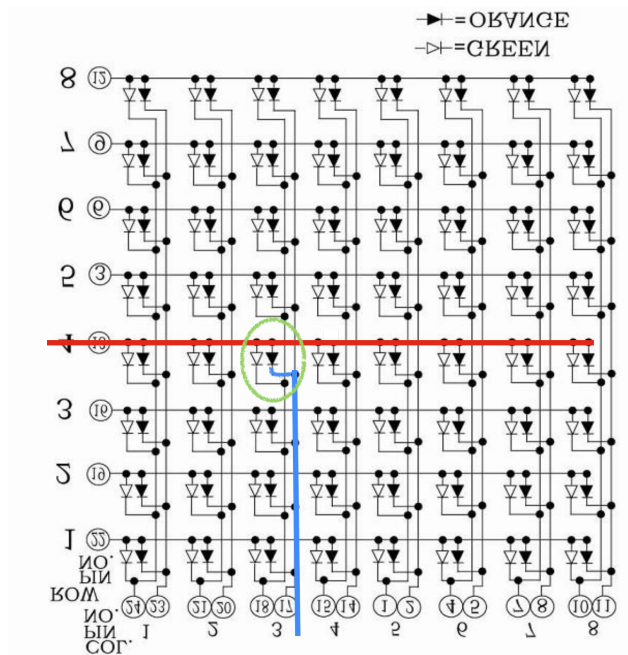
Äntligen är schemat läsbart med positiv potential uppåt, låg potential nedåt och signalflödet från vänster till höger. Som det ska vara!

Matrisens interna koppling är:

- Alla dioder oavsett färg i varje *rad* i matrisen hänger ihop i sina anoder.
- Alla dioder av samma sort och *kolumn* har sina katoder sammankopplade.

Lägg märke till vilke diodände som är anod respektive katod. De återkommer nedan.

För att tända en diod måste raden förses med positiv spänning samtidigt som diodens katod jordas. Till exempel, för en orange diod:



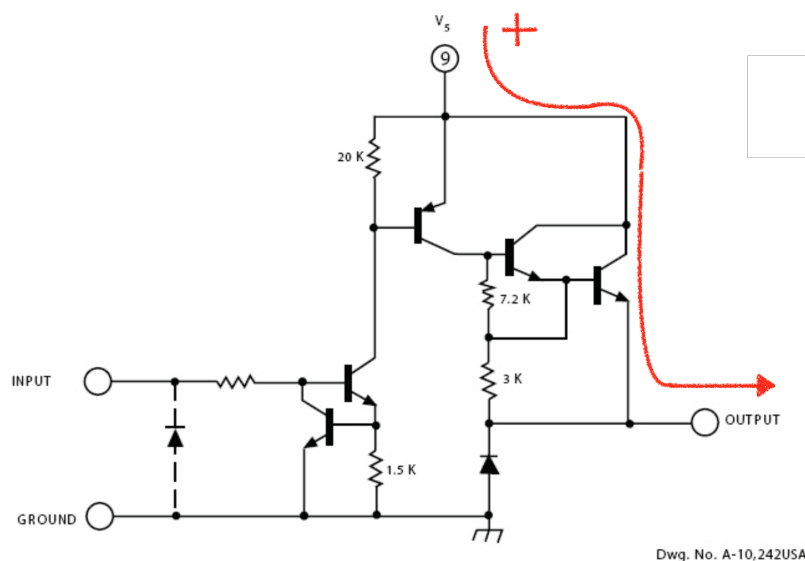
Med spänning på rad 4 (pinne 13) och jord på kolumn 3 (pinne 17) kommer den markerade lysdioden tändas. Diodmatrisens upplägg är en tydlig indikation på att den bör drivas med multiplexning av raderna.

För att tillfredsställa matrisens maximala strömbehov används speciella drivkretsar där *en försörjer raden* med tillräcklig ström och *en jordar kolumnen*.

Drivarna måste alltså vara av två sorter, en som kan leverera ström och en som kan suga i sig strömmen ner mot jord. Den första kallas *source*, den andra *sink*.

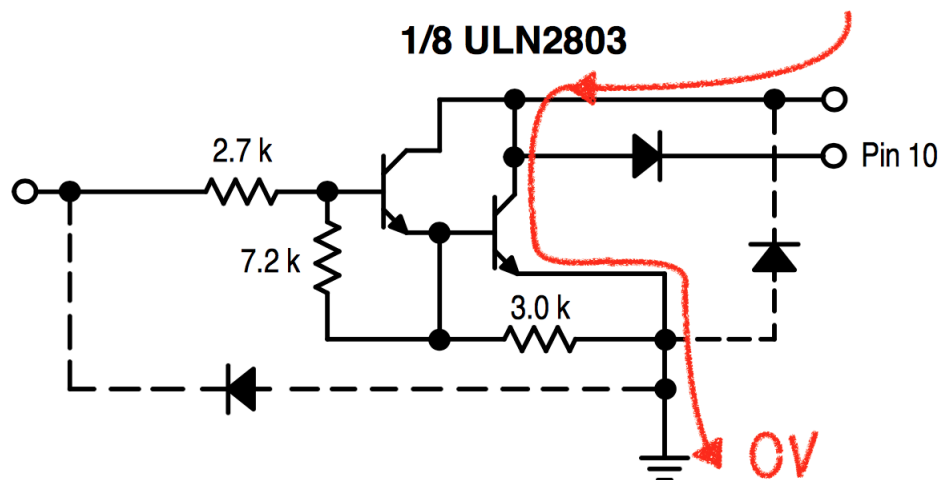
Source Kretsen 2981 innehåller 8 *source drivers*.

Ur dess datablad kan man läsa schemat (per pinne):

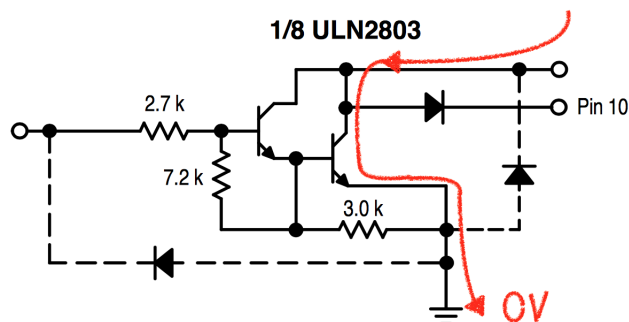
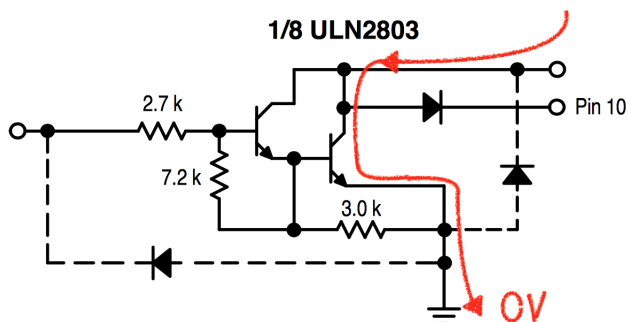
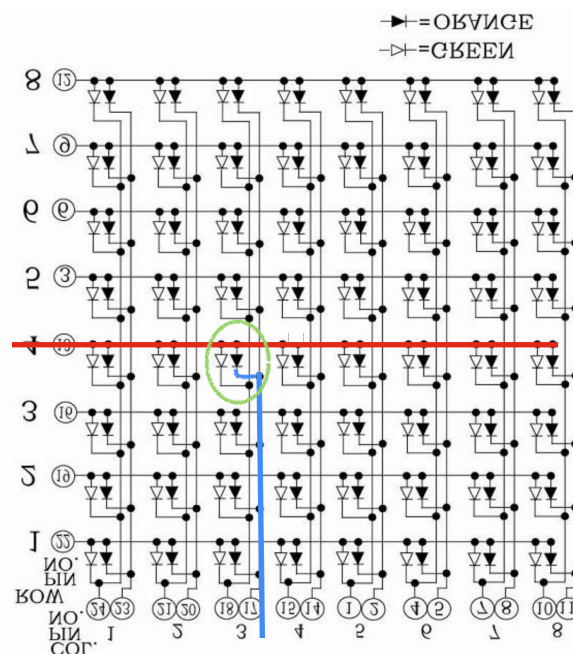
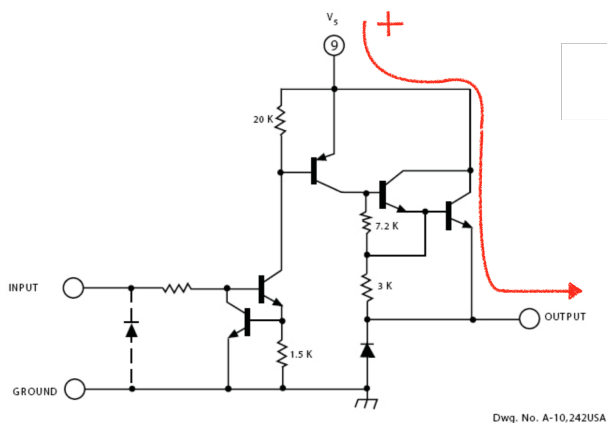


Funktionen är att mata ut (eller inte) ström via dess utgångspinnar. Varje utgångspinne måste anslutas till respektive rad på matrisen.

Sink Kretsen 2803 innehåller på motsvarande sätt 8 *sink drivers*. Funktionen framgår av schemat där man ser att den är konstruerad för att kunna sänka mot jord. Det är helt omöjligt att få denna att sourca, den kan enbart sinka:

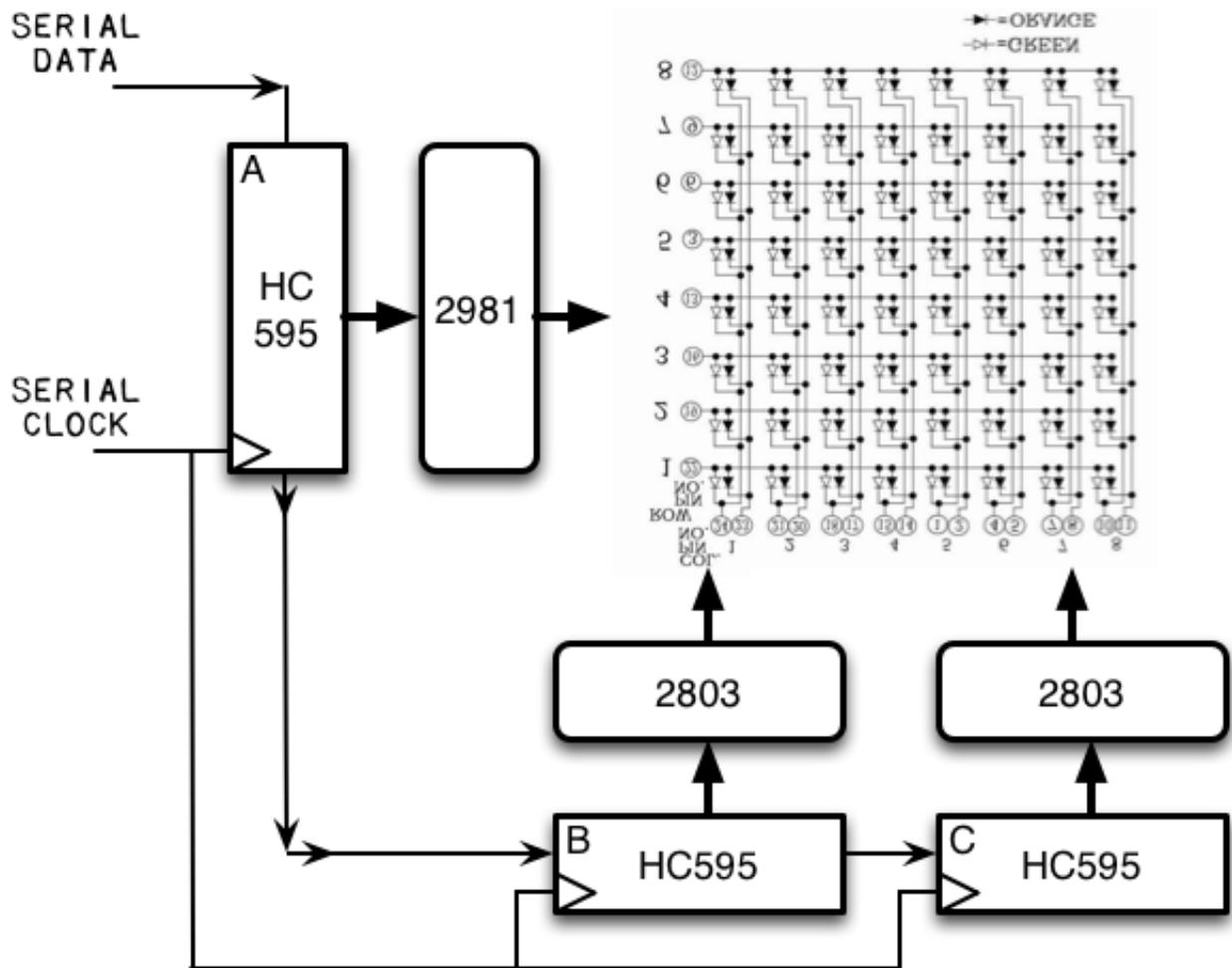


Sammantaget måste matrisen omges av en source-krets och två sink-krestar vilket antyds i layouten nedan. Där en 2803-kapsel används för vardera färg:



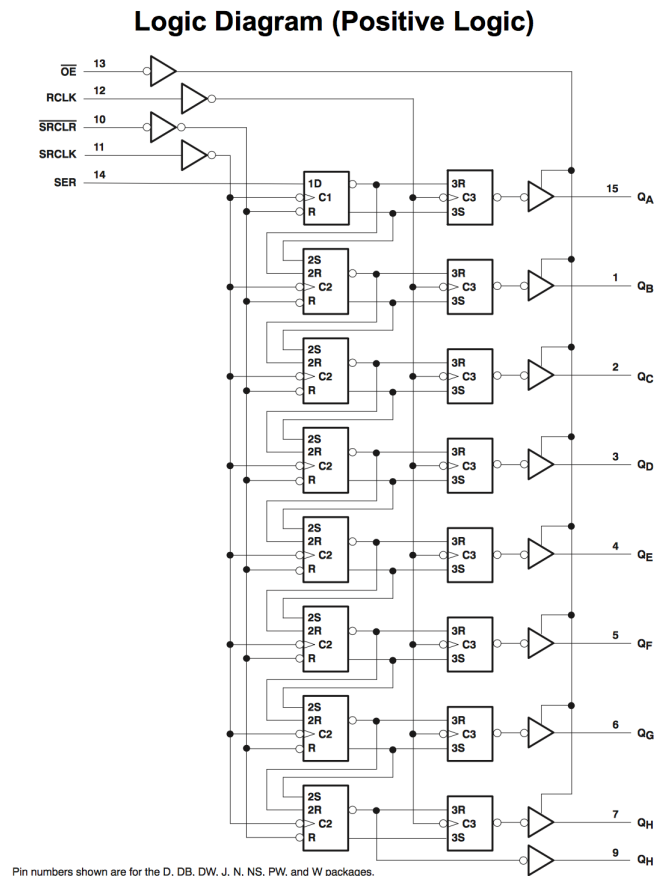
Seriell styrning Än så länge kräver konstruktionen en hel del port-ben för att kunna styras. Ett sätt att komma från denna *parallella* styrning är att ersätta den med en *seriell* sådan.

Principen är att låta insignalen till 2981/2803-orna ovan komma från var sitt skiftregister, här totalt tre. Skiftregistren kan seriekopplas till ett 24-bits skiftregister:



Från processorn anger man data och klockflank så att skiftregistren (HC595) förses med rätt information för att respektive diod skall lysa. För en multiplexad matris, där enbart en rad skall kunna vara tänd åt gången, betyder det att registret A enbart ska innehålla en enda *aktiv* bit, raden. Registren B och C innehåller *aktiva* bitar för de kolumner och färger som skall tändas i den raden.

Skiftregistret Anledningen till att just skiftregistret HC595 använts fås ur dess datablad¹:



Det seriella datat skiftas in (med SER och SRCLK) i vänstra uppsättningen vippor (C1 och C2). Först när detta data ligger på rätt plats i dessa vippor klockas de *gemensamt* över till utgångsvipporna (C3) med signalen RCLK. För att något skall hända på utgången måste utgångsbuffertarna aktiveras genom att lägga $\overline{OE} = 0$

Utan de isolerande utgångsvipporna skulle matrisen blinka hej vilt medans informationen klockades på plats.

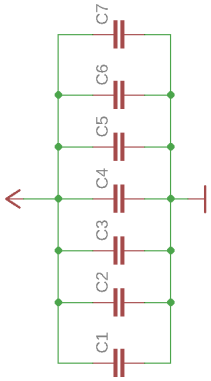
Schema och styrning För att kunna koppla upp alla dessa komponenter korrekt krävs stor noggrannhet. Enda rimliga metoden är att rita en komplett schema, gärna på A3-papper, med varje kapsel angiven och pinnarna numrerad. Låt sedan någon annan konstrollera schemat. Fel i schemat på det här stadiet tar några minuter att korrigera, med allt uppkopplat tar det **timmar!** Och så finns ju risken att någon komponent gått sönder vid felkoppling, vilket ger ännu mer felsökning.

För att undvika denna tidsödande process finns DAMATRIX som är utformad enligt ovan men med tre färger: Röd, Grön och Blå (RGB).

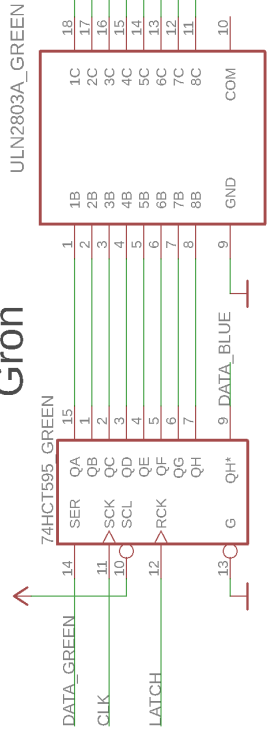
Schemat återfinns nedan och måste förstås för att kunna programmeras.

¹Och som vanligt är Texas Instruments datablad tydligast!

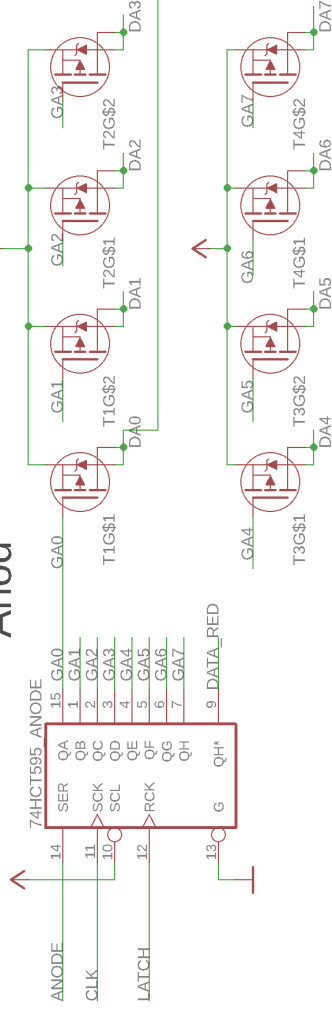
Avkopplingskondensatorer 0.1uF



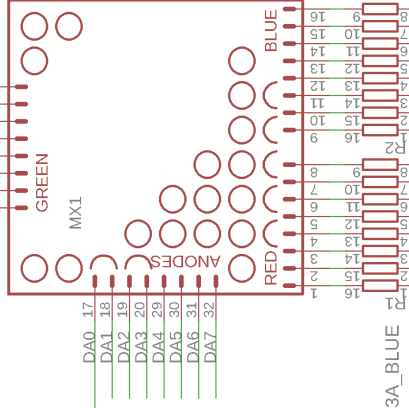
Grön



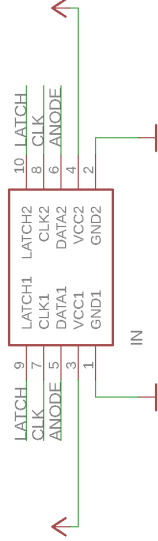
Anod



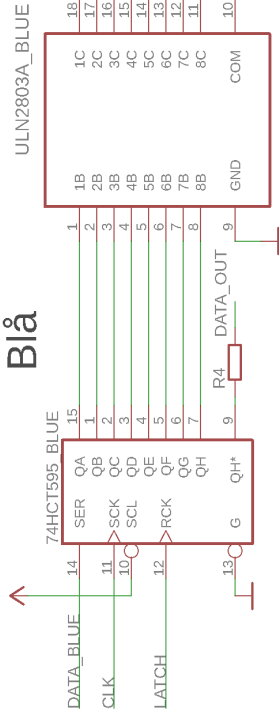
LED-matris



Kontakt IN (2x5)



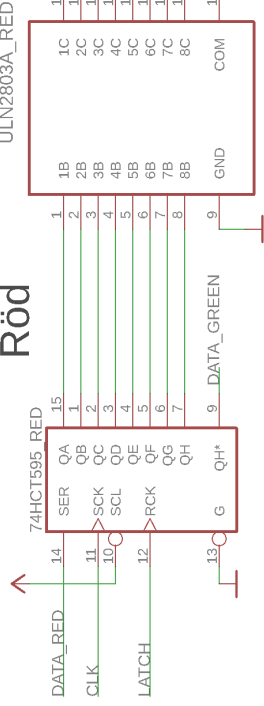
Blå



Kontakt OUT (2x5)



Röd



Kort rundvandring i schemat Den centrala komponenten i schemat är såklart LED-MATRIS. Till vänster om den återfinns tre drivare (*sink*) för respektive färg, GREEN, BLUE och RED.

För att kunna driva matrisens anoder används FET-transistorerna GA0–GA7. Dessa spelar rollen som *source* i denna krets.

Förutom matningsspänning om 5 V, VCC och GND matas data till kretsen in seriellt i takt med en klocka.

Yttre anslutning är via det 10-poliga donet KONTAKT IN där de intressanta signalerna är ANODE, CLK och LATCH. Notera att namngivningen i själva donet är, till exempel, DATA1 och DATA2 medan båda dessa heter ANODE i själva schemat; det är dock samma signal i schemat.

Följer man databitarna från kontaktens ANODE-pinne ser man att de först klockas in i ANODE på HCT595-kretsen, vidare till DATA_RED och så vidare. Slutlig ordning blir ANODE, DATA_RED, DATA_GREEN och DATA_BLUE.

En skrivning till displayen innehåller alltså totalt fyra byte, 32 bitar som klockas in bit för bit, för att kunna tända en rad i olika färger.

Ytterligare klockningar gör att datat klockas ut ur displayen via ett skyddsmotstånd R4 till en eventuell senare i kedjan liggande display. På detta sätt kan åtskilliga displayer anslutas i serie med en 10-polig bandkabel ansluten till KONTAKT UT.

Tips: Medan det är fullt möjligt att *bit-banga* alla bitar med egenskrivna rutiner är det inbyggda protokollet SPI till slut mycket enklare att använda. Protokollet SPI kan skriva en byte i taget med PRESCALER-valbar **hög** hastighet.