MEGA 65

Programmeergids



MUSEUM OF ELECTRONIC GAMES & ART

MEGA65-TEAM

Paul Gardner-Stephen, prof.
(highlander)
Oprichter
Software- en virtuele hardwarearchitect
Woordvoerder en hoofdwetenschapper

Martin Streit
(seriously)
Video- en fotoproductie
Fiscaal en organisatie
Sociale media

Dan Sanderson
(dddaaannn)
Media en documentatie
MEGA65.ROM

Edilbert Kirk, dr.
(Bit Shifter)
MEGA65.ROM
Handleiding en gereedschappen

Gábor Lénárt
(LGB)
Emulator

Farai Aschwanden
(Tayger)
Bestands-hosting en gereedschappen
Financieel adviseur

Falk Rehwagen
(bluewaysw)
GEOS

Robert Steffens
(kibo)
Netwerktechnologie
Jacht op bugs in onze cores

Detlef Hastik
(deft)
Mede-oprichter
Projectaansturing
Marketing en verkoop

Oliver Graf
(lydon)
Releasemanagement
VHDL- en platformverbeteringen

Antti Lukats
(antti-brain)
Host-hardwareontwerp en -productie

Dieter Penner
(doubleflash)
Host-hardwareondersteuning

Mirko H.
(sy2002)
Aanvullende platforms en advies

Gürçe Işıkyıldız
(gurce)
Gereedschappen en verbeteringen

Daniel England
(Mew Pokémon)
Aanvullende code en gereedschappen

Hernán Di Pietro
(indiocolifa)
Aanvullende emulatie en gereedschappen

Roman Standzikowski
(FeralChild)
Open ROMs

Anton Schneider-Michallek
(adtbm)
Presentatie en ondersteuning

Fouten en onvolkomenheden melden

Dit boek wordt voortdurend verbeterd door de MEGA65-gemeenschap. De versie van deze editie is:

```
commit c598652ac7f860872b8be62911f59ad5ad752169
date: Sat Oct 24 17:02:30 2026 +0200
```

We willen dat dit boek zo goed mogelijk wordt. Als u fouten ontdekt, iets mist of graag meer informatie wilt, meld dit dan via het issue-tracker van de MEGA65 User's Guide:

https://github.com/mega65/mega65-user-guide/issues

U kunt daar ook controleren of iemand anders al een vergelijkbaar probleem heeft gemeld, terwijl u wacht op een update van dit boek.

Ten slotte kunt u altijd de nieuwste versies van onze boekenserie downloaden via:

- https://mega65.org/mega65-book
- https://mega65.org/user-guide
- https://mega65.org/developer-guide
- https://mega65.org/basic65-ref
- https://mega65.org/chipset-ref
- https://mega65.org/docs

ONTWIKKELEN VOOR DE MEGA65

Uitgegeven door
het MEGA Museum of Electronic Games & Art e.V., Duitsland.

IN AANBOUW

Copyright ©2019 - 2026 Paul Gardner-Stephen, het MEGA Museum of Electronic Games & Art e.V. en medewerkers.

Dit boek is beschikbaar onder de GNU Free Documentation License v1.3 of latere versie. Dit betekent dat u dit boek vrij mag wijzigen, vermenigvuldigen en herdistribueren, onder bepaalde voorwaarden. De volledige tekst van de GNU Free Documentation License v1.3 is te vinden op https://www.gnu.org/licenses/fdl-1.3.en.html. Een Nederlandse vertaling van de licentie is te vinden op https://meta.wikimedia.org/wiki/GNU_Free_Documentation_License/nl.

Deze copyrightlicentie omvat ook de toestemming om dit document geheel of gedeeltelijk te kopiëren en/of te verspreiden voor gebruik in onderwijsomgevingen. Wij willen het onderwijs van toekomstige generaties ondersteunen, dus als u vragen of bezwaren heeft, neem dan contact met ons op.

oktober 29, 2026

Inhoudsopgave

1 Inleiding **1**
Welkom bij de MEGA65! 3
Overige boeken in deze reeks 4
Doe mee! 4

I INLEIDING **5**

2 Welkom en bedankt! **7**

3 Andere bronnen **11**

II PLATFORMONAFHANKELIJKE ONTWIKKELHULPMIDDELEN **15**

4 Emulators **17**
De Xmega65-emulator gebruiken 19
Het toetsenbord 19
Problemen met starten op Apple MacOS 20
Instellingen bijwerken 21
De Live ISO-image gebruiken 21
Een opstartbare USB-stick of DVD maken 21
Aan de slag 22
Andere functies van de Live ISO 23

5 Gegevensoverdracht en foutopsporing **25**
m65-hulpprogramma 27
Schermfoto's met het m65-hulpprogramma 27
Een programma laden en uitvoeren op de MEGA65 28
De FPGA herconfigureren om een andere bitstream uit te voeren . . . 28
Toetsenbordinvoer op afstand 28
Unit-testing en logboekondersteuning 29
Unit-tests gebruiken met C 29

Unit-tests gebruiken met BASIC 65 30
BASIC 65-voorbeeld 31
M65Connect 32
mega65_ftp 33
TFTP-server 33
Een BASIC-tekstbestandlisting omzetten naar een PRG-bestand 34

6 Assemblers **35**

7 C- en C-achtige compilers **39**
MEGA65 libc 41

8 MEGA65 Standard C Library **43**
Structuur en gebruik 45
conio.h 45
conionit 45
setscreenaddr 45
getscreenaddr 46
setcolramoffset 46
getcolramoffset 46
setcharsetaddr 46
getcharsetaddr 47
clrscr 47
getscreensize 47
setscreensize 47
set16bitcharmode 48
sethotregs 48
setextendedattrib 48
togglecase 48
togglecase 48
togglecase 49
bordercolor 49

bgcolor 49
textcolor 49
revers 49
highlight 50
blink 50
underline 50
altpal 50
clearattr 51
cellcolor 51
setpalbank 51
setpalbanka 51
getpalbank 52
getpalbanka 52
setmapedpal 52
getmapedpal 52
setpalentry 53
fillrect 53
box 53
hline 54
vline 54
gohome 55
gotoxy 55
gotox 55
gotoy 55
moveup 56
movedown 56
moveleft 56
moveright 56
wherex 57
wherey 57

pcputc 57
pcputsxy 57
cputcxy 57
pcputs 58
cputc 58
cputnc 58
cputhex 58
cputdec 59
cputs 59
cputsxy 59
cputcxy 60
cputncxy 60
cprintf 60
pcprintf 61
cgetc 61
kbhit 61
getkeymodstate 62
flushkeybuf 62
cinput 62
VIC_BASE 63

9 BASIC-tokenisatoren **65**

III PROGRAMMEERGEREEDSCHAP OP DE MEGA65 ZELF **69**

10 C65-geheugenmonitor **71**

11 MEGA65 Matrix Mode **73**

12 Turbo Assembler **75**

13 Omgaan met de C65- en C64-modus **77**

IV MEGA65 BESTURINGSSYSTEEM 81

14 Ontwikkeling van systeemprogramma's 83

Inleiding 87
Flashmenu 87
Format/FDISK-hulpprogramma 88
Toetsenbordtest-hulpprogramma 89
MEGA65-configuratiehulpprogramma 89
Freezer-menu 89
Hulpprogramma's van het Freezer-menu 90
Hypervisor 90
OpenROM 90

15 MEGA65 Hyppo-diensten 93

Inleiding 95
Hyppo-diensten: Beknopt overzicht 99
Algemene diensten 103
Station-/opslagservices 107
Schijfimage-services 136
Taak- en procesdiensten 140
Systempartitieservices 156
Freezerservices 157

V MULTIMEDIA EN GEGEVENSCONVERSIE 159

VI GEOS-ONTWIKKELING 161

VII HARDWARE 163

16 Nexys4-borden gebruiken als een MEGA65 165

Uw eigen MEGA65-compatibele computer bouwen 169

Werkende Nexys4-borden 170
Het Nexys4-bord 170
Het Nexys4DDR-bord 170
De Nexys A7 171
Stroom, jumpers, schakelaars en knoppen 172
Micro-USB-voeding 173
Externe voeding 173
Overige jumpers en schakelaars 174
Aansluitingen en randapparatuur 175
Communicatie met uw pc 175
Knoppen op het bord 176
Toetsenbord 176
Enkele toetstoewijzingen met een USB-toetsenbord 177
MicroSDHC-kaart voorbereiden 179
Voorbereidingsstappen 179
De bitstream laden vanuit QSPI 181
Voorbereidingsstappen 181
Widget-bord 181
PMOD-naar-joystick-adapter 184
Handige tips 185

VIII BIJLAGEN 187

A Het flashen van de FPGA's en CPLD's in de MEGA65 189
Aanbevolen specificaties voor de pc 191
Waarschuwing 192
Vivado installeren 192
De FTDI-stuurprogramma's installeren 208
Linux-drivers 208
Windows-drivers 208

De hoofd-FPGA flashen met Vivado 212
De CPLD in het toetsenbord van de MEGA65 flashen met Lattice Diamond . 225
Het flashen van de MAX10-FPGA op het MEGA65-moederbord met INTEL QUARTUS 232

B Problemen oplossen 239
Hardware 241
Geen digitale video-uitvoer 241
MEGA65 R4, R5, R6 241
Geen lampjes bij het inschakelen 241
Vivado 242
RAM-vereisten 242
mega65_ftp 243
Ontbrekende bibliotheek 243

C Elektronisch schema 245
MEGA65 R6 elektronisch schema 247
MEGA65 R3 elektronisch schema 271
MEGA65 R2 elektronisch schema 296
Nexys Widget Board elektronisch schema 319

D Steuners & donateurs 323
Organisaties 325
Bijdragers 326
Steuners 327

Index 337

HOOFDSTUK 1

Inleiding

- Welkom bij de MEGA65!
- Overige boeken in deze reeks
- Doe mee!

WELKOM BIJ DE MEGA65!

Van harte gefeliciteerd met de aanschaf van één van de meest langverwachte computers uit de geschiedenis van het computergebruik! De MEGA65 is door de community ontworpen en is gebaseerd op de nooit uitgebrachte Commodore® 65[1]-computer; een computer die in 1989 werd ontworpen en in 1990 voor het publiek verkrijgbaar zou worden. Inmiddels zijn er decennia verstreken en wij hebben ons ingespannen om herinneringen op te roepen aan een vroegere tijd waarin computers eenvoudig en vriendelijk waren. Zij waren niet alleen eenvoudig te bedienen en te begrijpen, doch ook vriendelijk en toegankelijk voor nieuwe gebruikers.

Deze computers uit de jaren tachtig inspireerden veel van hun eigenaren tot de opwindende en dankbare technologiecarrières die zij vandaag de dag hebben. Stelt u zich eens de opwinding voor die deze computerpioniers beleefden, toen zij ontdekten dat zij hun nieuwe computer konden gebruiken om problemen op te lossen, een brief te schrijven, belastingaangifte te doen, nieuwe dingen uit te vinden, te ontdekken hoe het heelal werkt en misschien zelfs een spannend spelletje of twee te spelen! Wij willen datzelfde niveau van opwinding weer aanwakkeren (dat helaas in het moderne computergebruik niet meer te vinden is) en daarom hebben wij de **MEGA65** gemaakt.

Het MEGA65-team is van mening dat het bezitten van een computer vergelijkbaar is met het bezitten van een huis. U gebruikt een huis niet alleen maar; u verandert dingen, groot en klein, om er uw eigen persoonlijke leefruimte van te maken. Na verloop van tijd, wanneer u zich er eenmaal thuis voelt, kunt u besluiten uw huis te verbouwen of uit te breiden om het comfortabeler of praktischer te maken. Beschouw de MEGA65 als uw eigen "computerthuis".

Deze handleiding leert u meer dan alleen plaatjes aan de muur te hangen; hij laat u zien hoe u uw droomhuis bouwt. Terwijl u deze handleiding leest, leert u de MEGA65 bedienen, programma's schrijven, extra programmatuur toevoegen en de mogelijkheden van de hardware uitbreiden. Wat niet meteen duidelijk zal zijn, is dat u onderweg ook meer leert over de geschiedenis van het computergebruik, terwijl u de vele facetten van BASIC-versie 65 en de commando's van het besturingssysteem verkent.

Computergrafiek en muziek maken het computergebruik leuker, en wij hebben de MEGA65 ontworpen om leuk te zijn! In deze handleiding leert u programma's schrijven die gebruikmaken van de ingebouwde **grafische** en **geluids**mogelijkheden van de MEGA65. U hoeft echter geen programmeur te zijn om plezier met de MEGA65 te beleven. Omdat de MEGA65 een complete Commodore® 64™[2] bevat, kan hij ook duizenden bestaande spellen, hulpprogramma's en zakelijke programmapakketten uitvoeren, evenals nieuwe programma's die op dit moment door liefhebbers van de Commodore-computer worden geschreven. Het enthousiasme voor de MEGA65 zal groeien naarmate wij getuige zijn van de programmeerwonderen die onze MEGA65-community voortbrengt, terwijl zij (en u!) de krachtige mogelijkheden van deze moderne herschepping van een Commodore-computer ontdekken en beheersen. Samen

[1] Commodore is een handelsmerk van C= Holdings
[2] Commodore 64 is een handelsmerk van C= Holdings

met u kunnen wij een nieuwe "homebrew"-community opbouwen, barstensvol proram-matuur en projecten die de mogelijkheden van de MEGA65 oprekken tot ver voorbij wat iemand ooit voor mogelijk had gehouden.

Wij heten u welkom op deze reis! Dank u dat u deel uitmaakt van de MEGA65-community van gebruikers, programmeurs en enthousiastelingen!

OVERIGE BOEKEN IN DEZE REEKS

Dit boek is één van de boeken uit de MEGA65-documentatiereeks. De reeks omvat:

- **De Mega65-gebruikershandleiding**
  Biedt een inleiding tot de MEGA65 en een beknopt overzicht van de BASIC 65-commando's
- **De Mega65 naslaggids BASIC65**
  Uitgebreide documentatie van alle BASIC 65-commando's, functies en opera-tors
- **De Mega65-chipset naslaggids**
  Gedetailleerde documentatie over de eigen chips van de MEGA65 en de C65
- **De Mega65 programmeergids**
  Informatie voor ontwikkelaars die programma's voor de MEGA65 willen schrijven
- **Alles over de Mega65**
  (Ook bekend als **Het grote Mega65-boek**)
  Alle delen in één enkel omvangrijk pdf-bestand voor eenvoudig zoeken. 1200 bladzijden en nog steeds groeiend!

DOE MEE!

Raak betrokken, leer meer over uw MEGA65 en sluit u online bij ons aan op:

- https://mega65.org/chat
- https://mega65.org/forum

DEEL I

INLEIDING

HOOFDSTUK 2

Welkom en bedankt!

Wij willen u bedanken voor uw interesse in de MEGA65 en voor het ontwikkelen van software die erop draait. Het succes van de MEGA65 als leuk retrocomputingplatform is afhankelijk van de beschikbaarheid van interessante software die mensen kunnen draaien. Dit vereist het vrijwillige werk van ontwikkelaars zoals u. Dus nogmaals, bedankt!

Ontwikkelen voor de MEGA65 is echter niet slechts een kwestie van geven ten bate van anderen. We hebben hard gewerkt aan een machine die net zo leuk is om op te programmeren als de Commodore 64, maar waarin enkele barrières en frustraties enigszins zijn weggewerkt. Dat wil zeggen, het is onze bedoeling dat mensen op de MEGA65 ontwikkelen omdat het leuk is, en dan genieten van het proces. De reis moet dus het doel op zichzelf zijn.

Dat gezegd hebbende, de MEGA65 is vrij nieuw en we zijn druk geweest met het maken van de machine. Dit betekent dat sommige ontwikkeltools nog niet zo gereed, eenvoudig of geïntegreerd zijn als we zouden willen. Het verbeteren van de tools is ook een gebied waarop we hopen dat velen van u met plezier zullen bijdragen, zodat ontwikkelen voor de MEGA65 toegankelijk kan worden voor zoveel mogelijk mensen.

Indien u problemen ondervindt bij het ontwikkelen voor de MEGA65, laat het ons dan weten en overweeg oplossingen te creëren die met anderen kunnen worden gedeeld. Dit zal iedereen helpen om te genieten van een actief onderdeel van het MEGA65-ecosysteem.

Tot slot onze excuses aan iedereen die hulpprogramma's of utilities voor de MEGA65 heeft ontwikkeld die aan onze herinnering zijn ontsnapt bij het schrijven van dit document. Indien dit het geval is, stuur ons dan een e-mail en wij zullen dit document bijwerken om ze op te nemen.

HOOFDSTUK 3

Andere bronnen

Dit boek richt zich op het proces van ontwikkelen voor de MEGA65. Om te voorkomen dat het duizend bladzijden lang wordt, beschrijft het niet de innerlijke werking van de MEGA65, behalve waar u dat moet weten om deel te kunnen nemen aan het ontwikkelingsproces. Indien u bijvoorbeeld wilt weten welke mogelijkheden de diverse custom chips in de MEGA65 hebben en welke registers welke taken uitvoeren, verwijzen we naar *het grote Mega65-boek* online op https://github.com/mega65/mega65-user-guide, of overweegt u een of meer van de naslagwerken voor de MEGA65 aan te schaffen, waaruit het Mega65-boek is samengesteld. Bijvoorbeeld de *Mega65-chipset naslaggids* of de *Mega65-assembleertaal naslaggids*.

We hopen ook dat u, de ontwikkelaarsgemeenschap, zal bijdragen aan die boeken, of uw eigen materiaal schrijft dat anderen kan helpen bij hun reis van het ontwikkelen voor de MEGA65.

DEEL II

PLATFORMONAFHANKELIJKE ONTWIKKELHULPMIDDELEN

HOOFDSTUK 4

Emulators

- De Xmega65-emulator gebruiken
- De Live ISO-image gebruiken

Op het moment van schrijven is er slechts één emulator voor de MEGA65, **xmega65**; de Xemu-emulatorsuite van LGB. De LGB-ontwikkelaars werken hard om bij te blijven met de ontwikkeling van de MEGA65; niettemin is sommige MEGA65-emulatie mogelijk niet nauwkeurig, maar zou deze voldoende moeten zijn voor softwareontwikkeling op de MEGA65.

Test software tijdens de ontwikkeling regelmatig op echte hardware, zoals een MEGA65 of een FPGA-board dat een MEGA65-core kan uitvoeren.

Download de broncode van de MEGA65-emulator van https://github.com/lgblgblgb/xemu.

Download voorgecompileerde versies van https://github.lgb.hu/xemu/. Installers zijn beschikbaar voor macOS, Windows en Linux.

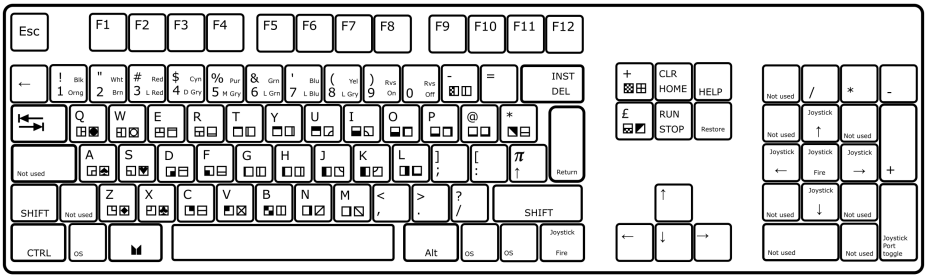
Een live ISO-image met de emulator, documentatie en andere hulpprogramma's is beschikbaar via Forum64.de op https://www.forum64.de/index.php?thread/104698-xemu-live-system-iso-file/&postID=1549927#post1549936. Indien u deze wilt gebruiken, vindt u de instructies op bladzijde 21.

DE XMEGA65-EMULATOR GEBRUIKEN

Het toetsenbord

De MEGA65-toetsenbordindeling is gebaseerd op de Commodore 65, wat afwijkt van een modern pc-toetsenbord. In de meeste gevallen herkent de Xmega65-emulator pc-toetsen als hun equivalente MEGA65-toetsen, zoals letters, cijfers en veelgebruikte modificatietoetsen zoals Shift. Xmega65 gaat uit van een Amerikaanse (VS-)indeling.

Sommige toetsen komen alleen op het MEGA65-toetsenbord voor en hebben geen pc-equivalenten. Xmega65 gebruikt de pc-toetsen Insert, Delete, Home, End, Page Up en Page Down voor +, £, CLR HOME, RUN STOP, HELP en RESTORE, respectievelijk.



Op de meeste pc-toetsenborden staan RUN STOP en RESTORE naast elkaar, zodat deze toetscombinatie eenvoudig is in te voeren.

Deze afbeelding toont de PETSCII-glyphen zoals ze op een MEGA65-toetsenbord verschijnen, met de  -glyphe links en de Shift-glyphe rechts. Merk op dat op een pc-toetsenbord de toets voor  rechts van de Shift staat.

Xemu biedt twee joysticks (een per keer) op het numerieke toetsenbord. Enter op het numerieke toetsenbord wordt gebruikt om bij herhaaldelijk drukken poorten 1 en 2 in en uit te schakelen.

Wees u ervan bewust dat de emulator standaard F9 F10 F11 als eigen sneltoetsen heeft gedefinieerd. Als u deze toetsen wilt gebruiken met de geëmulieerde MEGA65, moet u de toetsenbordindeling aanpassen. De bestanden staan in de map met de Xemu-instellingen zoals uitgelegd op bladzijde 21. Dit doet u door **keymap-default.cfg** te kopiëren naar **keymap.cfg** en vervolgens de volgende regels te verwijderen:

```
XEMU-EXIT F9
XEMU-FULLSCREEN F11
```

en te wijzigen in:

```
F9 Unknown
F11 Unknown
```

naar:

```
F9 F9
F11 F11
```

Nadat deze toetsen opnieuw zijn toegewezen, zijn de functies voor volledig scherm en afsluiten nog steeds beschikbaar via het menu. Klik met rechts (of Ctrl-klik) ergens in het venster om dit te openen.

Problemen met starten op Apple MacOS

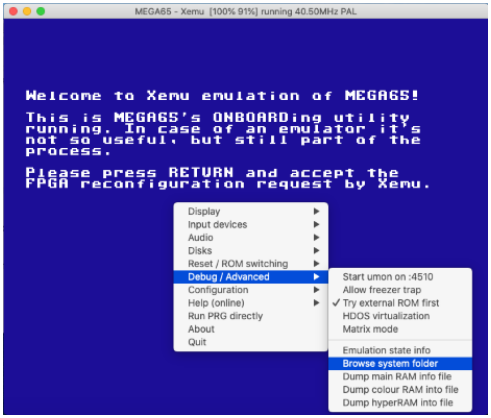
Als het bericht **macOS cannot verify that this app is free from malware** verschijnt, gebruik dan de volgende procedure wanneer u de app voor de eerste keer opent:

- Open Finder op uw Mac.
- Ctrl-klik (of rechtsklik) op het Xmega65-app-pictogram.
- Selecteer in het contextmenu **Open**.

- Het bericht verschijnt opnieuw, maar deze keer is er een knop **Open**. Klik op **Open**. U hoeft deze procedure slechts de eerste keer uit te voeren. Vanaf nu kunt u de app normaal openen.

Instellingen bijwerken

U kunt het Xemu-instellingsbestand bewerken om de toetsenbordindeling aan te passen of een reservekopie van het SD-kaart-imagebestand te maken. Open vanuit Xemu het menu om deze bestanden te openen: selecteer menu, selecteer **Debug / Advanced** en vervolgens **Browse system folder**. Bewerk de bestanden **keymap-default.cfg** en **mega65-default.cfg** met een teksteditor. Voor specifieke instructies over het wijzigen van **keymap-default.cfg** verwijzen we naar bladzijde 19.



De locatie van de Xemu-systeemmap verschilt per besturingssysteem:

- Windows (kan in de adresbalk van de verkenner worden gekopieerd): **%appdata%\xemu-lgb\mega65**
- Linux: **~/.xemu-lgb/**
- Macintosh: **~/Library/Application\ Support/xemu-lgb/mega65/**

DE LIVE ISO-IMAGE GEBRUIKEN

De Live ISO-image is het product van een vrijwillige gemeenschap; niet van het MEGA65-team. We nemen deze op voor uw gemak.

Een opstartbare USB-stick of DVD maken

Er zijn veel manieren om een live ISO-image te maken. De methode die u kiest hangt af van uw besturingssysteem en of u deze op een USB-station wilt installeren of op

een DVD wilt branden. Het branden op een DVD is eenvoudig, ervan uitgaande dat u een computer met een DVD-brander hebt. Als u een sneller opstartbaar USB-station wilt maken, probeert u een van de onderstaande methoden:

Als u Windows gebruikt, overweeg dan een hulpprogramma zoals http://www.isotousb.com/.

Op Linux kunt u de instructies op https://fossbytes.com/create-bootable-usb-media-from-iso-ubuntu/ volgen.

Voor Apple Macs zijn de instructies op https://ubuntu.com/tutorials/create-a-usb-stick-on-macos#1-overview beschikbaar.

Vergelijkbare instructies zijn beschikbaar voor andere populaire computers, zoals Amiga's (https://forum.hyperion-entertainment.com/viewtopic.php?t=3857) of Sun UltraSPARC-werkstations (https://forums.servethehome.com/index.php?threads/how-to-create-a-bootable-solaris-11-usb.1998/).

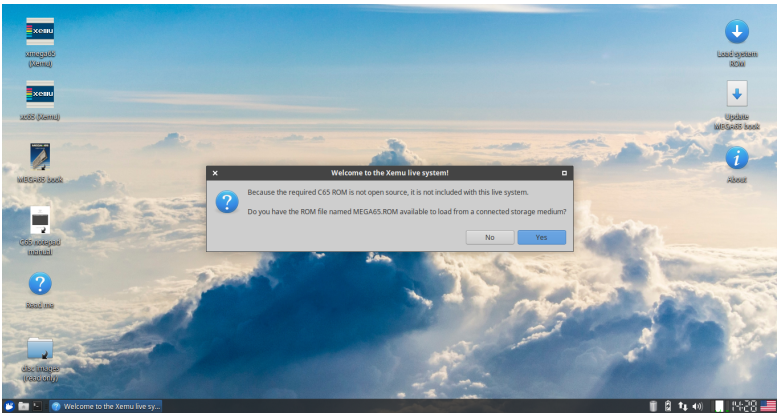
Tot slot is de populaire, eenvoudig te gebruiken en gratis cross-platform belanaEtcher beschikbaar op https://www.balena.io/etcher/.

Aan de slag

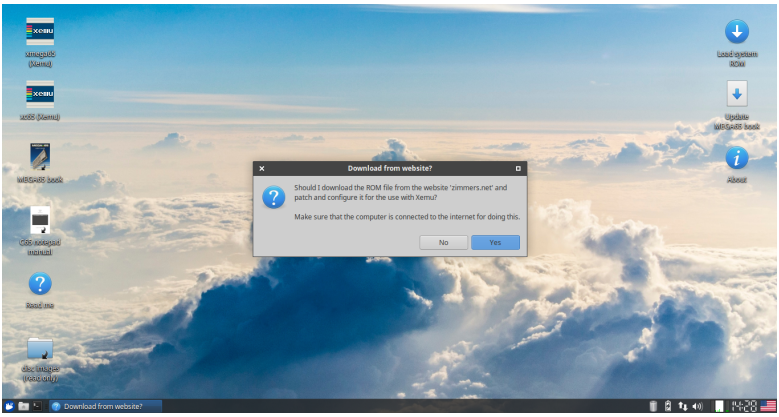
Om potentiële auteursrechtelijke kwesties te vermijden, bevat de opstartbare ISO-image geen propriëtaire ROM's voor de MEGA65, zoals oude C65-ROM's. Het bevat wel een vervangende open-source-ROM van ons OpenROMs-project. Deze ROM start op in een BASIC 2-omgeving waarmee u veel C64-programma's kunt laden en uitvoeren, zoals te zien is in de onderstaande afbeelding:



Als u een C65-ROM met BASIC 10 wilt gebruiken, download het juiste ROM-bestand en plaats het op een andere USB-stick met de naam **MEGA65.ROM**. Bij het opstarten zal de MEGA65 vragen of een ROM is gedownload; zoals te zien is in de onderstaande afbeelding:



Als de Live ISO geen ROM kan vinden, zal het u vragen om een ROM te downloaden; zoals hieronder te zien is:



Andere functies van de Live ISO

Zoals de bovenstaande schermafbeeldingen laten zien, biedt de Live ISO diverse handige bureaubladkortingen. Aan de linkerzijde staan koppelingen om de MEGA65-emulator en de C65-emulator te starten, zodat u kunt testen of programma's op beide platforms werken. Zoals eerder vermeld zijn beide emulators in ontwikkeling en mogelijk niet 100% compatibel.

Een andere koppeling geeft toegang tot het grote Mega65-boek. Deze omvattende band van ongeveer 800 pagina's bevat de officiële MEGA65-documentatie. Het grootste deel van deze handleiding voor ontwikkelaars is ook opgenomen in het grote Mega65-boek.

Deze ISO bevat ook documentatie voor de C65 Notepad; een programma voor de C65 en MEGA65 geschreven door Snoopy (de ontwikkelaar van de Live ISO-image). Een “lees mij”-bestand bevat verdere informatie over de Live ISO.

Tot slot zijn er aan de rechterzijde koppelingen om een C65-ROM te downloaden en het grote Mega65-boek bij te werken naar de nieuwste versie. Hiermee voorkomt u dat u steeds een nieuwe opstartbare image moet maken wanneer er een frequente update van het grote Mega65-boek wordt uitgebracht.

De volledige inhoud van de Live ISO-image is toegankelijk via de bestandsverkenner.

HOOFDSTUK 5

Gegevensoverdracht en foutopsporing

- m65-hulpprogramma
- M65Connect
- mega65_ftp
- TFTP-server
- Een BASIC-tekstbestandlisting omzetten naar een PRG-bestand

De sleutel tot effectieve cross-platform-ontwikkeling is het hebben van snelle en eenvoudige middelen om programmatuur op de MEGA65 te testen. Dit geldt zeker nu de MEGA65-emulator nog in ontwikkeling is. Zelfs wanneer de MEGA65-emulator voltooid is, zal het onwaarschijnlijk zijn dat deze volledige compatibiliteit op volledige snelheid kan bieden, omdat de MEGA65 veel veeleisender is om te emuleren dan de C64.

Er zijn verschillende hulpprogramma's die kunnen worden gebruikt voor gegevensoverdracht en foute opsporing. Deze maken doorgaans gebruik van ofwel de seriële monitorinterface van de MEGA65, of via de fast-ethernetadapter van de MEGA65. De seriële monitorinterface is beschikbaar via de UART-lijnen op de JB1- connector.

Als u geen toegang hebt tot de seriële monitorinterface, worden er hulpprogramma's ontwikkeld voor de fast-ethernetpoort die een deel, maar niet alles, van de mogelijkheden van de seriële monitor bieden. Deze worden gedocumenteerd zodra ze beschikbaar zijn. Het restant van dit hoofdstuk richt zich op methoden die de seriële monitorinterface gebruiken.

U kunt een 3.3V UART-adapter op de juiste lijnen aansluiten, of gemakkelijker, een TE-0790-03 JTAG-debugmodule op deze connector plaatsen. Dit geeft u een USB-verbinding die kan worden gebruikt voor het overzetten van software, op afstand fouten opsporen en geheugen inspecteren, alsmede het activeren of flashen van bitstreams. Met deze verbinding zijn de volgende hulpprogramma's beschikbaar:

M65-HULPPROGRAMMA

Het repository https://github.com/mega65/mega65-tools bevat een aantal hulpprogramma's, utilities en voorbeeldprogramma's. Deze hulpprogramma's zijn voornamelijk voor Linux maar kunnen worden gebruikt op Windows met Cygwin. Een van die hulpprogramma's is het **m65**-hulpprogramma. Dit is eigenlijk een zakmes verzameling van utilities in één. Veelgebruikte functies zijn onder meer:

Schermfoto's met het m65-hulpprogramma

Om een schermfoto van de MEGA65 te maken:

```
m65 -S
```

Dit maakt een bestand aan met de naam **mega65-screen-000000.png**, of als dit bestand al bestaat, wordt het eerste niet-gebruikte nummer gebruikt in plaats van **000000**.

Merk op dat deze schermfotofunctie werkt door **m65** de functionaliteit van de VIC-IV te laten emuleren. Hoewel het uitstekende digitale schermfoto's oplevert, komt het mogelijk niet exact overeen met het werkelijke scherm van de MEGA65. Op het moment van schrijven rendert het geen sprites of bitplanelen, alleen tekst- en bitmapgebaseerde videomodi.

Een programma laden en uitvoeren op de MEGA65

Om een programma op de MEGA65 te laden en uit te voeren, kunt u een commando als het volgende gebruiken:

```
m65 -F -4 -r foo.prg
```

De optie **-F** draagt **m65** op de MEGA65 te herstarten voordat het programma wordt geladen.

De optie **-4** draagt **m65** op de MEGA65 naar C64-modus te schakelen voordat het programma wordt geladen. Als dit wordt weggelaten, zal het proberen het programma in C65-modus te laden.

De optie **-r** draagt **m65** op het programma onmiddellijk na het laden uit te voeren.

Merk op dat dit commando werkt met het normale BASIC LOAD-commando en daarom beperkt is tot het laden van programma's in het onderste 64 KB van het geheugen.

De FPGA herconfigureren om een andere bitstream uit te voeren

Om een andere MEGA65-bitstream uit te proberen, kan een commando als het volgende worden gebruikt:

```
m65 -b bitstream.bit
```

Dit zal de genoemde bitstream naar de FPGA zenden. Omdat de FPGA door deze actie wordt herconfigureerd, zal het momenteel uitgevoerde programma niet alleen worden gestopt, maar ook het hoofdgeheugen zal worden gewist. Voor modellen van de MEGA65 die zijn uitgerust met 8 MB of 16 MB uitbreidingsgeheugen, worden deze uitbreidingsgeheugens geïmplementeerd in externe chips en dus zullen de inhoud daarvan niet worden gewist.

Voor niet-MEGA65-bitstreams (zoals zxunomega65 en gbc4mega65), gebruik in plaats daarvan het argument '-q':

```
m65 -q bitstream.bit
```

Toetsenbordinvoer op afstand

De toetsenbordinterface van de MEGA65 ondersteunt het overzetten van synthetische toetsgebeurtenissen via de registers $D615 - $D617. Het **m65**-hulpprogramma gebruikt dit om typen op afstand op de MEGA65 mogelijk te maken op een manier die transparant is voor software. Er zijn drie manieren om dit te gebruiken:

```
m65 -t ditiswattekst
```

Deze vorm typt de opgegeven tekst, in dit geval *ditiswattekst*, maar simuleert niet het indrukken van RETURN. Als u het indrukken van RETURN wilt simuleren, gebruikt u **-T** in plaats van **-t**, bijvoorbeeld:

```
m65 -T list
```

Dit zou het LIST-commando typen en uitvoeren.

Tot slot is het mogelijk om algemene toetsenbordbesturing op afstand te starten via:

```
m65 -t -
```

In deze modus wordt elke toets, die op het toetsenbord van de computer waarop **m65** draait, doorgegeven aan de MEGA65. Merk op dat niet alle speciale toetsen worden ondersteund en dat er enige vertraging is, dus het gebruik van de toetsherhalingsfunctie kan onverwachte resultaten opleveren. Voor besturing op afstand in het algemeen is het echter een zeer nuttige voorziening.

Unit-testing en logboekondersteuning

Het **m65**-hulpprogramma ondersteunt unit-testing direct op MEGA65-hardware. Wanneer de *unit testing-modus* actief is, wacht **m65** tot de MEGA65 bepaalde bytereeksen via de seriële interface stuurt die de huidige status (gestart, geslaagd, mislukt) signaleren van een gegeven test. Daarnaast is het mogelijk om logboekberichten te verzenden vanaf de MEGA65 naar de hostcomputer.

De unittestingmodus wordt betreden door **m65** aan te roepen met de vlag **-u**. Om een BASIC-programma op afstand in C65-modus uit te voeren en tegelijk **m65** in de *unittestingmodus* te zetten, kan het volgende commando worden gebruikt:

```
    m65 -Fur attic-ram.prg -w tests.log
```

De opties **-F** en **-r** vertellen **m65** om de MEGA65 te herstarten voordat het programma "attic-ram.prg" wordt geladen en automatisch uitgevoerd. De optie **-u** vertelt **m65** vervolgens om in de *unittestingmodus* te gaan in plaats van te stoppen na het starten van het programma. De optionele optie **-w** zorgt ervoor dat **m65** de testresultaten aan het bestand "test.log" toevoegt (het bestand aanmaakt als het nog niet bestaat).

Merk op dat **m65** automatisch de *unittestingmodus* verlaat als er gedurende meer dan 10 seconden geen teststatussignalen zijn ontvangen.

Er is ondersteuning voor het verzenden van unittestingsignalen naar de hostcomputer vanuit C- en BASIC 65-programma's:

Unit-tests gebruiken met C

De MEGA65 libc bevat ondersteuning voor unit-testing via functies gedefinieerd in **tests.h** en **tests.c**.

Om het begin van een test te signaleren, includeert u **tests.h** en gebruikt u:

```
unit_test_setup("testName",issueNumber);
```

waarbij *"testName"* een leesbare naam van de test is (bijvoorbeeld "VIC-II") en *issueNumber* een verwijzing naar het bijbehorende bug-issue (bijvoorbeeld het issue-nummer van GitHub).

Na het starten van een test kunnen geslaagde tests worden gesignaleerd met de functie unit_test_ok():

```
unit_test_ok();
```

Een mislukte test wordt gesignaleerd met unit_test_fail():

```
unit_test_fail("fail message");
```

Elke keer dat de functies unit_test_ok() of unit_test_fail() worden aangeroepen, wordt het *sub-issue* van de test (gerapporteerd op de hostcomputer) verhoogd. Dit maakt het eenvoudiger om meerdere tests in één bestand te combineren en te identificeren.

U kunt willekeurige logboekberichten verzenden via unit_test_log():

```
unit_test_log("hello world from mega65!");
```

...en ten slotte, wanneer alles klaar is, wordt het einde van unit-testing gesignaleerd door het gebruik van:

```
unit_test_done();
```

Unit-tests gebruiken met BASIC 65

b65support.bin is een machinetaalmodule die ondersteuning biedt voor unit-testing vanuit BASIC 65. Deze staat in het mega65-tools- repository op https://github.com/MEGA65/mega65-tools en kan worden gebouwd met:

```
make bin65/b65support.bin
```

Het bestand **b65support.bin** is dan beschikbaar in de map **bin65**.

Deze module werkt door de USR-vector om te leiden om de functies uit te voeren die nodig zijn om te communiceren met de test-host.

In een geautomatiseerd testsenario wilt u mogelijk het binaire bestand **b65support.bin** overzetten naar het MEGA65-geheugen door middel van **m65**:

```
m65 -@ mega65-tools/bin65/b65support.bin@15fe
```

Het is natuurlijk ook mogelijk om **b65support.bin** rechtstreeks vanaf de MEGA65 te laden door de **M65UTILS.D81**-image van de Freezer te koppelen en het volgende commando te geven:

```
BLOAD "B65SUPPORT.BIN"
```

Na het laden wordt **b65support.bin** geïnitialiseerd met:

```
SYS $1600
```

Na initialisatie biedt **b65support.bin** de volgende functies:

```
A=USR((issueNum))
```

bereidt een nieuwe test voor met nummer <issueNum> en herstelt het subissue-nummer naar 0.

```
A=USR("=(testName)")
```

stelt de testnaam in en stuurt het testsignaal; bijvoorbeeld: **A=USR("=VIC-III")** stelt de testnaam in op 'VIC-III' en signaleert de hostcomputer dat de test is gestart.

```
A=USR("/(logMessage)")
```

stuurt een logboekbericht naar de hostcomputer.

```
A=USR("P")
```

stuurt het signaal 'geslaagd' naar de hostcomputer en verhoogt het subissue-nummer.

```
A=USR("F")
```

stuurt het signaal 'test mislukt' naar de hostcomputer en verhoogt het subissue-nummer.

```
A=USR("D")
```

stuurt het signaal 'test voltooid' naar de hostcomputer.

Alle aanroepen retourneren het huidige subissue-nummer of **?ILLEGAL QUANTITY ERROR** in geval van een ongeldig commando.

BASIC 65-voorbeeld

Het volgende is een volledig BASIC 65-voorbeeld dat laat zien hoe de unittesting-functies van **m65** worden gebruikt:

```
100 rem attic ram cache test
110 poke $bfffff2,$e0          : rem schakel cache van zolder-ram in
120 sys $1600                  : rem initialiseer testmodule
130 a=usr(379)                 : rem zet issue-nummer
140 a=usr("=attic-ram-cache")  : rem zet testnaam
150 bank128:poke0,65           : rem uit voorzorg
160 b0=$8000000 : b1=$8000100 : rem zolden-ram-gebieden om te testen
170 for r=0 to $ff
180   poke b0+r,0              : rem vul gebied 1 met 0
190   poke b1+r,$ff            : rem vul gebied 2 met $ff
200 next r
210 for t=1 to 10              : rem 10 pogingen
220   poke b0,32               : rem schrijf naar b0
230   for x=0 to $ff:t1=b1+x
240     a=peek(b0)             : rem lees uit b0
250     b=peek(t1):b=peek(t1)  : rem lees tweemaal uit t1
260     ifb()255 thenf=t:t=11:x=256   : rem dit mag niet gebeuren
270   next x
280 next t
290 if f=0 then begin
300   print "geen fouten gevonden na";t;"pogingen."
310   a=usr("p")               : rem signal 'test passed' to host
320 bend : else begin
330   a=usr("f")               : rem signal 'test failed' to host
340   print "hyper-ram-fout gedetecteerd na";f;"pogingen."
350   print "peek($";hex$(t1);") [t1] is";b;"maar moet 255 zijn"
360 bend
370 a=usr("d")                 : rem test gereed
```

M65CONNECT

Dit is een cross-platform grafisch hulpprogramma dat beschikbaar is voor Windows, Linux en MacOSX, waarmee toegang wordt geboden tot de meeste functies van het **m65**-hulpprogramma, zonder dat een commandoregel nodig is, of zonder dat u het hulpprogramma voor uw favoriete besturingssysteem hoeft te compileren.

Het repository voor M65Connect is: https://github.com/MEGA65/m65connect

De nieuwste binaire versie is beschikbaar op https://files.mega65.org.

Met de MEGA65 of Nexys FPGA uitgeschakeld, sluit een USB-kabel aan van uw computer op de MEGA65 of Nexys FPGA-board. Voer het *M65Connect*-programma uit en volg de aanwijzingen om verbinding te maken. Het programma helpt u bij het identificeren van welke USB Serial Port moet worden gebruikt.

Met dit hulpprogramma kunt u eenvoudig PRG-programma's en diverse andere bestanden overdragen. M65Connect kan de overdracht, het schakelen naar C64-modus en de uitvoering van programma's afhandelen.

MEGA65_FTP

Het **mega65_ftp**-hulpprogramma van het repository https://github.com/mega65/mega65-tools is enigszins misleidend genoemd: hoewel het een File Transfer Program is, gebruikt het niet het File Transfer Protocol (FTP). Het gebruikt in plaats daarvan de seriële monitorinterface om de MEGA65 op afstand te besturen en heeft directe toegang tot de SD-kaart om het kopiëren van bestanden tussen de MEGA65 en de hostcomputer mogelijk te maken.

Merk op dat het de staat van de MEGA65 niet perfect herstelt bij het afsluiten, en dus alleen moet worden gebruikt wanneer de MEGA65 bij de READY.-prompt staat, zodat eventuele draaiende software niet in de war raakt. Met name moet u het gebruik vermijden wanneer een gevoelig programma draait, zoals het Freeze-menu, het MEGA65-configuratiehulpprogramma of het MEGA65 Format/FDISK-hulpprogramma. (Dit probleem kan met weinig inspanning worden opgelost, als iemand de tijd en interesse heeft om het te fixen).

Bij uitvoering biedt het een FTP-achtige interface die de commando's **get**, **put**, **rename** en **dir** ondersteunt. Merk op dat u bij het plaatsen van een bestand moet zorgen dat het een naam heeft die volledig in hoofdletters staat en een DOS-compatibele 8.3-teken bestandsnaam heeft. Dit komt door beperkingen in zowel **mega65_ftp** als de VFAT32-bestandssysteemcode van de Hypervisor van de MEGA65. Opnieuw kunnen deze problemen worden opgelost met een bescheiden hoeveelheid inspanning van de kant van een gemotiveerd lid van de gemeenschap.

Tot slot is het programma **mega65_ftp** *zeer* langzaam bij het pushen van nieuwe bestanden naar de MEGA65, met doorgaans snelheden van rond de 5 KB/sec. Dit komt deels doordat de seriële interface in staat is om gegevens over te dragen met slechts 40 KB/sec (bij 4.000.000 bits per seconde), en deels doordat het besturingsproces op afstand resulteert in veel heen-en-weerverkeer waarbij hulpoutines op de MEGA65 worden uitgevoerd om sectoren op de SD-kaart te lezen, schrijven en verifiëren. Het zou heel haalbaar zijn om dit te verbeteren tot bijna 40 KB/sec, en mogelijk sneller met een combinatie van gegevenscompressie, deduplicatie van identieke sectoren (met name bij het uploaden van schijf- images) en andere technieken. Opnieuw zou dit een zeer welkome bijdrage zijn die iemand uit de gemeenschap zou kunnen leveren ten algemene nutte.

TFTP-SERVER

Er is een echte TFTP-server voor de MEGA65 in ontwikkeling die snelle TFTP-overdrachten via de 100 Mbit-ethernet ondersteunt en die kan worden gebruikt om

zeer snel bestanden van de MEGA65 te lezen. Snelheden van bijna 1 MB/sec zijn mogelijk, afhankelijk van de prestaties van de SD-kaart. In plaats van DHCP te gebruiken, zal dit hulpprogramma reageren op *elk* IP-adres dat eindigt op .65. Het gebruikt altijd het MAC-adres 40:40:40:40:40:40. Echte DHCP- ondersteuning en het gebruik van het geconfigureerde ethernet-MAC-adres van de MEGA65 worden mogelijk in de toekomst toegevoegd.

Belangrijker nog, de ondersteuning voor het schrijven van bestanden naar de SD-kaart is nog niet volledig en wordt geblokkeerd door de behoefte aan de implementatie van de nodige functies in de Hypervisor van de MEGA65 voor het aanmaken en vergrootten van bestanden. Een bijzondere uitdaging is het mogelijk maken van het aanmaken van bestanden met aaneengesloten clusters zoals vereist voor D81-schijfimages: als een D81-bestand gefragmenteerd is, kan het niet worden gekoppeld, omdat het koppelingsmechanisme een pointer naar het aaneengesloten blok van de SD-kaart vereist dat de schijfimage bevat. In de tussentijd kan **`mega65_ftp`** als alternatief worden gebruikt.

EEN BASIC-TEKSTBESTANDLISTING OMZETTEN NAAR EEN PRG-BESTAND

Als u een ongetokeniseerd BASIC-programma in platte-tekstformaat hebt dat afkomstig is van bijvoorbeeld een internetbericht, en u wilt het op de MEGA65 proberen zonder het in te typen, dan is het mogelijk het om te zetten naar een PRG.

C64List is een Windows-gebaseerd commandoregelhulpprogramma waarmee u de conversie kunt uitvoeren. Zodra u een *.PRG*-bestand hebt, kunt u een hulpprogramma zoals M65Connect gebruiken om het naar de MEGA65 of Nexys FPGA te uploaden.

C64List is beschikbaar voor download op `http://www.commodoreserver.com/Downloads.asp`

Zorg ervoor dat u een programmalijst in een bestand op uw lokale computer hebt opgeslagen (bijvoorbeeld *program.txt*) gecodeerd als ANSI of UTF8.

Gebruik C64List om het bestand om te zetten naar een PRG-bestand:

```
C64List program.txt -prg
```

Nu kunt u uw nieuw geconverteerde programma uploaden naar de MEGA65 met M65Connect of een van de eerder beschreven hulpprogramma's.

Het is vermeldenswaard dat deze methode niet 100% effectief zal zijn bij listings met speciale PETSCII-tekens. Programma's met PETSCII vereisen enige bewerking op de MEGA65 zelf voordat ze op schijf worden opgeslagen.

HOOFDSTUK 6

Assemblers

Onderstaande tabel geeft een overzicht van assemblers waarvan bekend is dat ze met de MEGA65 werken. Voor algemeen gebruik raden we **ACME** aan, omdat het goede ondersteuning biedt voor de 45GS02-instructieset, open source is en bovendien in C is geschreven. Dat laatste betekent dat de assembler op termijn naar de MEGA65 geporteerd zou kunnen worden om er native te draaien.

Naam	45GS02	Bron	Verwijzing
ACME	ja	C	https://sourceforge.net/projects/acme-crossass
KickAss	ja	Java	gitlab.com/jespergravgaard/kickassembler65ce02
LLVM-MOS	ja	C++	https://llvm-mos.org/wiki/Welcome
Ophis	ja	Python	https://github.com/michaelcmartin/Ophis
BSA	ja	C	https://github.com/Edilbert/BSA
CA65	nee[1]	C	https://github.com/mega65/cc65

De **BSA**-assembler wordt momenteel gebruikt om de **MEGA65.ROM** te bouwen. Het grootste deel van de broncode is geschreven in de syntaxis van de oude **BSO**-assembler (Boston Systems Office), die in de jaren 1989-1991 werd gebruikt door softwareontwikkelaars aan de C65. De **BSA**-assembler heeft een compatibiliteitsmodus waarmee deze oude broncodes met weinig of geen aanpassingen kunnen worden geassembleerd. De **BSA**-assembler bevat momenteel alleen een beschrijving van de commando's in de C-broncode van de assembler.

Het **LLVM-MOS**-project wordt geleverd met een GNU-compatibele macro-assembler en -disassembler die alle mnemonieken van de 45GS02 ondersteunt. Het kan als zelfstandig hulpprogramma worden gebruikt met het commando **llvm-mc**.

[1] Onze fork van CA65 (onderdeel van CC65) herkent de CPU van de MEGA65 correct, maar heeft geen expliciete ondersteuning voor de mogelijkheden van de processor

HOOFDSTUK 7

C- en C-achtige compilers

- MEGA65 libc

Kort antwoord: CC65 en KickC werken beide op de MEGA65.

Zowel CC65 als KickC werken op de MEGA65, maar beide hebben standaard alleen een C64-geheugenmodel en gebruiken uitsluitend 6502-opcodes. Het zou geweldig zijn als iemand een C65-geheugenconfiguratie voor CC65 zou ontwikkelen; dat zou niet al te moeilijk moeten zijn.

CC65 ondersteunt overlays, wat krachtig kan worden ingezet met het extra geheugen van de MEGA65 om programma's groter dan 64 KB mogelijk te maken. Daarvoor is echter een geschikte loader nodig, die nog niet bestaat.

Op dezelfde manier zou het aanpassen van de codegenerator van CC65 om 45GS02-mogelijkheden te gebruiken niet bijzonder moeilijk zijn en zou het helpen om de anders vreselijk trage en omvangrijke code die CC65 produceert te overwinnen. Het toevoegen van volledige ondersteuning voor de 45GS02-CPU-mogelijkheden in CA65 (of nog beter: het laten produceren van ACME-compatibele assembleeruitvoer door CC65) zou van overweldigend voordeel zijn en niet bijzonder moeilijk te realiseren. Dit zouden allemaal geweldige taken zijn om mee bezig te zijn terwijl u wacht op uw MEGA65 DevKit!

Een voorbeeldsjabloon voor een C-programma dat met CC65 kan worden gecompileerd en op de MEGA65 kan worden uitgevoerd, staat in het repository `https://github.com/MEGA65/hello-world`. Dit repository kan zelfs CC65 downloaden en compileren als u dat nog niet op uw systeem hebt geïnstalleerd. Het repository werkt op Linux en Mac en op Windows via het Windows Subsystem for Linux (WSL).

Het LLVM-MOS-project (`https://llvm-mos.org`) heeft een MEGA65-doelplatform dat is getest met C99, C++11 en Rust. De LLVM MOS-compiler heeft ondersteuning voor 45GS02-instructies (`-mcpu=mos45gs02`) en wordt geleverd met een assembler en disassembler. Voorgecompileerde binaire bestanden van de SDK zijn beschikbaar voor Linux, Windows en macOS.

MEGA65 LIBC

Er wordt een C-bibliotheek ontwikkeld voor de MEGA65, die al een aantal nuttige functies bevat. Deze bibliotheek is beschikbaar via `http://github.com/mega65/mega65-libc`. De procedures, functies en definities die het biedt, zijn gedocumenteerd in een apart hoofdstuk.

De MEGA65 libc is momenteel alleen beschikbaar voor LLVM en CC65, hoewel we een KickC-port hiervan zeer op prijs zouden stellen.

HOOFDSTUK 8

MEGA65 Standard C Library

- **Structuur en gebruik**
- **conio.h**

Er wordt een C-bibliotheek ontwikkeld voor de MEGA65, die al een aantal nuttige functies bevat. Deze bibliotheek is beschikbaar via http://github.com/mega65/mega65-libc. De procedures, functies en definities die zij biedt, worden gedocumenteerd in een apart hoofdstuk.

STRUCTUUR EN GEBRUIK

De MEGA65-libc wordt bewust alleen als broncode aangeboden, met functiegroepen in aparte bestanden en afzonderlijke headerbestanden voor inclusie. Het idee is dat u alleen de headerbestanden opneemt die u nodig heeft, en alleen de benodigde bronbestanden toevoegt aan de lijst van bronbestanden van het programma dat u compileert. Daarmee voorkomt u dat de compiler functies in uw gecompileerde programma opneemt die nooit worden gebruikt, waardoor kostbaar geheugen verspild wordt.

Let op dat sommige bronbestanden van de bibliotheek in C geschreven zijn en dus bestanden met de extensie `.c` betreffen, terwijl andere in assemblerertaal geschreven zijn, hetzij vanwege efficiëntie, hetzij uit noodzaak, en de extensie `.s` hebben.

Gangbaar gebruik is de broncode van mega65-libc uit te checken in een naburige map of binnen de bronmap van uw eigen project. In het laatste geval kan dit met behulp van de git submodule-faciliteit.

De volgende secties documenteren elk van de headerbestanden en de bijbehorende functies die zij bieden.

CONIO.H

conionit

Beschrijving: Initialiseert de interne status van de bibliotheek

Syntaxis: `void conioinit(void)`

Opmerkingen:
U moet deze functie aanroepen voordat u een conio-bibliotheekfunctie gebruikt.

setscreenaddr

Beschrijving: Stelt het startadres van de screen RAM in

Syntaxis: `void setscreenaddr(long addr);`

Parameters: **addr:** Het adres dat als begin van de screen RAM ingesteld moet worden

Opmerkingen:
Er wordt geen grenscontrole uitgevoerd op het geselecteerde adres

getscreenaddr

Beschrijving: Retourneert het startadres van de screen RAM

Syntaxis: `long getscreenaddr(void);`

Resultaat: Het huidige startadres van de screen RAM.

setcolramoffset

Beschrijving: Stelt de offsetwaarde van het begin van de color RAM in

Syntaxis: `void setcolramoffset(long offset);`

Parameters: **addr:** De offset vanaf het begin van de color RAM ($FF80000)

Opmerkingen:
Er wordt geen grenscontrole uitgevoerd op het resulterende adres. Overschrijd de beschikbare grootte van de color RAM niet

getcolramoffset

Beschrijving: Retourneert de offsetwaarde van het begin van de color RAM

Syntaxis: `long getscreenaddr(void);`

Resultaat: De huidige offsetwaarde van het begin van de color RAM.

setcharsetaddr

Beschrijving: Stelt het startadres van de tekenset in

Syntaxis: `void setcharsetaddr(long addr);`

Parameters: **addr:** Het adres dat als begin van de tekenset ingesteld moet worden

Opmerkingen:
Er wordt geen grenscontrole uitgevoerd op het geselecteerde adres

getcharsetaddr

Beschrijving: Retourneert het huidige startadres van de tekenset

Syntaxis: `long getscreenaddr(void);`

Resultaat: Het huidige startadres van de tekenset.

clrscr

Beschrijving: Maakt het tekstscherm leeg.

Syntaxis: `void clrscr(void)`

Opmerkingen:
De color RAM wordt met de huidige tekstkleur leeggemaakt

getscreensize

Beschrijving: Retourneert de afmetingen van het tekstscherm

Syntaxis: `void getscreensize(unsigned char* width, unsigned char* height)`

Parameters: **width:** Aanwijzer naar de locatie waar de breedte wordt teruggegeven

height: Aanwijzer naar de locatie waar de hoogte wordt teruggegeven

setscreensize

Beschrijving: Stelt de afmetingen van het tekstscherm in

Syntaxis: `void setscreensize(unsigned char width, unsigned char height)`

Parameters: **width:** De breedte in kolommen (40 of 80)

height: De hoogte in rijen (25 of 50)

Opmerkingen:
Momenteel worden alleen 40/80 en 25/50 aanvaard. Andere waarden worden genegeerd.

set16bitcharmode

Beschrijving: Schakelt de 16-bits tekenmodus in of uit

Syntaxis: `void set16bitcharmode(unsigned char f)`

Parameters: **f:** Stel true in om de 16-bits tekenmodus in te schakelen

Opmerkingen:
Dit leidt tot een reset van de videoparameters indien HOTREG is ingeschakeld. Zie de functie sethotregs.

sethotregs

Beschrijving: Schakelt het hot-registergedrag van de VIC-IV-chip in of uit.

Syntaxis: `void set16bitcharmode(unsigned char f)`

Parameters: **f:** Stel true in om het hotreg-gedrag in te schakelen

Opmerkingen:
Indien deze modus is ingeschakeld, wordt een reset van de videomodus geactiveerd bij toegang tot $D011, $D016, $D018, $D031 of de VIC-II-bankbits van $DD00.

setextendedattrib

Beschrijving: Schakelt de uitgebreide attributenmodus van de VIC-III in of uit ter ondersteuning van knipperen, onderstrepen, vet en markeren.

Syntaxis: `void setextendedattrib(unsigned char f)`

Parameters: **f:** Stel true in om de uitgebreide attributenmodus in te schakelen

togglecase

Beschrijving: Stelt de tekenset met kleine letters in

Syntaxis: `void setlowercase(void)`

togglecase

Beschrijving: Stelt de tekenset met hoofdletters in

Syntaxis: `void setuppercase(void)`

togglecase

Beschrijving: Wisselt de huidige tekenset tussen kleine en hoofdletters

Syntaxis: `void togglecase(void)`

bordercolor

Beschrijving: Stelt de huidige randkleur in

Syntaxis: `void bordercolor(unsigned char c)`

Parameters: **c:** De in te stellen kleur

bgcolor

Beschrijving: Stelt de huidige schermkleur (achtergrondkleur) in

Syntaxis: `void bgcolor(unsigned char c)`

Parameters: **c:** De in te stellen kleur

textcolor

Beschrijving: Stelt de huidige tekstkleur in

Syntaxis: `void textcolor(unsigned char c)`

Parameters: **c:** De in te stellen kleur

Opmerkingen:
Deze functie behoudt de attributen in de bovenste 4 bits indien uitgebreide attributen zijn ingeschakeld. Zie setextendedattrib.

revers

Beschrijving: Schakelt het reverse-attribuut in

Syntaxis: `void revers(unsigned char c)`

Parameters: **enable:** 0 om uit te schakelen, 1 om in te schakelen

Opmerkingen:
De uitgebreide attributenmodus moet actief zijn. Zie setextendedattrib.

highlight

Beschrijving: Schakelt het highlight-attribuut in

Syntaxis: `void highlight(unsigned char c)`

Parameters: **enable:** 0 om uit te schakelen, 1 om in te schakelen

Opmerkingen:
De uitgebreide attributenmodus moet actief zijn. Zie setextendedattrib.

blink

Beschrijving: Schakelt het knipperattribuut in

Syntaxis: `void blink(unsigned char c)`

Parameters: **enable:** 0 om uit te schakelen, 1 om in te schakelen

Opmerkingen:
De uitgebreide attributenmodus moet actief zijn. Zie setextendedattrib.

underline

Beschrijving: Schakelt het onderstrepingsattribuut in

Syntaxis: `void underline(unsigned char c)`

Parameters: **enable:** 0 om uit te schakelen, 1 om in te schakelen

Opmerkingen:
De uitgebreide attributenmodus moet actief zijn. Zie setextendedattrib.

altpal

Beschrijving: Schakelt het attribuut van het alternatieve palet in

Syntaxis: `void altpal(unsigned char c)`

Parameters: **enable:** 0 om uit te schakelen, 1 om in te schakelen

Opmerkingen:
De uitgebreide attributenmodus moet actief zijn. Zie setextendedattrib.

clearattr

Beschrijving: Wist alle tekstattributen

Syntaxis: `void clearattr())`

Opmerkingen:
De uitgebreide attributenmodus moet actief zijn. Zie setextendedattrib.

cellcolor

Beschrijving: Stelt de kleur van een tekencel in

Syntaxis: `void cellcolor(unsigned char x, unsigned char y, unsigned char c)`

Parameters: **x:** De X-coördinaat van de cel

y: De Y-coördinaat van de cel

c: De in te stellen kleur

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

setpalbank

Beschrijving: Stelt de huidige tekst-/bitmappaletbank in (BTPALSEL).

Syntaxis: `void setpalbank(unsigned char bank)`

Parameters: **bank:** De in te stellen paletbank. Geldige waarden zijn 0, 1, 2 of 3.

Opmerkingen:
Gebruik setpalbanka om een alternatief tekst-/bitmappalet in te stellen

setpalbanka

Beschrijving: Stelt de alternatieve tekst-/bitmappaletbank in.

Syntaxis: `void setpalbanka(unsigned char bank)`

Parameters: **bank:** De in te stellen paletbank. Geldige waarden zijn 0, 1, 2 of 3.

Opmerkingen:
Gebruik setpalbank om het hoofdtekst-/bitmappalet in te stellen

getpalbank

Beschrijving: Haalt de geselecteerde tekst-/bitmappaletbank op.

Syntaxis: `unsigned char getpalbank(void)`

Opmerkingen:
Gebruik getpalbanka om het geselecteerde alternatieve tekst-/bitmappalet op te halen

Resultaat: De huidige geselecteerde hoofdtekst-/bitmappaletbank.

getpalbanka

Beschrijving: Haalt de geselecteerde alternatieve tekst-/bitmappaletbank op.

Syntaxis: `unsigned char getpalbanka(void)`

Opmerkingen:
Gebruik getpalbank om het geselecteerde hoofdtekst-/bitmappalet op te halen

Resultaat: De huidige geselecteerde alternatieve tekst-/bitmappaletbank.

setmapedpal

Beschrijving: Stelt de ingemapte paletbank in op $D100-$D3FF.

Syntaxis: `void setmapedpal(unsigned char bank)`

Parameters: **bank:** De in te mappen paletbank. Geldige waarden zijn 0, 1, 2 of 3.

getmapedpal

Beschrijving: Haalt de ingemapte paletbank op $D100-$D3FF.

Syntaxis: `unsigned char getmapedpal(void)`

setpalentry

Beschrijving: Stelt een paletkleur in voor het ingemapte palet

Syntaxis: `void setpalentry(unsigned char c, unsigned char r, unsigned char g, unsigned char b)`

Parameters: **c:** De index van de paletkleur (0-255)

r: De rode kleurcomponent

g: De groene kleurcomponent

b: De blauwe kleurcomponent

Opmerkingen:
Gebruik setmapedmal om het te wijzigen palet in te mappen

fillrect

Beschrijving: Vult een rechthoekig gebied met een teken en een kleurwaarde

Syntaxis: `void fillrect(const RECT *rc, unsigned char ch, unsigned char col)`

Parameters: **rc:** Een RECT-structuur die de coördinaten van de rechthoek opgeeft

ch: Een tekencode om de rechthoek mee te vullen

col: De kleur om mee te vullen

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

box

Beschrijving: Tekent een kader met grafische tekens

Syntaxis: `void box(const RECT *rc, unsigned char color, unsigned char style, unsigned char clear, unsigned char shadow)`

Parameters: **rc:** Een RECT-structuur die de coördinaten van het kader opgeeft

color: De kleur die voor de grafische tekens gebruikt wordt

style: De stijl voor de randen van het kader. Kan worden ingesteld op BOX_STYLE_NONE, BOX_STYLE_ROUNDED, BOX_STYLE_INNER, BOX_STYLE_OUTER, BOX_STYLE_MID

clear: Zet op 1 om het binnenvlak van het kader te wissen met de geselecteerde kleur

shadow: Zet op 1 om een slagschaduw te tekenen

Opmerkingen:

Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

hline

Beschrijving: Tekent een horizontale lijn.

Syntaxis: `void hline(unsigned char x, unsigned char y, unsigned char len, unsigned char style)`

Parameters:

x: De X-coördinaat van het begin van de lijn

y: De Y-coördinaat van het begin van de lijn

len: De lengte van de lijn

style: De stijl voor de lijn. Zie de HLINE_-constanten voor de beschikbare stijlen.

Opmerkingen:

Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

vline

Beschrijving: Tekent een verticale lijn.

Syntaxis: `void vline(unsigned char x, unsigned char y, unsigned char len, unsigned char style)`

Parameters:

x: De X-coördinaat van het begin van de lijn

y: De Y-coördinaat van het begin van de lijn

len: De lengte van de lijn

style: De stijl voor de lijn. Zie de VLINE_-constanten voor de beschikbare stijlen.

Opmerkingen:

Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

gohome

Beschrijving: Stelt de huidige positie in op de thuispositie (coördinaat 0,0)

Syntaxis: `void gohome(void)`

gotoxy

Beschrijving: Stelt de huidige positie in op de coördinaten X,Y

Syntaxis: `void gotoxy(unsigned char x, unsigned char y)`

Parameters: **x:** De nieuwe X-coördinaat

y: De nieuwe Y-coördinaat

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

gotox

Beschrijving: Stelt de X-coördinaat van de huidige positie in

Syntaxis: `void gotox(unsigned char x)`

Parameters: **x:** De nieuwe X-coördinaat

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

gotoy

Beschrijving: Stelt de Y-coördinaat van de huidige positie in

Syntaxis: `void gotoy(unsigned char y)`

Parameters: **y:** De nieuwe Y-coördinaat

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

moveup

Beschrijving: Verplaatst de huidige positie naar boven

Syntaxis: `void moveup(unsigned char count)`

Parameters: **count:** Het aantal te verplaatsen posities

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

movedown

Beschrijving: Verplaatst de huidige positie naar beneden

Syntaxis: `void movedown(unsigned char count)`

Parameters: **count:** Het aantal te verplaatsen posities

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

moveleft

Beschrijving: Verplaatst de huidige positie naar links

Syntaxis: `void moveleft(unsigned char count)`

Parameters: **count:** Het aantal te verplaatsen posities

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

moveright

Beschrijving: Verplaatst de huidige positie naar rechts

Syntaxis: `void moveright(unsigned char count)`

Parameters: **count:** Het aantal te verplaatsen posities

Opmerkingen:
Er worden geen grenscontroles op de schermgrenzen uitgevoerd; het gedrag buiten het scherm is ongedefinieerd

wherex

Beschrijving: Retourneert de X-coördinaat van de huidige positie

Syntaxis: `unsigned char wherex(void)`

Resultaat: De X-coördinaat van de huidige positie

wherey

Beschrijving: Retourneert de Y-coördinaat van de huidige positie

Syntaxis: `unsigned char wherey(void)`

Resultaat: De Y-coördinaat van de huidige positie

pcputc

Beschrijving: Geeft één PETSCII-teken weer op het scherm op de huidige positie

Syntaxis: `void cputc(unsigned char c)`

Parameters: **c:** Het weer te geven PETSCII-teken

pcputsxy

Beschrijving: Geeft een PETSCII-string weer op de coördinaten X,Y

Syntaxis: `void pcputsxy (unsigned char x, unsigned char y, const unsigned char* s)`

Parameters: **x:** De X-coördinaat waar de string wordt weergegeven

y: De Y-coördinaat waar de string wordt weergegeven

s: De weer te geven PETSCII-string

Opmerkingen:

Er wordt geen pointercontrole uitgevoerd. Indien s null of ongeldig is, is het gedrag ongedefinieerd

cputcxy

Beschrijving: Geeft één PETSCII-teken weer op de coördinaten X,Y

Syntaxis: `void pcputcxy (unsigned char x, unsigned char y, unsigned char c)`

Parameters: **x:** De X-coördinaat waar het teken wordt weergegeven

y: De Y-coördinaat waar het teken wordt weergegeven

c: Het weer te geven PETSCII-teken

pcputs

Beschrijving: Geeft een PETSCII-string weer op de huidige positie

Syntaxis: `void pcputs(const unsigned char* s)`

Parameters: **s:** De weer te geven string

Opmerkingen: Er wordt geen pointercontrole uitgevoerd. Indien s null of ongeldig is, is het gedrag ongedefinieerd

cputc

Beschrijving: Geeft één schermcodeteken weer op het scherm op de huidige positie

Syntaxis: `void cputc(unsigned char c)`

Parameters: **c:** De schermcode van het weer te geven teken

cputnc

Beschrijving: Geeft N exemplaren van een teken weer op de huidige positie

Syntaxis: `void cputnc(unsigned char count, unsigned char c)`

Parameters: **c:** De schermcode van de weer te geven tekens

count: Het aantal weer te geven tekens

cputhex

Beschrijving: Geeft een hexadecimaal opgemaakt getal weer op de huidige positie

Syntaxis: `void cputhex(long n, unsigned char prec)`

Parameters: **n:** Het te schrijven getal

prec: De precisie van het hexadecimale getal, in cijfers. Voorloopnullen worden dienovereenkomstig weergegeven

Opmerkingen:

Het $-teken wordt automatisch aan het begin van de string toegevoegd

cputdec

Beschrijving: Geeft een decimaal getal weer op de huidige positie

Syntaxis: `void cputdec(long n, unsigned char padding, unsigned char leadingZ)`

Parameters: **n:** Het te schrijven getal

padding: De opvullingsruimte die vóór het getal moet worden toegevoegd

leadingZ: De voorloopnullen die moeten worden weergegeven

cputs

Beschrijving: Geeft schermcodes weer op de huidige positie

Syntaxis: `void cputs(const unsigned char* s)`

Parameters: **s:** Een array van schermcodes die moeten worden weergegeven

Opmerkingen:

Deze functie werkt uitsluitend met schermcodes. Voor het weergeven van gewone ASCII-/PETSCII-strings gebruikt u de macro "pcputs". Er wordt geen pointercontrole uitgevoerd. Indien s null of ongeldig is, is het gedrag ongedefinieerd.

cputsxy

Beschrijving: Geeft meerdere schermcodes weer op de X,Y-coördinaten

Syntaxis: `void cputsxy (unsigned char x, unsigned char y, const unsigned char* s)`

Parameters: **x:** De X-coördinaat waar de string wordt weergegeven

y: De Y-coördinaat waar de string wordt weergegeven

s: Een array van schermcodes die moeten worden weergegeven

Opmerkingen:

Deze functie werkt uitsluitend met schermcodes. Voor het weergeven van gewone ASCII-/PETSCII-strings gebruikt u de macro "pcputsxy". Er wordt geen pointercontrole uitgevoerd. Indien s null of ongeldig is, is het gedrag ongedefinieerd.

cputcxy

Beschrijving: Geeft één teken weer op de X,Y-coördinaten

Syntaxis: `void cputcxy (unsigned char x, unsigned char y, unsigned char c)`

Parameters: **x:** De X-coördinaat waar het teken wordt weergegeven

y: De Y-coördinaat waar het teken wordt weergegeven

c: De schermcode van het weer te geven teken

cputncxy

Beschrijving: Geeft N exemplaren van een teken weer op de X,Y-coördinaten

Syntaxis: `void cputncxy (unsigned char x, unsigned char y, unsigned char count, unsigned char c)`

Parameters: **x:** De X-coördinaat waar het teken wordt weergegeven

y: De Y-coördinaat waar het teken wordt weergegeven

count: Het aantal weer te geven tekens

c: De schermcode van de weer te geven tekens

cprintf

Beschrijving: Drukt opgemaakte uitvoer af.
Escape-strings kunnen worden gebruikt om attributen te wijzigen, de cursor te verplaatsen, enz., vergelijkbaar met PRINT in CBM BASIC.

Syntaxis: `unsigned char cprintf (const unsigned char* format, ...)`

Parameters: **format:** De weer te geven string. De beschikbare escape-codes zijn:

Cursorpositionering

\t Ga naar de volgende tabpositie (veelvoud van 8)
\r Carriage Return
\n Nieuwe regel

{clr}	Scherm wissen	{home}	Cursor naar homepositie (linksboven)
{d}	Cursor omlaag	{u}	Cursor omhoog
{r}	Cursor naar rechts	{l}	Cursor naar links

Attributen

{rvson}	Omgekeerd-attribuut AAN	{rvsoff}	Omgekeerd-attribuut U
{blon}	Knipperattribuut AAN	{bloff}	Knipperattribuut UIT
{ulon}	Onderstrepingsattribuut AAN	{uloff}	Onderstrepingsattribuu

Kleuren (standaardpalet)

{blk}	{wht}	{red}	{cyan}
{pur}	{grn}	{blu}	{yel}
{ora}	{brn}	{pink}	{gray1}
{gray2}	{lblu}	{lgrn}	{gray3}

Opmerkingen:

Deze functie werkt uitsluitend met schermcodes! Gebruik de "pcprintf"-macro om gewone ASCII/PETSCII-tekenreeksen weer te geven. Momenteel wordt geen vervanging van argumenten uitgevoerd bij de variabele argumenten.

pcprintf

Beschrijving: Drukt opgemaakte PETSCII-tekenreeksuitvoer af.

Syntaxis: `see cprintf`

cgetc

Beschrijving: Wacht totdat een teken in de toetsenbordbuffer staat en retourneert dit

Syntaxis: `unsigned char cgetc (void);`

Resultaat: Het laatste teken in de toetsenbordbuffer

Opmerkingen:

Geretourneerde waarden zijn ASCII-tekencodes

kbhit

Beschrijving: Retourneert het teken in de toetsenbordbuffer

Syntaxis: `unsigned char kbhit (void);`

Resultaat: De tekencode in de toetsenbordbuffer, 0 indien de buffer leeg is.

Opmerkingen:
Geretourneerde waarden zijn ASCII-tekencodes

getkeymodstate

Beschrijving: Retourneert de toetsmodificatiestatus.

Syntaxis: `unsigned char getkeymodstate(void)`

Resultaat: Een byte met de toetsmodificatiestatusbits, waarbij de bits:

Bit	Betekenis	Constante
0	Rechts SHIFT-status	KEYMOD_RSHIFT
1	Links SHIFT-status	KEYMOD_LSHIFT
2	CTRL-status	KEYMOD_CTRL
3	MEGA-status	KEYMOD_MEGA
4	ALT-status	KEYMOD_ALT
5	NOSCRL-status	KEYMOD_NOSCRL
6	CAPSLOCK-status	KEYMOD_CAPSLOCK
7	Gereserveerd	-

flushkeybuf

Beschrijving: Leegt de toetsenbordbuffer

Syntaxis: `void flushkeybuf(void)`

cinput

Beschrijving: Ontvangt invoer van het toetsenbord en drukt binnenkomende tekens af op de huidige positie.

Syntaxis: `unsigned char cinput(char* buffer, unsigned char buflen, unsigned char flags)`

Parameters: **buffer:** Doeltekenbuffer die vooraf door de aanroeper is toegewezen

buflen: Lengte van de doelbuffer in tekens, inclusief het afsluitende nulteken

flags: Vlaggen voor invoer: (standaard worden alle afdrukbare tekens aanvaard) CINPUT_ACCEPT_NUMERIC
Aanvaardt numerieke tekens.

CINPUT_ACCEPT_LETTER
Aanvaardt letters.

`CINPUT_ACCEPT_SYM`
Aanvaardt symbolen.

`CINPUT_ACCEPT_ALL`
Aanvaardt alles. Is gelijk aan `CINPUT_ACCEPT_NUMERIC |CINPUT_ACCEPT_LETTER |CINPUT_ACCEPT_SYM`

`CINPUT_ACCEPT_ALPHA`
Aanvaardt alfanumerieke tekens. Is gelijk aan `CINPUT_ACCEPT_NUMERIC |CINPUT_ACCEPT_LETTER`

`CINPUT_NO_AUTOTRANSLATE`
Schakelt de functie uit waarmee cinput automatisch hoofdletters weergeeft wanneer de standaard kleinelettertekenset is geselecteerd en u letters intoetst zonder de SHIFT-toets; zonder deze functie geeft de computer grafische tekens weer in plaats van alfabetische tekens.

Resultaat: Aantal succesvol gelezen tekens in de buffer

VIC_BASE

VIC_BASE is een preprocessor-macro die het basisadres van de VIC-IV-chip opgeeft, namelijk $D000.

IS_H640 is een preprocessor-macro welke 0 retourneert wanneer de huidige VIC-III/IV-videomodus is ingesteld op 320 pixels breed (40-kolomsmodus) en een waarde ongelijk aan nul wanneer deze is ingesteld op 640 pixels breed (80-kolomsmodus).

HOOFDSTUK 9

BASIC-tokenisatoren

Er bestaan verschillende tokenisatoren voor C64-BASIC, zoals https://github.com/catseye/hatoucan, https://www.c64-wiki.com/wiki/C64list of het hulpprogramma **petcat** dat deel uitmaakt van VICE. Als u Ubuntu Linux gebruikt, kunt u **petcat** installeren met het volgende commando:

```
sudo apt-get install vice
```

We raden **petcat** aan, omdat het zowel C64 BASIC 2 als C65 BASIC 10 ondersteunt.

Sommige IDE's bieden BASIC 65-tokenisatorens, zoals:

Eleven

https://files.mega65.org?id=8b189d0b-ea1e-45a7-a4de-87bcb0b11696

C64 Studio

https://www.georg-rottensteiner.de/files/C64StudioRelease.zip

CBM prg Studio

https://www.ajordison.co.uk

DEEL III

PROGRAMMEERGEREEDSCHAP OP DE MEGA65 ZELF

HOOFDSTUK 10

C65-geheugenmonitor

HOOFDSTUK 11

MEGA65 Matrix Mode

HOOFDSTUK 12

Turbo Assembler

HOOFDSTUK 13

Omgaan met de C65- en C64-modus

De MEGA65 start normaal in C65-modus en het is aantrekkelijk om programma's te maken die direct vanuit C65-modus kunnen starten. De verschillende versies van de C65-ROM maken dit echter moeilijk, omdat er verschillen zijn in de geheugenlocatie van diverse systeemvariabelen en ROM- routines. Indien u een programma in assembleertaal of C ontwikkelt, is het doorgaans eenvoudiger om het voor C64-modus te ontwikkelen.

Dat zadelt u evenwel op met het probleem hoe u het programma vanuit C65-modus kunt laten starten. Onze oplossing hiervoor is het programma **c65toc65wrapper.asm**. Dit korte programma staat in de map **src/utilities** van het https://github.com/mega65/mega65-tools-repository. Wanneer het gecompileerd wordt, produceert het een kort programma dat u voor uw C64-modus-georiënteerde MEGA65-programma kunt plaatsen, zodat het automatisch C64- of C65-modus detecteert, indien nodig naar C64-modus schakelt en uw programma laat draaien. Het schakelt ook de CPU naar 40 MHz. De enige aanname is dat uw programma gestart moet worden met **SYS 2061**.

Dit programma is bijzonder handig bij het gebruik van exomizer, omdat het instellen van de CPU op 40 MHz vóór het uitpakken ervoor zorgt dat het uitpakken bijna onmiddellijk gebeurt. Dit is belangrijk op de MEGA65, omdat het laden van een programma doorgaans slechts een fractie van een seconde duurt, maar het uitpakken van een geëxomized programma op 1 MHz enkele seconden kan duren.

Tot slot kunt u deze benadering gebruiken om opstartbare schijven te maken voor de MEGA65, door het bestand op de schijf te plaatsen met de naam **AUTOBOOT.C65**. Hiermee kunt u programma's voor de C64-modus laten autobooten op de MEGA65, wat op zichzelf zeer handig is.

DEEL IV

MEGA65 BESTURINGSSYSTEEM

HOOFDSTUK 14

Ontwikkeling van systeemprogramma's

- Inleiding
- Flashmenu
- Format/FDISK-hulpprogramma
- Toetsenbordtest-hulpprogramma
- MEGA65-configuratiehulpprogramma
- Freezer-menu
- Hulpprogramma's van het Freezer-menu
- Hypervisor

- OpenROM

INLEIDING

De MEGA65 heeft een aantal systeemprogramma's en hulpprogramma's die bij diverse gelegenheden verschillende functies vervullen. Daartoe behoren de hulpprogramma's via het hulpprogrammamenu, het Freezer-menu en de eigen hulpprogramma's, alsmede het flashmenu.

Een aantal van deze systeemprogramma's is vooraf in de bitstream van de MEGA65 geladen, terwijl andere op de SD-kaart staan. Bij de versies die vooraf in de bitstream zijn geladen, gebeurt dit door middel van voorgeïnitialiseerde geheugengebieden die het juiste programma bevatten. De hulpprogramma's die via het hulpprogrammamenu toegankelijk zijn, bevinden zich bijvoorbeeld allemaal in het colour RAM, terwijl het flashmenu op het adres $50000 - $57FFF staat.

De eenvoudigste manier om nieuwe versies van deze hulpprogramma's te testen, is het genereren van een nieuwe bitstream met de bijgewerkte versies. Het synthesiseren van een nieuwe bitstream is echter tijdrovend: het duurt doorgaans een uur op een redelijk snelle computer. Dit hoofdstuk legt daarom de procedure uit voor het laden van een alternatieve versie van elk van deze systeemprogramma's, en verstrekt enkele nuttige gegevens over deze programma's, hun werking en de omgeving waarin zij werken in vergelijking met gewone programma's in C64- of C65-modus.

FLASHMENU

Het flashmenu bevindt zich in voorgeïnitialiseerd RAM op $50000 - $57FFF. Het wordt iedere keer uitgevoerd tijdens de eerste opstart na inschakelen van de MEGA65. Het programma is buitenissig in de zin dat het in de Hypervisor-context draait. Dit is nodig om toegang te krijgen tot het QSPI-flashgeheugen, dat buiten de Hypervisor-modus niet beschikbaar is, zodat gebruikersprogramma's de cores die in de flash zijn opgeslagen niet kunnen beschadigen.

Het is ook belangrijk op te merken dat het flashmenu-programma *volledig* onder $8000 moet passen wanneer het wordt geladen *en* wordt uitgevoerd, omdat de Hypervisor op $8000 - $BFFF in het geheugen geladen is en gemakkelijk kan worden beschadigd door een flashmenu-programma dat zich misdraagt. In dit opzicht kan het flashmenu worden beschouwd als een uitbreiding van de Hypervisor die na de eerste opstart wordt verwijderd. Dit verschilt van alle andere systeemprogramma's, die in een afzonderlijk geheugencontext werken waarin de Hypervisor beschermd is tegen beschadiging. Het betekent ook dat u het flashmenu niet kunt comprimeren om het te laten passen, omdat het tijdens het uitpakken de Hypervisor zou overschrijven.

Ook wordt het flashmenu zeer vroeg in het opstartproces uitgevoerd, zodat alleen het vooraf opgenomen OpenROM ROM-image beschikbaar is. U moet er daarom voor zorgen dat uw flashmenu-programma compatibel is met die ROM.

De Hypervisor houdt een vlag bij die aangeeft of het flashmenu al dan niet is uitgevoerd. Deze vlag wordt bijgewerkt op het moment dat de Hypervisor voor de eerste

keer naar de gebruikersmodus overschakelt, omdat de inhoud van $50000 - $57FFF vanaf dat moment niet meer betrouwbaar is als opslagplaats van het flashmenu. Dit betekent dat u, indien u de Hypervisor een nieuwe versie van het flashmenu wilt laten uitvoeren die u hebt geladen, eerst moet voorkomen dat de Hypervisor naar de gebruikersmodus overschakelt.

De eenvoudigste manier om dit te bereiken, is de ALT-toets ingedrukt te houden terwijl u de MEGA65 inschakelt. Hierdoor toont de Hypervisor het hulpprogrammamenu in plaats van naar de gebruikersmodus over te schakelen. Op dit moment kunt u het `m65`-hulpprogramma gebruiken om het vervangende flashmenu-programma te laden met een commando als het volgende:

```
m65 -@ newflashmenu.prg@50000
```

Dat commando laadt het bestand `newflashmenu.prg` op geheugenadres $50000 geladen.

Daarna kunt u eenvoudig de resetknop aan de zijkant van de MEGA65 indrukken terwijl u NO SCROLL ingedrukt houdt. De computer start opnieuw op en omdat tijdens de vorige opstartcyclus niet uit de Hypervisor-modus is overschakeld, wordt uw bijgewerkte flashmenu-programma uitgevoerd.

Het zou ook mogelijk moeten zijn om dit proces volledig te automatiseren, door eerst `m65 -b` te gebruiken om een nieuwe bitstream te laden (een koude start nabootsen), vervolgens snel `m65` opnieuw aan te roepen om het indrukken van de ALT-toets na te bootsen (of eenvoudig de processor te stoppen), daarna `m65 -@ ...` en ten slotte `m65 -F` om de computer te resetten. Het schrijven van een script of hulpprogramma dat deze automatisering correct implementeert, laten wij over als een oefening voor de lezer.

FORMAT/FDISK-HULPPROGRAMMA

Het Format/FDISK-hulpprogramma is toegankelijk via het hulpprogrammamenu. Deze hulpprogramma's worden gecompileerd, gecomprimeerd en gelinkt met het `utilpacker`-programma. Indien u de broncode van de mega65-core hebt uitgecheckt, kunt u het colour RAM-image opnieuw bouwen met:

```
make bin/COLOURRAM.BIN
```

U moet uiteraard eerst het Format/FDISK-hulpprogramma hebben aangepast, dat zich normaal gesproken in de subdirectory `src/mega65-freeze-menu` bevindt.

Vervolgens moet u dit aangepaste colour RAM-image in de draaiende computer laden. Net als bij het bijwerken van het flashmenu, toont de Hypervisor het hulpprogrammamenu alleen bij de eerste opstart, voordat er voor de eerste keer naar de

gebruikersmodus wordt overgeschakeld, omdat de Hypervisor anders niet zeker kan zijn dat het colour RAM de geldige hulpprogramma's bevat.

Net als bij het flashmenu schakelt u de MEGA65 uit en schakelt u deze vervolgens weer in terwijl u de ALT-toets ingedrukt houdt, zodat het hulpprogrammamenu wordt weergegeven. Op dat moment kunt u het volgende commando gebruiken om uw aangepaste bestand `COLOURRAM.BIN` te laden:

```
m65 -c COLOURRAM.BIN
```

U kunt nu ALT ingedrukt houden en de resetknop aan de linkerkant van de MEGA65 indrukken. Het hulpprogrammamenu verschijnt dan opnieuw, ditmaal met uw aangepaste Format/FDISK-hulpprogramma.

TOETSENBORDTEST-HULPPROGRAMMA

De procedure voor het bijwerken van het toetsenbordtest-hulpprogramma is in beginsel dezelfde als voor het Format/FDISK-hulpprogramma, omdat het zich in het colour RAM bevindt.

MEGA65-CONFIGURATIEHULPPROGRAMMA

De procedure voor het bijwerken van het MEGA65-configuratiehulpprogramma is in beginsel dezelfde als voor het Format/FDISK-hulpprogramma, omdat het zich in het colour RAM bevindt.

FREEZER-MENU

Het Freezer-menu is een gewoon programma dat als `FREEZER.M65` op het FAT32-bestandssysteem van de SD-kaart staat.

Om het Freezer-menu bij te werken, gebruikt u het `m65ftp`-hulpprogramma of een andere methode om uw bijgewerkte bestand FREEZER.M65 naar het FAT32-bestandssysteem van de SD-kaart te uploaden. Het formaat van het programma is eenvoudigweg een PRG-bestand in C64-modus, hernoemd naar FREEZER.M65.

HULPPROGRAMMA'S VAN HET FREEZER-MENU

De hulpprogramma's van het Freezer-menu worden op dezelfde manier bijgewerkt als het Freezer-menu zelf.

HYPERVISOR

De Hypervisor wordt doorgaans gebouwd als HICKUP.M65, een bestand van 16 KB dat het volledige Hypervisor-programma bevat. MEGA65-bitstreams bevatten een vooraf gebouwde versie op het adres $FFF8000 – $FFFBFFF. Bijgewerkte versies van de Hypervisor kunnen op twee manieren worden getest:

- 1. Plaats het bijgewerkte bestand HICKUP.M65 op het FAT32-bestandssysteem van de SD-kaart en schakel de MEGA65 vervolgens uit en weer in. Dit werkt omdat de Hypervisor code bevat die een bijgewerkte versie van zichzelf zoekt en deze, indien gevonden, laadt. Deze aanpak is echter problematisch doordat bij het installeren van een nieuwere bitstream de Hypervisor alsnog wordt teruggedraaid naar de versie die in het bestand HICKUP.M65 op de SD-kaart staat. Deze methode wordt alleen aanbevolen voor ontwikkelaars die hun aangepaste Hypervisor-code vanuit een koude start moeten testen. Ook dan wordt aanbevolen het bestand HICKUP.M65 onmiddellijk na het testen te verwijderen om onverwachte terugval in de toekomst te voorkomen.
- 2. Gebruik de `-k`-optie van het m65-commando om de Hypervisor ter plekke te vervangen en reset de MEGA65 vervolgens met de resetknop aan de linkerkant van de behuizing. Dit dient te gebeuren wanneer de Hypervisor *niet* actief is, zodat beschadiging van de huidige uitvoering niet kan optreden. Het moet echter ook gebeuren voordat er een ROM is geladen die het standaard OpenROM-image vervangt. De Hypervisor zal bij de eerste opstart namelijk een aanroep doen naar de ROM ter voorbereiding op het flashmenu, en ervan uitgaan dat de OpenROM aanwezig is, omdat hij een speciale, voor OpenROM bestemde aanroep gebruikt om delen van de systeemtoestand voor het flashmenu te initialiseren. Het beste doet u dit met een commando als `m65 -k bin/HICKUP.M65 -R bin/MEGA65.ROM` om zowel een nieuw Hypervisor-programma te laden als het OpenROM-image opnieuw te laden.

OPENROM

Om een nieuwe versie van een ROM te laden, bestaan er verschillende mogelijkheden, waaronder het tegelijkertijd vervangen van de Hypervisor en de ROM zoals hierboven beschreven. Doorgaans is de eenvoudigste methode het kopiëren van de

nieuwe ROM naar het FAT32-bestandssysteem van de SD-kaart als MEGA65.ROM of MEGA65x.ROM, waarbij x wordt vervangen door een cijfer tussen 0 en 9. Bij het resetten van de MEGA65 wordt dan normaal gesproken MEGA65.ROM geladen, of indien een cijfer tussen 0 en 9 wordt ingedrukt op het toetsenbord tijdens het resetten, laadt de Hypervisor in plaats daarvan MEGA65x.ROM, waarbij x het cijfer is dat wordt ingedrukt.

HOOFDSTUK 15

MEGA65 Hyppo-diensten

- Inleiding
- Hyppo-diensten: Beknopt overzicht
- Algemene diensten
- Station-/opslagservices
- Schijfimage-services
- Taak- en procesdiensten
- Systempartitieservices
- Freezerservices

INLEIDING

Een onderdeel van de MEGA65 is het systeemprogramma Hyppo dat:

- De MEGA65 opstart.
- De ROM's en andere bestanden van de SD-kaart laadt.
- Geheugenbanken 2 en 3 ROM-achtig maakt door ze tegen schrijven te beschermen.
- De diskettestationcontroller virtualiseert zodat u schijfimages kunt gebruiken.
- Diverse hulpprogramma's start zoals de Freezer en de Matrix Mode Debugger.
- Diensten biedt die specifiek zijn voor de MEGA65 en die u in uw programma's kunt gebruiken.

Wanneer u bekend bent met hypervisors en virtuele machines, is Hyppo een zeer beperkte hypervisor. Verwacht niet dat u meerdere virtuele machines tegelijkertijd met volledige isolatie kunt uitvoeren. Hyppo voert taken uit die meer doen denken aan de taken en processen van een modern besturingssysteem dan aan de virtuele machines van een hypervisor zoals u die wellicht kent.

Hyppo biedt drie bedrijfsmodi.

- **De C64-achtige bedrijfsmodus** voert C64-programma's en MEGA65-programma's uit die draaien in de C64-modus van de MEGA65. Wanneer u opstart met de  ingedrukt of het **GO64**-commando gebruikt, start Hyppo een proces in de C64-achtige bedrijfsmodus om BASIC 2 of het MEGA65-programma uit te voeren.
- **De C65-achtige bedrijfsmodus** is de normale bedrijfsmodus van de MEGA65. Hier worden de reguliere MEGA65-programma's uitgevoerd, waaronder BASIC 65-programma's.
- **De MEGA65-bedrijfsmodus** voert de systeemprogramma's van de MEGA65 uit, zoals de Freezer, het configuratiehulpprogramma en de Matrix Mode Debugger. Wellicht verrassend draaien reguliere MEGA65-programma's niet in de MEGA65-bedrijfsmodus. Zij draaien in de C65-achtige bedrijfsmodus. De MEGA65-bedrijfsmodus is uitsluitend ontworpen voor de MEGA65 en probeert niet compatibel te zijn met of zelfs maar te lijken op eerdere systemen.

In tegenstelling tot de C128 kan een programma de bedrijfsmodus effectief wijzigen terwijl het draait, door eenvoudigweg de verschillende hardwarefuncties in of uit te schakelen.

Hyppo biedt zeer beperkte virtualisatie van de hardware van de MEGA65. Het kan de diskettestationcontroller virtualiseren. Er bestaan plannen om de seriële bus te virtualiseren zodat de MEGA65 schijfimages kan gebruiken voor apparaten zoals de 1541.

Er zijn delen van de hardware die alleen Hyppo kan benaderen. Het is het enige onderdeel dat directe toegang heeft tot de interne en externe SD-kaarten. U dient de diensten van Hyppo te gebruiken als u vanuit uw programma's de bestanden en directories op de SD-kaarten wilt benaderen.

Terminologie

Wanneer u kennismaakt met Hyppo, kunt u in aanraking komen met termen die verwarrend kunnen zijn als u al bekend bent met andere onderdelen van de MEGA65.

Op de SD-kaart bevindt zich waarschijnlijk een bestand met de naam HICKUP.M65. Dit bestand werkt Hyppo bij naar nieuwe versies zonder dat een bijgewerkte core geïnstalleerd hoeft te worden. Het zou kunnen voorkomen dat u Hyppo soms Hickup hoort noemen vanwege deze sterke associatie.

Er zijn drie verschillende disk-besturingssystemen in de MEGA65.

- Binnen Hyppo bevindt zich Hyppo DOS, of kortweg HDOS. HDOS is bedoeld voor toegang tot het FAT32-bestandssysteem op de SD-kaarten. HDOS weet niets van Commodore-bestandssystemen. Het kan een image van een Commodore-bestandssysteem koppelen, maar het begrijpt niet wat er in de image staat.
- Binnen de KERNAL bevindt zich CBDOS. CBDOS is bedoeld voor toegang tot 1581-achtige bestandssystemen. CBDOS gebruikt de 45IO27 multifunctionele I/O-controller om de sectoren van een fysieke schijf te benaderen. CBDOS weet niets van SD-kaarten en het FAT32-bestandssysteem daarop. Hyppo virtualiseert een deel van de 45IO27 zodat CBDOS schijfimages kan benaderen alsof het fysieke schijven zijn.
- De externe schijfstations die op de seriële bus zijn aangesloten, hebben elk hun eigen DOS. Deze worden gebruikt voor toegang tot de bestandssystemen op hun respectievelijke fysieke schijven.

Het woord drive heeft voor elk van deze DOS's een andere betekenis.

- De drives in Hyppo zijn de partities van de interne en externe SD-kaarten. Wanneer de MEGA65 opstart, wijst Hyppo nummers toe aan de partities die het kan lezen.
- De drives in CBDOS zijn de fysieke schijfstations die op de 45IO27 multifunctionele I/O-controller zijn aangesloten — zoals het interne schijfstation — of de schijfimages die op de gevirtualiseerde 45IO27 zijn gekoppeld. De CBDOS-drives zijn doorgaans zichtbaar als stations 8 en 9.
- De drives in een extern station dat op de seriële bus is aangesloten, zijn de schijfstations binnen dat station.

Versies

Dit hoofdstuk beschrijft de diensten die beschikbaar zijn in Hyppo 1.2.

Er kunnen nieuwe Hyppo-diensten beschikbaar komen en bestaande Hyppo-diensten kunnen wijzigen of komen te vervallen. Een robuust programma gebruikt de hyppo_getversion-dienst om te controleren of het compatibel is met de Hyppo in de MEGA65 waarop het wordt uitgevoerd.

Gebruik

Wanneer u een Hyppo-dienst wilt gebruiken, gebruikt u geen JSR. Hyppo bevindt zich namelijk in een afzonderlijke ruimte ten opzichte van gewone code. Om er toegang toe te krijgen, moet de CPU overschakelen naar zijn hypervisor-modus.

Op de adressen $D640 - $D67F staat een reeks hypervisor-traps. Het schrijven naar deze adressen werkt niet zoals het schrijven naar andere adressen. In plaats van naar geheugen of I/O te schrijven, schakelt de CPU over naar de hypervisor-modus en start een Hyppo-dienst. Hoe de CPU dit doet, wordt beschreven in *het grote Mega65-boek*, 45GS02-microprocessor (Bijlage K).

Welke Hyppo-dienst wordt gestart, hangt af van welke waarde u uit het A-register schrijft en welke trap u aanspreekt. Elke dienst die in dit hoofdstuk wordt beschreven, vermeldt welke waarde u moet schrijven en welke trap u moet gebruiken. U moet het A-register gebruiken wanneer u een trap activeert. Het schrijven van dezelfde waarde vanuit een ander register werkt niet.

Wanneer de Hyppo-dienst is afgerond, schakelt de CPU terug naar uw programma. Met uitzondering van de registers die een dienst gebruikt om waarden terug te geven, blijven de registers ongewijzigd.

Belangrijk De CPU kan het volgende byte in uw programma uitvoeren nadat de Hyppo-dienst is afgerond, maar dit is niet gegarandeerd. Plaats een CLV-instructie na de STA. Of de CPU de CLV nu wel of niet uitvoert, mag geen invloed hebben op uw programma. Indien uw programma afhankelijk is van de V-vlag, kunt u in plaats daarvan de NOP-instructie gebruiken. Wanneer u NOP gebruikt, moet u erop letten wanneer de CPU de NOP interpreteert als een prefix voor de volgende instructie. Om die reden verdient het de voorkeur om CLV boven NOP te gebruiken.

Foutcodes:

Indien de dienst succesvol was, wordt de C-vlag gezet.

Indien de dienst niet succesvol was, wordt de C-vlag gewist en komt een foutcode in het A-register. Er staat een tabel met foutcodes in de beschrijving van hyppo_geterrorcode.

Voorbeelden

De voorbeelden gebruiken de ACME-assembler. De ACME-assembler is niet verplicht. De Hyppo-diensten kunnen met elke assembler worden gebruikt.

De voorbeelden zijn doorgaans niet volledig. Ze gaan ervan uit dat ergens een foutverwerker met de naam error is gedefinieerd. Ze gaan er eveneens van uit dat ergens een overdrachtsgebied is gedefinieerd. hyppo_setup_transfer_area en hyppo_setname laten zien hoe u een overdrachtsgebied definieert.

HYPPO-DIENSTEN: BEKNOPT OVERZICHT

Hieronder volgt een beknopt overzicht van de Hyppo-diensten. Raadpleeg de volledige naslag voor details en voorbeelden.

Hyppo-naam	Categorie	Invoer	STA	Beschrijving
geterrorcode	Algemeen	A=$38	$D640	Huidige foutcode van Hyppo
getversion	Algemeen	A=$00	$D640	Versie van Hyppo, HDOS
setup_transfer_area	Algemeen	A=$3A, Y	$D640	Stelt het gebied in dat Hyppo gebruikt om gegevens over te dragen
chdir	SD-kaart	A=$0C	$D640	Wijzigt de huidige werkmap
cdrootdir	SD-kaart	A=$3C, X	$D640	Voert selectdrive uit en stelt de huidige werkmap in op de root
closeall	SD-kaart	A=$22	$D640	Sluit alle bestandsdescriptoren
closedir	SD-kaart	A=$16, X	$D640	Sluit een bestandsdescriptor voor een directory
closefile	SD-kaart	A=$20, X	$D640	Sluit een bestandsdescriptor voor een bestand
filedate	SD-kaart	A=$2C	$D640	Niet geïmplementeerd
findfile	SD-kaart	A=$34	$D640	Zoekt het eerste bestand waarvan de bestandsnaam overeenkomt met de huidige Hyppo-bestandsnaam
findfirst	SD-kaart	A=$30	$D640	Zoekt het eerste bestand waarvan de bestandsnaam overeenkomt met de huidige Hyppo-bestandsnaam
findnext	SD-kaart	A=$32	$D640	Zoekt een volgend bestand waarvan de bestandsnaam overeenkomt met de huidige Hyppo-bestandsnaam
fstat	SD-kaart	A=$28	$D640	Niet geïmplementeerd
getcurrentdrive	SD-kaart	A=$04	$D640	Het nummer van de momenteel geselecteerde SD-kaartpartitie
getcwd	SD-kaart	A=$0A	$D640	Niet geïmplementeerd
getdefaultdrive	SD-kaart	A=$02	$D640	De SD-kaartpartitie die Hyppo heeft geselecteerd tijdens het opstarten

vervolg ...

...vervolg

Hyppo-naam	Categorie	Invoer	STA	Beschrijving
getdrivesize	SD-kaart	A=$08	$D640	Niet geïmplementeerd
loadfile	SD-kaart	A=$36, XYZ	$D640	Laadt een bestand in chip-geheugen
loadfile_attic	SD-kaart	A=$3E, XYZ	$D640	Laadt een bestand in hyper-geheugen
mkdir	SD-kaart	A=$0E	$D640	Niet geïmplementeerd
mkfile	SD-kaart	A=$1E	$D640	Niet geïmplementeerd
opendir	SD-kaart	A=$12	$D640	Opent de huidige werkmap om de bestandsvermeldingen daarin te lezen
openfile	SD-kaart	A=$18	$D640	Opent een bestand op een station
readdir	SD-kaart	A=$14, XY	$D640	Leest de volgende FAT-maprecord in een doelgebied
readfile	SD-kaart	A=$1A	$D640	Leest de volgende sector van het huidige bestand in de sectorbuffer
rename	SD-kaart	A=$2A	$D640	Niet geïmplementeerd
rmdir	SD-kaart	A=$10	$D640	Niet geïmplementeerd
rmfile	SD-kaart	A=$26	$D640	Verwijdert een bestand
seekfile	SD-kaart	A=$24	$D640	Niet geïmplementeerd
selectdrive	SD-kaart	A=$06, X	$D640	Stelt de huidig geselecteerde SD-kaartpartitie in
setname	SD-kaart	A=$2E, Y	$D640	Stelt de huidige Hyppo-bestandsnaam in
writefile	SD-kaart	A=$1C	$D640	Schrijft de sectorbuffer naar het huidige bestand
attach	Schijfimage	A=$4A, X	$D640	Koppelt een schijfimage aan of los van het gevirtualiseerde F011-station
d81attach0	Schijfimage	A=$40	$D640	Verouderd
d81attach1	Schijfimage	A=$46	$D640	Verouderd
d81detach	Schijfimage	A=$42	$D640	Verouderd
d81write_en	Schijfimage	A=$44	$D640	Maakt schrijven naar alle schijfimages die aan de gevirtualiseerde F011-stations 0 en 1 zijn gekoppeld, mogelijk
create_task_c64	Taak	A=$66	$D640	Niet geïmplementeerd
create_task_c65	Taak	A=$68	$D640	Niet geïmplementeerd
create_task_native	Taak	A=$62	$D640	Niet geïmplementeerd
exit_and_switch_to_task	Taak	A=$6A	$D640	Niet geïmplementeerd

vervolg ...

...vervolg

Hyppo-naam	Categorie	Invoer	STA	Beschrijving
exit_task	Taak	A=$6E	$D640	Niet geïmplementeerd
get_mapping	Taak	A=$74	$D640	Kopieert de huidige 45GS02-geheugenindeling naar een doelgebied
get_proc_desc	Taak	A=$58	$D640	Kopieert het huidige taak-blok naar een doelgebied
gettasklist	Taak	A=$50	$D640	Niet geïmplementeerd
load_into_task	Taak	A=$64	$D640	Niet geïmplementeerd
readoutoftask	Taak	A=$58	$D640	Niet geïmplementeerd
receivemessage	Taak	A=$54	$D640	Niet geïmplementeerd
reset	Taak	A=$7E	$D640	Voert een warme herstart van de MEGA65 uit
rom_writeenable	Taak	A=$02	$D641	Maakt schrijven naar bank 2 en 3 mogelijk
rom_writeprotect	Taak	A=$00	$D641	Schakelt schrijven naar bank 2 en 3 uit
sendmessage	Taak	A=$52	$D640	Niet geïmplementeerd
serial_monitor_wait _and_write	Taak	A=byte	$D643	Wacht tot de seriële debug-ger gereed is en schrijft een teken naar deze debugger
serial_monitor_write	Taak	A=$7C, Y=byte	$D640	Schrijft een teken naar de seriële debugger, zonder te wachten
set_mapping	Taak	A=$76, Y	$D640	Kopieert het brongebied naar de huidige 45GS02-geheugenindeling
switch_to_task	Taak	A=$6C	$D640	Niet geïmplementeerd
terminateothertask	Taak	A=$60	$D640	Niet geïmplementeerd
toggle_force_4502	Taak	A=$72	$D640	Wisselt de CPU tussen de 45GS02- en 6502-modus
toggle_rom_writeprotect	Taak	A=$70	$D640	Schakelt de schrijfbeveili-ging van bank 2 en 3 om
writeintotask	Taak	A=$56	$D640	Niet geïmplementeerd
configsector_apply	Configuratie	A=$04	$D642	Past het systeemconfigura-tiesector toe dat momenteel in het geheugen is geladen
configsector_read	Configuratie	A=$00	$D642	Leest het systeemconfigura-tiesector in het geheugen
configsector_write	Configuratie	A=$02	$D642	Schrijft het systeemcon-figuratiesector vanuit het geheugen

vervolg ...

...vervolg

Hyppo-naam	Categorie	Invoer	STA	Beschrijving
dmagic_autoset	Configuratie	A=$06	$D642	Stelt de DMAgic-revisie in op basis van de geladen ROM
freeze_self	Freezer	A=lukraak	$D67F	Start de Freezer
get_slot_count	Freezer	A=$16	$D642	Vraagt het aantal freeze-slots op
locate_freeze_slot	Freezer	A=$10	$D642	Zoekt de eerste sector van een freeze-slot
read_freeze_region_list	Freezer	A=$14	$D642	Leest de lijst met freezeregio's
unfreeze_from_slot	Freezer	A=$12	$D642	Herstelt vanuit een freeze-slot

ALGEMENE DIENSTEN

hyppo_geterrorcode

Trap: LDA #$38 : STA $D640 : CLV

Dienst: Geeft de huidige foutcode van Hyppo terug.

Voorbeding: De C-vlag was gewist door de vorige dienst.

Resultaat: **A** De foutcode van de voorgaande mislukte dienst.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: De foutcode is alleen geldig als de vorige Hyppo-dienst de C-vlag heeft gewist. Indien de C-vlag was gezet, is er geen fout opgetreden en is de Hyppo-foutcode ongedefinieerd.

De betekenissen hier zijn algemeen. Zie de secties van de diensten voor meer specifieke betekenissen.

Error codes: Dit is mogelijk geen volledige lijst.

Hex	Dec	Naam	Algemene betekenis
$01	1	partition not interesting	De partitie heeft geen ondersteund type.
$02	2	bad signature	De handtekeningsbytes aan het einde van een partitietabel of van de eerste sector van een partitie ontbraken of waren onjuist.
$03	3	is small FAT	Dit is een FAT12- of FAT16-partitie. Er wordt alleen FAT32 ondersteund.
$04	4	too many reserved clusters	De partitie heeft meer dan 65.535 gereserveerde sectoren.
$05	5	not two FATs	De partitie heeft niet precies twee kopieën van de FAT-structuur.
$06	6	too few clusters	De partitie bevat te weinig clusters.
$07	7	read timeout	Het duurde te lang om van de SD-kaart te lezen.
$08	8	partition error	Er is een ongespecificeerde fout opgetreden bij het verwerken van een partitie.
$10	16	invalid address	Er is een ongeldig adres opgegeven in een parameter.
$11	17	illegal value	Er is een ongeldige waarde opgegeven in een parameter.

Hex	Dec	Naam	Algemene betekenis
$20	32	read error	Er is een ongespecificeerde fout opgetreden bij het lezen.
$21	33	write error	Er is een ongespecificeerde fout opgetreden bij het schrijven.
$80	128	no such drive	Het opgegeven Hyppo-stationnummer bestaat niet.
$81	129	name too long	De opgegeven bestandsnaam was te lang.
$82	130	not implemented	De Hyppo-service is niet geïmplementeerd.
$83	131	file too long	Het bestand is groter dan 16 MB.
$84	132	too many open files	Alle bestandsdescriptoren zijn in gebruik.
$85	133	invalid cluster	Het opgegeven clusternummer is ongeldig.
$86	134	is a directory	Er is geprobeerd een handeling uit te voeren op een directory, terwijl een normaal bestand werd verwacht.
$87	135	not a directory	Er is geprobeerd een handeling uit te voeren op een normaal bestand, terwijl een directory werd verwacht.
$88	136	file not found	Het bestand kon niet worden gevonden in de huidige directory van het huidige station.
$89	137	invalid file descriptor	Er is een ongeldige of gesloten bestandsdescriptor opgegeven.
$8A	138	image wrong length	Het schijfimage-bestand heeft een onjuiste lengte.
$8B	139	image fragmented	Het schijfimage is niet aaneengesloten opgeslagen op de SD-kaart.
$8C	140	no space	De SD-kaart heeft geen vrije ruimte voor de gevraagde bewerking.
$8D	141	file exists	Er bestaat al een bestand met de opgegeven naam.
$8E	142	directory full	De directory kan geen extra vermeldingen bevatten.
$8F	143	double attach	Het image is al gekoppeld aan het andere station.
$FF	255	eof	Het einde van een bestand of directory werd bereikt.
$FF	255	no such trap	Er is geen Hyppo-service beschikbaar voor de trap. Het programma is mogelijk onverenigbaar met deze versie van Hyppo.

hyppo_getversion

Trap: LDA #$00 : STA $D640 : CLV

Dienst: Geeft de versie van Hyppo **A.X** en HDOS **Y.Z** weer.

Resultaat: **A** Het hoofdversienummer van Hyppo

X Het subversienummer van Hyppo

Y Het hoofdversienummer van HDOS

Z Het subversienummer van HDOS

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: De HDOS in Hyppo staat los van de CBDOS binnen de KERNAL of de DOS in de schijfstations die op de seriële poort zijn aangesloten.

Voorbeeld: Test of de versie van Hyppo $\geq$ 1.2 is en $<$ 2.0.

```
; Verkrijg de versienummers
LDA #$00 : STA $D640 : CLV
; Controleer of het hoofdversienummer van Hyppo in A gelijk is aan
; 1
CMP #1 : BNE incompatible
; Controleer of het subversienummer van Hyppo in X gelijk is aan 2
; of hoger
TXA : CMP #2 : BMI incompatible
; Deze Hyppo-versie is compatibel
```

hyppo_setup_transfer_area

Trap: LDA #$3A : STA $D640 : CLV

Dienst: Stelt het gebied in dat Hyppo gebruikt om gegevens naar uw programma over te dragen en ervan te ontvangen.

Invoer: **Y** Het hoogste byte van het adres van het overdrachtsgebied

Foutcodes: **$10 invalid address** Het adres van het overdrachtsgebied in Y > $7E

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Het overdrachtsgebied moet tussen $0000 en $7E00 liggen. Het moet ook op een paginagrens beginnen. Het laagste byte van het adres moet $00 zijn.

Het overdrachtsgebied is 256 bytes lang voor de meeste services.

Het overdrachtsgebied wordt aangeduid met het huidige geheugenindeling van de CPU op het moment dat een service wordt uitgevoerd. Het is echter goed gebruik om het altijd in de onderste 32 KB van bank 0 te plaatsen.

Voorbeeld: Reserveert 256 bytes op een paginagrens en stelt deze in als het overdrachtsgebied.

```
;-------------------------------------------------------------------------------
; Ergens in het gegevensgebied van het programma

    ; Uitlijnen op de nieuwe paginagrens
    !align 255, 0

transferarea:
    ; 256 bytes reserveren
    !skip 256

;-------------------------------------------------------------------------------
; Elders in de programmacode

setuptransferarea:
    ; Stel de transferarea in als het Hyppo-overdrachtsgebied
    LDY #>transferarea : LDA #$3A : STA $D640 : CLV : BCC error
```

STATION-/OPSLAGSERVICES

In Hyppo zijn stations de partities van de interne en externe SD-kaarten. Het zijn niet de station 0 en station 1 van de F011-floppycontroller. Het zijn ook niet de station 0 en station 1 van dual-drive-units die op de seriële bus zijn aangesloten.

hyppo_chdir

Trap: LDA #$0C : STA $D640 : CLV

Dienst: Wijzigt de huidige werkmap.

Voorbeding: De FAT-dir-vermelding van de directory waarnaar u wilt overschakelen, is gevonden. hyppo_findfile wordt doorgaans gebruikt om een FAT-dir-vermelding te vinden. hyppo_findfirst, hyppo_findnext en hyppo_readdir kunnen ook worden gebruikt.

Foutcodes: **$87 not a directory** De laatst gevonden FAT-dir-vermelding is geen directory. Bit 4 van het attribuutbyte van de FAT-dir-vermelding is gezet voor directories.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: U kunt naar de bovenliggende directory navigeren door de FAT-dir-vermelding .. te vinden.

U kunt niet meer dan één directory tegelijk omhoog of omlaag navigeren.

Gebruik hyppo_cdrootdir om direct terug te keren naar de hoofdmap.

Voorbeeld: Wijzigt het pad naar een willekeurig pad op het huidige station. Roep aan met Y:X als het adres van het pad. Het laatste teken van elk onderdeel in het pad moet bit 7 ingesteld hebben. Het gehele pad wordt afgesloten met een $00. Om bijvoorbeeld naar DIR1 en vervolgens DIR2 te wisselen, is het pad `!text "DIR", '1'+$80, "DIR" '2'+$80, 0`.

Bij succes wordt de C-vlag ingesteld. Als een deel van het pad niet bestaat, wordt de C-vlag gewist en is de huidige werkmap de map die het laatst succesvol is bereikt.

```
    !addr pathptr = $C4
chdirpath:
    STX pathptr : STY pathptr+1 ; Stel pathptr in op Y:Z
@chdirpath10:
    JSR @copycomponent ; Haal het volgende onderdeel van het pad op
    BEQ @chdirpath20 ; Stop bij een nulbyte in het pad
```

... gaat verder op de volgende bladzijde ...

```
    JSR @trychdir ; Probeer van map te veranderen
    BCC @chdirpath30 ; Stop als we niet van map kunnen veranderen
    BRA @chdirpath10 ; Herhaal met het volgende onderdeel
@chdirpath20: SEC
@chdirpath30: RTS
;-------------------------------------------------------------------------------
@trychdir:
    ; Stel de hyppo-bestandsnaam in vanuit transferbuffer
    LDY #>transferbuffer : LDA #$2E : STA $D640 : CLV : BCC @trychdir10
    ; Zoek de FAT-dir-entry
    LDA #$34 : STA $D640 : CLV : BCC @trychdir10
    ; Verander van map
    LDA #$0C : STA $D640 : CLV
@trychdir10:
    RTS
;-------------------------------------------------------------------------------
    ; Kopieer het volgende onderdeel van het pad naar transferbuffer
@copycomponent:
    LDY #0
@copycomponent10:
    LDA (pathptr),Y
    BEQ @copycomponent20
    TAX
    AND #$7F : STA transferbuffer,Y
    INY
    TXA : BPL @copycomponent10
    ; Voeg een nulbyte toe aan transferbuffer
    LDA #0 : STA transferbuffer,Y
    ; Verplaats pathptr
    TYA
    CLC : ADC pathptr : STA pathptr : LDA #0 : ADC pathptr+1 : STA pathptr+1
@copycomponent20:
    RTS
```

hyppo_cdrootdir

Trap: LDA #$3C : STA $D640 : CLV

Dienst: Wisselt het station door hyppo_selectdrive aan te roepen en stelt de huidige map in op de hoofdmap.

Invoer: **X** Het stationnummer dat het nieuwe huidige station wordt

Voorbeding: Geen.

Resultaat: **A** Een foutmelding die door hyppo_selectdrive kan worden geretourneerd

Geschiedenis: Beschikbaar vanaf Hyppo 1.16

hyppo_closeall

Trap: LDA #$22 : STA $D640 : CLV

Dienst: Sluit alle bestandsdescriptors.

Nawerking: Het gebruik van een willekeurige bestandsbestandsdescriptor met hyppo_closedir of hyppo_closefile slaagt.

Het gebruik van een willekeurige bestandsdescriptor met hyppo_readdir of hyppo_readfile mislukt.

hyppo_opendir en hyppo_openfile hergebruiken de bestandsdescriptor.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: U kunt ook afzonderlijke bestandsdescriptors sluiten met hyppo_closedir of hyppo_closefile.

hyppo_closedir

Trap: LDA #$16 : STA $D640 : CLV

Dienst: Sluit een bestandsdescriptor voor een map.

Voorbeding: De bestandsdescriptor die is opgegeven in het X-register is geopend met hyppo_opendir.

Invoer: **X** De bestandsdescriptor voor de map

Nawerking: Het opnieuw gebruiken van de bestandsdescriptor met hyppo_closedir slaagt.

Het opnieuw gebruiken van de bestandsdescriptor met hyppo_readdir mislukt.

hyppo_opendir en hyppo_openfile hergebruiken de bestandsdescriptor.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: U kunt ook alle geopende bestandsdescriptors sluiten met hyppo_closeall.

Voorbeeld: Zie het voorbeeld in hyppo_opendir.

hyppo_closefile

Trap: LDA #$20 : STA $D640 : CLV

Dienst: Sluit een bestandsdescriptor voor een bestand.

Voorbeding: De bestandsdescriptor die is opgegeven in het X-register is geopend met hyppo_openfile.

Invoer: **X** De bestandsdescriptor voor het bestand

Nawerking: Het opnieuw gebruiken van de bestandsdescriptor met hyppo_closefile slaagt.

Het opnieuw gebruiken van de bestandsdescriptor met hyppo_readfile mislukt.

hyppo_opendir en hyppo_openfile hergebruiken de bestandsdescriptor.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: U kunt ook alle geopende bestandsdescriptors sluiten met hyppo_closeall.

hyppo_filedate

Trap: LDA #$2C : STA $D640 : CLV

Dienst: Stelt de tijdstempel van een bestand in.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_findfile

Trap: LDA #$34 : STA $D640 : CLV

Dienst: Zoekt het eerste bestand waarvan de bestandsnaam overeenkomt met de huidige Hyppo-bestandsnaam.

Voorbeding: De huidige Hyppo-bestandsnaam is ingesteld met hyppo_setname.

Nawerking: Er zijn geen extra bestandsdescriptors geopend.

Foutcodes: **$88 file not found** Er is geen overeenkomstig bestand gevonden in de huidige map van het huidige station.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Hyppo vindt alleen bestanden waarvan de lange bestandsnaam geheel in hoofdletters staat. Hyppo converteert de huidige bestandsnaam naar ASCII-hoofdletters voordat het een overeenkomst probeert te vinden. Bytes $61 - $7B worden $41 - $5A.

Hyppo ondersteunt nog niet de jokertekens * en ?. Ondersteuning daarvoor is gepland in een toekomstige versie.

Dit vindt alleen het eerste overeenkomende bestand. U kunt meerdere overeenkomsten vinden met hyppo_findfirst en hyppo_findnext.

Voorbeeld: Zie het voorbeeld in hyppo_openfile.

hyppo_findfirst

Trap: LDA #$30 : STA $D640 : CLV

Dienst: Zoekt het eerste bestand waarvan de bestandsnaam overeenkomt met de huidige Hyppo-bestandsnaam.

Voorbeding: De huidige Hyppo-bestandsnaam is ingesteld met hyppo_setname.

Resultaat: **A** De bestandsdescriptor voor het lezen van de huidige werkmap. U bent mogelijk verantwoordelijk voor het sluiten van deze bestandsdescriptor met hyppo_closedir. Zie de opmerkingen.

Nawerking: hyppo_findnext vindt het volgende overeenkomende bestand of mislukt met een foutmelding dat het bestand niet is gevonden.

Neveneffect: Stelt de huidige bestandsdescriptor in.

Foutcodes: **$88 file not found** Er is geen overeenkomstig bestand gevonden in de huidige map van het huidige station.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Als Hyppo een eerste overeenkomend bestand vindt, stelt het de C-vlag in en retourneert het een bestandsdescriptor in het A-register. Dit is een bestandsdescriptor voor het lezen van de huidige werkmap. U bent verantwoordelijk voor het sluiten ervan met hyppo_closedir. Het is een standaard mapbestandsdescriptor. U kunt hyppo_readdir gebruiken om de FAT-mapitems na het gevonden bestand te lezen.

Als Hyppo geen overeenkomende bestanden vindt, mislukt het met een foutmelding dat het bestand niet is gevonden. In dat geval heeft Hyppo de bestandsdescriptor al gesloten en hoeft u die niet te sluiten.

Hyppo vindt alleen bestanden waarvan de lange bestandsnaam geheel in hoofdletters staat. Hyppo converteert de huidige bestandsnaam naar ASCII-hoofdletters voordat het een overeenkomst probeert te vinden. Bytes $61 - $7B worden $41 - $5A.

Hyppo ondersteunt nog niet de jokertekens * en ?. Ondersteuning daarvoor is gepland in een toekomstige versie.

Als u alleen de eerste overeenkomst zoekt, kunt u hyppo_findfile gebruiken. hyppo_findfile sluit de bestandsdescriptor altijd voor u. U kunt het echter niet gebruiken om meerdere overeenkomende bestanden te vinden.

Voorbeeld: Zie het voorbeeld in hyppo_findnext.

hyppo_findnext

Trap: LDA #$32 : STA $D640 : CLV

Dienst: Zoekt een volgend bestand waarvan de bestandsnaam overeenkomt met de huidige Hyppo-bestandsnaam.

Voorbeding: De huidige Hyppo-bestandsnaam is ingesteld met hyppo_setname.

Het eerste overeenkomende bestand is reeds succesvol gevonden met hyppo_findfirst.

Nawerking: Het opnieuw gebruiken van hyppo_findnext vindt het volgende overeenkomende bestand of mislukt met een foutmelding dat het bestand niet is gevonden.

Foutcodes: **$88 file not found** Er is geen volgend overeenkomend bestand gevonden in de huidige map van het huidige station.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Als Hyppo geen volgend overeenkomend bestand vindt, mislukt de aanroep met een file not found-fout. Hyppo sluit tevens de bestandsdescriptor die in hyppo_findfirst is uitgegeven.

Indien u de zoekopdracht niet uitput door hyppo_findnext te blijven aanroepen totdat deze met een file not found-fout mislukt, bent u verplicht de bestandsdescriptor zelf te sluiten via hyppo_closedir.

Voorbeeld: Geeft het X-register terug met het aantal bestanden dat overeenkomt met de huidige Hyppo-bestandsnaam in de huidige werkmap van het huidige station. Hoewel een directory in theorie meerdere bestanden met een identieke naam kan bevatten, wordt dit voorbeeld nuttiger zodra Hyppo jokertekens met * en ? ondersteunt.

```
    ; Neem aan dat de huidige hyppo-bestandsnaam al ingesteld is.
    ; Herstel de teller in X.
    LDX #0
    ; Probeer het eerste overeenkomende bestand te vinden.
    LDA #$30
@10:
    STA $D640 : CLV : BCC @20
    ; Verhoog de teller in X
    INX
    ; Probeer het volgende overeenkomende bestand te vinden.
    LDA #$32
    BRA @10
@20:
```

... gaat verder op de volgende bladzijde ...

```
; Als de foutcode in A $88 is, zijn er geen overeenkomende
; bestanden meer; anders is er een fout opgetreden.
CMP #$88 : BNE error
; Sluiten is niet nodig omdat de zoekactie voltooid is.
```

hyppo_fstat

Trap: LDA #$28 : STA $D640 : CLV

Dienst: Geeft informatie over een bestand.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_getcurrentdrive

Trap: LDA #$04 : STA $D640 : CLV

Dienst: Geeft het nummer van het huidige geselecteerde station (SD-kaartpartitie) terug.

Resultaten: **A** Het huidige stationsnummer

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: hyppo_selectdrive wijzigt het huidige stationsnummer. hyppo_cdrootdir kan het eveneens wijzigen.

Voorbeeld: Drukt het nummer van het huidige geselecteerde station in de linkerbovenhoek van het scherm. Dit voorbeeld gaat ervan uit dat er niet meer dan 10 stations (stations 0 tot en met 9) zijn. Het gaat er tevens vanuit dat het schermgeheugen niet is verplaatst vanaf $800.

```
; Verkrijg het huidige station
LDA #$04 : STA $D640 : CLV : BCC error

; Zet het stationsnummer in A om naar een schermcode
CLC : ADC #$30

; Zet de schermcode in de linkerbovenhoek van het schermgeheugen
STA $0800
```

hyppo_getcwd

Trap: LDA #$0A : STA $D640 : CLV

Dienst: Geeft informatie over de huidige geselecteerde directory of subdirectory.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_getdefaultdrive

Trap: LDA #$02 : STA $D640 : CLV

Dienst: Geeft het stationsnummer (SD-kaartpartitie) terug dat Hyppo tijdens het opstarten heeft geselecteerd.

Resultaten: **A** Het standaard stationsnummer

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Voorbeeld: Selecteert het standaardstation.

```
; Verkrijg het standaardstation
LDA #$02 : STA $D640 : CLV : BCC @error

; Kopieer het stationsnummer van A naar X
TAX

; Selecteer het standaardstation
LDA #$06 : STA $D640 : CLV : BCC @error
```

hyppo_getdrivesize

Trap: LDA #$08 : STA $D640 : CLV

Dienst: Geeft informatie over de grootte van het huidige geselecteerde station (SD-kaartpartitie).

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_loadfile

Trap: LDA #$36 : STA $D640 : CLV

Dienst: Laadt een bestand in chip memory.

Voorwaarden: De naam van het te laden bestand is ingesteld met hyppo_setname.

Invoer: **X** Het LSB van het adres waarmee het laden begint

Y Het middelste byte van het adres waarmee het laden begint

Z Het MSB van het adres waarmee het laden begint

Nawerkingen: Er zijn geen extra bestandsdescriptors geopend.

Foutcodes: **$84 too many open files** hyppo_loadfile maakt intern gebruik van één bestandsdescriptor, maar alle bestandsdescriptors zijn in gebruik. Sluit enkele of alle bestandsdescriptors met hyppo_closedir, hyppo_closefile of hyppo_closeall.

$88 file not found Het bestand is niet gevonden in de huidige directory van het huidige station.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Deze dienst kan bestanden tot 16 MB groot laden in de eerste 16 MB van chip memory. Chip memory is het geheugen van 384 KB of meer in de CPU-module.

Het laden begint op het 28-bit-adres $00ZZYYXX. Indien het laden voorbij $00FFFFFF probeert te gaan, slaat het om en gaat verder op $00000000.

U kunt hyppo_loadfile_attic gebruiken om een bestand in hyper memory te laden. Hyper memory is het geheugen van 8 MB of meer in de externe RAM-chips.

Voorbeeld: Laadt een bestand in het geheugen vanaf $48000.

```
; Neem aan dat de huidige hyppo-bestandsnaam al is ingesteld.
; Het is niet nodig het bestand eerst te zoeken.
LDZ #$04 ; Meest significante byte
LDY #$80 ; Middelste byte
LDX #$00 ; Minst significante byte
LDA #$36 : STA $D640 : CLV : BCC error
```

hyppo_loadfile_attic

Trap: LDA #$3E : STA $D640 : CLV

Dienst: Laadt een bestand in hyper memory.

Voorwaarden: De naam van het te laden bestand is ingesteld met hyppo_setname.

Invoer: **X** Het LSB van het adres waarmee het laden begint

Y Het middelste byte van het adres waarmee het laden begint

Z Het MSB van het adres waarmee het laden begint

Nawerkingen: Er zijn geen extra bestandsdescriptors geopend.

Foutcodes: **$84 too many open files** hos_loadfile_attic maakt intern gebruik van één bestandsdescriptor, maar alle bestandsdescriptors zijn in gebruik. Sluit enkele of alle bestandsdescriptors met hyppo_closedir, hyppo_closefile of hyppo_closeall.

$88 file not found Het bestand is niet gevonden in de huidige directory van het huidige station.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Deze dienst kan bestanden tot 16 MB groot laden in de eerste 16 MB van hyper memory. Hyper memory is het geheugen van 8 MB of meer in de externe RAM-chips.

Het laden begint op het 28-bit-adres $08ZZYYXX. Indien het laden voorbij $08FFFFFF probeert te gaan, slaat het om en gaat verder op $08000000.

U kunt hyppo_loadfile gebruiken om een bestand in chip memory te laden. Chip memory is het geheugen van 384 KB of meer in de CPU-module.

hyppo_mkdir

Trap: LDA #$0E : STA $D640 : CLV

Dienst: Maakt een subdirectory aan.

Foutcodes: **$8D file exists** Er bestaat al een subdirectory of bestand met de huidige Hyppo-bestandsnaam in de huidige werkmap van het huidige station.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_mkfile

Trap: LDA #$1E : STA $D640 : CLV

Dienst: Maakt een bestand aan.

Foutcodes: **$8D file exists** Er bestaat al een subdirectory of bestand met de huidige Hyppo-bestandsnaam in de huidige werkmap van het huidige station.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_opendir

Trap: LDA #$12 : STA $D640 : CLV

Dienst: Opent de huidige werkmap om de bestandsvermeldingen erin te lezen.

Voorwaarden: Het station en de directory die u wilt lezen, zijn indien nodig al ingesteld met hyppo_selectdrive en hyppo_chdir.

Resultaten: **A** De bestandsdescriptor voor het lezen van de directory. U bent zelf verantwoordelijk voor het sluiten van deze bestandsdescriptor met hyppo_closedir.

Nawerkingen: hyppo_readdir leest de eerste FAT-dir-entry in de directory.

Foutcodes: **$84 too many open files** Alle bestandsdescriptors zijn in gebruik. hyppo_opendir en hyppo_openfile maken gezamenlijk gebruik van dezelfde kleine voorraad bestandsdescriptors. Sluit enkele of alle bestandsdescriptors met hyppo_closedir, hyppo_closefile of hyppo_closeall.

$87 not a directory De laatst gevonden FAT-dir-entry is van een bestand. Gebruik hyppo_openfile voor bestanden.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Voorbeeld: Roept processdirentry aan voor elke FAT-dir-entry in de huidige werkmap. processdirentry wordt elders gedefinieerd verondersteld.

```
    ; Open de huidige werkmap
    LDA #$12 : STA $D640 : CLV : BCC error
    ; Verplaats de bestandsdescriptor naar X
    TAX
    ; Stel Y in op het MSB van het overdrachtsgebied
    LDY #)transferarea
@10:
    ; Lees het directory-item
    LDA #$14 : STA $D640 : CLV : BCC @20
    ; Roep processdirentry aan (wordt elders gedefinieerd)
    PHX : PHY : JSR processdirentry : PLY : PLX
    BRA @10
@20:
    ; Als de foutcode in A $85 is, hebben we het einde van de map
    ; bereikt; anders is er een fout opgetreden
    CMP #$85 : BNE error
    ; Sluit de bestandsindicator in X
    LDA #$16 : STA $D640 : CLV : BCC error
```

hyppo_openfile

Trap: LDA #$18 : STA $D640 : CLV

Dienst: Opent een bestand op een station.

Voorwaarden: Het bestand is al gevonden. Bestanden kunnen worden gevonden met hyppo_findfile, hyppo_findfirst en hyppo_findnext. hyppo_readdir kan ook worden gebruikt om een bestand te vinden.

Resultaten: **A** Het bestandsdescriptor voor toegang tot het bestand. U bent zelf verantwoordelijk voor het sluiten van deze bestandsdescriptor met hyppo_closefile.

Nawerkingen: Het gebruik van hyppo_readfile met deze bestandsdescriptor leest de eerste sector van het bestand.

Neveneffecten: Stelt het huidige bestand in op het nieuw geopende bestand. hyppo_readfile leest vanuit het huidige bestand.

Foutcodes: **$84 too many open files** Alle bestandsdescriptors zijn in gebruik. hyppo_opendir en hyppo_openfile maken gezamenlijk gebruik van dezelfde kleine voorraad bestandsdescriptors. Sluit enkele of alle bestandsdescriptors met hyppo_closedir, hyppo_closefile of hyppo_closeall.

$86 is a directory De laatst gevonden FAT-dir-entry is van een directory. Gebruik hyppo_opendir voor directories.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: U kunt dit niet gebruiken om een bestand binnen een schijfimage te openen. Gebruik in plaats daarvan hyppo_attach om de schijfimage te koppelen en gebruik vervolgens de KERNAL om het bestand te lezen of programmeer de virtuele F011-floppycontroller.

Voorbeeld: Zoekt en opent een bestand.

```
; Neem aan dat de huidige hyppo-bestandsnaam al is ingesteld.
; Zoek het bestand
LDA #$34 : STA $D640 : CLV : BCC error
; Open het bestand
LDA #$18 : STA $D640 : CLV : BCC error
```

hyppo_readdir

Trap: LDA #$14 : STA $D640 : CLV

Dienst: Leest de volgende FAT-dir-entry in een bestemmingsgebied.

Voorwaarden: De bestandsdescriptor die in het X-register is opgegeven, is geopend met hyppo_opendir en hyppo_closedir is sindsdien niet gebruikt om deze te sluiten.

Het bestemmingsgebied bevindt zich op een paginagrens tussen $0000 en $7E00 en is ten minste 87 bytes groot.

Invoer: **X** De bestandsdescriptor voor de directory.

Y Het MSB van het bestemmingsgebied.

Resultaten: Vanaf $YY00 heeft de FAT-dir-entry de volgende structuur.

Offset	Type	Omschrijving
$00	asciiz	De lange bestandsnaam
$40	byte	De lengte van de lange bestandsnaam
$41	ascii	De "8.3"-bestandsnaam. Het naamgedeelte wordt aangevuld met spaties zodat het precies 8 bytes is. De 3 bytes van de extensie volgen. Er staat geen punt tussen de naam en de extensie. Er is geen NULL-byte.
$4E	dword	Het clusternummer waar het bestand begint. Voor sub-directories is dit de plaats waar de FAT-dir-entries voor die subdirectory beginnen.
$52	dword	De lengte van het bestand in bytes.
$56	byte	Het type- en attribuutbyte.

Dit is de betekenis van de bits in het laatste byte. Bits 6 en 7 zijn ongedefinieerd.

Bit	Betekenis indien bit ingesteld
0	Alleen lezen
1	Verborgen
2	Systeem
3	Volumelabel
4	Sub-directory
5	Archief

Nawerkingen: Het opnieuw gebruiken van hyppo_readdir leest de volgende FAT-dir-entry in de directory.

Foutcodes: **$08 partition error** Er is een ongespecificeerde fout opgetreden bij het verwerken van de huidig geselecteerde partitie.

$10 invalid address Het Y-register is > $7E.

$85 invalid cluster Er is geprobeerd voorbij het einde van de directory te lezen.

Opmerkingen: Indien de lange bestandsnaam in de FAT-dir-entry te lang is om in het bestemmingsgebied te kopiëren, slaat Hyppo de entry geheel over.

De bestandsnamen in FAT zijn gecodeerd als UTF-16. Hyppo leest alleen het LSB van elk 16-bit teken. Hyppo converteert bestandsnamen niet naar PETSCII.

Zie hyppo_setup_transfer_area voor meer details over de waarde van het Y-register.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Voorbeeld: Zie het voorbeeld bij hyppo_opendir.

hyppo_readfile

Trap: LDA #$1A : STA $D640 : CLV

Dienst: Leest het volgende sector van het huidige bestand in de sectorbuffer.

Voorwaarden: Er is een huidig bestand geopend. Bestanden kunnen worden geopend met hyppo_openfile.

Resultaten: **X** Het LSB van het aantal gelezen bytes

Y Het MSB van het aantal gelezen bytes

Nawerkingen: De volgende aanroep van hyppo_readfile leest het volgende sector van het huidige bestand of geeft het einde van het bestand aan.

Foutcodes: **$89 invalid file descriptor** Er is geen huidig bestand.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Om de gegevens te benaderen, moet u het volgende doen:

- de sectorbuffer in de 16-bit adresruimte te plaatsen;
- een verbeterde DMA-overdracht te gebruiken om de sectorbuffer op $FFD6E00 - $FFD6FFF naar een buffer te kopiëren die al in de 16-bit adresruimte is geplaatst; of
- 32-bit laadinstructies te gebruiken om de sectorbuffer direct te benaderen.

Indien een volledig sector is gelezen, is Y:X gelijk aan $0200. Voor het laatste sector van het bestand kan Y:X kleiner zijn. Bytes in de sectorbuffer na Y:X zijn ongedefinieerd en niet noodzakelijk nul.

Indien u voorbij het einde van het laatste sector leest, is Y:X gelijk aan $0000, is het A-register gelijk aan $FF en is de C-vlag ingesteld.

Hoewel meerdere bestanden tegelijkertijd kunnen worden geopend, kan alleen het huidige bestand worden gelezen. Het huidige bestand is meestal het laatst geopende bestand, maar niet altijd.

Voorbeeld: Plaatst de sectorbuffer op $DE00 en leest vervolgens elk sector van het bestand door voor elk gelezen sector een aanroep te doen naar proccesssector. proccesssector wordt elders gedefinieerd verondersteld.

```
; Neem aan dat het bestand al geopend is.
; Koppel kleuren-RAM los van $DC00 om de sectorbuffer te kunnen
; inladen
```

... gaat verder op de volgende bladzijde ...

```
    LDA $D030 : PHA : AND #%11111110 : STA $D030
@10:
    ; Lees de volgende sector
    LDA #$1A : STA $D640 : CLV : BCC @20
    ; Breng de sectorbuffer naar $DE00
    LDA #$81 : STA $D680
    ; Roep processsector aan (wordt elders gedefinieerd)
    JSR processsector
    ; Koppel de sectorbuffer los van $DE00
    LDA #$82 : STA $D680
    BRA @10
@20:
    ; Als de foutcode in A $FF is, hebben we het einde van het bestand
    ; bereikt; anders is er een fout opgetreden
    CMP #$FF : BNE error
    ; Breng kleuren-RAM terug naar $DC00 als het daarvoor in het
    ; geheugen stond
    PLA : STA $D030
```

hyppo_rename

Trap: LDA #$2A : STA $D640 : CLV

Dienst: Hernoemt een bestand of subdirectory.

Foutcodes: **$8D file exists** Er bestaat al een subdirectory of bestand met de huidige Hyppo-bestandsnaam in de huidige werkmap van het huidige station.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_rmdir

Trap: LDA #$10 : STA $D640 : CLV

Dienst: Verwijdert een subdirectory.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_rmfile

Trap: LDA #$26 : STA $D640 : CLV

Dienst: Verwijdert een bestand.

Voorbeding: Het bestand is reeds gevonden. Bestanden kunnen worden gevonden met hyppo_findfile, hyppo_findfirst en hyppo_findnext. hyppo_readdir kan ook worden gebruikt om een bestand te vinden.

Nawerking: Het bestand is verwijderd en de sectoren zijn vrijgegeven.

Geschiedenis: Beschikbaar vanaf Hyppo 1.3

Opmerkingen: Deze documentatie is onvolledig en in opbouw!

hyppo_seekfile

Trap: LDA #$24 : STA $D640 : CLV

Dienst: Zoekt naar een bepaalde sector in een bestand.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_selectdrive

Trap: LDA #$06 : STA $D640 : CLV

Dienst: Stelt het momenteel geselecteerde station (SD-kaartpartitie) in.

Voorbeding: Hyppo heeft een stationsnummer toegewezen aan de SD-kaartpartitie.

Invoer: **X** Het stationsnummer dat het nieuwe huidige station wordt

Nawerking: hyppo_getcurrentdrive geeft de waarde terug die in het X-register stond.

Hyppo-services werken op het nieuw geselecteerde station.

Foutcodes: **$80 no such drive** Het station in het X-register bestaat niet. Hyppo wijst alleen stationsnummers toe aan SD-kaartpartities die het kan lezen.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Voorbeeld: Test of station 2 bestaat door het te selecteren. Keert terug met de C-vlag ingesteld als station 2 bestaat.

```
doesdrive2exist:
    ; Bewaar het huidige station zodat we het later kunnen herstellen
    LDA #$04 : STA $D640 : CLV : BCC error
    PHA
    ; Probeer station 2 te selecteren
    LDX #2 : LDA #$06 : STA $D640 : CLV : BCC @10
    ; Herstel het eerder geselecteerde station
    PLX
    LDA #$06 : STA $D640 : CLV : BCC error
    ; De C-vlag was reeds gezet door de Hyppo-service
    RTS

@10:
    ; Als de foutcode in A $80 is, bestaat het station niet; anders is
    ; een andere fout opgetreden
    CMP #$80
    BNE error
    ; Negeer het huidige station dat we bewaarden, want het is niet
    ; gewijzigd
    PLX
    ; Wis de C-vlag omdat het station niet bestaat
    CLC
    RTS
```

hyppo_setname

Trap: LDA #$2E : STA $D640 : CLV

Dienst: Stelt de huidige Hyppo-bestandsnaam in.

Voorbeding: De bestandsnaam is opgeslagen in ASCII en eindigt op een $00-byte.

De bestandsnaam begint op een paginagrens tussen $0000 en $7E00 en is korter dan 63 tekens, exclusief de $00-byte.

Invoer: **Y** Het MSB van het bestandsnaamadres.

Nawerking: Hyppo heeft de bestandsnaam gekopieerd naar het eigen gegevensgebied.

De hyppo_find*- en hyppo_load*-services gebruiken deze bestandsnaam.

Neveneffect: Stelt het overdrachtsgebied in op $YY00.

Foutcodes: **$10 invalid address** Het Y-register is > $7E.

$81 name too long De bestandsnaam is langer dan 63 tekens.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: De bestandsnaam moet tussen $0000 en $7E00 liggen. Deze moet ook op een paginagrens beginnen. Het adres moet dus eindigen op $00. De huidige geheugenindeling wordt gebruikt. Het is echter aan te raden om de bestandsnaam in de onderste 32KB van bank 0 te plaatsen.

De bestandsnamen in FAT worden gecodeerd in UTF-16. Hyppo leest alleen het LSB van elk 16-bit teken. Hyppo voert geen conversie uit tussen ASCII en PETSCII.

Hyppo geeft toegang tot de bestanden in het FAT-bestandssysteem op de interne en externe SD-kaarten. Het geeft geen toegang tot bestanden op schijven in floppy-stations of in schijfimages.

Voorbeeld: Stel de huidige Hyppo-bestandsnaam in op GAME.MAP.

```
; Ergens in het gegevensgebied van het programma
   !align 255, 0 ; Uitlijnen op de volgende paginagrens
filename:
   !text "GAME.MAP", 0 ; Moet eindigen met een nulbyte

; Elders in de programmacode
setfilename:
   LDY #>filename : LDA #$2E : STA $D640 : CLV : BCC error
```

hyppo_writefile

Trap: LDA #$1C : STA $D640 : CLV

Dienst: Schrijft de sectorbuffer naar het huidige bestand.

Voorbeding: Er is een huidig bestand geopend. Bestanden kunnen worden geopend met hyppo_openfile.

Nawerking: De volgende aanroep van hyppo_writefile schrijft de volgende sector van het huidige bestand.

Foutcodes: **$89 invalid file descriptor** Er is geen huidig bestand.

Geschiedenis: Beschikbaar vanaf Hyppo 1.3

Opmerkingen: Deze documentatie is onvolledig en in opbouw!

Om de gegevens aan te leveren, heeft u een van de volgende mogelijkheden:

- plaats de sectorbuffer in de 16-bit adresruimte;
- gebruik een verbeterde DMA-overdracht om de sectorbuffer van $FFD6E00 - $FFD6FFF te kopiëren naar een buffer die reeds in de 16-bit adresruimte is geladen; of
- gebruik 32-bit laadinstructies om direct toegang tot de sectorbuffer te krijgen.

SCHIJFIMAGE-SERVICES

De 45IO27 multifunctionele I/O-controller bevat een F011-compatibele floppycontroller. Het interne floppy-station is hierop aangesloten als station 0.

Hyppo kan de F011 floppy-controller virtualiseren zodat schijfimages kunnen worden gekoppeld in plaats van floppy-stations. Zodra een schijfimage is gekoppeld, vangt Hyppo de I/O-registers van de F011 af en emuleert de commando's op de schijfimage.

U kunt BASIC, de KERNAL en de F011 I/O-registers gebruiken om een schijfimage net als een fysieke schijf te bewerken. De virtualisatie gedraagt zich niet in alle gevallen hetzelfde als een floppy-station. Indien u wilt dat uw programma met zowel schijfimages als fysieke schijven werkt, test het dan met beide.

hyppo_attach

Trap: LDA #$4A : STA $D640 : CLV

Dienst: Koppelt of ontkoppelt een schijfimage aan het virtuele F011-station.

Voorbeding: (koppelen) De huidige Hyppo-bestandsnaam is ingesteld met hyppo_setname.

Invoer: **X.0** Selecteer station 0 of 1.

X.1 Bij X.0=1, selecteer beide stations (alleen ontkoppelen).

X.6 Herkoppel geen fysieke stations (alleen ontkoppelen).

X.7 Werkingsmodus: 0=koppelen, 1=ontkoppelen.

Foutcodes: **$81 name to long** De naam van de schijfimage is te lang voor de procesdescriptor (max. 31 tekens).

$88 file not found De schijfimage is niet gevonden in de huidige map van het huidige station.

$8A image wrong length De lengte van de image komt niet overeen met een van de ondersteunde imagetypen.

$8B image fragmented De gegevens van de image bevinden zich niet in één aaneengesloten reeks van sectoren; dit wordt niet ondersteund.

$8F double attach De gevraagde image is al gekoppeld aan het andere station.

Geschiedenis: Beschikbaar vanaf Hyppo 1.3

Opmerkingen: Tenzij dit is gewijzigd, is station 0 van de virtuele F011 floppy-controller apparaat 8 en station 1 apparaat 9.

Voorbeeld: Koppelt en ontkoppelt de schijfimage DISK2.D81 aan virtueel F011 station 0.

```
; Ergens in het gegevensgebied van het programma
    !align 255, 0 ; Lijn uit op de volgende paginagrens
disk2name:
    !text "DISK2.D81", 0 ; Moet eindigen met een nulbyte

; Elders in de code van het programma
attachdisk:
    ; Stel de hyppo-bestandsnaam in op DISK2.D81
    LDY #>disk2name : LDA #$2E : STA $D640 : CLV : BCC error
    ; Koppel de schijfimage aan
```

... gaat verder op de volgende bladzijde ...

```
    LDX #$00 : LDA #$4A : STA $D640 : CLV : BCC error
    RTS

detachdisk:
    ; Ontkoppel de schijfimage en herstel het fysieke station
    LDX #$80 : LDA #$4A : STA $D640 : CLV : BCC error
    RTS

good:
    LDX #$53 : STX $800 : RTS

error:
    LDX #$56 : STX $800 : STA $801 : RTS

main:
    JSR attachdisk
    JSR detachdisk
```

hyppo_d81attach0

Trap: LDA #$40 : STA $D640 : CLV

Dienst: Koppelt een D81-schijfimage aan virtueel F011 station 0.

Voorbeding: De huidige Hyppo-bestandsnaam is ingesteld met hyppo_setname.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2
VEROUDERD sinds Hyppo 1.3. Deze dienst kan in een toekomstige versie worden verwijderd. Nieuwe programma's moeten in plaats daarvan attach gebruiken.

Opmerkingen: Deze service aanroept eigenlijk hyppo_attach met **X** = $00

hyppo_d81attach1

Trap: LDA #$46 : STA $D640 : CLV

Dienst: Koppelt een D81-schijfimage aan virtueel F011 station 1.

Voorbeding: De huidige Hyppo-bestandsnaam is ingesteld met hyppo_setname.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

VEROUDERD sinds Hyppo 1.3. Deze dienst kan in een toekomstige versie worden verwijderd. Nieuwe programma's moeten in plaats daarvan attach gebruiken.

Opmerkingen: Deze dienst roept eigenlijk hyppo_attach aan met **X** = $01

hyppo_d81detach

Trap: LDA #$42 : STA $D640 : CLV

Dienst: Koppelt alle schijfimages los van de virtuele F011-stations 0 en 1.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

VEROUDERD sinds Hyppo 1.3. Deze dienst kan in een toekomstige versie worden verwijderd. Nieuwe programma's moeten in plaats daarvan attach gebruiken.

Opmerkingen: Deze dienst roept eigenlijk hyppo_attach aan met **X** = $C2

hyppo_d81write_en

Trap: LDA #$44 : STA $D640 : CLV

Dienst: Schakelt schrijven in voor alle schijfimages die gekoppeld zijn aan virtuele F011-stations 0 en 1.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Images worden standaard gekoppeld met schrijven ingeschakeld, zodat deze aanroep niet nodig is.

TAAK- EN PROCESDIENSTEN

hyppo_create_task_c64

Trap: LDA #$66 : STA $D640 : CLV

Dienst: Maakt een Hyppo-taak aan in de C64-achtige werking.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_create_task_c65

Trap: LDA #$68 : STA $D640 : CLV

Dienst: Maakt een Hyppo-taak aan in de C65-achtige werking.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_create_task_native

Trap: LDA #$62 : STA $D640 : CLV

Dienst: Maakt een Hyppo-taak aan in de MEGA65-werking.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_exit_and_switch_to_task

Trap: LDA #$6A : STA $D640 : CLV

Dienst: Sluit de huidige Hyppo-taak en schakelt de context over naar een andere Hyppo-taak.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_exit_task

Trap: LDA #$6E : STA $D640 : CLV

Dienst: Sluit de huidige Hyppo-taak.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_get_mapping

Trap: LDA #$74 : STA $D640 : CLV

Dienst: Kopieert de huidige 45GS02-geheugenindeling naar een bestemmingsgebied.

Voorwaarden: Het bestemmingsgebied begint op een paginagrens tussen $0000 en $7E00 en is minstens 6 bytes lang.

Invoer: **Y** Het hoogste byte van het bestemmingsgebied.

Uitvoer: Beginnend op $YY00 heeft de huidige indelingsinformatie de volgende structuur.

Offset	Type	Omschrijving
0	word	MAPLO (big-endian)
2	word	MAPHI (big-endian)
4	byte	Het megabyte-offset voor MAPLO
5	byte	Het megabyte-offset voor MAPHI

Foutcodes: **$10 invalid address** Het Y-register is > $7E.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: MAPLO is de indeling voor $0000 - $7FFF.

MAPHI is de indeling voor $8000 - $FFFF.

Zie *het grote Mega65-boek*, 45GS02-microprocessor (Bijlage K) voor meer informatie over geheugenindeling en banken van de MEGA65.

hyppo_get_proc_desc

Trap: LDA #$48 : STA $D640 : CLV

Dienst: Kopieert het huidige taakblok naar een bestemmingsgebied.

Voorwaarden: Het bestemmingsgebied begint op een paginagrens tussen $0000 en $7E00 en is minstens 256 bytes lang.

Invoer: **Y** Het hoogste byte van het bestemmingsgebied.

Uitvoer: Beginnend op $YY00 heeft het huidige taakblok de volgende structuur.

Offset	Type	Omschrijving
$00	byte	Het ID van de huidige taak.
$01	text	De naam van de huidige taak. Maximaal 16 tekens. Opgevuld met $00-bytes. Bij 16 tekens staan er geen afsluitende $00-bytes.
$11	byte	Vlaggen voor de D81-schijfimage die gekoppeld is aan station 0 van de virtuele F011-floppycontroller.
$12	byte	Idem maar voor station 1.
$13	byte	De lengte van de bestandsnaam van de D81-schijfimage die gekoppeld is aan station 0.
$14	byte	Idem maar voor station 1.
$15	text	De bestandsnaam van de D81-schijfimage die gekoppeld is aan station 0. Maximaal 32 tekens. Opgevuld met $20-bytes. Er staan geen afsluitende $00-bytes.
$35	text	Idem maar voor station 1.
$55		De betekenis van deze bytes is ongedefinieerd en kan veranderen.
$80		Bestandsdescriptor 0.
$A0		Bestandsdescriptor 1.
$C0		Bestandsdescriptor 2.
$E0		Bestandsdescriptor 3.

Elke bestandsdescriptor heeft de volgende structuur.

Offset	Type	Omschrijving
$00	byte	Het nummer van de SD-kaartpartitie waarin het bestand staat. $FF betekent dat de bestandsdescriptor gesloten is.
$01	dword	Het cluster waar het bestand begint
$05	dword	Het huidige cluster
$09	byte	De huidige sector binnen het huidige cluster
$0A	dword	De lengte van het bestand
$0E	dword	De huidige positie binnen de buffer van het bestand
$12	dword	Het cluster van de directory waarin het bestand staat
$16	word	De index van het bestand binnen zijn directory
$18	dword	Het absolute 32-bits adres van de buffer van het bestand
$1C	word	Het aantal bytes dat in de buffer van het bestand wordt gebruikt
$1E	word	De huidige offset binnen de buffer van het bestand

Foutcodes: **$10 invalid address** Het Y-register is > $7E.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_gettasklist

Trap: LDA #$50 : STA $D640 : CLV

Dienst: Haalt de lijst met taken op uit Hyppo.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_load_into_task

Trap: LDA #$64 : STA $D640 : CLV

Dienst: Laadt een bestand vanaf een SD-kaartpartitie in het geheugen van een Hyppo-taak.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_readoutoftask

Trap: LDA #$58 : STA $D640 : CLV

Dienst: Leest uit het geheugen van een andere Hyppo-taak.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_receivemessage

Trap: LDA #$54 : STA $D640 : CLV

Dienst: Ontvangt berichten die door andere Hyppo-taken zijn verzonden.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_reset

Trap: LDA #$7E : STA $D640 : CLV

Dienst: Voert een warme start van de MEGA65 uit.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_rom_writeenable

Trap: LDA #$02 : STA $D641 : CLV

Dienst: Verandert $20000 - $3FFFF zodat ze zich als RAM gedragen door de schrijfbeveiliging uit te schakelen.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: $20000 - $3FFFF bevat normaal de KERNAL-, BASIC-, CBDOS- en lettertype-ROM's.

hyppo_rom_writeprotect schakelt de schrijfbeveiliging in en blokkeert schrijfbewerkingen. hyppo_toggle_rom_writeprotect wisselt de schrijfbeveiliging om.

hyppo_rom_writeprotect

Trap: LDA #$00 : STA $D641 : CLV

Dienst: Verandert $20000 - $3FFFF zodat ze zich als ROM gedragen door de schrijfbeveiliging in te schakelen.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: $20000 - $3FFFF bevat normaal de KERNAL-, BASIC-, CBDOS- en lettertype-ROM's.

hyppo_rom_writeenable schakelt de schrijfbeveiliging uit en staat schrijfbewerkingen toe. hyppo_toggle_rom_writeprotect wisselt de schrijfbeveiliging om.

hyppo_sendmessage

Trap: LDA #$52 : STA $D640 : CLV

Dienst: Verstuurt een bericht naar een andere Hyppo-taak.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_serial_monitor_wait_and_write

Trap: LDA #$xx : STA $D643 : CLV

Dienst: Wacht totdat de seriële monitor of de Matrix Mode Debugger gereed is om te ontvangen en schrijft vervolgens een teken naar het apparaat.

Invoer: **A** Het ASCII-teken dat geschreven moet worden.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: De service wacht tot de seriële monitor gereed is om te ontvangen. Dit kan uw programma vertragen of vast laten lopen. Indien u dat niet wilt en het geen bezwaar vindt dat het teken verloren gaat wanneer de seriële monitor niet gereed is, gebruik dan hyppo_serial_monitor_write.

hyppo_serial_monitor_write

Trap: LDA #$7C : STA $D640 : CLV

Dienst: Schrijft een teken naar de seriële monitor of de Matrix Mode Debugger.

Voorwaarden: De seriële monitor is gereed om te ontvangen.

Invoer: **Y** Het ASCII-teken dat geschreven moet worden.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Het teken gaat verloren indien de seriële monitor niet gereed is om te ontvangen. Indien u niet wilt dat het teken verloren gaat, gebruik dan hyppo_serial_monitor_wait_and_write.

hyppo_set_mapping

Trap: LDA #$76 : STA $D640 : CLV

Dienst: Kopieert het brongebied naar de huidige 45GS02-geheugenindeling.

Voorwaarden: Het brongebied begint op een paginagrens tussen $0000 en $7E00 en is minimaal 6 bytes.

Invoer: **Y** Het meest significante byte van het brongebied.

Vanaf $YY00 heeft de huidige mappingsinformatie de volgende structuur.

Offset	Type	Omschrijving
0	word	MAPLO (big-endian)
2	word	MAPHI (big-endian)
4	byte	Het megabyte-offset voor MAPLO
5	byte	Het megabyte-offset voor MAPHI

Nawerkingen: De CPU gaat verder met de nieuwe geheugenindeling.

Foutcodes: **$10 invalid address** Het Y-register is > $7E.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: Wees voorzichtig bij het wijzigen van de geheugenindeling. Hyppo zal geen maatregelen nemen om te garanderen dat de instructies na de STA worden uitgevoerd ongeacht de indeling. Indien u de indeling van het blok waar het programmateller naar wijzigt, zal de CPU hervatten met de instructies in het nieuw toegewezen blok.

MAPLO is de mapping voor $0000 - $7FFF.

MAPHI is de mapping voor $8000 - $FFFF.

Zie *het grote Mega65-boek*, 45GS02-microprocessor (Bijlage K) voor meer informatie over de geheugenindeling en bankindeling van de MEGA65.

hyppo_switch_to_task

Trap: LDA #$6C : STA $D640 : CLV

Dienst: Schakelt de context over naar een andere Hyppo-taak.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_terminateothertask

Trap: LDA #$60 : STA $D640 : CLV

Dienst: Beëindigt een andere Hyppo-taak.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

hyppo_toggle_force_4502

Trap: LDA #$72 : STA $D640 : CLV

Dienst: Schakelt de CPU-persoonlijkheid om tussen 45GS02 en 6502.

Uitvoer: **A** Indien bit 5 is gezet, bevindt de CPU zich in de 45GS02-persoonlijkheid. Indien bit 5 is gewist, bevindt de CPU zich in de 6502-persoonlijkheid.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: De overige bits van het A-register zijn ongedefinieerd. Verwacht niet dat ze nul zijn.

In de 6502-persoonlijkheid zijn geen van de nieuwe opcodes van de 65C02, 65CE02, 4510 of 45GS02 beschikbaar. Deze worden vervangen door het oorspronkelijke — en vaak vreemde — gedrag van de ongedefinieerde opcodes van de 6502.

Waarschuwing Deze functie is onvolledig en niet getest. De meeste ongedocumenteerde 6502-opcodes werken niet correct wanneer de 6502-persoonlijkheid is ingeschakeld.

Voorbeeld: Schakelt de 45GS02-persoonlijkheid in ongeacht de huidige persoonlijkheid van de CPU.

```
@10
    ; Wissel de CPU-persoonlijkheid
    LDA #$72 : STA $D640 : CLV : BCC error
    ; Wissel opnieuw indien bit 5 van A niet ingesteld is
    AND #%00100000 : BEQ @10
```

hyppo_toggle_rom_writeprotect

Trap: LDA #$70 : STA $D640 : CLV

Dienst: Schakelt de schrijfbescherming voor $20000 - $3FFFF om.

Uitvoer: **A** Indien bit 2 is gezet, kan naar $20000 - $3FFFF niet worden geschreven.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

Opmerkingen: De overige bits van het A-register zijn ongedefinieerd. Verwacht niet dat ze nul zijn.

Indien u de bescherming eenvoudig wilt uitschakelen of inschakelen, kunt u hyppo_rom_writeenable en hyppo_rom_writeprotect gebruiken.

hyppo_writeintotask

Trap: LDA #$56 : STA $D640 : CLV

Dienst: Schrijft in het geheugen van een andere Hyppo-taak.

Opmerkingen: **NIET GEÏMPLEMENTEERD**

SYSTEMPARTITIESERVICES

hyppo_configsector_apply

Trap: LDA #$04 : STA $D642 : CLV

Dienst: Past de systeemconfiguratie-sector toe die momenteel in het geheugen is geladen.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_configsector_read

Trap: LDA #$00 : STA $D642 : CLV

Dienst: Leest de systeemconfiguratie-sector in het geheugen.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_configsector_write

Trap: LDA #$02 : STA $D642 : CLV

Dienst: Schrijft de systeemconfiguratie-sector vanuit het geheugen.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_dmagic_autoset

Trap: LDA #$06 : STA $D642 : CLV

Dienst: Stelt de DMAgic-revisie in op basis van de geladen ROM.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

FREEZERSERVICES

hyppo_freeze_self

Trap: LDA #$xx : STA $D67F : CLV

Dienst: Start de Freezer.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_get_slot_count

Trap: LDA #$16 : STA $D642 : CLV

Dienst: Geeft het aantal freeze-slots op.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_locate_freeze_slot

Trap: LDA #$10 : STA $D642 : CLV

Dienst: Zoekt de eerste sector van een freeze-slot.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_read_freeze_region_list

Trap: LDA #$14 : STA $D642 : CLV

Dienst: Leest de freeze-regiolijst.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

hyppo_unfreeze_from_slot

Trap: LDA #$12 : STA $D642 : CLV

Dienst: Maakt een freeze-slot ongedaan.

Geschiedenis: Beschikbaar vanaf Hyppo 1.2

DEEL V

MULTIMEDIA EN GEGEVENSCONVERSIE

DEEL VI

GEOS-ONTWIKKELING

DEEL VII

HARDWARE

HOOFDSTUK 16

Nexys4-borden gebruiken als een MEGA65

- Uw eigen MEGA65-compatibele computer bouwen
- Werkende Nexys4-borden
- Stroom, jumpers, schakelaars en knoppen
- Toetsenbord
- MicroSDHC-kaart voorbereiden
- De bitstream laden vanuit QSPI
- Widget-bord
- PMOD-naar-joystick-adapter

- **Handige tips**

UW EIGEN MEGA65-COMPATIBELE COMPUTER BOUWEN

U kunt uw eigen MEGA65-compatibele computer bouwen met een Nexys4DDR-bord (ook wel Nexys A7) of een ouder Nexys4-bord (zonder DDR). Deze bijlage beschrijft het proces om een Nexys4DDR (Nexys A7)-bord hiervoor in te stellen (het nieuwere, bij voorkeur te gebruiken bord). Het oudere non-DDR Nexys4-bord wordt ook ondersteund en de instructies zijn hetzelfde, met uitzondering van het feit dat u een bitstream moet gebruiken die voor dat bord ontworpen is. Het gebruik van een Nexys4DDR-bitstream op een non-DDR Nexys4-bord of andersom kan onherstelbare schade aan uw bord veroorzaken. Zorg er daarom voor dat u de juiste bitstream voor uw bord heeft.

DISCLAIMER: M.E.G.A kan geen enkele verantwoordelijkheid aanvaarden voor schade die aan uw Nexys4DDR/NexysA7/Nexys4-borden kan ontstaan.

WERKENDE NEXYS4-BORDEN

Er zijn momenteel drie Nexys FPGA-borden die als MEGA65 ingesteld kunnen worden:

Het Nexys4-bord

Niet meer geproduceerd, maar nog wel verkrijgbaar via sommige websites met oude voorraad.



Documentatie:

- https://reference.digilentinc.com/reference/programmable-logic/nexys-4/reference-manual
- https://reference.digilentinc.com/_media/reference/programmable-logic/nexys-4/nexys4_rm.pdf

Het Nexys4DDR-bord

Niet meer geproduceerd, maar nog wel verkrijgbaar via sommige websites met oude voorraad.



Documentatie:

- https://reference.digilentinc.com/reference/programmable-logic/nexys-4-ddr/reference-manual
- https://reference.digilentinc.com/_media/reference/programmable-logic/nexys-4-ddr/nexys4ddr_rm.pdf

De Nexys A7

Dit is het bovenstaande Nexys4 DDR-bord, opnieuw uitgebracht onder een andere naam:

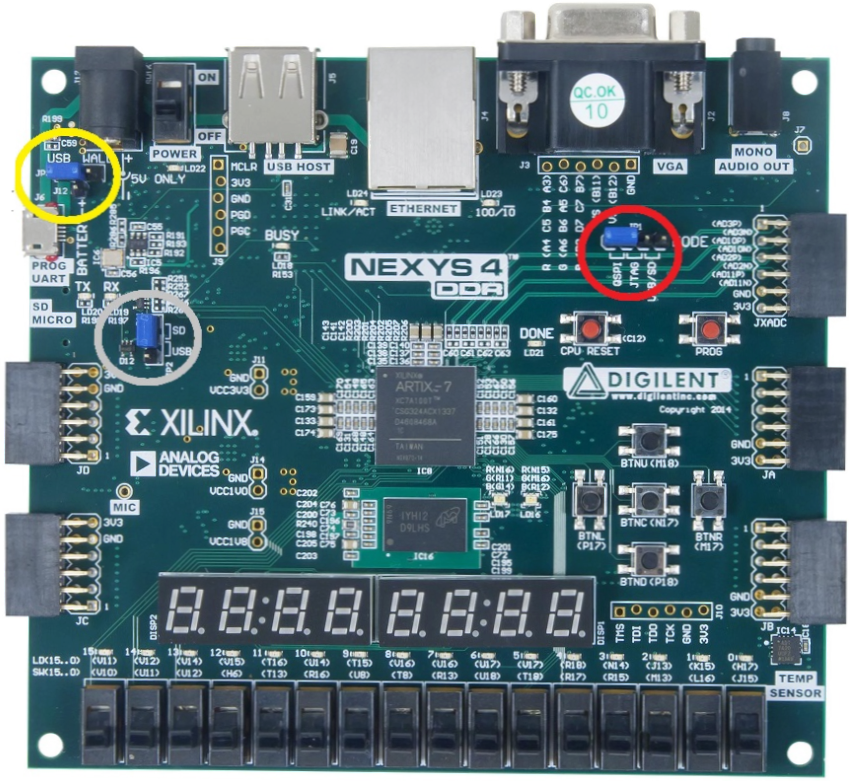


Documentatie:

- https://reference.digilentinc.com/reference/programmable-logic/nexys-a7/reference-manual
- https://reference.digilentinc.com/_media/reference/programmable-logic/nexys-a7/nexys-a7_rm.pdf

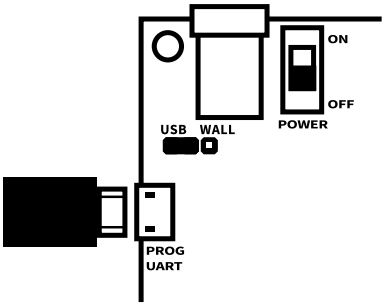
STROOM, JUMPERS, SCHAKELAARS EN KNOPPEN

Deze foto van bovenaf toont de belangrijkste jumperposities op het Nexys4-bord:



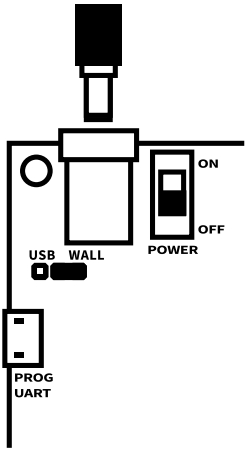
De Nexys4-borden kunnen op twee manieren van stroom worden voorzien: via een externe voeding of via een standaard USB-poort.

Micro-USB-voeding



Sluit uw micro-USB-kabel aan op een USB-poort van een USB-lader of pc om stroom te leveren. Sluit het andere uiteinde aan op de micro-USB-connector van de Nexys4. Plaats de JP3-jumper op pennen 1 en 2 om USB-voeding te selecteren. Gebruik de schakelaar om de Nexys4 in te schakelen.

Externe voeding



De MEGA65-core kan veel stroom verbruiken en een standaard USB-poort kan in potentie te weinig vermogen leveren voor het Nexys4-bord. Het schrijven naar de SD-kaart kan met name vastlopen of vreemd gedrag vertonen. U wordt dan aangeraden een voeding van 5V te gebruiken.

Digilent verkoopt een voeding voor het Nexys4-bord en wij raden u aan deze te gebruiken om het risico op schade aan uw Nexys4-bord te vermijden. De gekozen voeding moet een center-positieve stekker met een binnendiameter van 2.1 mm hebben en 4.5VDC tot 5.5VDC leveren met een stroomsterkte van ten minste 1 ampère.

Sluit de voedingskabel aan op de voedingsaansluiting van de Nexys4. Plaats de JP3-jumper op pennen 2 en 3 om netvoeding te selecteren. Gebruik de schakelaar om de Nexys4 in te schakelen.

Overige jumpers en schakelaars

Voor uw eerste installatie raden wij u aan de volgende jumpers op uw Nexys4-bord in te stellen:

- JP1 - USB/SD
- JP2 - SD

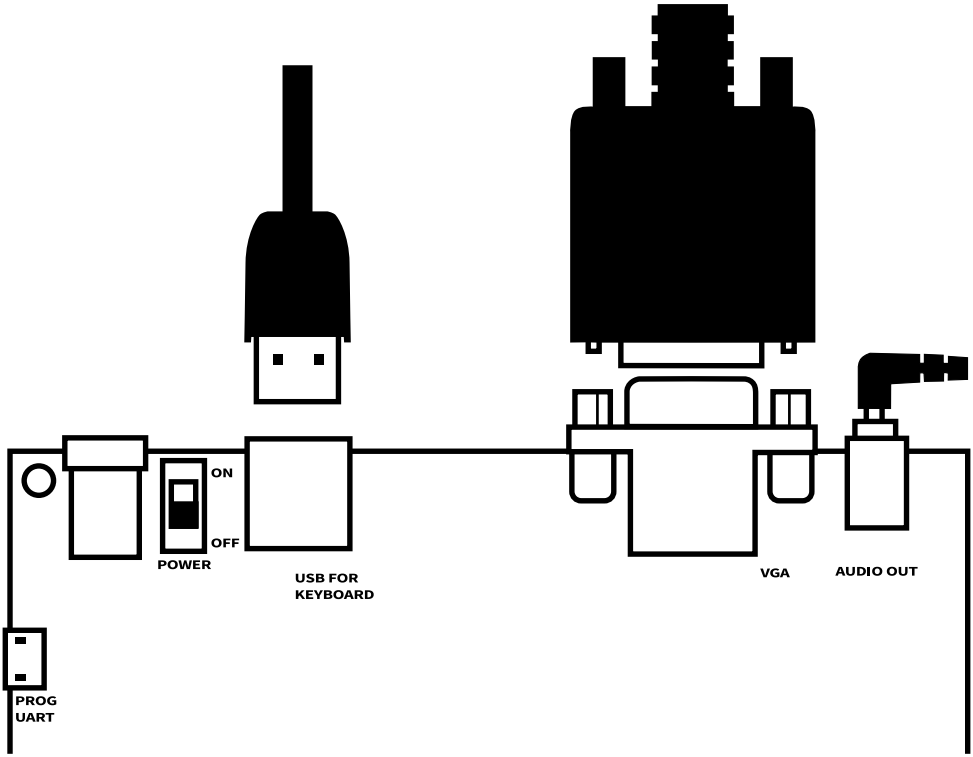


Dit zorgt ervoor dat de bitstream-bestanden bij het opstarten vanaf uw SD-kaart geladen worden.

Op een later moment kunt u ervoor kiezen de bitstream vanaf de ingebouwde QSPI-flash te laden. In dat geval kunt u de instelling van de JP1-jumper aanpassen naar de QSPI-positie.

Alle 16 schakelaars aan de onderkant van het bord moeten op de uit-positie staan.

Aansluitingen en randapparatuur



Aan de USB-poort kan een USB-toetsenbord worden aangesloten. Alleen een toetsenbord zonder USB-hub werkt met het Nexys4-bord. Over het algemeen werken zeer goedkope toetsenborden, terwijl duurdere toetsenborden vaak een USB-hub bevatten en niet werken. Het zou kunnen voorkomen dat u meerdere toetsenborden moet proberen voordat u er een vindt die werkt.

Aan de VGA-poort kan een VGA-beeldscherm worden aangesloten.

De mono-audio-uitgang kan worden aangesloten op de line-in van een versterker.

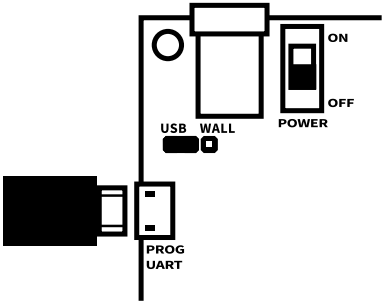
Communicatie met uw pc

Er kunnen situaties zijn waarin u vanaf uw pc met het Nexys4-bord wilt communiceren, bijvoorbeeld om de volgende activiteiten uit te voeren:

- Het QSPI-flashgeheugen flashen via Vivado
- Bitstream-bestanden rechtstreeks vanaf uw pc uploaden (via het m65-hulpprogramma)

- Hulpprogramma's gebruiken zoals M65Connect, m65, mega65_ftp, m65dbg, enz.

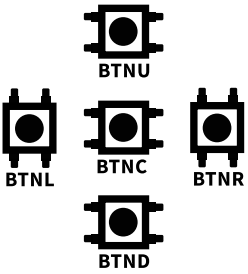
In zulke gevallen moet u de micro-USB-kabel op uw pc aansluiten.



Knoppen op het bord



De "CPU RESET"-toets reset de MEGA65 wanneer u deze indrukt, terwijl de "PROG"-toets de FPGA zelf de MEGA65-core laat herladen. Het belangrijkste verschil tussen de twee is dat CPU RESET sneller is en de inhoud van het geheugen niet wist, terwijl de FPGA-knop langzamer is en de inhoud van het geheugen wel reset.



Twee van de vijf toetsen in de kruisopstelling kunnen eveneens worden gebruikt: BTND gedraagt zich alsof u RESTORE heeft ingedrukt, terwijl BTNC een interrupt activeert, alsof de IRQ-lijn naar massa is getrokken.

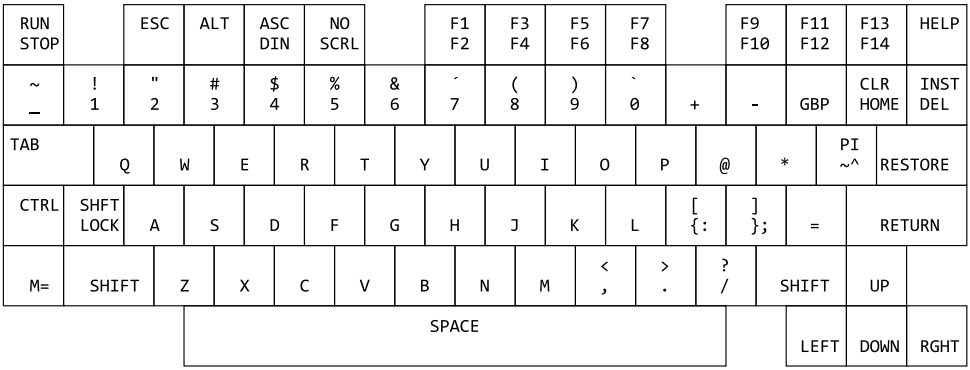
TOETSENBORD

De toetsenbordindeling is positioneel en niet logisch. Dit betekent dat toetsen die zich in dezelfde positie bevinden als de toetsen op een C65-toetsenbord een vergelijkbare

functie hebben. Deze relatie gaat ervan uit dat uw USB-toetsenbord een Amerikaanse toetsenbordindeling gebruikt.

Om u te helpen bij het vinden van de toetstoewijzingen van de diverse MEGA65-toetsen, beschikt de MEGA65 over een ingebouwde virtuele toetsenbordtest. Deze is op twee manieren te bereiken.

De eenvoudigste manier is om ALT ingedrukt te houden terwijl u de Nexys4 inschakelt, of de Nexys4 reset met de "PROG"-toets. Het configuratiemenu wordt weergegeven en door op 3 te drukken, verschijnt het virtuele toetsenbord op een zwarte achtergrond.



Wanneer u een toets op het USB-toetsenbord indrukt, wordt de betreffende toets op het virtuele toetsenbord gemarkeerd weergegeven om u te helpen bij het herkennen van de toetstoewijzing.

De andere manier om het virtuele toetsenbord te bereiken is vanuit de MEGA65 zelf. Houd M ingedrukt en druk op TAB om de Matrix Mode Debugger te openen. Voer vervolgens het volgende in:

```
s ffd3615 ff
```

Dit opent een halfdoorzichtig virtueel toetsenbord aan de bovenkant van het scherm. Als alternatief:

```
s ffd3615 ff ff
```

Dit opent een halfdoorzichtig virtueel toetsenbord in het midden van het scherm.

Houd M ingedrukt en druk op TAB om de Matrix Mode Debugger te verlaten en terug te keren naar de MEGA65.

Enkele toetstoewijzingen met een USB-toetsenbord

RESTORE is toegewezen aan de Page Up-toets.

RUN STOP is toegewezen aan ESC.

MICROSDHC-KAART VOORBEREIDEN

De MEGA65 vereist een SDHC-kaart met een capaciteit van 4 GB tot 64 GB. Sommige SDXC-kaarten kunnen werken, maar dit wordt niet officieel ondersteund.

De voorbereidingsstappen voor de SD-kaart van het Nexys4-bord komen grotendeels overeen met de stappen die nodig zijn voor echte MEGA65-hardware. Het is dan ook de moeite waard om een kijkje te nemen in *het grote Mega65-boek*, Configureren van uw MEGA65 (paragraaf 5) als u ooit meer details nodig heeft.

In dit onderdeel bespreken we de specifieke stappen uitgebreider en beperken we ons tot een beknopte behandeling van de gemeenschappelijke stappen.

Een verschil tussen het Nexys-bord en de echte MEGA65-hardware is dat laatstgenoemde standaard een bitstream/core bevat waarmee u uw SD-kaart in de specifieke indeling kunt formateren die door de MEGA65 wordt vereist.

Voor eigenaren van het Nexys4-bord is er echter geen standaard bitstream; raadpleeg daarom het **grote Mega65-boek** voor meer informatie over waar u de juiste bestanden "nexys4.bit of "nexys4ddr-widget.bit voor uw apparaat kunt downloaden.

Voorbereidingsstappen

De stappen zijn:

- Formateer de SD-kaart op een handige computer met het FAT32-bestandssysteem. De MEGA65- en Nexys4-borden ondersteunen geen andere bestandssystemen, met name niet het exFAT-bestandssysteem.
- Kopieer uw bitstream-bestand (met een bestandsnaam die eindigt op ".bit") naar de SD-kaart.
- Plaats de SD-kaart in de SD-kaartsleuf aan de onderkant van het Nexys4-bord.
- Schakel het Nexys4-bord in.
- Open het hulpprogrammamenu door ALT ingedrukt te houden op het USB-toetsenbord dat op het Nexys4-bord is aangesloten wanneer het MEGA65-startscherm wordt weergegeven.
- Open het FDISK/FORMAT-hulpprogramma door op 2 te drukken wanneer de optie verschijnt op het MEGA65-opstartscherm.
- Volg de aanwijzingen in het FDISK/FORMAT-programma om de SD-kaart opnieuw te formatteren voor gebruik door de MEGA65.

  Het FDISK-hulpprogramma maakt twee partities op uw SD-kaart aan en formatteert deze.

  - De ene is type $41 = MEGA65-systeempartitie, waar de opslagslots, configuratiegegevens en andere bestanden staan.
    (Deze partitie is onzichtbaar op bijvoorbeeld Windows-pc's).
  - De andere partitie met type $0C = VFAT32, waar KERNAL, ondersteuningsbestanden, spellen enzovoort later naartoe worden gekopieerd.
    (Deze partitie is zichtbaar op bijvoorbeeld Windows-pc's).
- Zodra het formatteren is voltooid, schakelt u het Nexys4-bord uit en verwijdert u de microSDHC-kaart uit het Nexys-bord en plaatst u deze terug in uw pc
- Kopieer deze keer de volgende items naar de SD-kaart:
  - Het bitstream-bestand
  - De uitgepakte bestanden uit het "**SD essentials.rar**"-of het "**SD essentialsNoROM.rar**"-bestand dat u van de MEGA65-filehost hebt gedownload.
  - Als u uw eigen voorkeurs-ROM-bestand hebt (bijvoorbeeld "**911001.BIN**"), kopieer dit ook naar de SD-kaart en hernoem het naar "**MEGA65.ROM**" (hoofdletters zijn verplicht).
  - Eventuele .D81-schijfimage-bestanden waarvan u gebruik wilt maken.
    * Let op: wanneer een bestand met de naam MEGA65.D81 aan de SD-kaart wordt toegevoegd, wordt dit automatisch gemonteerd bij het opstarten.
    * Zorg ervoor dat alle .D81-bestanden namen hebben die voldoen aan de oude DOS 8.3-tekenlimiet en in hoofdletters zijn geschreven. Deze beperking wordt in een toekomstige versie verwijderd.
- Verwijder de SD-kaart en plaats deze terug in uw Nexys4-bord.
- Schakel het Nexys4-bord weer in. De MEGA65 zou binnen 15 seconden moeten opstarten.
- "Bij de eerste opstart bevindt u zich op het onboarding-scherm, waarvan meer details te vinden zijn in *het grote Mega65-boek*, Configureren van uw MEGA65 (paragraaf 5) .

Gefeliciteerd. Uw MEGA65 is ingesteld en gereed voor gebruik.

Houd er rekening mee dat bij de bovenstaande methode het bitstream-bestand naar de SD-kaart wordt gekopieerd, waardoor de Nexys-FPGA de bitstream bij elke opstartcyclus laadt; het opstarten duurt dan ongeveer 13 seconden. De bitstream kunt u ook met Vivado-software naar het QSPI-flashgeheugen flitsen; de opstarttijd verkort u daarmee tot 0.3 seconden.

Voor meer gedetailleerde informatie over het voorbereiden en configureren van uw MEGA65 verwijzen wij u naar *het grote Mega65-boek*, Configureren van uw MEGA65 (paragraaf 5) hoofdstuk.

DE BITSTREAM LADEN VANUIT QSPI

Hoewel het laden van de bitstream vanaf de SD-kaart de aanbevolen (en veelbewandelde) route is die dit document heeft gekozen, hebben de laatste tijd meer Nexys4-gebruikers de alternatieve weg verkend om de bitstream vanuit het QSPI-flashgeheugen te laden. Enkele mogelijke redenen hiervoor zijn:

- Snellere opstarttijden (0,3 seconden versus 13 seconden)
- Sommige gebruikers waren geïnteresseerd in de mogelijkheid om meerdere cores op hun QSPI te flitsen (via stappen die beschreven staan in de *het grote Mega65-boek*, Hoe u een MEGA65 kunt upgraden (paragraaf 6) Hoofdstuk)
- Sommige gebruikers hebben last gehad van hardnekkige problemen met de route via de SD-kaart, zoals:
  - Het systeem kan niet opnieuw opstarten vanaf het onboarding-scherm
  - Het systeem kan niet opnieuw opstarten vanuit het freeze-menu na het wisselen tussen PAL/NTSC

Indien dit in de toekomst een populairdere route blijkt te zijn, kunnen wij onze documentatie hierop aanpassen. Hier volgen enkele stappen in het kort.

Voorbereidingsstappen

Als u deze route wilt proberen, moet u de JP1-jumperinstelling aanpassen om QSPI te gebruiken en vervolgens de stappen volgen in het hoofdstuk Het flashen van de FPGA's en CPLD's in de MEGA65 met betrekking tot Vivado installeren en De hoofd-FPGA flashen met Vivado.

Houd er rekening mee dat de installatie van Vivado een langdurig proces is (zowel wat betreft de downloadtijd als de installatietijd).

Nadat u Slot0 van uw QSPI-chip hebt geflitst via Vivado, kunt u de stappen volgen die beschreven staan in *het grote Mega65-boek*, Configureren van uw MEGA65 (paragraaf 5) voor de aangepaste SD-kaartformattering, het installeren van ROM- en ondersteuningsbestanden en de onboarding.

WIDGET-BORD

Als u een Nexys-bord bezit (alle modellen), vindt u het wellicht interessant om een widget-bord aan uw Nexys-apparaat toe te voegen, zodat u verbinding kunt maken met:

- een originele C64- of C65-toetsenbord
- 2 DB-9-joysticks

- Paddles of een 1351-muis
- Cartridges (werken nog niet)

Het widget-bord wordt via de PMOD-connectoren met uw Nexys-bord verbonden.

Momenteel is het alleen verkrijgbaar als onbewerkte printplaat (zonder componenten), hier aan te schaffen:

- https://oshpark.com/shared_projects/Y37xg9N7

De firmware, de printplaatdiagrammen, het schema en de stuklijst zijn in het volgende GitHub-project te vinden:

- https://github.com/sy2002/DM65PIC

U moet de onderdelen uit de stuklijst apart aanschaffen en de componenten zelf op het bord monteren.

Dit forum64.de-thread kan voor u van belang zijn wanneer u de ervaringen van anderen die dit proces hebben doorlopen wilt lezen of wanneer u zelf vragen heeft:

- https://www.forum64.de/index.php?thread/90465-mega-65-ports-add-on-card-s

De printplaatbestanden (c65Keyb.brd) en schematische bestanden (c65Keyb.sch) bevinden zich in de map "eagle/" van het DM65PIC-project en kunnen worden bekeken met het gratis programma AutoDesk Eagle, hier beschikbaar:

- https://www.autodesk.com/products/eagle/free-download

Hier volgen enkele foto's van het widget-bord in gebruik:






MEGA65

Voor het gemak zijn de printplaat- en schema's van het widget-bord eveneens opgenomen in Hoofdstuk/Bijlage C op bladzijde 319.

Enkele aanvullende achtergrondnotities over het bord:

- Het widget-bord is oorspronkelijk bedoeld om in een C65-behuizing te passen
- De uitbreidingspoort is niet bruikbaar en er moet een stuk worden weggezaagd voor de C64-behuizing
- Het zou redelijk zijn het bord aan te passen voor gebruik in een C64-behuizing (wat zeker het meest voorkomende gebruik zal zijn)

PMOD-NAAR-JOYSTICK-ADAPTER

Als alternatief (en goedkoper) voor wie alleen een DB9-joystick-poort via de PMOD-connectoren wil toevoegen, heeft een community-lid, TheChief, een manier bedacht om dat te doen. Meer informatie is te vinden in het volgende Discord-thread:

- https://discord.com/channels/719326990221574164/903079038015389716/929369283119685643



Joystick-actie	PMOD-JA-aansluitingen	DB9-aansluitingen
Vuur	jahi(9)	pin 6
Omhoog	jalo(1)	pin 1
Links	jalo(2)	pin 3
Omlaag	jahi(7)	pin 2
Rechts	jahi(8)	pin 4

HANDIGE TIPS

Hier volgen enkele handige tips om vertrouwd te raken met de MEGA65:

- Druk op & houd  (of de Commodore-toets als u een Commodore 64- of 65-toetsenbord gebruikt) ingedrukt tijdens het opstarten om in C64-modus op te starten in plaats van C65-modus.
- Druk op & houd  ingedrukt tijdens het opstarten om de machine-taalmonitor te openen in plaats van BASIC te starten.
- Druk op  en houd deze ongeveer een halve tot een seconde ingedrukt om het MEGA65 Freeze-menu te openen. Dit menu biedt handige hulpmiddelen om de processorsnelheid te wijzigen, te wisselen tussen PAL & NTSC-videomodus, audio-instellingen aan te passen, freeze-states te beheren en D81-schijfimages te selecteren. U kunt bovendien het geheugen van het bevroren programma inspecteren en bewerken, en nog veel meer. Dit menu is in vele opzichten het hart van de MEGA65 en is het ontdekken zeker waard.
- Typ `POKE0,65` in C64-modus om de processor op volle snelheid (40 MHz) te zetten. Sommige software gedraagt zich in deze modus onjuist, terwijl andere software juist sneller werkt.
- Typ `POKE0,64` in C64-modus om de processor op 1 MHz te zetten.
- Typ `SYS58552` in C64-modus om naar C65-modus te schakelen.
- Typ `GO64` in C65-modus en bevestig door `Y` in te drukken om naar C64-modus te schakelen, hetzelfde als op een C128.
- De C65-ROM maakt apparaat 8 tot standaard, zodat u het **,8** aan het einde van LOAD- en SAVE-commando's normaal gesproken kunt weglaten.
- Het indrukken van  +  in de C64- of C65-modus probeert van schijf op te starten.

Veel plezier! Veel plezier! Er is jarenlang met liefde aan de MEGA65 gewerkt voor uw plezier. Wij hopen dat u er net zoveel plezier aan beleeft als wij bij het maken hebben gehad!

Het MEGA Museum of Electronic Games & Art verwelkomt uw feedback, suggesties en bijdragen aan dit open-source-project voor het bewaren van digitaal erfgoed.

DEEL VIII

BIJLAGEN

BIJLAGE A

Het flashen van de FPGA's en CPLD's in de MEGA65

- Aanbevolen specificaties voor de pc
- Waarschuwing
- Vivado installeren
- De FTDI-stuurprogramma's installeren
- De hoofd-FPGA flashen met Vivado
- De CPLD in het toetsenbord van de MEGA65 flashen met Lattice Diamond
- Het flashen van de MAX10-FPGA op het MEGA65-moederbord met INTEL QUARTUS

De MEGA65 is een computer met open-source hardware en open-source ontwerp. Dit betekent dat u niet alleen programma's kunt schrijven die op de MEGA65 als een voltooide computer draaien, maar ook de programmeerbare chips in de MEGA65 kunt gebruiken om er allerlei andere dingen van te maken.

Als u alleen een upgradecore voor de MEGA65 wilt installeren, of een core waarmee u uw MEGA65 als een ander type computer kunt gebruiken, kijkt u beter in *het grote Mega65-boek*, Hoe u een MEGA65 kunt upgraden (paragraaf 6).

Dit hoofdstuk is vooral bedoeld voor mensen die willen helpen bij het ontwikkelen van cores voor de MEGA65. Het kan ook interessant zijn voor eigenaren van Nexys4-borden die hun apparaten vanaf het QSPI-flashgeheugen op het bord willen opstarten (in plaats van een bitstreambestand vanaf de SD-kaart). Hiervoor moet een .mcs-bestand op de QSPI-chip van het bord geflashed worden om een initiële bitstream in de positie "Slot 0" te plaatsen.

Deze programmeerbare chips heten Field Programmable Gate Arrays (FPGA's) of Complex Programmable Logic Devices (CPLD's) en kunnen een breed scala aan circuits implementeren. Ze worden doorgaans geprogrammeerd met een taal zoals VHDL of Verilog. Dit zijn talen die de meeste mensen niet vaak tegenkomen. Ze zijn ook op een aantal punten heel anders dan "normale" programmeertalen en het kan even duren voordat u begrijpt hoe ze werken. Met enige inzet en volharding kunt u echter spannende dingen ermee creëren.

AANBEVOLEN SPECIFICATIES VOOR DE PC

Reken erop dat u vele gigabytes aan software moet installeren op een Linux- of Windows-pc voordat u programma's kunt schrijven voor de FPGA's en CPLD's in de MEGA65. Het "compileren" van complexe ontwerpen kan ook enkele uren duren, afhankelijk van de snelheid en het geheugen van uw computer. Wij raden een computer met ten minste 12 GB RAM (bij voorkeur 16 GB) aan als u programma's voor FPGA's en CPLD's wilt schrijven. Als u daarentegen alleen programma's op de FPGA's en CPLD's van uw MEGA65 wilt laden die door anderen zijn geschreven, volstaan de meeste computers met een recente versie van Windows of Linux.

- Besturingssysteem: Linux of Windows
- Snelheid van de processor: zo snel als u kunt vinden!
- Aantal cores: bij voorkeur 8 of meer, omdat de gratis licentie van Vivado maximaal 8 cores kan gebruiken.
- Harde schijf: ongeveer 70 GB of meer. De exacte hoeveelheid hangt af van hoeveel onderdelen van Vivado u installeert (hou er rekening mee dat het volledige installatiebestand zelf ongeveer 50 GB is)

- Geheugen: minimaal 12 GB (bij voorkeur meer, voor de zekerheid)

WAARSCHUWING

Voordat we verdergaan, moeten we u waarschuwen voor het opnieuw programmeren van de FPGA's en CPLD's in de MEGA65. Het opnieuw programmeren van de FPGA van de MEGA65 kan schade veroorzaken of uw MEGA65 in een onbereikbare staat achterlaten waarvan het moeilijk is te herstellen, oftewel "gebrickt". Als u ervoor kiest om uw MEGA65 te openen en een van de FPGA's erin opnieuw te programmeren, kunnen wij de correcte werking van het apparaat niet langer garanderen. Wij kunnen de fabrieksgarantie van het apparaat als computer derhalve niet meer redelijkerwijs honoreren. U bent gewaarschuwd!

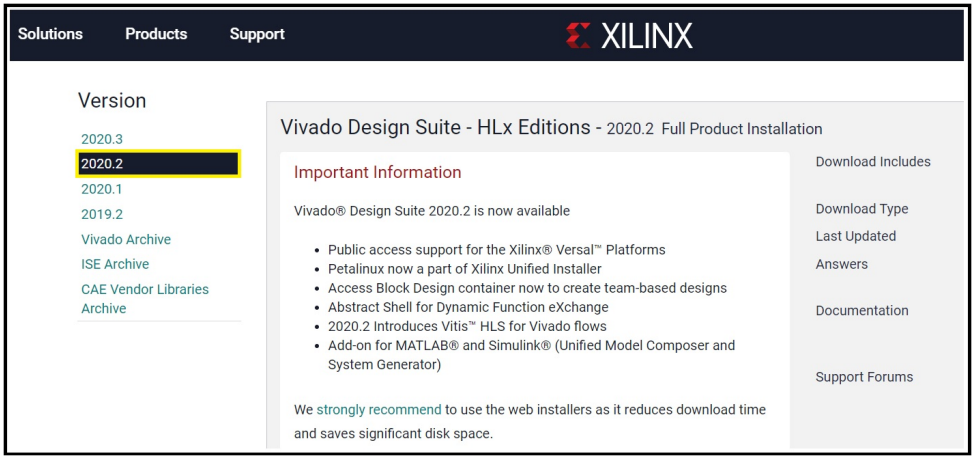
VIVADO INSTALLEREN

U moet Vivado installeren om het QSPI-flashgeheugen in uw doelapparaat te flashen, of het nu een MEGA65 R2/R3/R3A/R4, Nexys4/Nexys4DDR/NexysA7, MEGAphone of ander apparaat is.

Vivado is ook het hulpprogramma dat wordt gebruikt voor het compileren (beter bekend als synthese) van FPGA-bitstreams.

Ga naar https://www.xilinx.com/support/download.html

Selecteer versie 2020.2



Opmerking: sommige gebruikers hebben nog succes met oudere versies, omdat het hoofddoel hier is om een versie te installeren die de FPGA van uw doelhardware ondersteunt.

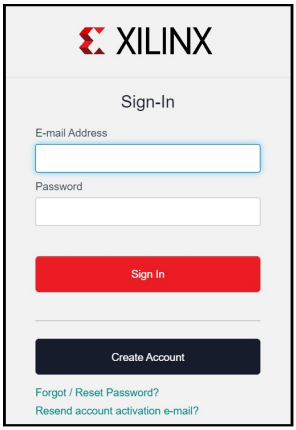
Dat wil zeggen de Artix7 100T (voor Nexys en R2) of 200T (R3/R3A/R4).

Klik op Xilinx Unified Installer 2020.2: Windows Self Extracting Web Installer EXE - 248.44MB

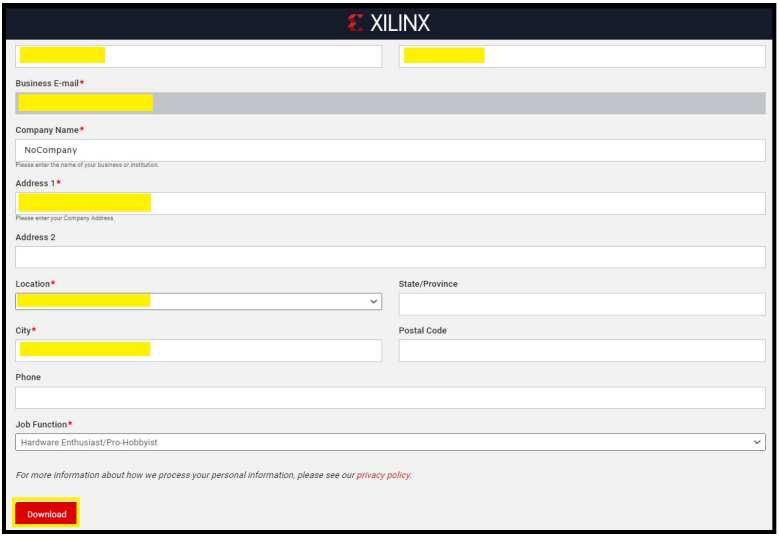


U wordt gevraagd een account aan te maken om in te kunnen loggen en het installatieprogramma te kunnen downloaden.

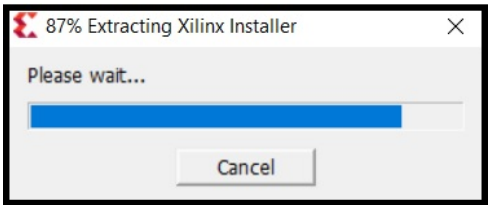
Uw inloggegevens worden ook gevraagd tijdens de installatie.



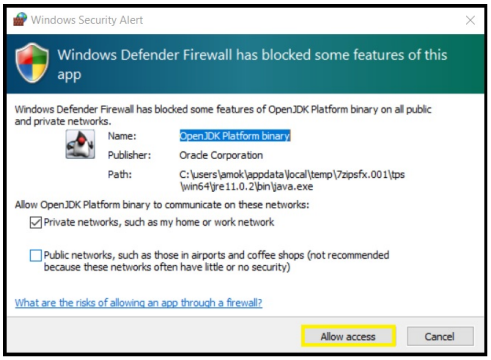
Nadat u ingelogd bent, moet u enkele persoonlijke gegevens invullen en dan op Download klikken.



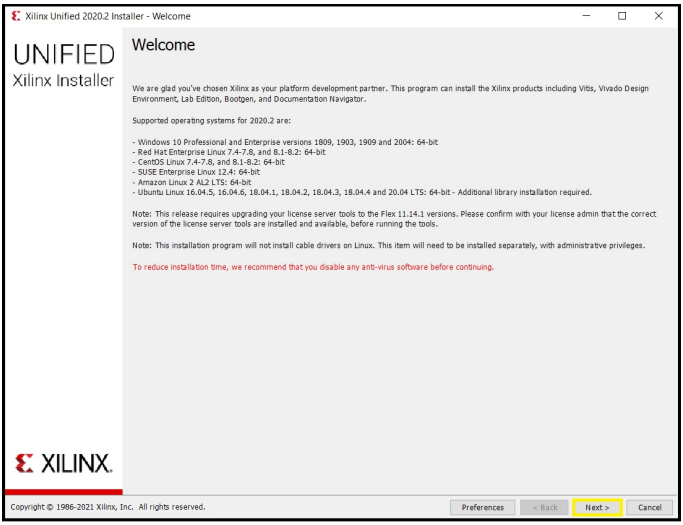
Voer het installatieprogramma uit als beheerder (Xilinx_Unified_2020.2_1118_1232_Win64.exe).



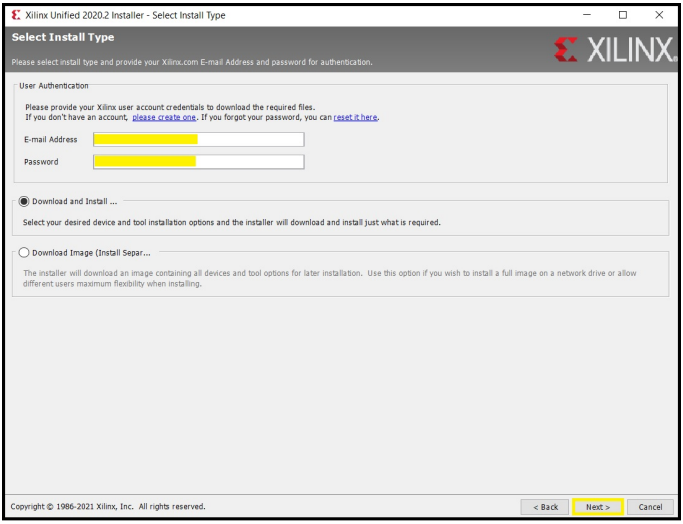
Klik op Allow Access.



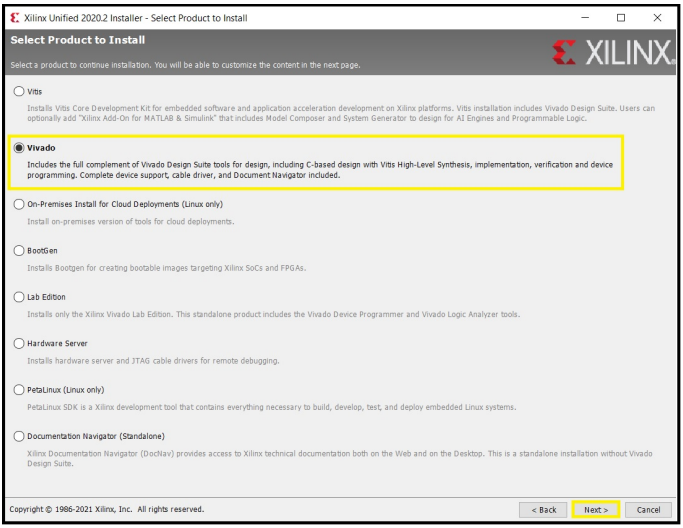
Klik op Next.



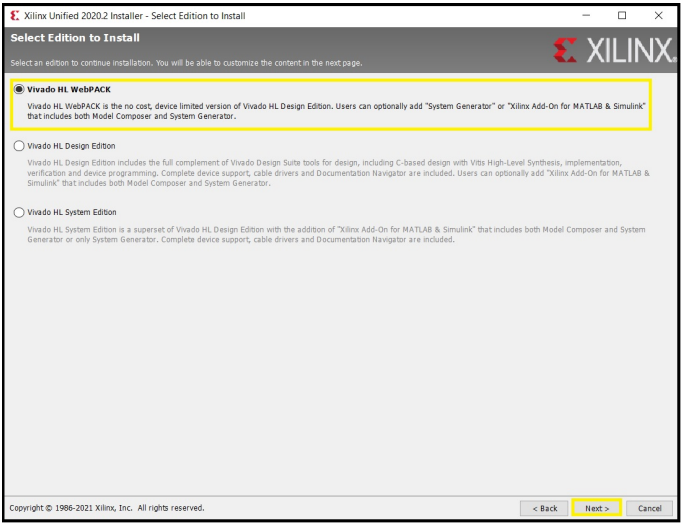
Voer uw inloggegevens in en klik op Next.



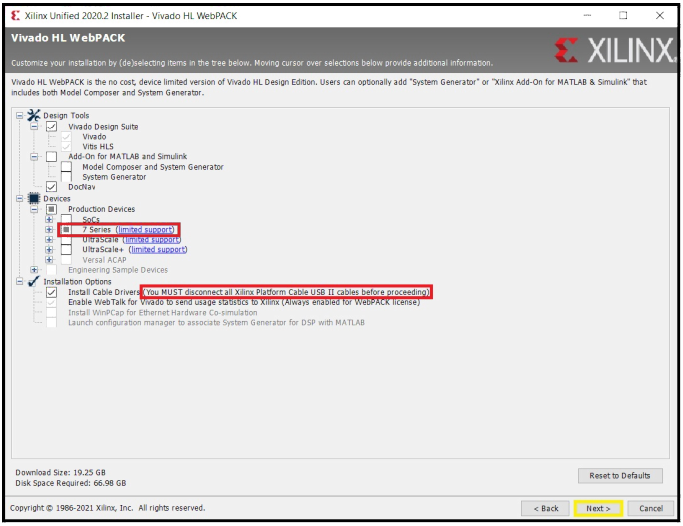
Selecteer Vivado en klik op Next.



Selecteer “Vivado HL WebPACK” en klik op “Next”.

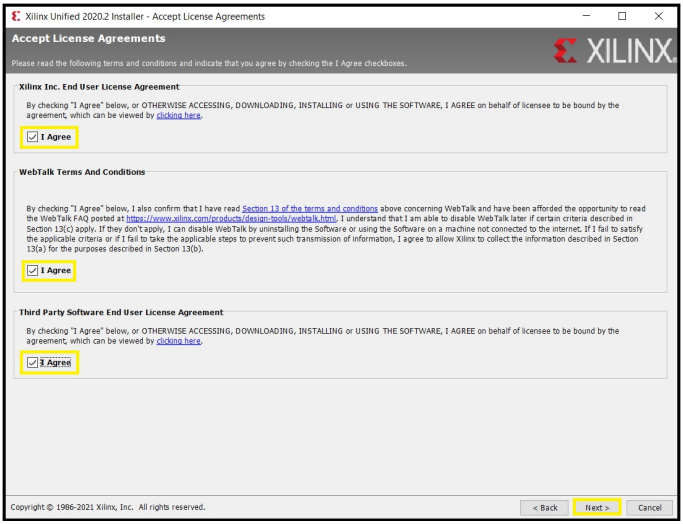


Wij raden aan om alleen de apparaten uit de “**7 Series**” te selecteren, omdat onze gekozen FPGA in deze serie valt, en het deselecteren van de andere series bespaart u ongeveer 6 GB aan downloadgrootte. Klik dan op “Next”.

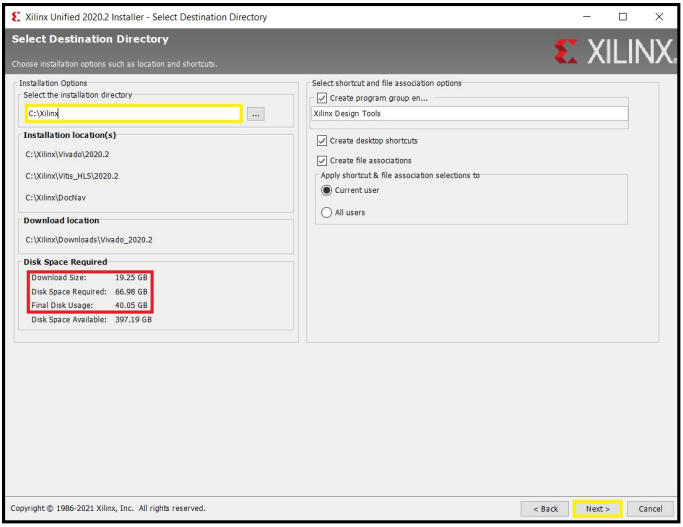


Waarschuwing: zoals gezegd, ontkoppel elke USB-kabel die op uw pc is aangesloten van het Nexys-bord.

Ga akkoord met alle bepalingen van de eindgebruikerslicentieovereenkomst en de voorwaarden en klik op “Next”.

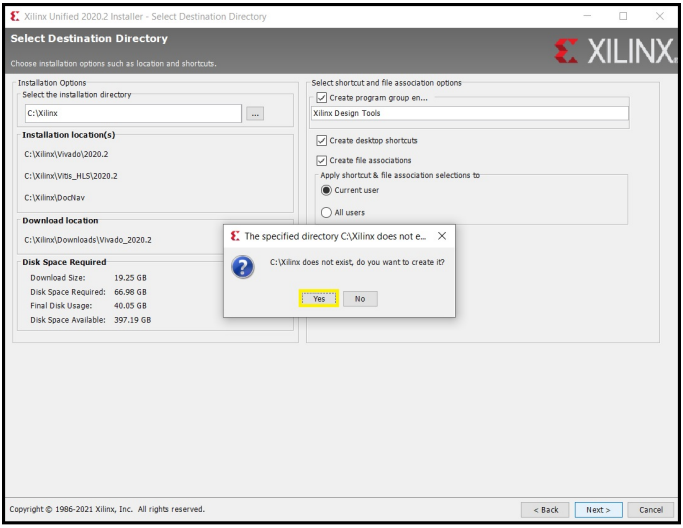


Kies de locatie waar u de software wilt installeren en klik op “Next”.

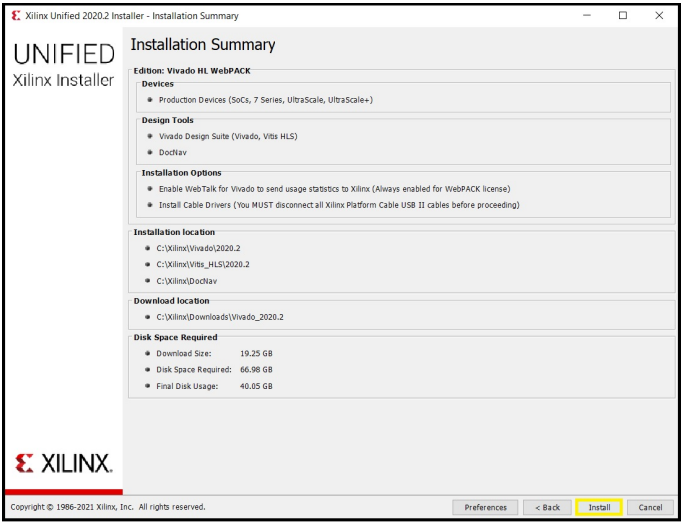


Waarschuwing: u staat op het punt om 20 GB aan software te downloaden en u hebt 70 GB nodig om de installatie uit te voeren.

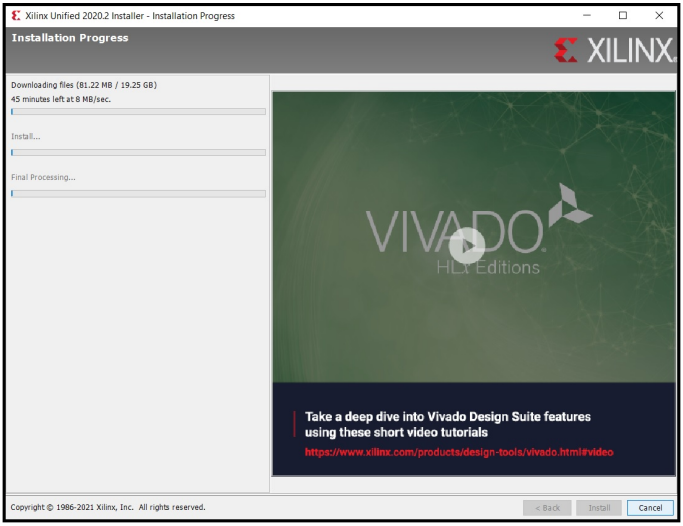
Klik op “Yes”.



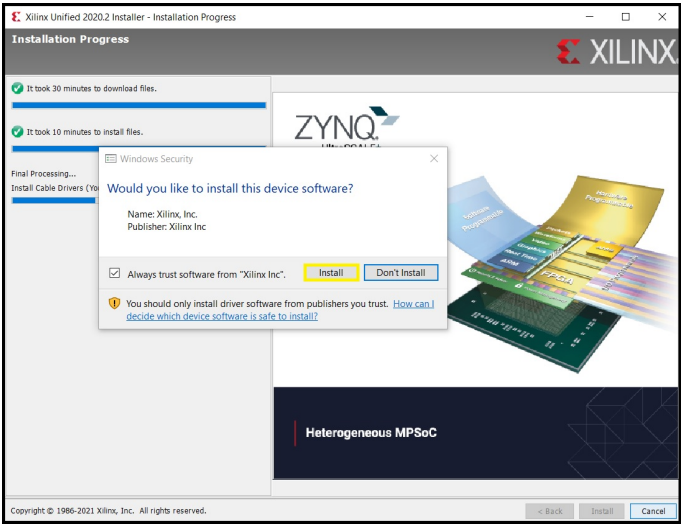
Klik op “Install”.



Wacht tot de installatie is voltooid. Aan het einde van de installatie wordt u gevraagd of u de Xilinx-apparatensoftware wilt installeren.

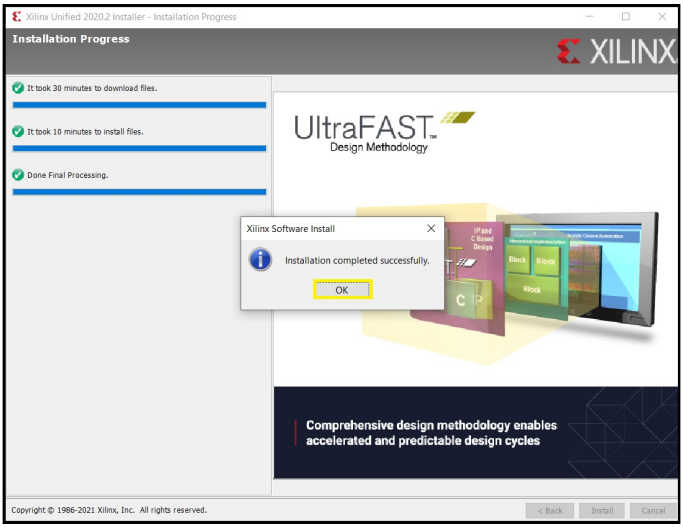


Klik op “Install”.



Laat de installatie voltooien.

De installatie is voltooid. Klik op “OK”.

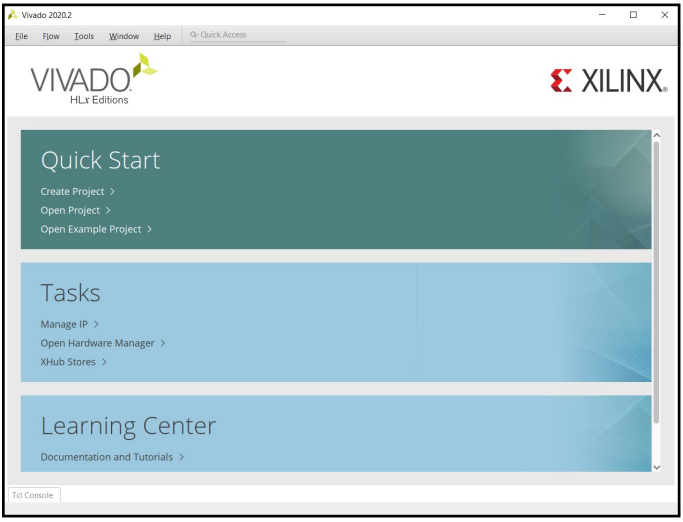


Op uw bureaublad verschijnen nu de volgende pictogrammen:

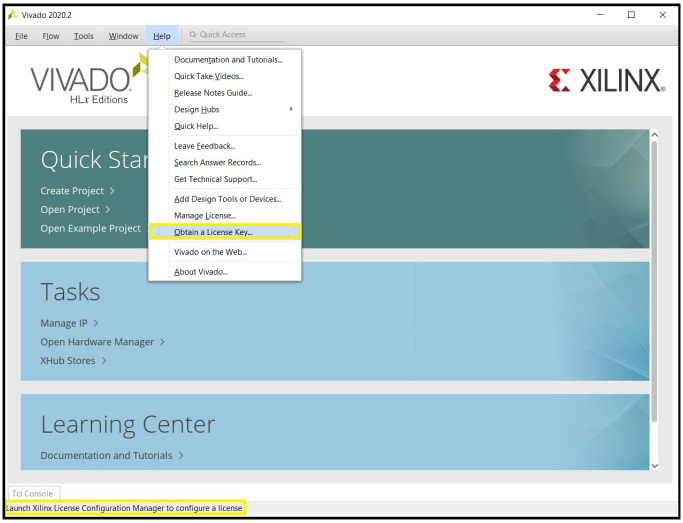


Let op: Ubuntu-gebruikers moeten mogelijk een ontbrekende afhankelijkheid installeren met: `sudo apt install libtinfo5`

Start Vivado 2020.2



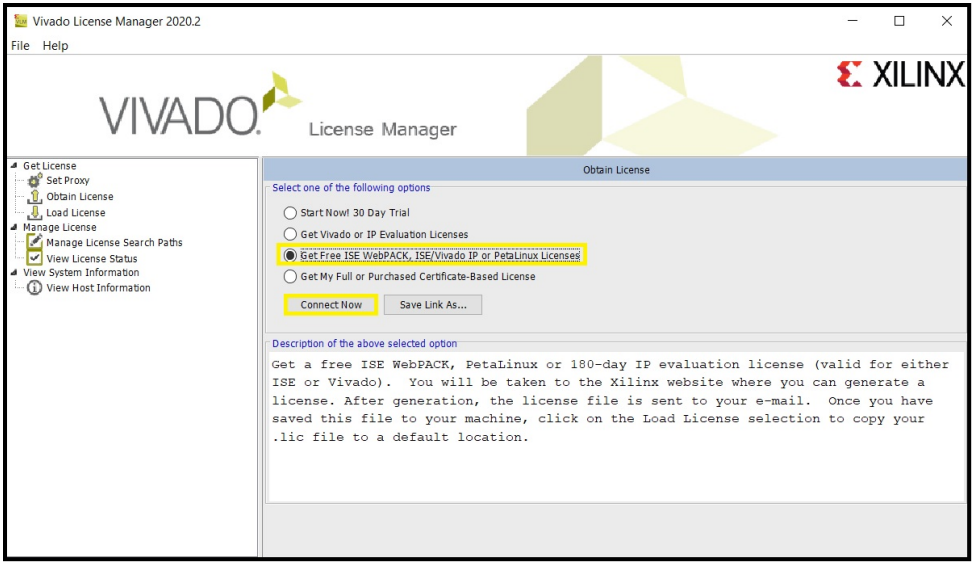
Klik op “Help”->“Obtain a licence Key”.



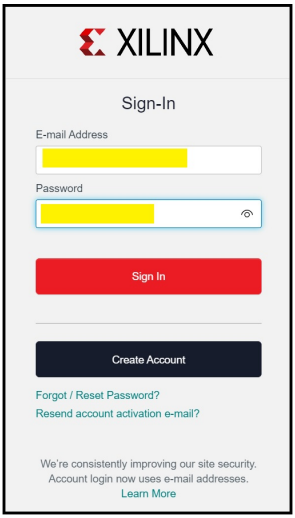
Hiermee opent u de Vivado-licentiemanager.

Selecteer “Get Free ISE WebPACK, ISE/Vivado IP or PetaLinux Licenses”

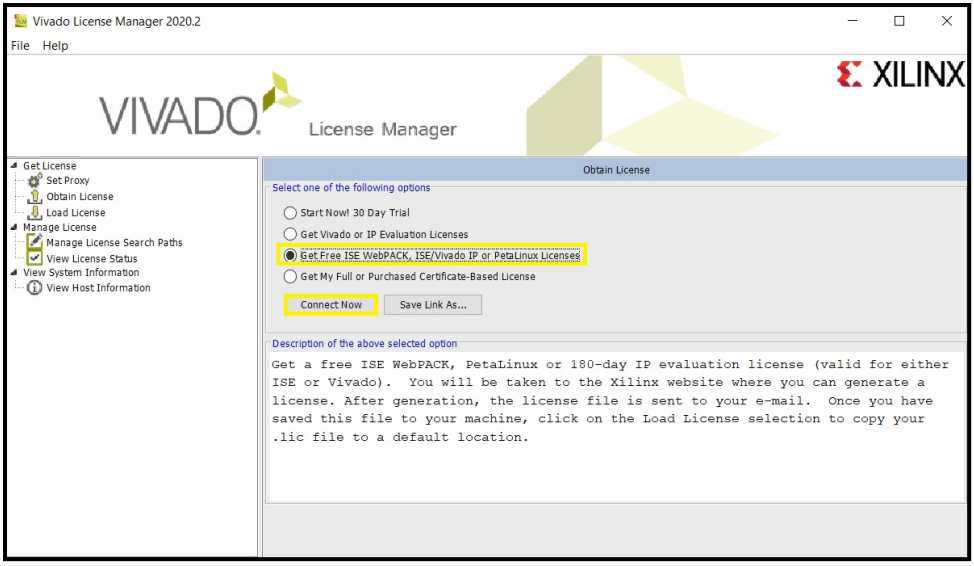
Klik op “Connect Now”.



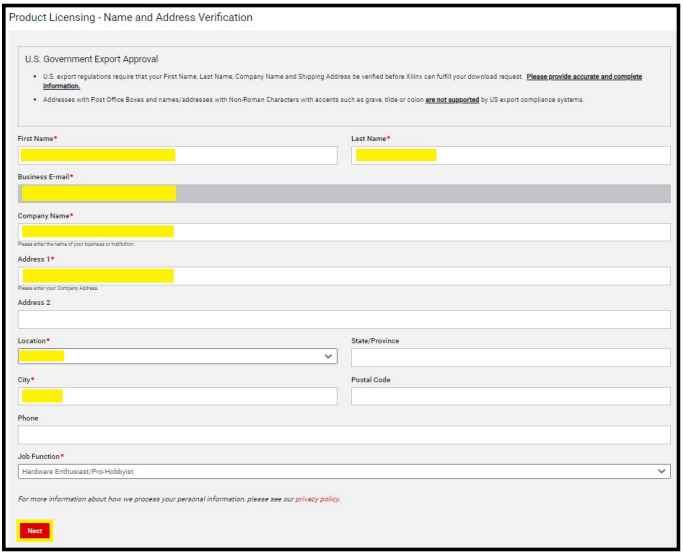
Verbind met het gebruikersaccount dat u hebt aangemaakt om de Vivado-software te kunnen downloaden. Als u nog niet op de website van Xilinx was ingelogd, wordt u naar de hoofdpagina geleid. Keer terug naar de licentiemanager (die niet is gesloten).



Klik opnieuw op “Connect Now” (controleer of “Get Free ISE WebPACK, ISE/Vivado IP or PetaLinux Licenses” nog steeds is geselecteerd).

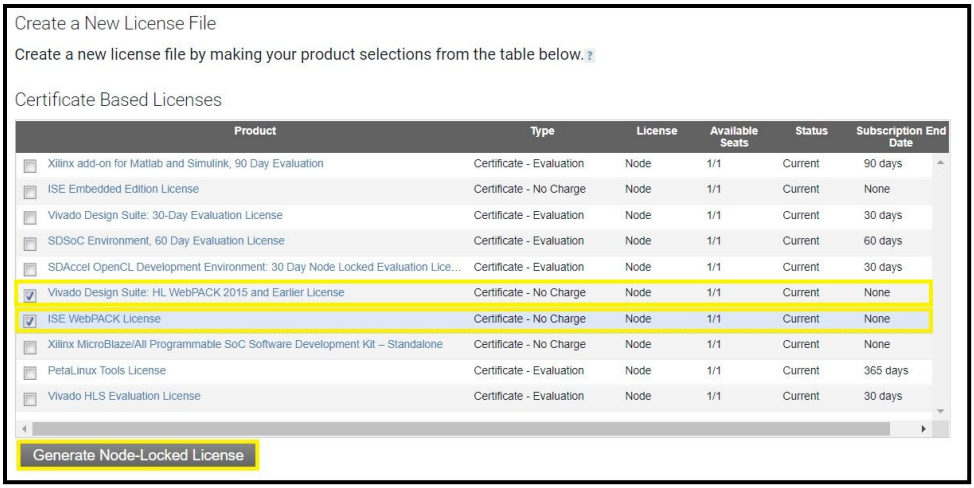


Vervolgens registreert u uw persoonlijke gegevens op de Vivado-website en klikt u op “Next”.

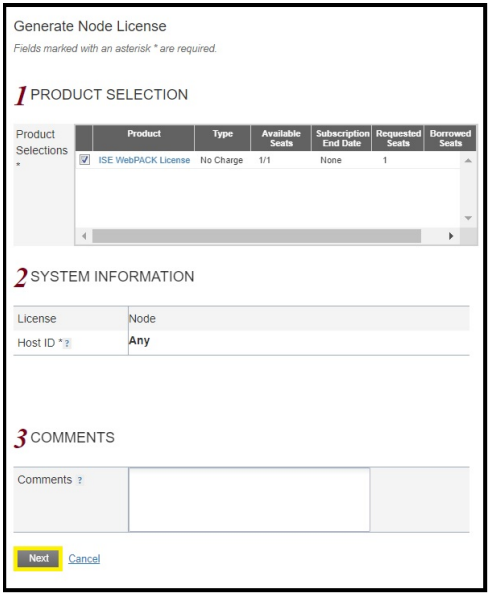


Selecteer “ISE WebPACK Licence” en “Vivado Design Suite: HL WebPACK 2015 and Earlier License”

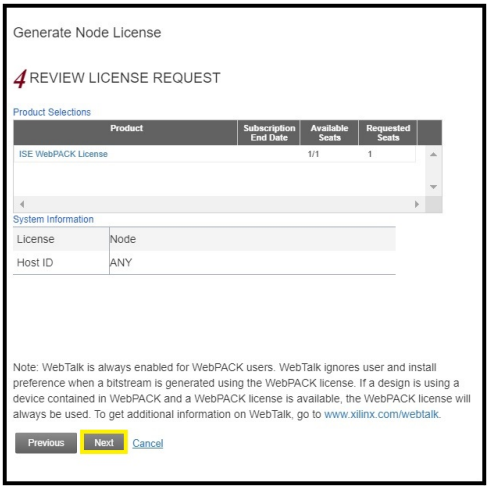
Klik vervolgens op “Generate Node-Locked Licence”.



Klik op “Next”.



Klik op “Next”.



Controleer uw e-mail: u hebt een e-mail ontvangen van Xilinx, Inc. met een licentie-bestand met de naam “Xilinc.lic” als bijlage.

Sla dit bestand op uw computer op en bewaar het op een veilige plaats.

Congratulations

Your new license file has been successfully generated and e-mailed to [redacted]. You can also view the license file under the Manage Licenses tab.

Please add this sender (xilinx.notification@entitlenow.com) to your address book.

License File Details
Node License
Host ID: ANY

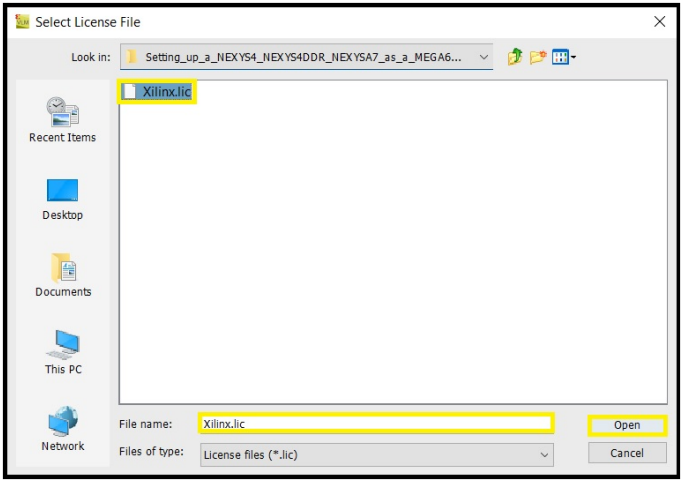
Products
ISE WebPACK License (No Charge): 1 seats

Ga terug naar de licentiemanager (die nog steeds actief is).

Selecteer “Load License” en klik op “Copy License”.



Blader naar de locatie waar u het bestand “Xilinc.lic” heeft opgeslagen, selecteer het en klik op “Open”.



Klik op “OK” en sluit de Vivado-licentiemanager.



Uw Vivado-software is geregistreerd en u kunt deze nu gebruiken.

DE FTDI-STUURPROGRAMMA’S INSTALLEREN

De FTDI-drivers zijn nodig om uw PC te laten communiceren met de JTAG-poort en de seriële communicatiepoort van de hardware (merk op dat de enkele fysieke USB-verbinding met uw PC feitelijk deze twee poorten aanbiedt).

Linux-drivers

Gebruikers van Linux melden dat zij de FTDI-drivers standaard in hun Linux-distributie aangetroffen hebben, terwijl anderen aangeven dat het volgende extra commando na de installatie van Vivado uitgevoerd moet worden:

```
cd /opt/Xilinx/Vivado/2018.3/data/xicom/cable_drivers/lin64/install_script/install_driv
sudo ./install_drivers
```

Windows-drivers

Download het volgende archief om de drivers te installeren:

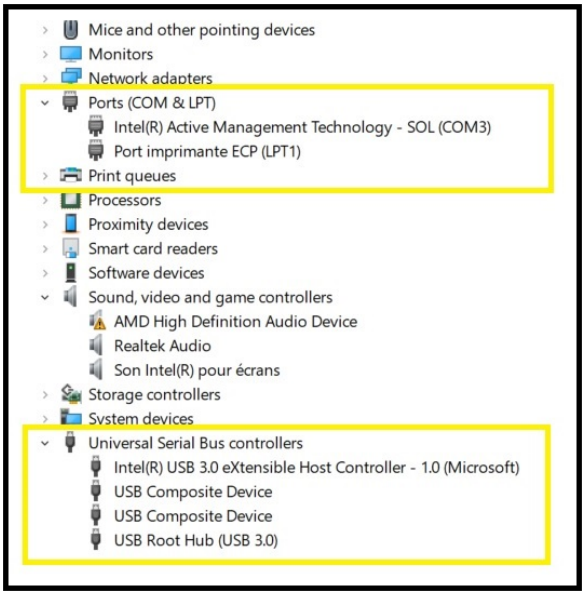
- https://ftdichip.com/wp-content/uploads/2021/08/CDM212364_Setup.zip

Pak het bestand **CDM212364_Setup.zip** uit; u verkrijgt het bestand **CDM212364_Setup.exe**.

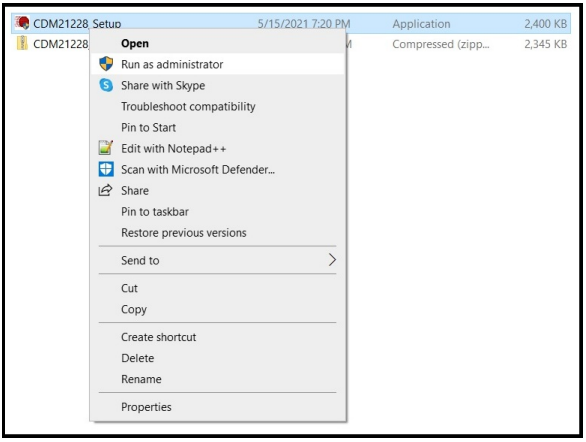
Waarschuwing:

Voordat u de drivers installeert, moet u de Nexys4-plaat uitschakelen en de USB-kabel loskoppelen van de PC.

Controleer welke apparaten er voor de installatie al aanwezig zijn:

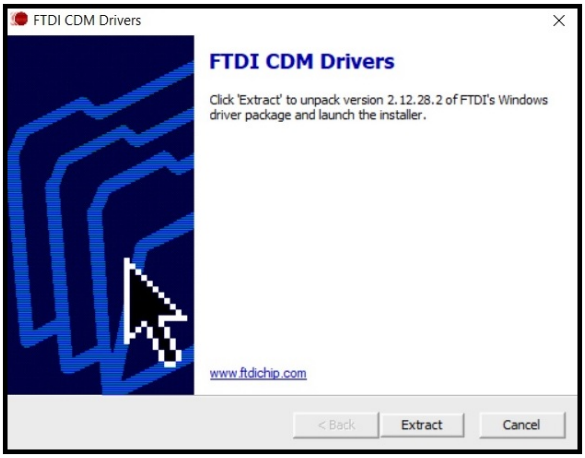


Voer het bestand CDM212364_Setup.exe uit als administrator:

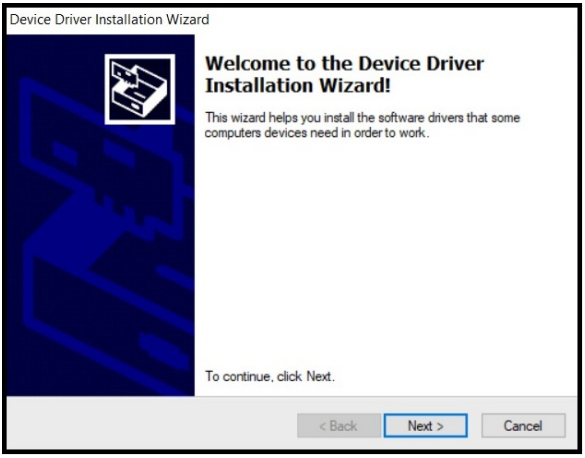


Bevestig dat u het programma wilt uitvoeren.

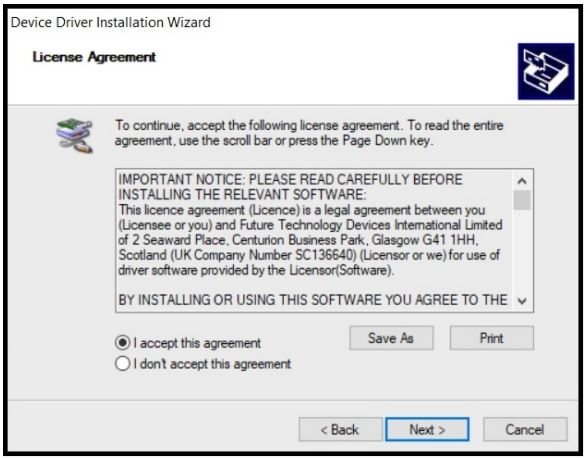
Klik op “Extract”.



Klik op “Next >”.



Aanvaard de overeenkomst en klik op “Next >”.



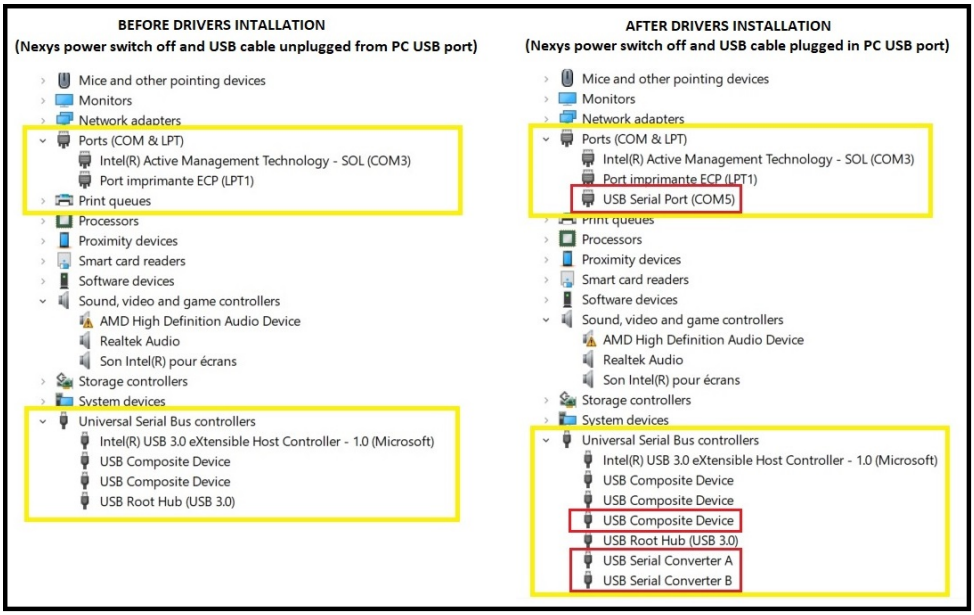
De installatie van de drivers wordt gestart.

Klik op “Finish”.



Sluit de USB-kabel aan op een USB-poort op de PC zonder de Nexys4-plaat in te schakelen.

Het aansluiten van de USB-kabel veroorzaakt het verschijnen van nieuwe apparaten.



- Er is een extra COM-poort geïnstalleerd: dit is de COM-poort die gebruikt wordt om met de Nexys4-plaat te communiceren.
- Er is een extra USB-samengesteld apparaat geïnstalleerd.
- Er zijn twee USB-serieel-omzetters geïnstalleerd.

Op dit moment is de Nexys4-plaat nog niet van stroom voorzien.

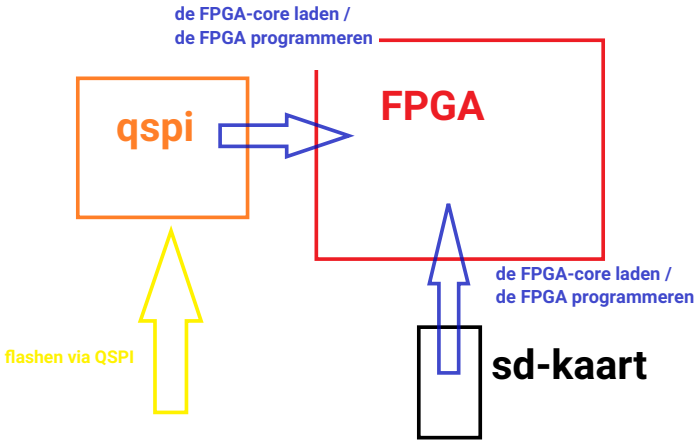
Voor meer informatie over de geïnstalleerde drivers kunt u de bijbehorende documentatie downloaden:

- https://ftdichip.com/wp-content/uploads/2020/08/AN_396-FTDI-Drivers-Installation-Guide-for-Windows-10.pdf
- https://ftdichip.com/wp-content/uploads/2021/01/AN_119_FTDI_Drivers_Installation_Guide_for_Windows7.pdf

DE HOOFD-FPGA FLASHEN MET VIVADO

Ter verduidelijking: wanneer wij spreken over het “flashen van de FPGA”, bedoelen wij in feite het flashen van het QSPI-flashgeheugen waarvan de FPGA bij het opstarten gebruikmaakt om de bitstream snel te laden.

De afbeelding hieronder toont twee veelgebruikte manieren waarop de FPGA bij het opstarten een bitstream kan laden:



- We kunnen eerst een bitstream-/core-bestand op het QSPI-flashgeheugen flashen, waarna de FPGA dit bij het opstarten snel kan laden. Het flashen van het QSPI is tamelijk traag, maar het voordeel van een korte opstarttijd is een duidelijke meerwaarde.
- Gebruikers van een Nexys-plaat kunnen een bitstream-bestand op hun SD-kaart plaatsen en de FPGA dit (enigszins langzamer) bij het opstarten vanaf daar laten laden. Hierdoor kunnen bitstreams snel gewisseld of bijgewerkt worden zonder een TE0790-03 JTAG-programmeermodule. Deze functionaliteit is echter voor later gepland voor de MEGA65.

In dit onderdeel beschrijven wij de werkwijze die gebruikmaakt van de QSPI.

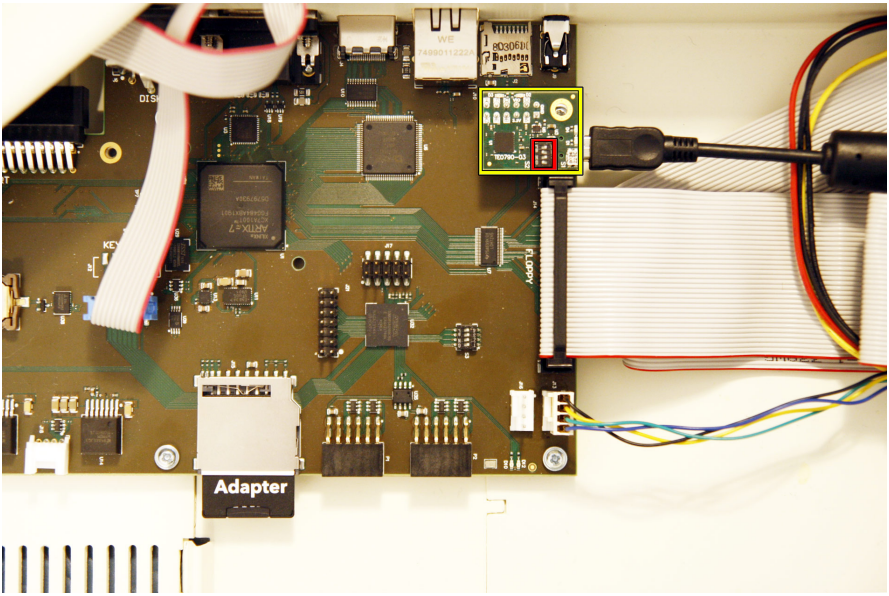
Veel van de volgende stappen in dit onderdeel zijn van toepassing op eigenaren van zowel de MEGA65 R2/R3/R3A/R4 als op eigenaren van een Nexys4-plaat. Er zijn onderweg enkele verschillen waarvan de lezer op de hoogte gesteld zal worden.

Indien u ervoor kiest om verder te gaan, heeft u een functionele installatie van het Vivado-programma van Xilinx nodig en de FTDI-drivers zoals beschreven in de voorgaande onderdelen.

U moet ook een .mcs-bitstreambestand downloaden of zelf bouwen (en een optioneel .prm-controlesomverificatiebestand) dat u via Vivado op de QSPI-chip wilt flashen. Zie het **grote Mega65-boek** voor meer details over waar dergelijke bestanden gedownload kunnen worden.

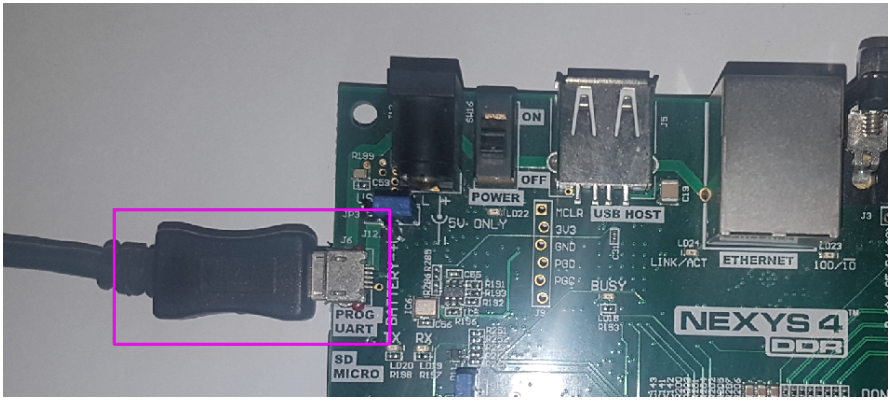
Voor eigenaren van de MEGA65 R2/R3/R3A/R4:

U heeft een TE0790-03 JTAG-programmeermodule nodig. Het is ook noodzakelijk dat dip-schakelaars 1 en 3 op de AAN-stand staan en dip-schakelaars 2 en 4 op de UIT-stand op de TE-0790. Met de MEGA65 losgekoppeld van de voeding moet de TE-0790 op de JB1-connector worden geïnstalleerd, die zich tussen de diskettedatakabel en de audiojack bevindt. De vergulde opening van de TE-0790 moet in lijn liggen met het schroefgat eronder. De mini-USB-kabel verbindt zich vervolgens aan de kant richting het 3.5" diskettestation. De volgende afbeelding toont de juiste positie: de TE0790 is omringd door de gele kast en de dip-schakelaars door de rode kast. Dip-schakelaar 1 is de dichtstbijzijnde bij de diskettedatakabel.

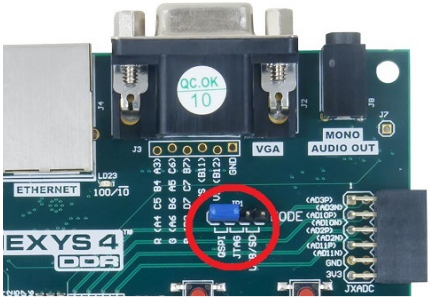


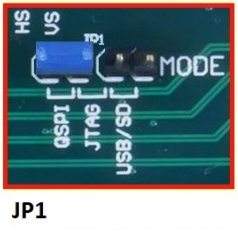
Voor eigenaren van een Nexys4-bord:

Sluit uw micro-USB-kabel aan tussen uw Nexys4-bord en uw pc via de poort met het label “PROG UART” (J6), zoals getoond:



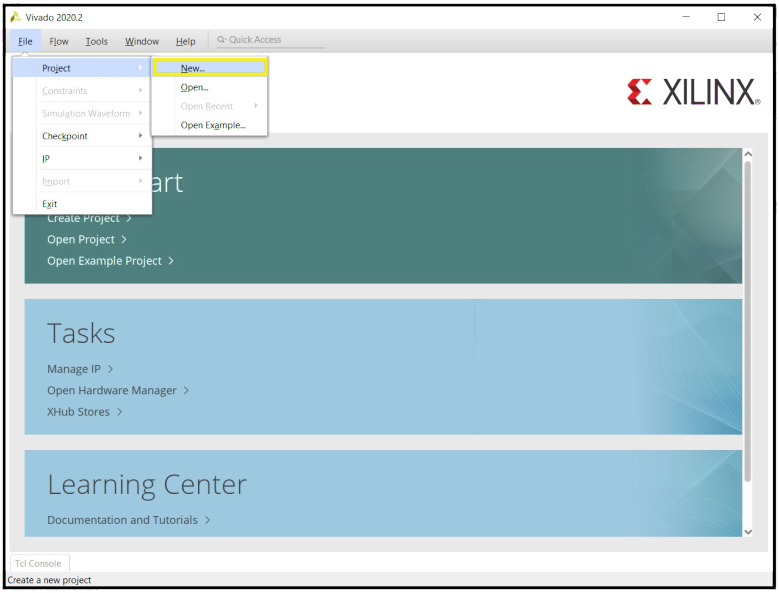
Zet bovendien de J1-jumper op de QSPI-stand:



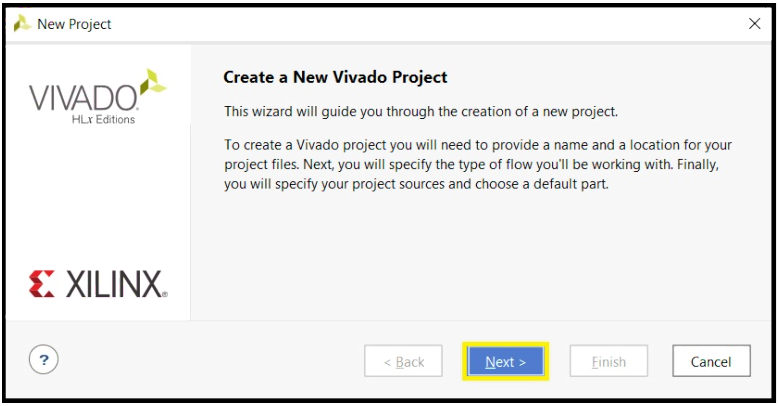


- Sluit uw niet-8-bit computer aan op het FPGA-programmeerapparaat met behulp van de juiste USB-kabel.
- Schakel de MEGA65-computer in.
- Open Vivado.

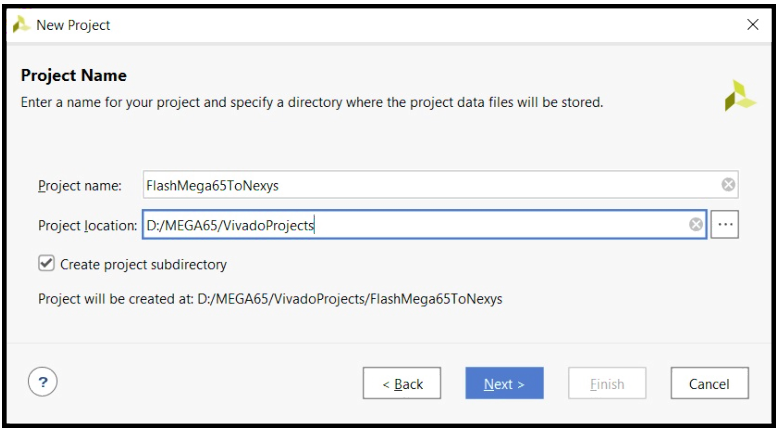
Stap 1a: Maak een nieuw Vivado-project aan via “File”, “Project”, “New...”.
LET OP: Bij toekomstige gelegenheden waarop u de QSPI moet flashen, heropent u dit project (het is niet nodig elke keer een nieuw project aan te maken).



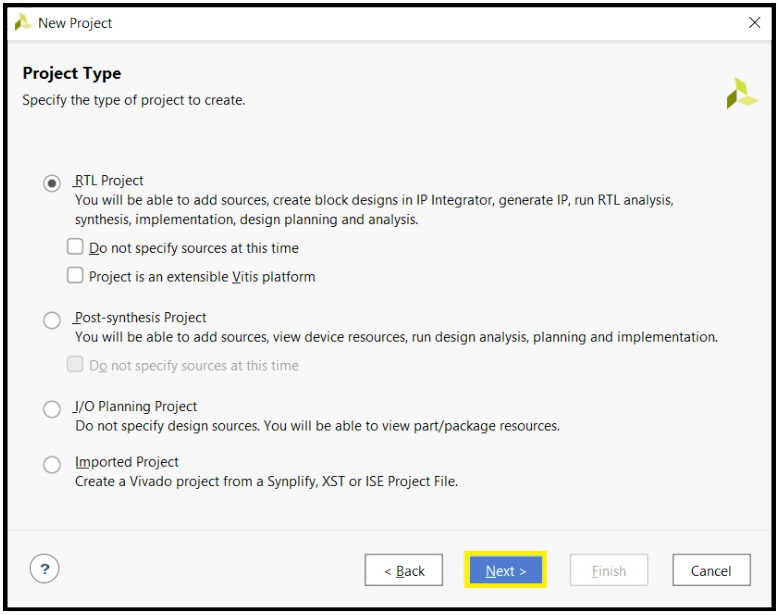
Stap 1b: De wizard “New Project” verschijnt. Klik op “Next”:



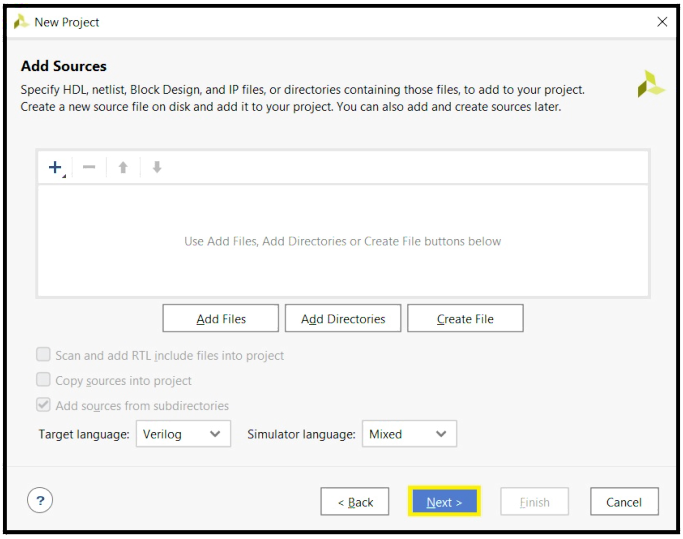
Stap 1c: Geef uw project een naam en kies de gewenste locatie en klik op “Next”:



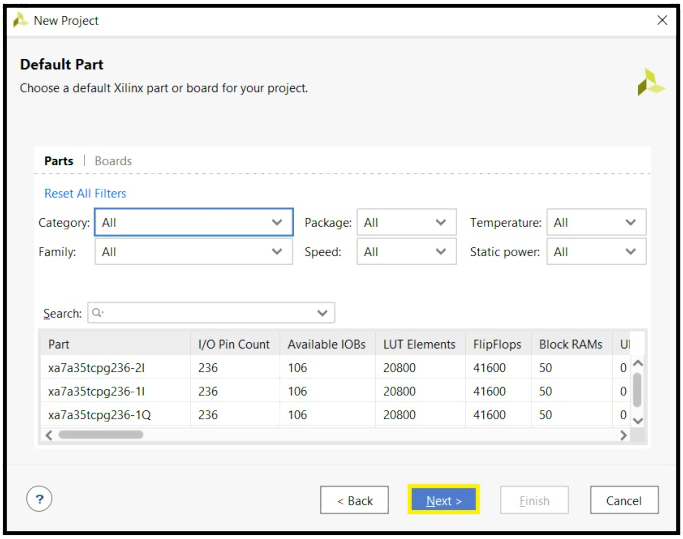
Stap 1d: Laat de standaard geselecteerde opties behouden en klik op “Next”:



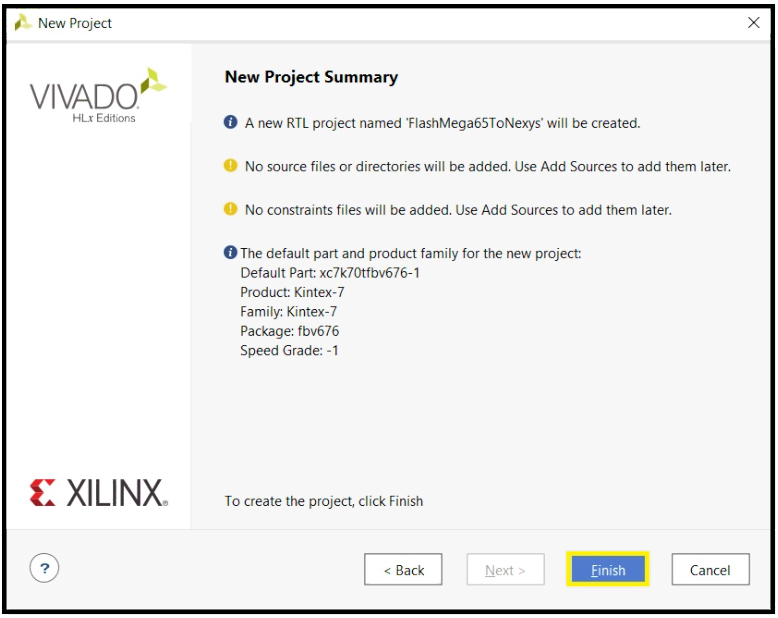
Stap 1e: Voeg geen bronnen toe, laat de standaard geselecteerde opties behouden en klik op “Next”:



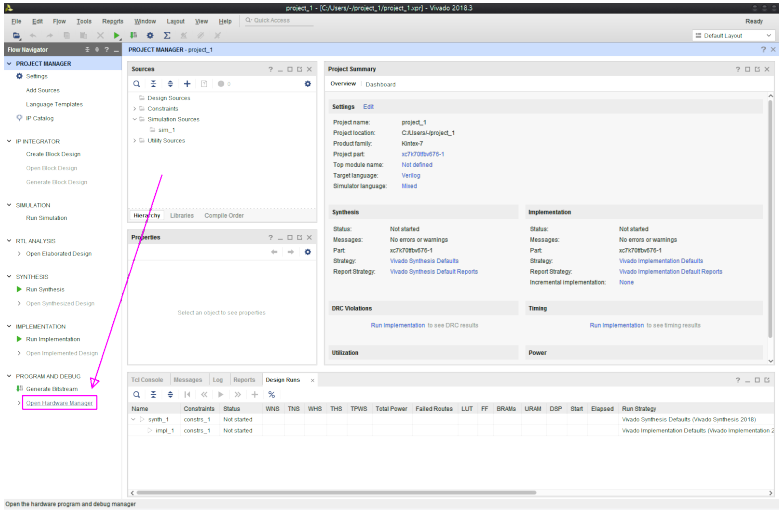
Stap 1f: Laat de standaard geselecteerde opties behouden en klik op “Next”:



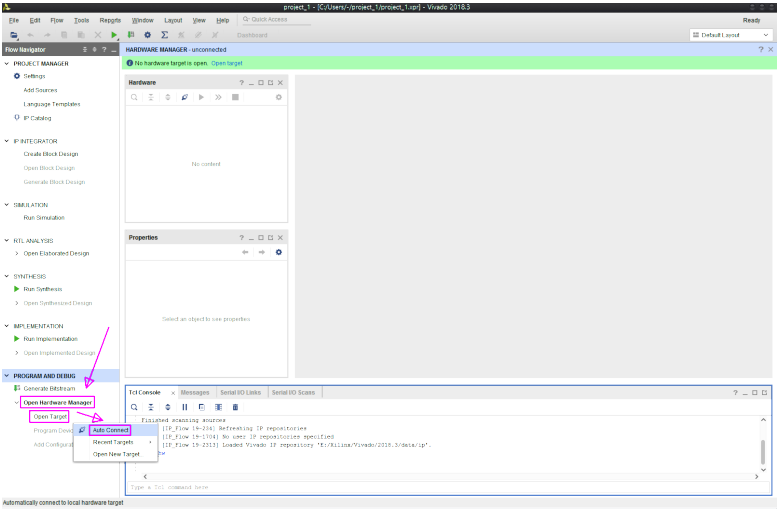
Stap 1g: Klik op “Finish”:



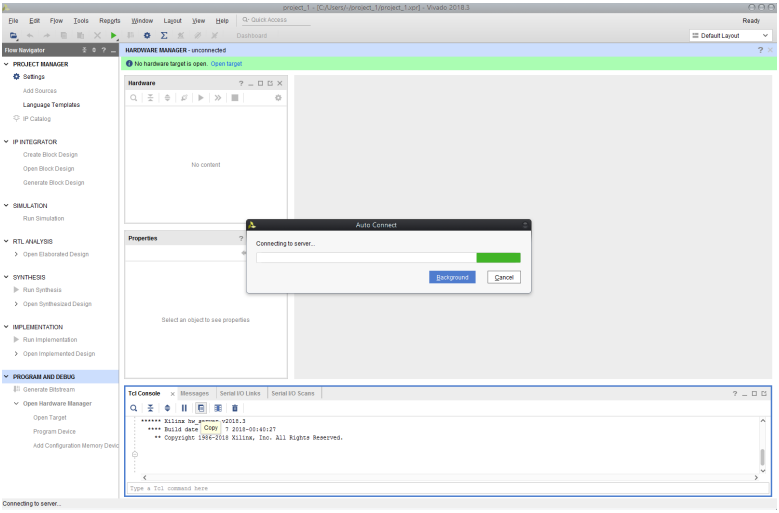
Stap 2: Selecteer in de linker kolom “Open Hardware Manager” helemaal onderaan.



Stap 3: Verbind met de FPGA:
Onder “Open Hardware Manager” kiest u “Open Target” en vervolgens “Auto Connect”.

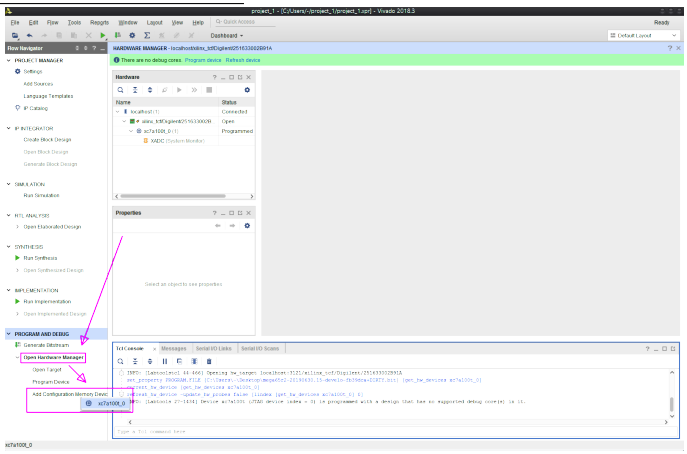


Stap 4: Wacht even; “Connecting to server...” zou automatisch moeten sluiten zonder een foutmelding aan de console te geven.



Stap 5: Onder “Open Hardware Manager” kiest u “Add Configuration Memory Device” en vervolgens:

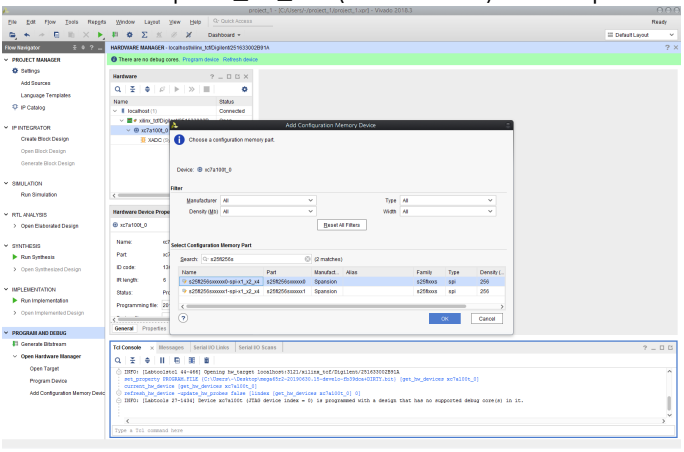
- Voor MEGA65R3/R3A/R4: “xc7a200t_0”
- Voor Nexys4 en MEGA65R2: “xc7a100t_0”.



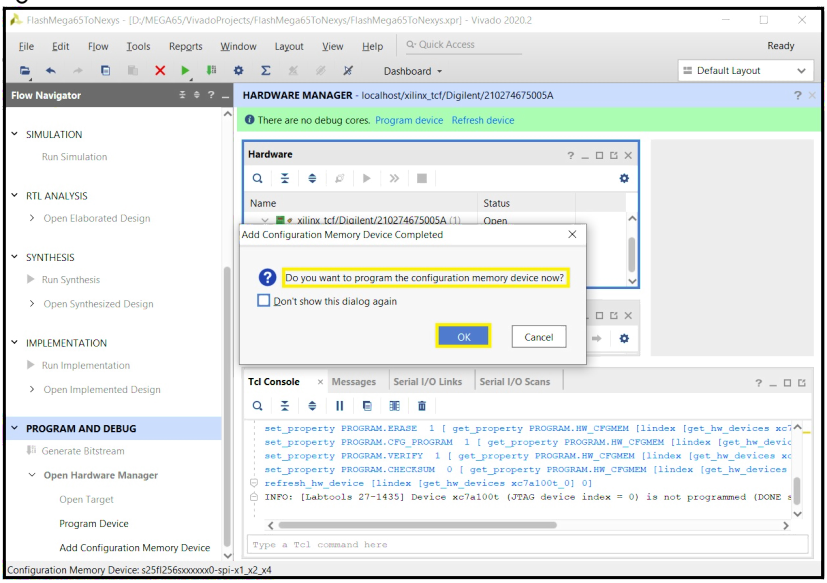
Stap 6a: Selecteer Memory Part:

In het nieuw geopende dialoogvenster:

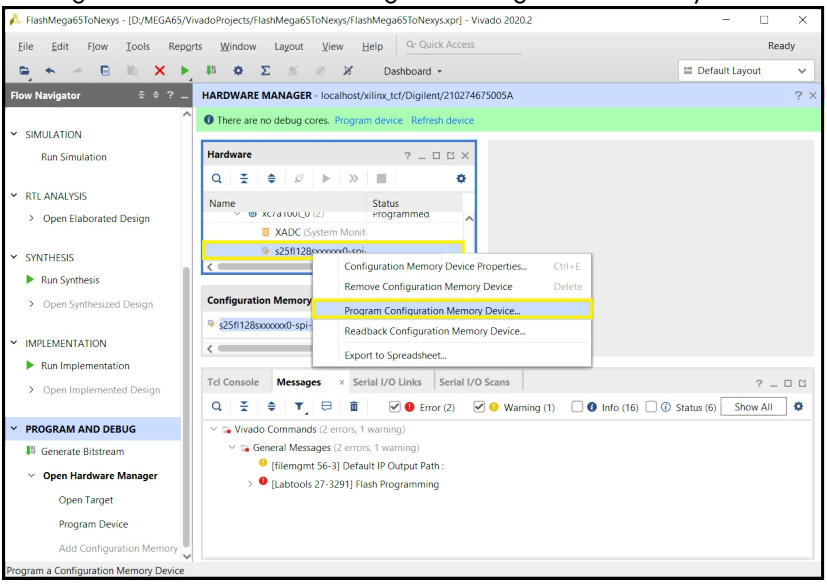
- Voor MEGA65R2/R3: typ “S25fl256s” (zonder aanhalingstekens), selecteer vervolgens “s25fl256sxxxxxxx0-spi-x1_x2_x4” (de bovenste) en klik op “OK”.
- Voor MEGA65R3A/R4: Vivado kan het grotere flashgeheugen op deze borden niet flashen. Gebruik het flashmenu op de MEGA65.
- Voor Nexys4: typ “S25fl128s” (zonder aanhalingstekens), selecteer vervolgens “s25fl128sxxxxxxx0-spi-x1_x2_x4” (de bovenste) en klik op “OK”.



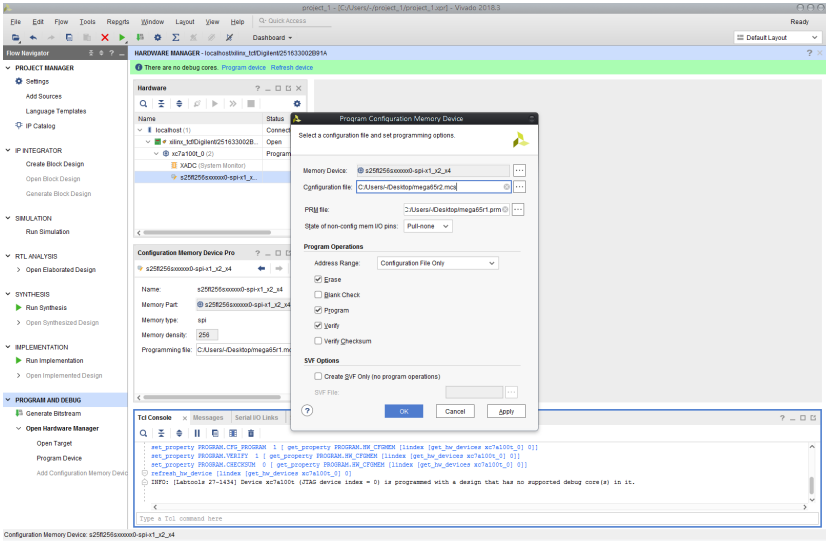
Stap 6b: Klik op “OK” om te bevestigen dat u het configuratiegeheugenapparaat nu wilt programmeren.



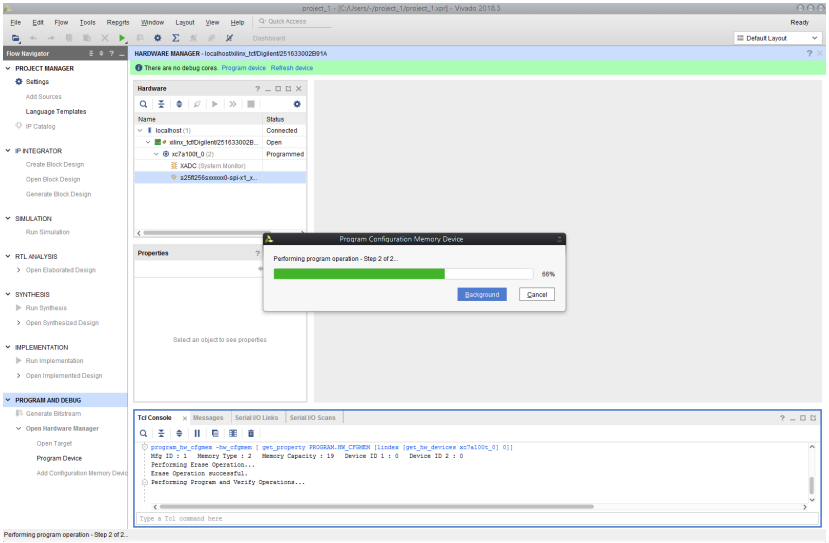
Stap 6c: Als u zo’n pop-up niet ziet of de QSPI op een later tijdstip opnieuw wilt programmeren, klikt u in het “Hardware”-venster met de rechtermuisknop op de geheugenconfiguratie en selecteert u “Program Configuration Memory Device”:



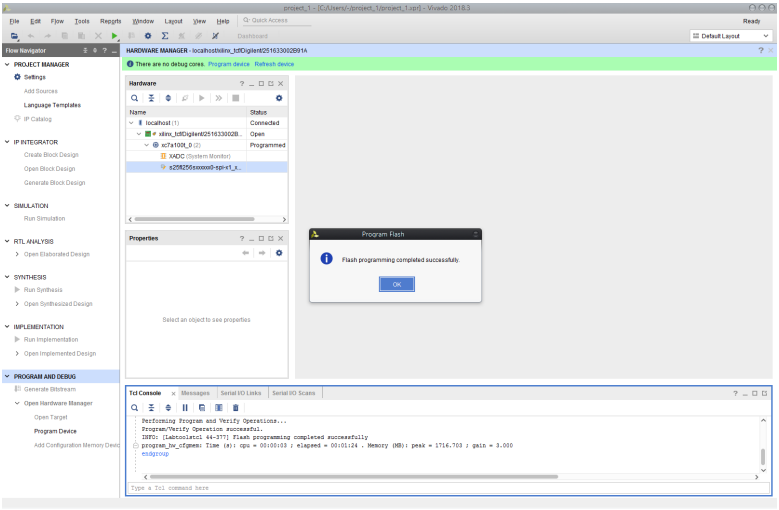
Stap 7: Stel de programmeeropties in:
Stel in het volgende dialoogvenster “Configuration file” in op het pad van uw “.mcs”-bitstream-bestand. U kunt optioneel ook het “PRM file”-veld instellen op het pad van uw “.prm”-bestand. Laat alle overige parameters zoals ze zijn (zie onderstaande screenshot).



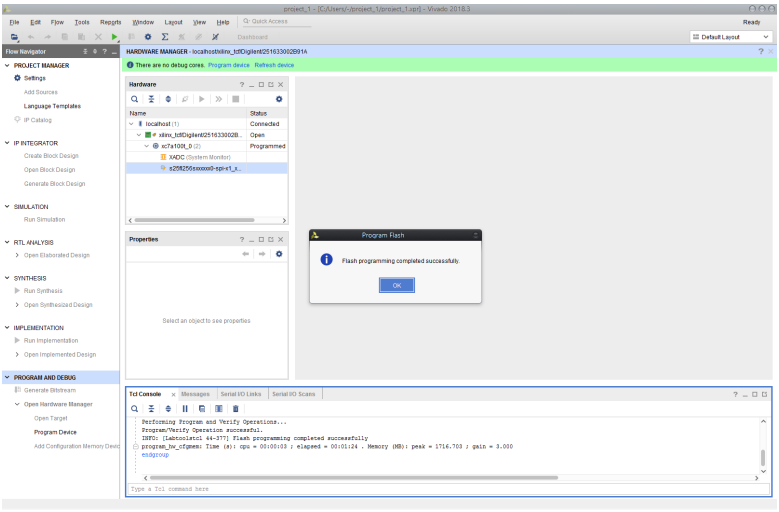
Stap 8: Wacht geduldig tot het programmeren is voltooid. Dit kan enkele minuten duren, omdat de Vivado-software het flashgeheugen dat wordt gebruikt om de FPGA bij het inschakelen te initialiseren, eerst wist en vervolgens opnieuw programmeert.



Stap 9: Als uw scherm eruitziet als de onderstaande screenshot, is uw nieuwe bitstream succesvol geflasht in Slot 0 van uw QSPI-flashgeheugen!

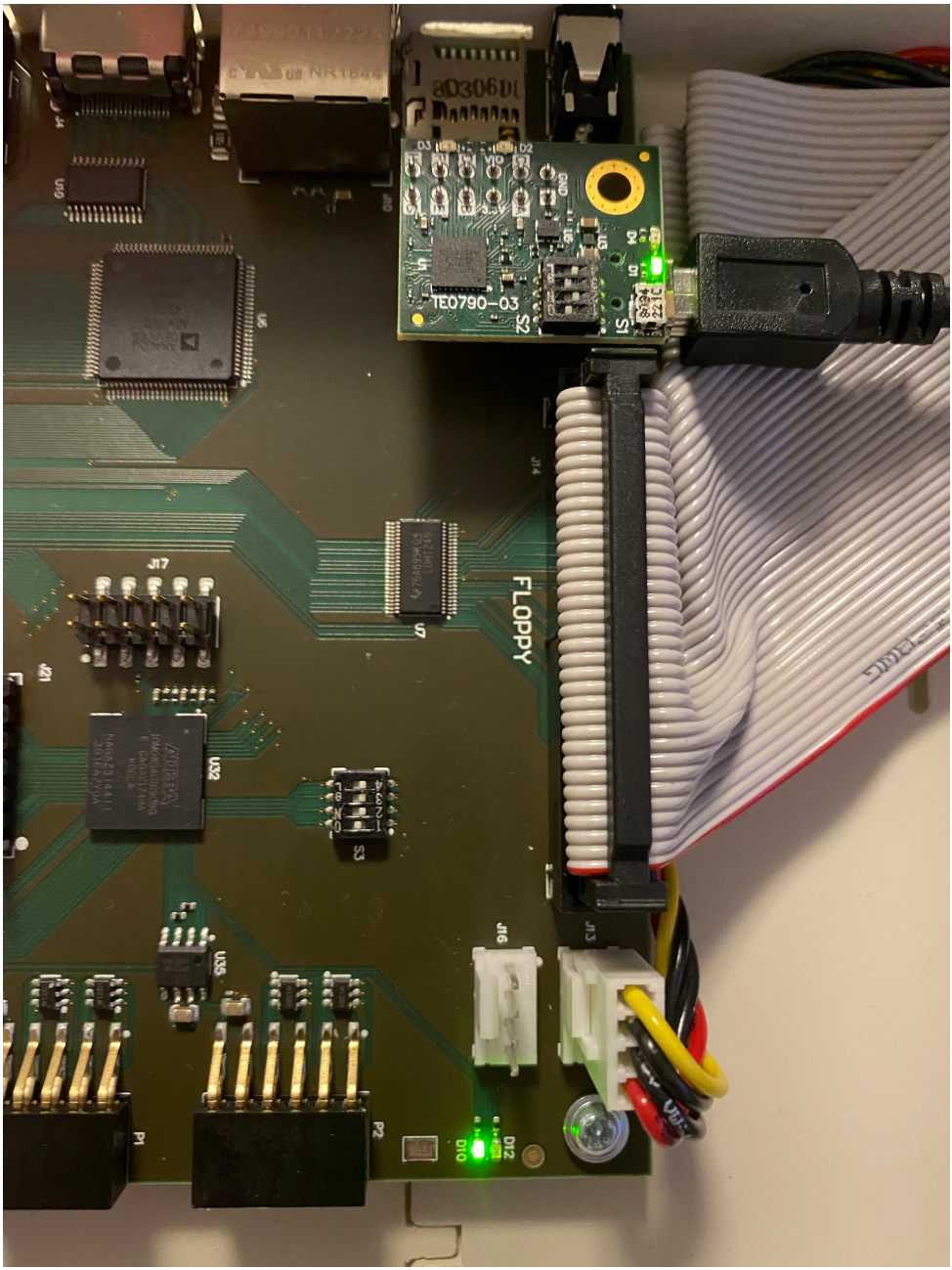


Stap 10: Als u de FPGA opnieuw wilt flashen, kan het zijn dat de optie “Add Configuration Memory Device” uit stap 5 grijs weergegeven wordt. Selecteer in dat geval “s25fl256sxxxxxxx0-spi-x1_x2_x4” in het “Hardware”-venster, druk op de rechtermuisknop en selecteer “Program Configuration Memory Device” om te flashen.



DE CPLD IN HET TOETSENBORD VAN DE MEGA65 FLASHEN MET LATTICE DIAMOND

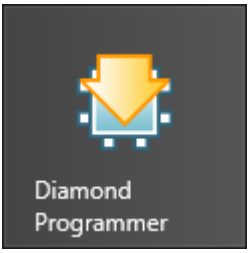
Indien u ervoor kiest om door te gaan, heeft u een TE0790-03 JTAG-programmeermodule en een werkende installatie van Lattice Diamond Programmer-software nodig. Dit kan op zowel Windows als Linux, maar in beide gevallen moet u de benodigde USB-stuurprogramma’s installeren. Het is ook noodzakelijk dat dip-schakelaars 1 en 3 op de AAN-stand staan en dip-schakelaars 2 en 4 op de UIT-stand op de TE-0790. Met de MEGA65 losgekoppeld van de voeding moet de TE-0790 op de JB1-connector worden geïnstalleerd, die zich tussen de diskettedatakabel en de audiojack bevindt. De vergulde opening van de TE-0790 moet in lijn liggen met het schroefgat eronder. De mini-USB-kabel verbindt zich vervolgens aan de kant richting het 3.5” diskettestation. De volgende afbeelding toont de juiste positie: de TE0790 is omringd door de gele kast en de dip-schakelaars door de rode kast. Dip-schakelaar 1 is de dichtstbijzijnde bij de diskettedatakabel.


FLOPPY
TE0790-03
S1
S2
J17
J21
U32
U33
S3
J16
J3
J14
P1
P2
D12
U9
U8
U6
U4

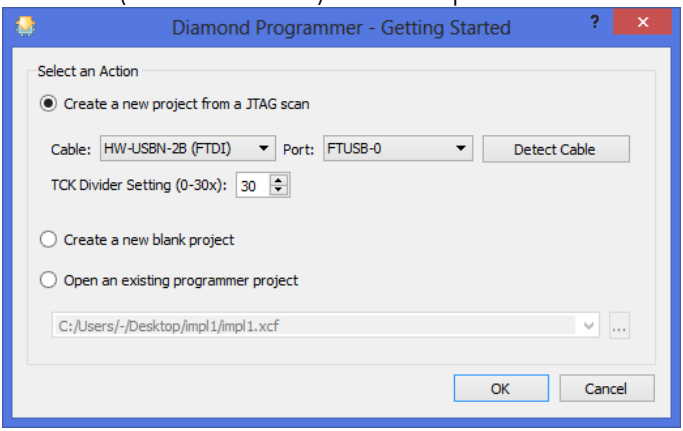
Op de PCB van een R2/R3/R3A/R4 MEGA65-moederbord moet dip-schakelaar 1 (de dichtstbijzijnde bij de gebruiker die voor de computer zit) op de AAN-stand staan. De overige schakelaars moeten UIT staan. Het toetsenbord gaat in de "ambulance-modus" (blauwe flitsende lichten) wanneer het correct is ingesteld.

Sluit uw niet-8-bit computer aan op het FPGA-programmeerapparaat met behulp van een mini-USB-kabel. Schakel de MEGA65-computer in. Open de Diamond Programmer, die van het internet kan worden gedownload.

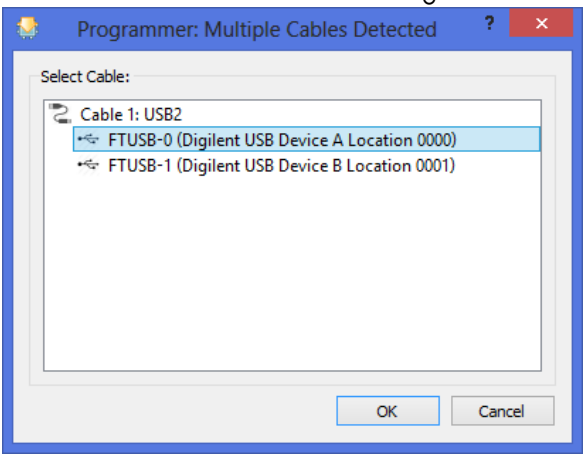
Stap 1: Open DIAMOND PROGRAMMER:
Selecteer "Create a new project from a JTAG scan". Indien het veld onder "Cable:" leeg is, klikt u op "Detect Cable".



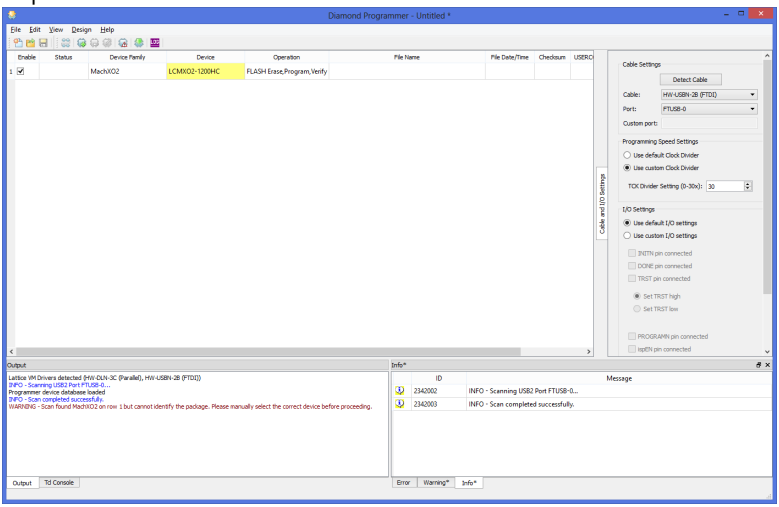
Stap 2: Maak een nieuw project aan:
Indien het dialoogvenster "Programmer: Multiple Cables Detected" verschijnt, selecteert u het eerste item ("Location 0000") en klikt u op "OK".



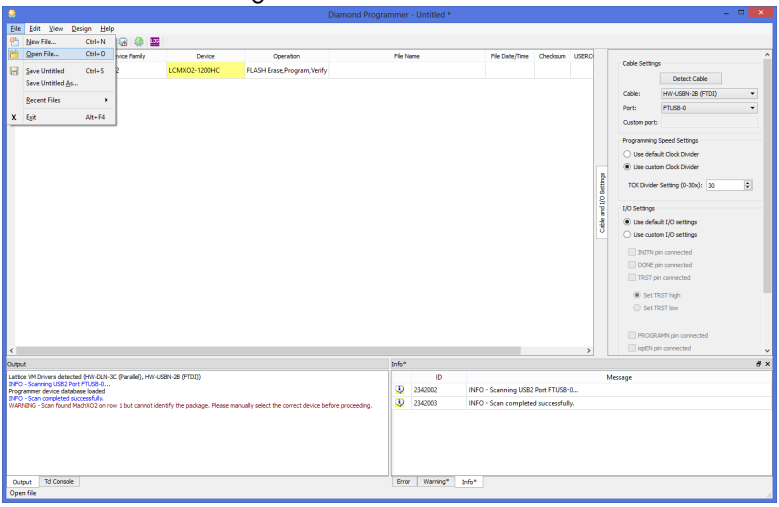
Stap 3: Selecteer kabel:
U heeft nu een nieuw project aangemaakt dat “MachXO2” onder “Device Family” en “LCMXO2-1200HC” onder “Device” zou moeten weergeven.



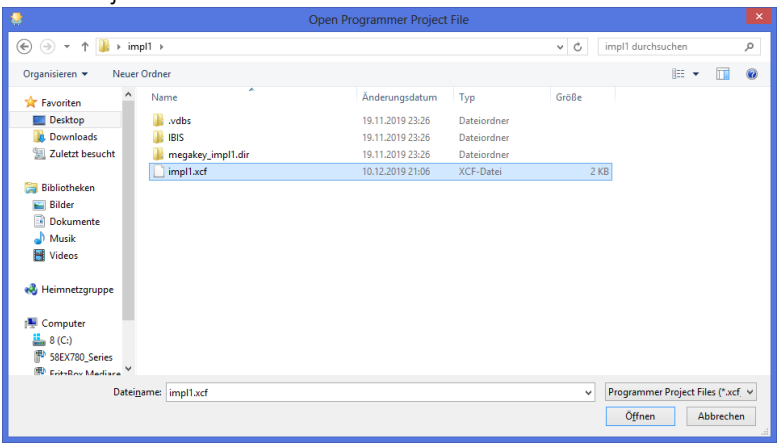
Stap 4: Nieuw Diamond Programmer-project:
Kies “File” en vervolgens “Open File” om het Diamond Programmer-project met de firmware-update voor het MEGA65-toetsenbord te laden.



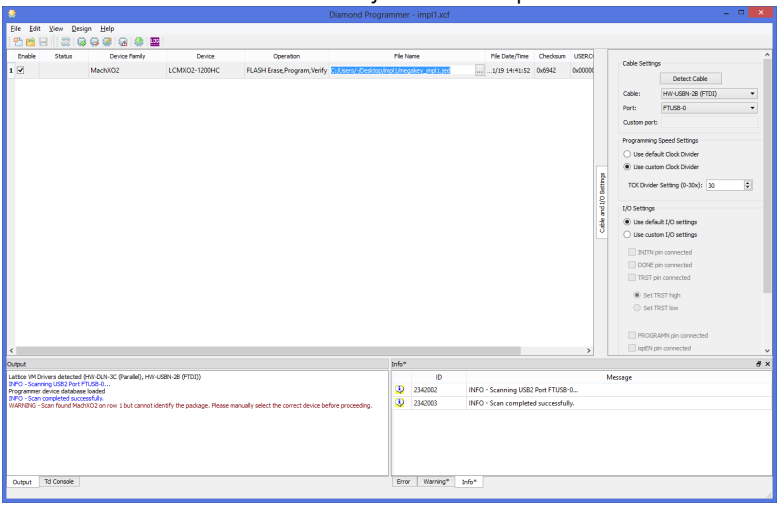
Stap 5: Open project:
Navigeer naar de map met de uitgepakte firmwarebestanden voor het MEGA65-toetsenbord die u hebt ontvangen en selecteer het bestand met de extensie “.xcf”.



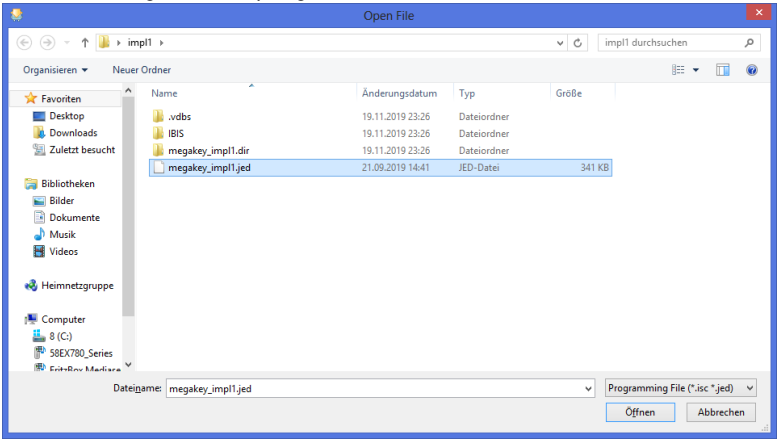
Stap 6: Selecteer projectbestand:
Klik op de drie puntjes onder “File Name” om het juiste pad in te stellen en het bestand met de extensie “.jed” te vinden.



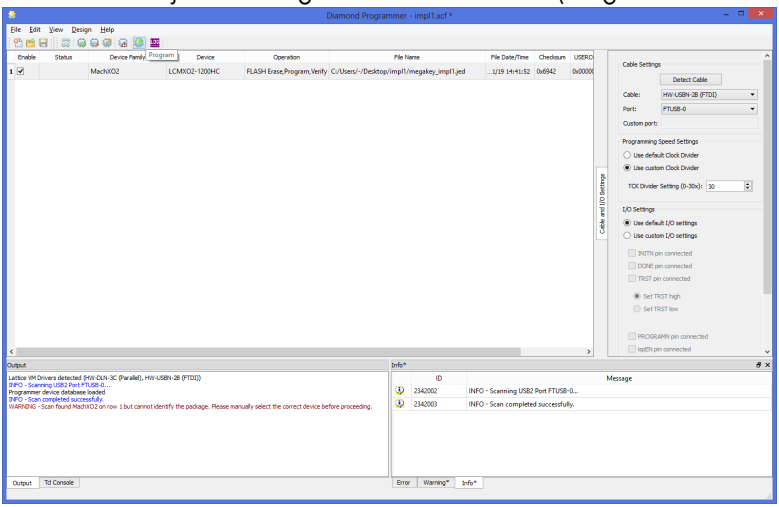
Stap 7: Kies het juiste pad van het .jed-bestand:
Selecteer het bestand met de extensie ".jed" en klik op "OK".



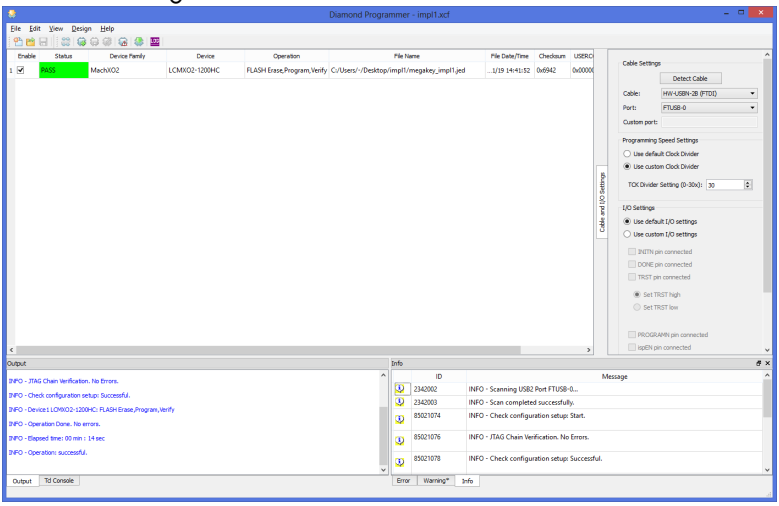
Stap 8: Selecteer .jed-bestand:
Klik op het icoon met de groene pijl naar beneden "PROGRAM", dat lijkt op het icoon van het Diamond Programmer-programma.



Stap 9: Selecteer kabel:
Na een moment zou het uitvoervenster “INFO - Operation: successful.” moeten weergeven en zou het celletje “Status” groen moeten worden (dit gebeurt niet altijd).



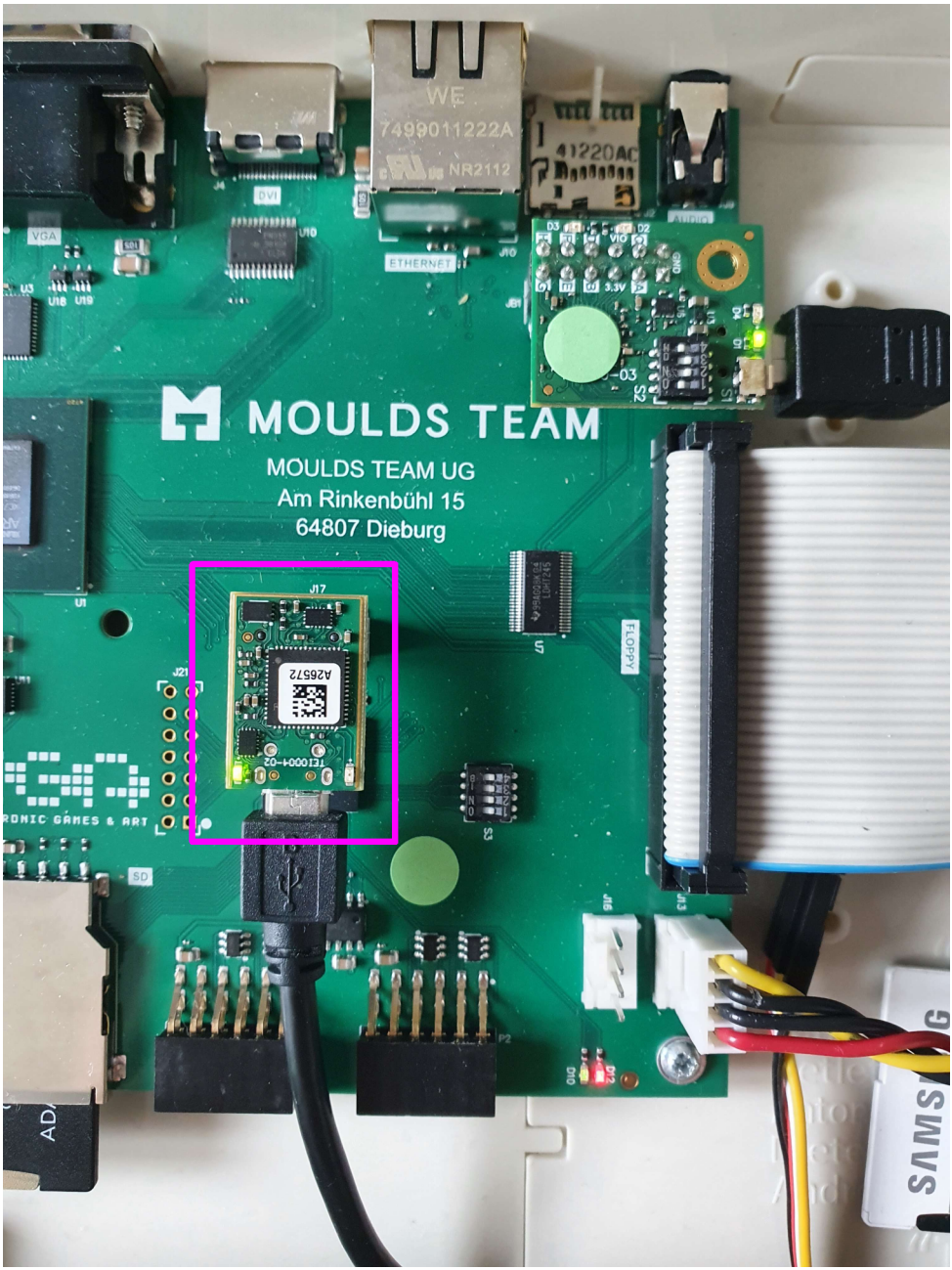
Stap 10: Operatie geslaagd:
U heeft het MEGA65-toetsenbord nu succesvol geflasht. Desgewenst kunt u het project opslaan voor later gebruik.



HET FLASHEN VAN DE MAX10-FPGA OP HET MEGA65-MOEDERBORD MET INTEL QUARTUS

Indien u verder wilt gaan, heeft u een TEI0004 - Arrow USB Programmer2-module nodig met het bijbehorende TEI0004-stuurprogramma geïnstalleerd en een goed functionerende installatie van Quartus Prime Programmer Lite Edition. Dit kan zowel op Windows als op Linux, maar in beide gevallen moet u de nodige USB-stuurprogramma's installeren. Sluit de MEGA65 niet aan op de netspanning en installeer de TEI0004 op de J17-connector, die zich tussen de datakabel van de diskette en de ARTIX 7-FPGA op het moederbord bevindt. De micro-USB-poort van de TEI0004 wijst in de tegenovergestelde richting van de HDMI- en LAN-aansluitingen, naar het klepje. De volgende afbeelding toont de juiste positie.

Op het PCB van een R2/R3/R3A/R4-MEGA65-moederbord moeten alle dipschakelaars in de positie uit staan. De hoofd-FPGA van de MEGA65 mag geen geldige bitstream bevatten. De eenvoudigste manier om dit te bereiken is met een TE0790-JTAG-adapter: u kunt dan het m65-hulpprogramma vanaf een aangesloten computer gebruiken om een bitstream via JTAG te verzenden en vervolgens Control-C gebruiken om het m65-hulpprogramma te onderbreken voordat de bitstream volledig is geladen.

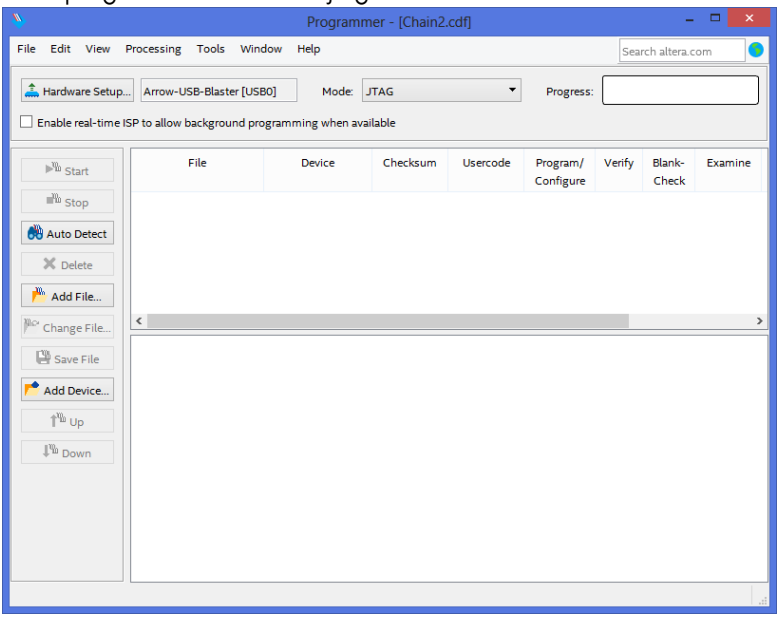

MOULDS TEAM
MOULDS TEAM UG
Am Rinkenbühl 15
64807 Dieburg
J17
A26572
J21
FLOPPY
SD
ETHERNET
DVI
VGA

Sluit uw niet-8-bit computer aan op het FPGA-programmeerapparaat met behulp van een micro-USB-kabel. Open Quartus Prime Programmer Lite Edition, dat van internet kan worden gedownload.

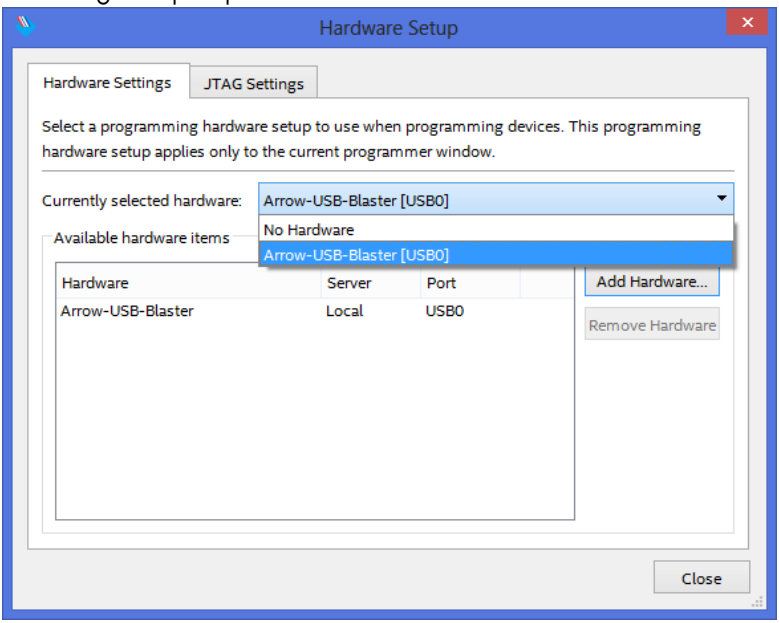
Stap 1: Open Quartus Prime Programmer Lite Edition:
Klik op de knop “Hardware Setup” in de linkerbovenhoek van het Quartus Prime Programmer-venster.



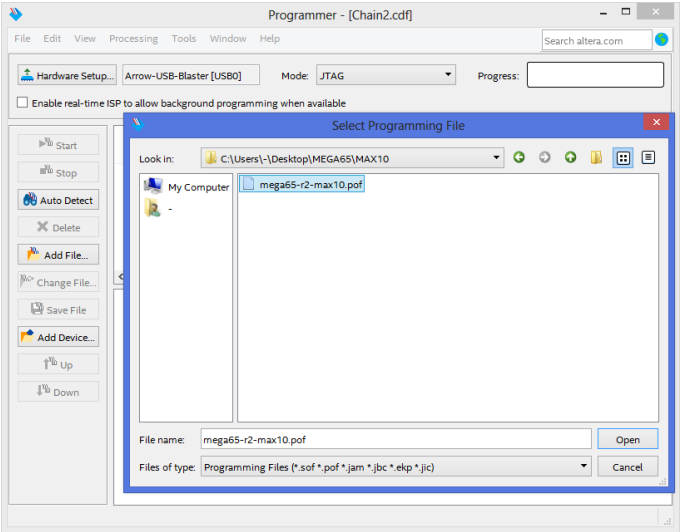
Stap 2: Hardware-instellingen invoeren:
In het nieuw verschenen venster kiest u onder “Currently selected hardware” voor “Arrow-USB-Blaster”. Indien “Arrow-USB-Blaster” niet verschijnt, controleert u of de kabel en stuurprogramma’s correct zijn geïnstalleerd.



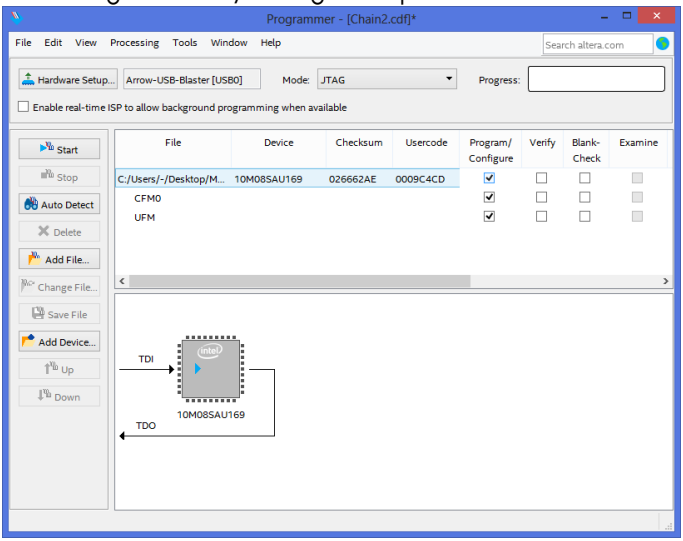
Stap 3: Selecteer Arrow USB-Blaster:
Klik op de knop “Add File” in de linkerrij en kies het nieuwste bestand met de extensie “.pof”. Klik vervolgens op “Open”.



Stap 4: Selecteer programmeerbestand:
Vink ten minste de drie vakjes onder “Program/Configure” aan. Het inschakelen van alle vakjes onder “Verify” en “Blank-Check” maakt het proces betrouwbaarder.



Stap 5: Selecteer Programmeer-/Configureeropties:

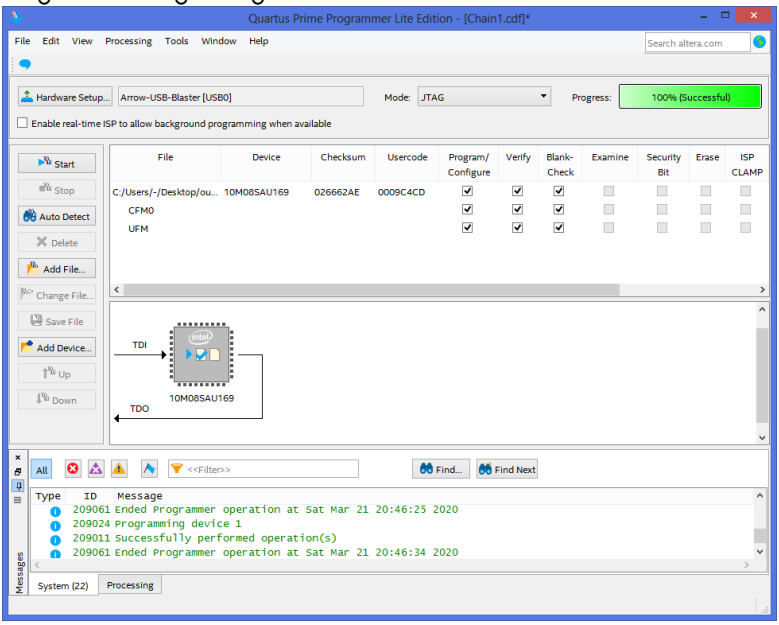


Houd de Reset-knop ingedrukt en schakel de MEGA65-computer in. Het toetsenbord gaat in de “ambulance mode” (blauwe flitsende lichten). Indien dit niet het geval is, is de hoofd-FPGA niet leeg: start het gehele proces opnieuw.

Klik nu op “Start” in de linkerrij van knoppen. De voortgangsbalk rechtsboven zou snel naar 100 procent moeten gaan en groen moeten worden. U heeft de MAX10-FPGA nu succesvol bijgewerkt.

Indien u in plaats daarvan een foutmelding ontvangt, controleer dan of de bitstream van de hoofd-FPGA is gewist en of u de reset-knop op de MEGA65 niet te vroeg heeft losgelaten. Schakel de MEGA65 uit en herhaal deze stap.

Stap 6: Programmeren geslaagd:



BIJLAGE B

Problemen oplossen

- Hardware
- Vivado
- mega65_ftp

HARDWARE

Geen digitale video-uitvoer

MEGA65 R4, R5, R6

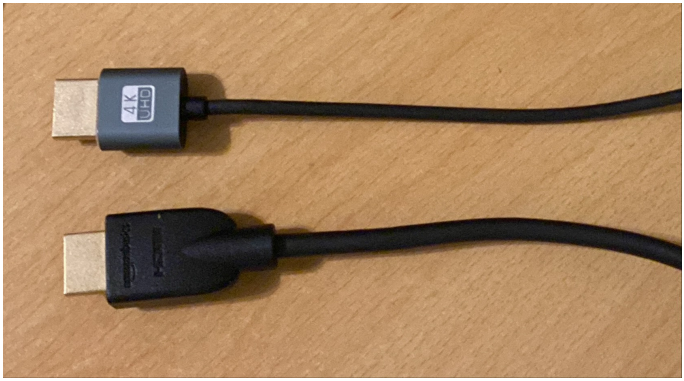
De digitale videopoort heeft een verbeterde isolatie tegen terugvoeding gekregen om het probleem "Geen lampjes bij het inschakelen" op te lossen. Een neveneffect hiervan is dat sommige digitale videokabels ten onrechte alleen de behuizing van de connector gebruiken om het grondsignaal door te geven, in plaats van de grondsignalen van de pinnen van de connector te gebruiken. Kort samengevat: als u geen beeld krijgt op de digitale videopoort, kunt u een van de volgende oplossingen proberen:

- Probeer verschillende digitale videokabels die u heeft. Dit probleem komt alleen voor bij bepaalde merken kabels, vooral dunnere exemplaren, waarvan wij sterk vermoeden dat de interne grondleidingen ontbreken om kosten te besparen. Dit is de beste oplossing, omdat de kabel daadwerkelijk de oorzaak van het probleem is doordat deze niet aan de specificaties voldoet. Zelfs als u een van de volgende oplossingen toepast, moet u toch op zoek gaan naar een betere HDMI-kabel die het probleem bij de bron oplost.
- Sluit een VGA-kabel aan op de MEGA65 en op hetzelfde beeldscherm. Hierdoor wordt een grondverbinding via de VGA-connector tot stand gebracht, waardoor de HDMI op mysterieuze wijze plotseling lijkt te werken.
- Sluit een TE0790 JTAG-adapter aan. Een gebruiker meldde dat dit het probleem eveneens op mysterieuze wijze omzeilde.
- Verbind de behuizing van de digitale video-connector met de behuizing van de interne SD-kaart op vol formaat.
- Sluit een splitter of repeater aan op de digitale videopoort.

Geen lampjes bij het inschakelen

Als de MEGA65 bij het inschakelen soms geen lampjes laat zien, heeft dit te maken met bepaalde digitale video-apparaten die op het moment dat de MEGA65 uitstaat aangesloten zijn en onvoldoende vermogen leveren om de CPLD van het toetsenbord correct van stroom te voorzien, maar wel genoeg om te voorkomen dat deze goed kan resetten wanneer de MEGA65 wordt ingeschakeld. Het verwijderen van de digitale videokabel en het opnieuw inschakelen van het apparaat lost het probleem op.

Figuur B.1: Voorbeeld van een slechte (boven) en goede (onder) digitale videokabel. Let op hoe de slechte kabel verdacht dunner is dan de goede. Dit is een teken dat waarschijnlijk dedicated grondleidingen ontbreken, wat problemen veroorzaakt op de MEGA65 R4 en latere revisies. Elke kabel van "normale dikte" zou echter goed moeten werken. U hoeft geen van die peperdure "met zilverdraad gevlochten, bij maanlicht door elfen gesmeede" kabels te kopen. Die kabels bieden ongeveer evenveel praktisch nut als het strooien van glitter of het tekenen van een eenhoorn aan één uiteinde, maar dan met veel minder klasse.



VIVADO

RAM-vereisten

```
INFO: [Synth 8-256] done synthesizing module 'ram32x1024' [/home/....]
INFO: [Synth 8-256] synthesizing module 'charrom' [/home/....]
   /opt/Xilinx/Vivado/2019.2/bin/loader: line 280: 2317 killed
   WARNING: [Vivado 12-8222] Failed run(s) : 'synth\_1'
       ERROR: Application Exception: failed to launch run 'impl\_1' due
           to failures in the following run(s):
       synth\_1
       These failed run(s) need to be reset prior to launching 'impl\_1'
           again.
```

Deze fout wordt veroorzaakt doordat Vivado crasht omdat het apparaat onvoldoende RAM heeft om Vivado uit te voeren. Vivado heeft minimaal 4 GB RAM nodig om het MEGA65-doel te synthetiseren, maar 8 GB is beter.

MEGA65_FTP

Ontbrekende bibliotheek

```
/usr/bin/ld: cannot find -lncurses
collect2: error: ld returned 1 exit status
Makefile:474: recipe for target 'bin/mega65_ftp' failed
make: *** [bin/mega65_ftp] Error 1
```

Deze fout treedt op wanneer de ncurses-bibliotheek op de computer ontbreekt bij het bouwen van het mega65_ftp-programma. Om dit probleem op te lossen, moet u deze afhankelijkheid installeren.

```
sudo apt-get install libncurses5-dev libncursesw5-dev
```

BIJLAGE C

Elektronisch schema

- MEGA65 R6 elektronisch schema
- MEGA65 R3 elektronisch schema
- MEGA65 R2 elektronisch schema
- Nexys Widget Board elektronisch schema

MEGA65 R6 ELEKTRONISCH SCHEMA

U_B13
B13.SchDoc

U_B14
B14.SchDoc

U_B15
B15.SchDoc

U_B16
B16.SchDoc

U_B34
B34.SchDoc

U_FPGA-CFG
FPGA-CFG.SchDoc

U_FPGA-MGT
FPGA-MGT.SchDoc

U_FPGA-PWR
FPGA-PWR.SchDoc

U_PowerMain
PowerMain.SchDoc

U_POWER
POWER.SchDoc

U_HDMI
HDMI.SchDoc

U_VGA
VGA.SchDoc

U_Ethernet
Ethernet.SchDoc

U_SOUND
SOUND.SchDoc

U_KEYBOARD
KEYBOARD.SchDoc

U_Floppy
Floppy.SchDoc

U_JOY
JOY.SchDoc

U_EXT_HEADER
EXT_HEADER.SchDoc

U_EXP_Slot
EXP_Slot.SchDoc

U_LED_SW_BUT
LED_SW_BUT.SchDoc

PM1 FIDU-DOT - small
PM2 FIDU-DOT - small
PM3 FIDU-DOT - small
PM4 FIDU-DOT - small
PM5 FIDU-DOT - small
PM6 FIDU-DOT - small

Address 0x40 I/O Expander
Address 0x61 DC/DC Converter
Address 0x67 DC/DC Converter

KK1
ATS-CPX025025015-138-C2-R0
R234 0R
R235 0R
GND GND

Serial1
Serial
Serialnumber 6,3 x 6.3mm

LOGO1
TE Logo PRINT Layer
LOGO PRINT

CE
CE Logo on Top Overlay
CE-TOPOVERLAY

Shielding

DDR
SHIELD4
BMI-S-202-C

FPGA
SHIELD5
BMI-S-230-C

DCDC
SHIELD6
BMI-S-205-C

Shield1
GND
BMI-S-202-F

Shield2
GND
BMI-S-230-F-R

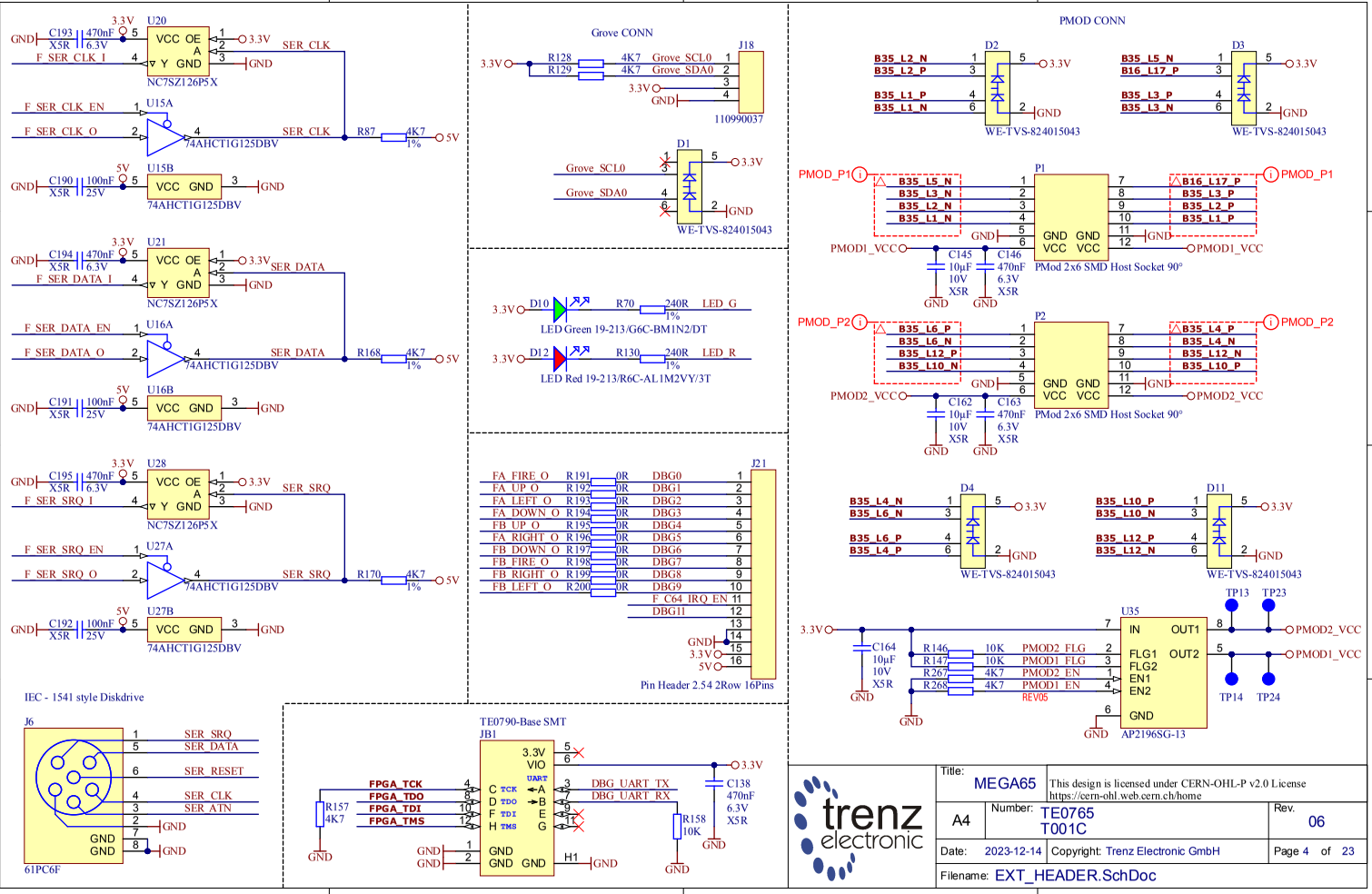
Shield3
GND
GND
BMI-S-205-F

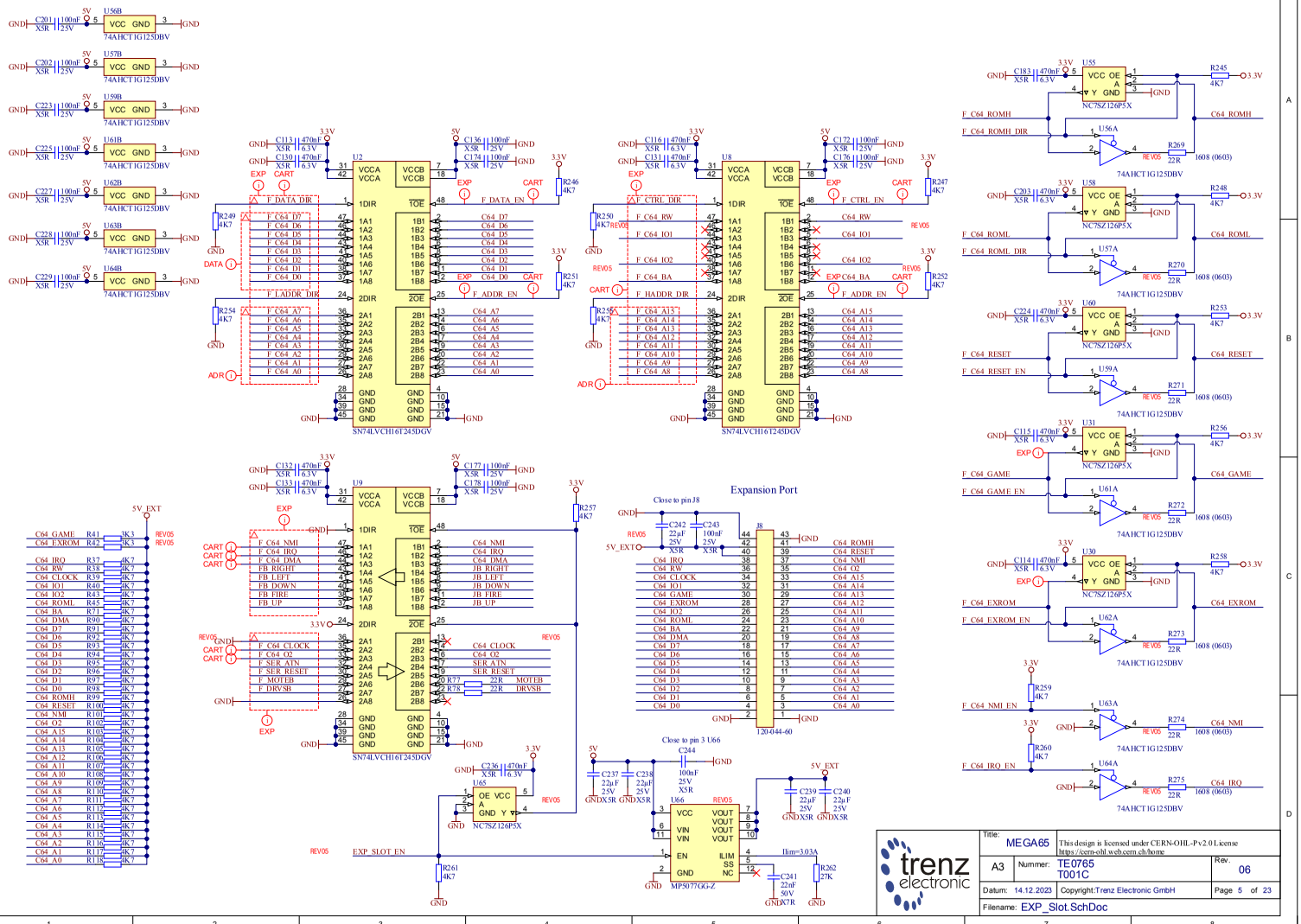
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm

Design drawn by:	VY
Checked by:	IG
Assembly variant:	T001C
Created by:	VY
Modified by:	VY
Modified at:	2023-07-14

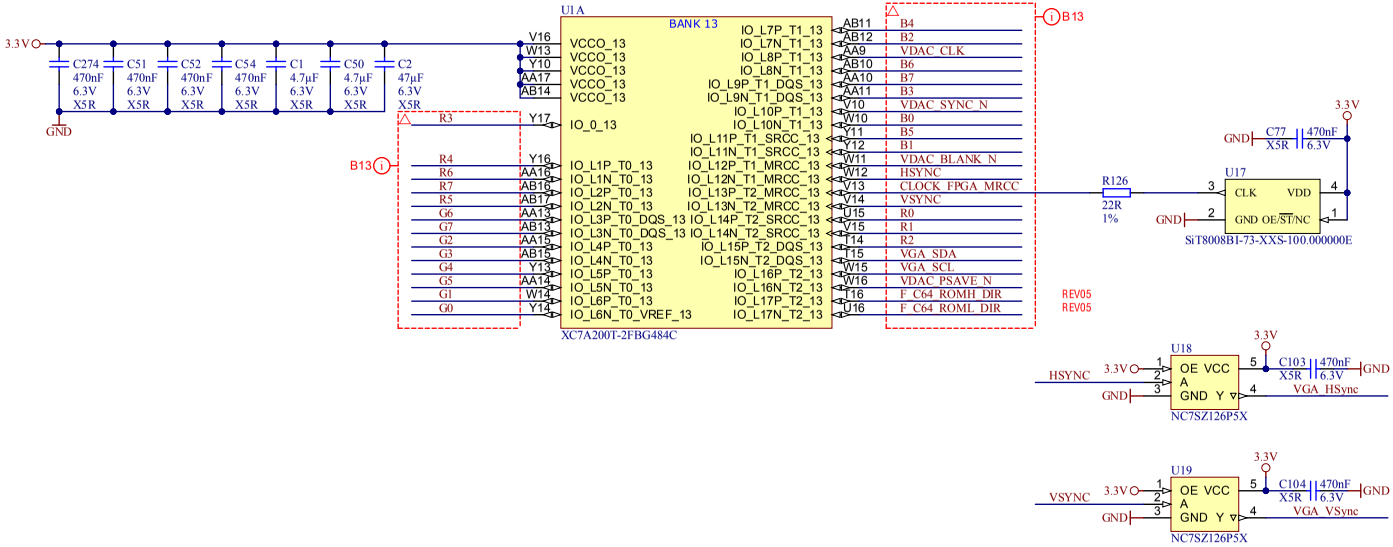
trenz electronic

Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
A4	Number: TE0765 T001C	Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 3 of 23
Filename: TE0765.SchDoc		


Grove CONN
PMOD CONN
IEC - 1541 style Diskdrive
Pin Header 2.54 2Row 16Pins
LED Green 19-213/G6C-BM1N2/DT
LED Red 19-213/R6C-AL1M2VY/3T
PMod 2x6 SMD Host Socket 90°
WE-TVS-824015043
NC7SZ126P5X
74AHCT1G125DBV
110990037
61PC6F
AP2196SG-13
TE0790-Base SMT
MEGA65
trenz electronic
This design is licensed under CERN-OHL-P v2.0 License
https://cern-ohl.web.cern.ch/home
TE0765
T001C
06
2023-12-14
Trenz Electronic GmbH
Page 4 of 23
EXT_HEADER.SchDoc

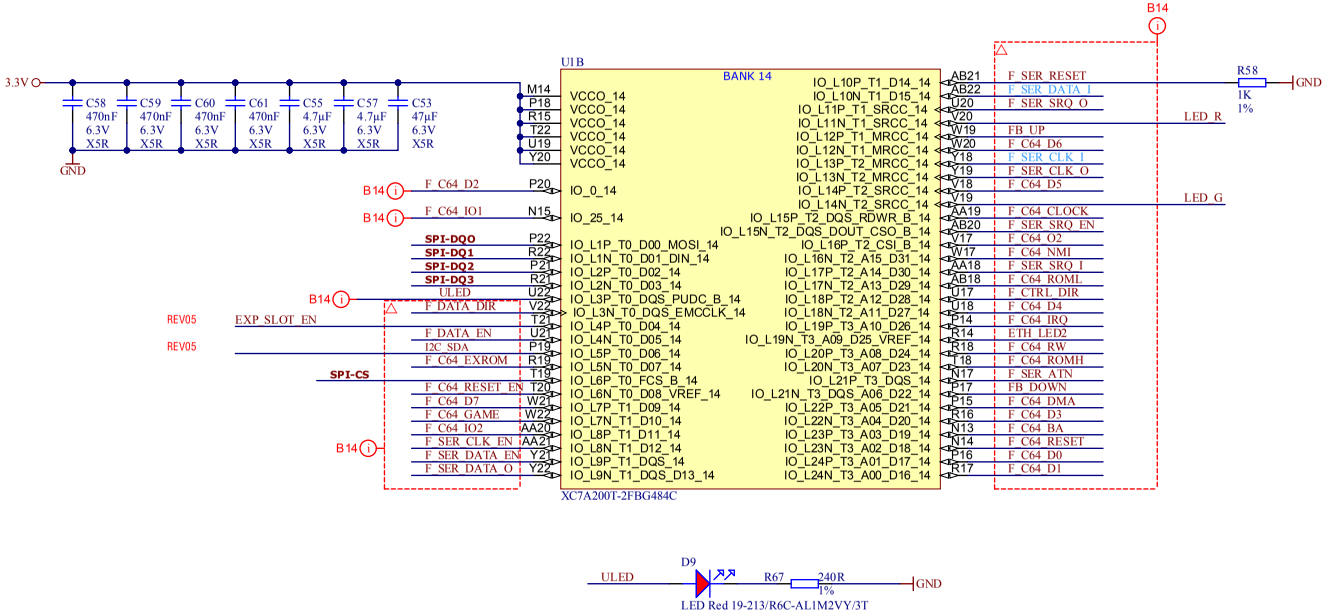


Title:	MEGA65	This design is licensed under CERN-OHL-Pv2.0 License https://cern-ohl.web.cern.ch/home	
A3	Nummer: TE0765 T001C		Rev: 06
Datum: 14.12.2023	Copyright:Trenz Electronic GmbH		Page 5 of 23
Filename: EXP_Slot.SchDoc			

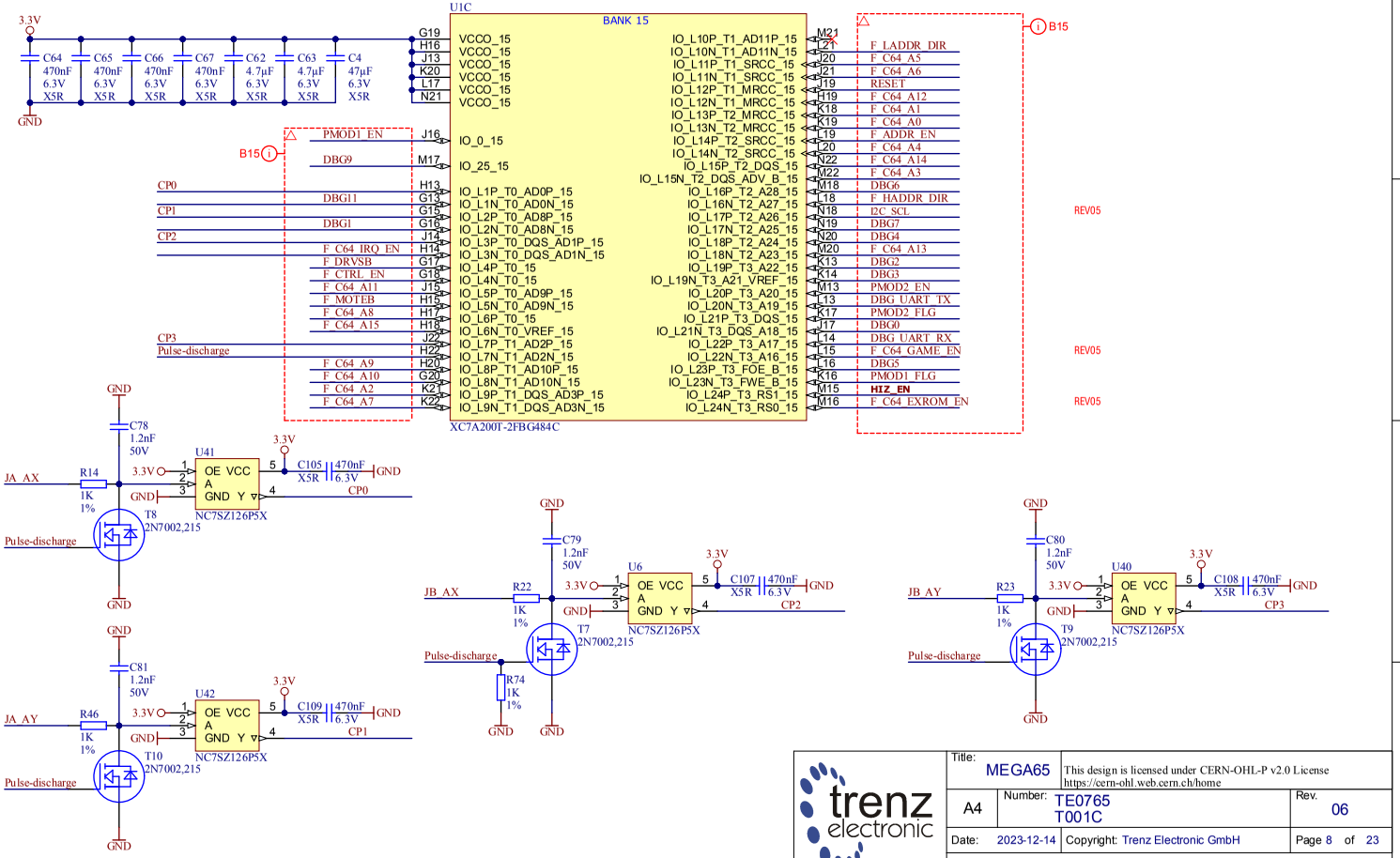


trenz electronic

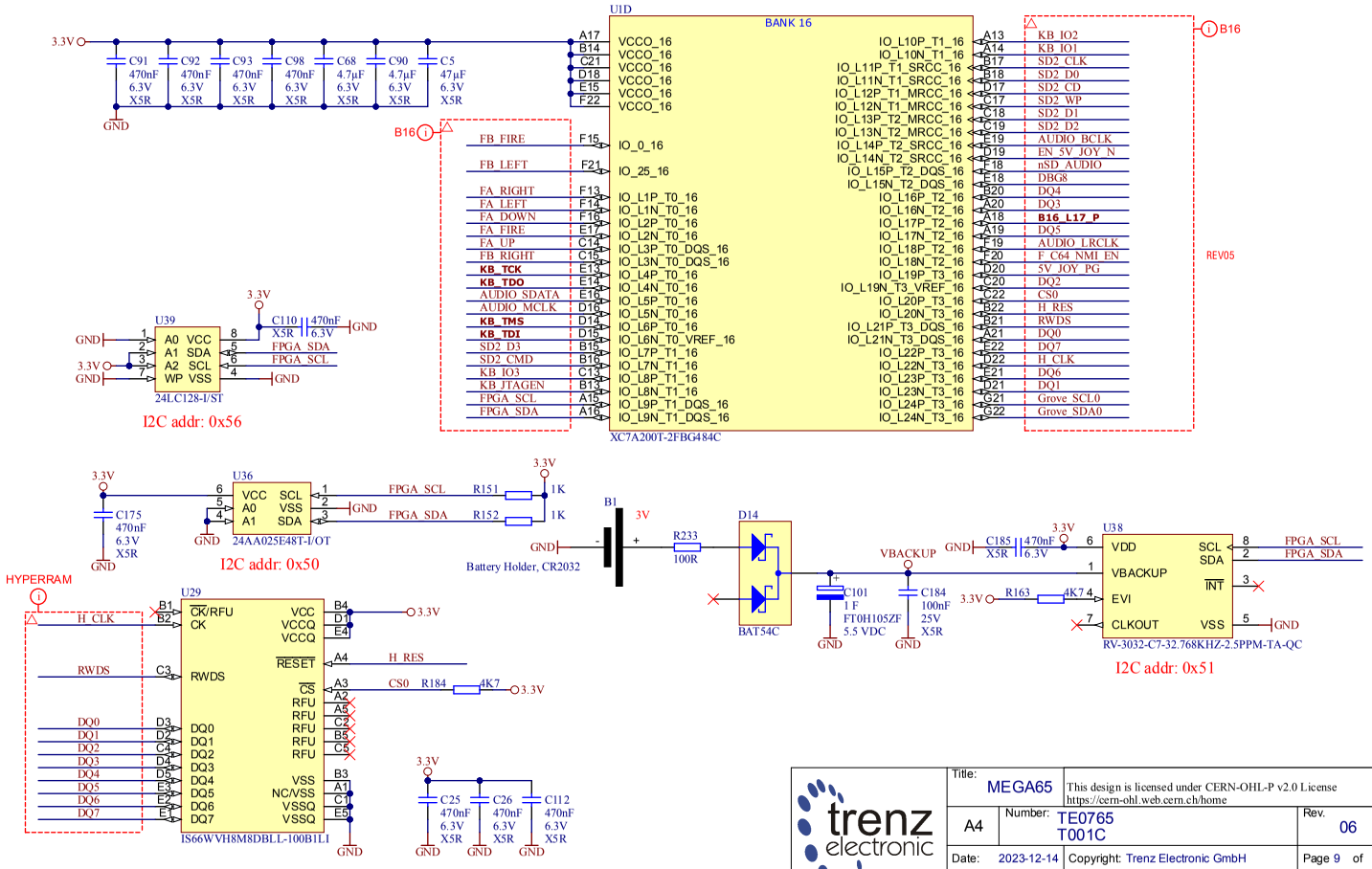
Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home		
A4	Number: TE0765 T001C		Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH		Page 6 of 23
Filename: B13.SchDoc			



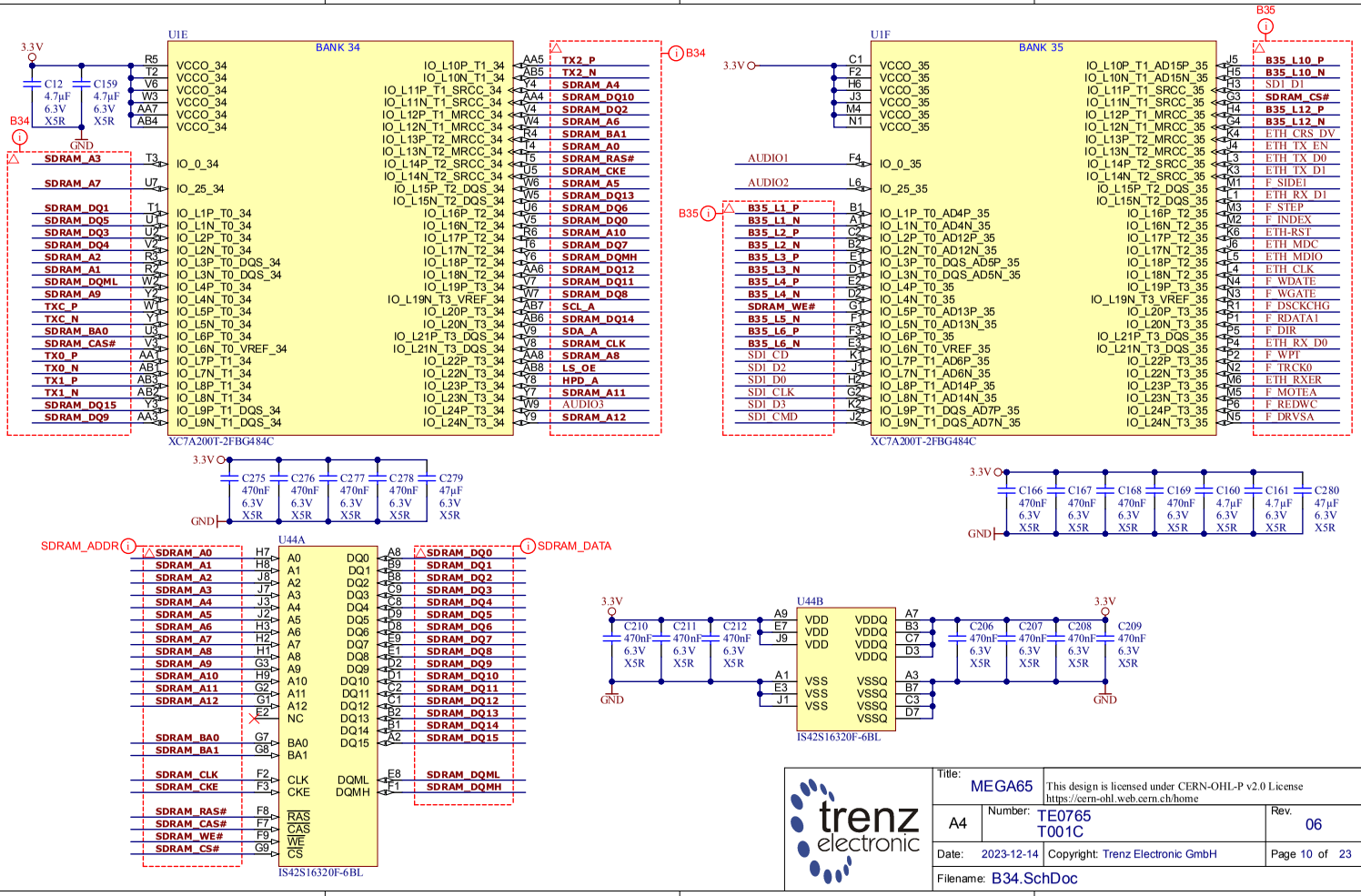
trenz electronic	Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
	A4	Number: TE0765 T001C	Rev. 06
	Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 7 of 23
	Filename: B14.SchDoc		

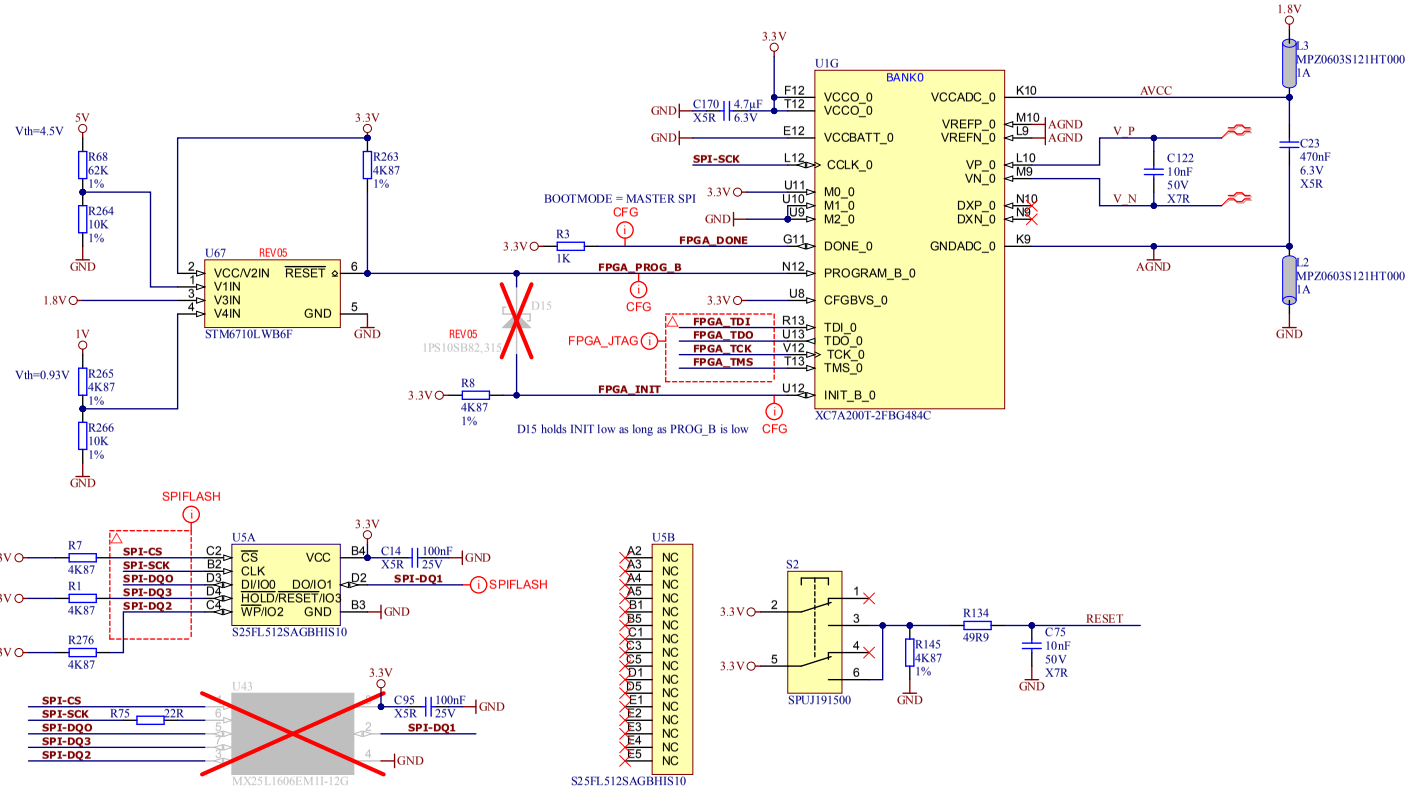


Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home		
A4	Number: TE0765 T001C		Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH		Page 8 of 23
Filename: B15.SchDoc			

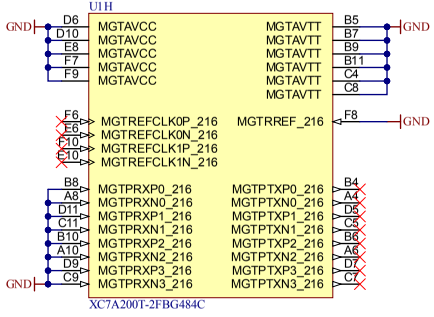


Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home		
A4	Number: TE0765 T001C		Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH		Page 9 of 23
Filename: B16.SchDoc			


U1E
BANK 34
XC7A200T-2FBG484C
U1F
BANK 35
XC7A200T-2FBG484C
B34
B35
3.3V
GND
U44A
IS42S16320F-6BL
U44B
IS42S16320F-6BL
SDRAM_ADDR
SDRAM_DATA
trenz electronic
MEGA65
This design is licensed under CERN-OHL-P v2.0 License
https://cern-ohl.web.cern.ch/home
A4
TE0765
T001C
06
2023-12-14
Copyright: Trenz Electronic GmbH
Page 10 of 23
B34.SchDoc

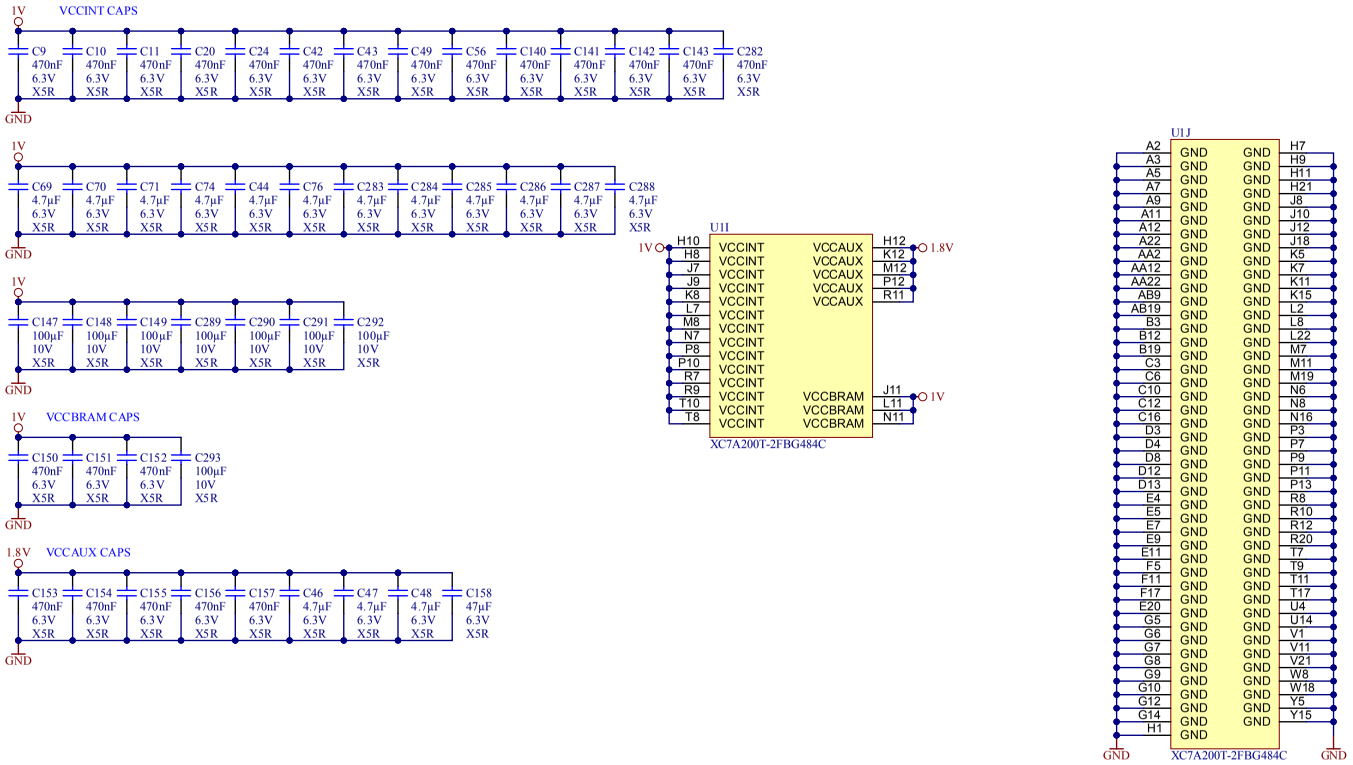


trenz electronic	Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
	A4	Number: TE0765 T001C	Rev. 06
	Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 11 of 23
	Filename: FPGA-CFG.SchDoc		



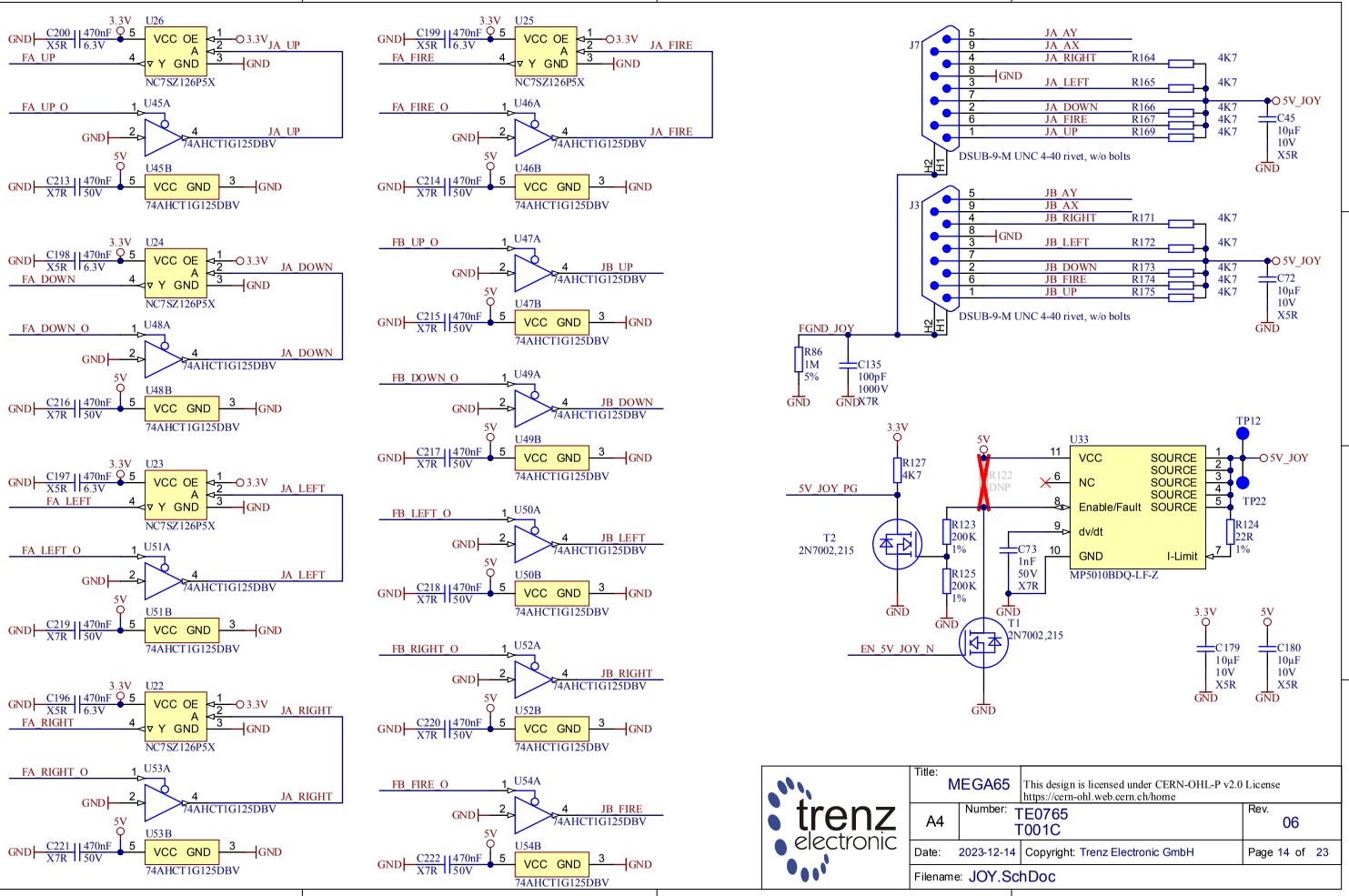
trenz electronic

Title: MEGA65		This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
A4	Number: TE0765 T001C		Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH		Page 12 of 23
Filename: FPGA-MGT.SchDoc			



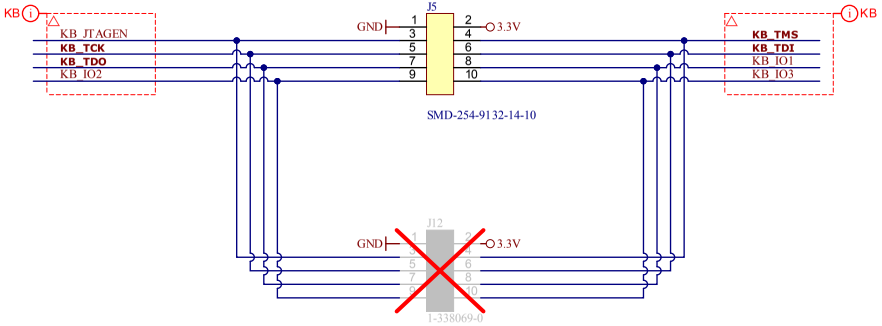
trenz electronic

Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
A4	Number: TE0765 T001C	Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 13 of 23
Filename: FPGA-PWR.SchDoc		

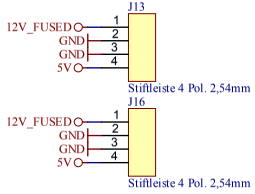


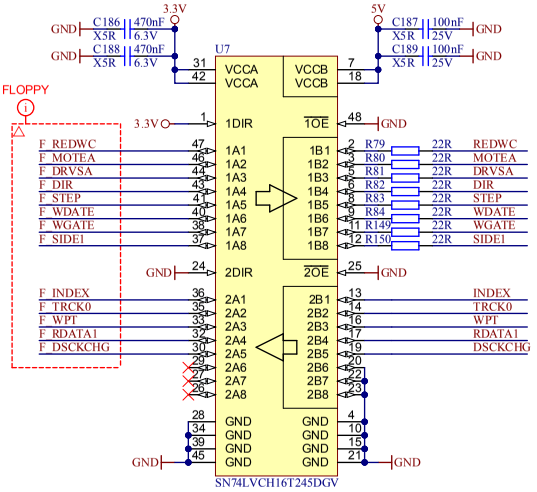
trenz electronic

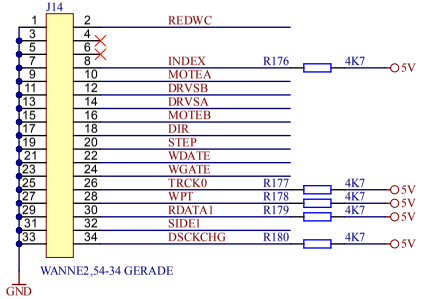
Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
A4	Number: TE0765 T001C	Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 14 of 23
Filename: JOY.SchDoc		



Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home		
A4	Number: TE0765 T001C		Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH		Page 15 of 23
Filename: KEYBOARD.SchDoc			

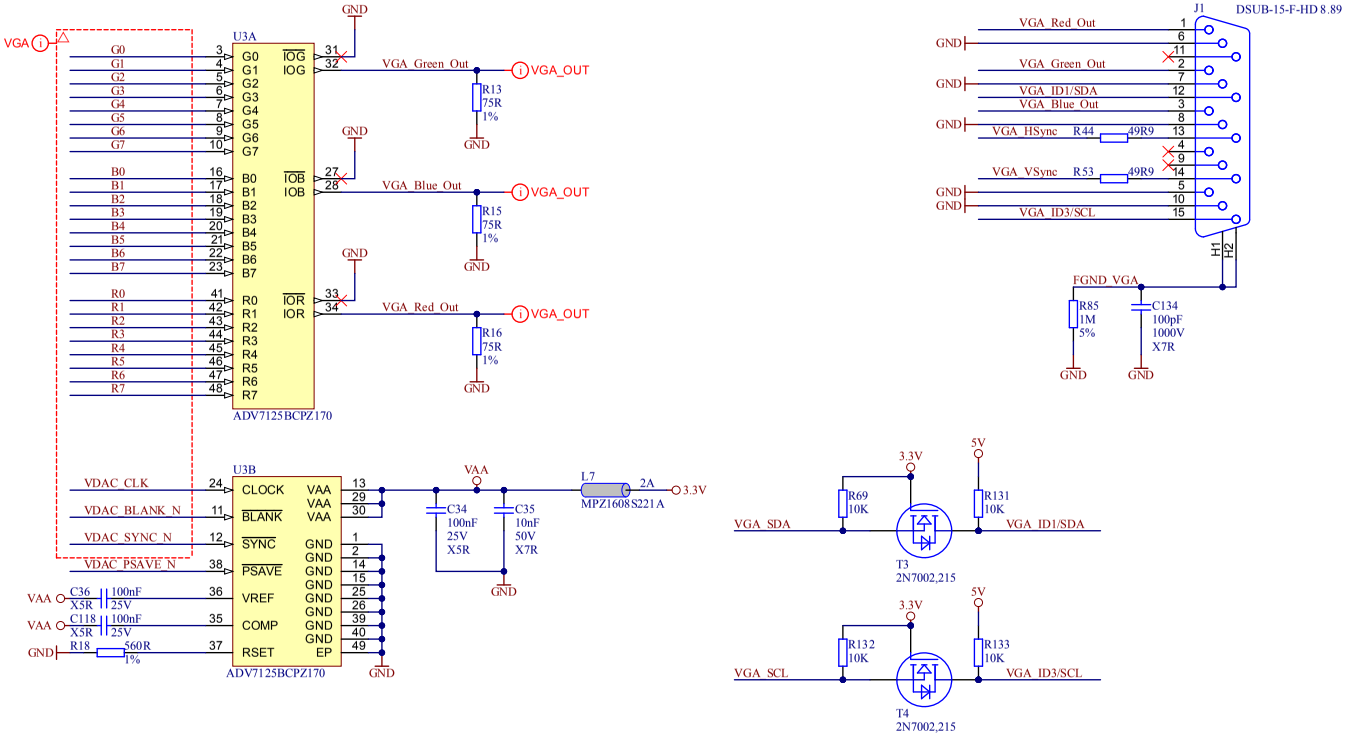




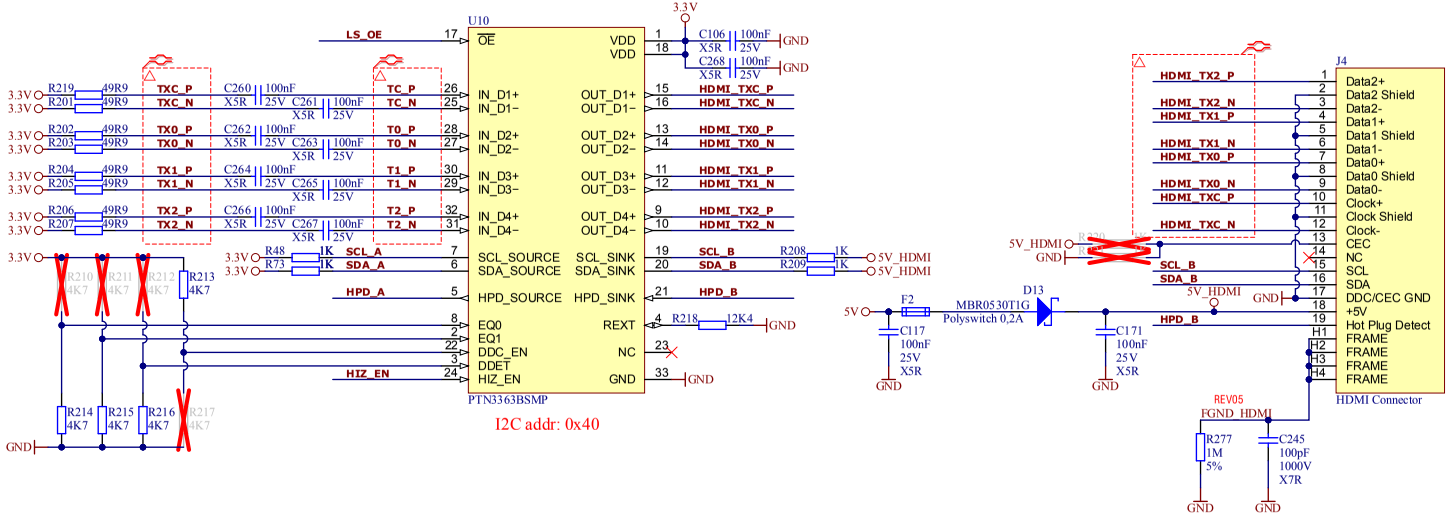


trenz electronic

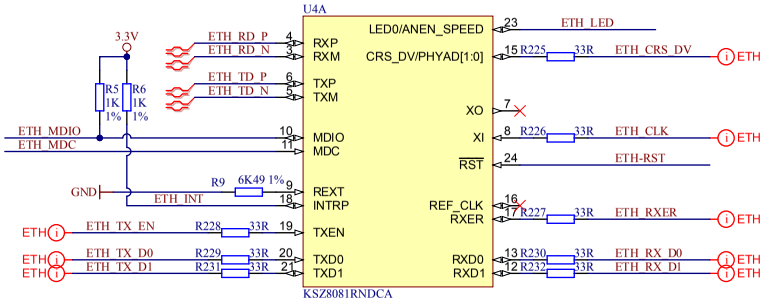
Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home		
A4	Number: TE0765 T001C		Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH		Page 16 of 23
Filename: Floppy.SchDoc			

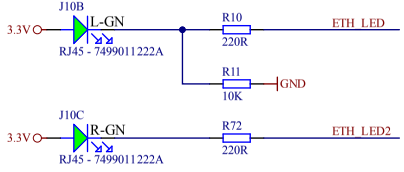


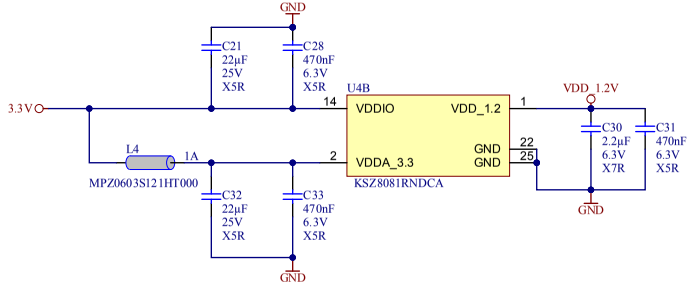
trenz electronic	Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
	A4	Number: TE0765 T001C	Rev. 06
	Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 17 of 23
	Filename: VGA.SchDoc		

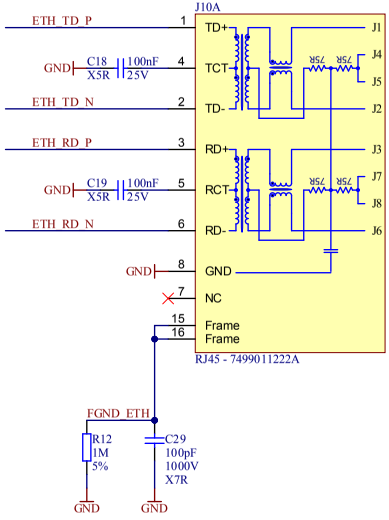


trenz electronic	Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
	A4	Number: TE0765 T001C	Rev. 06
	Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 18 of 23
	Filename: HDMI.SchDoc		



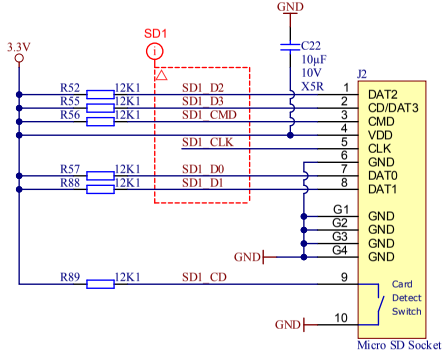


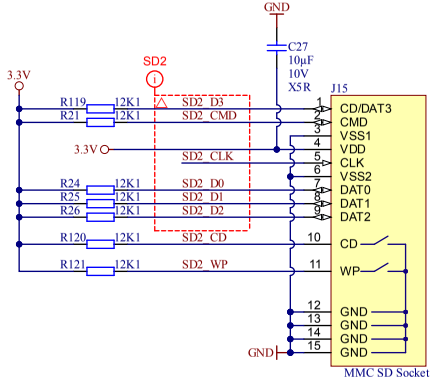


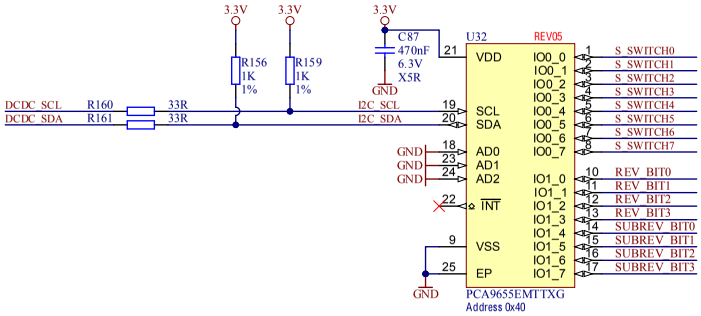


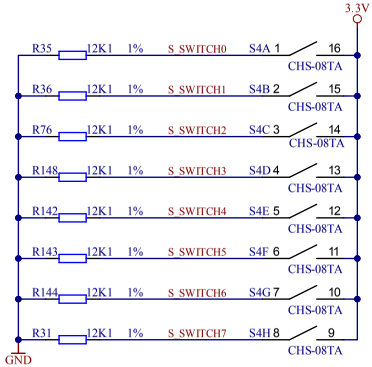
trenz electronic

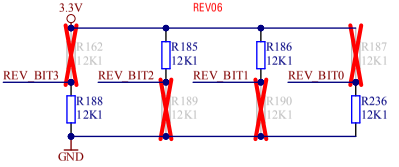
Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home		
A4	Number: TE0765 T001C		Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH		Page 19 of 23
Filename: Ethernet.SchDoc			



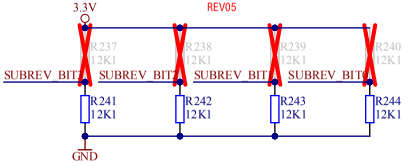








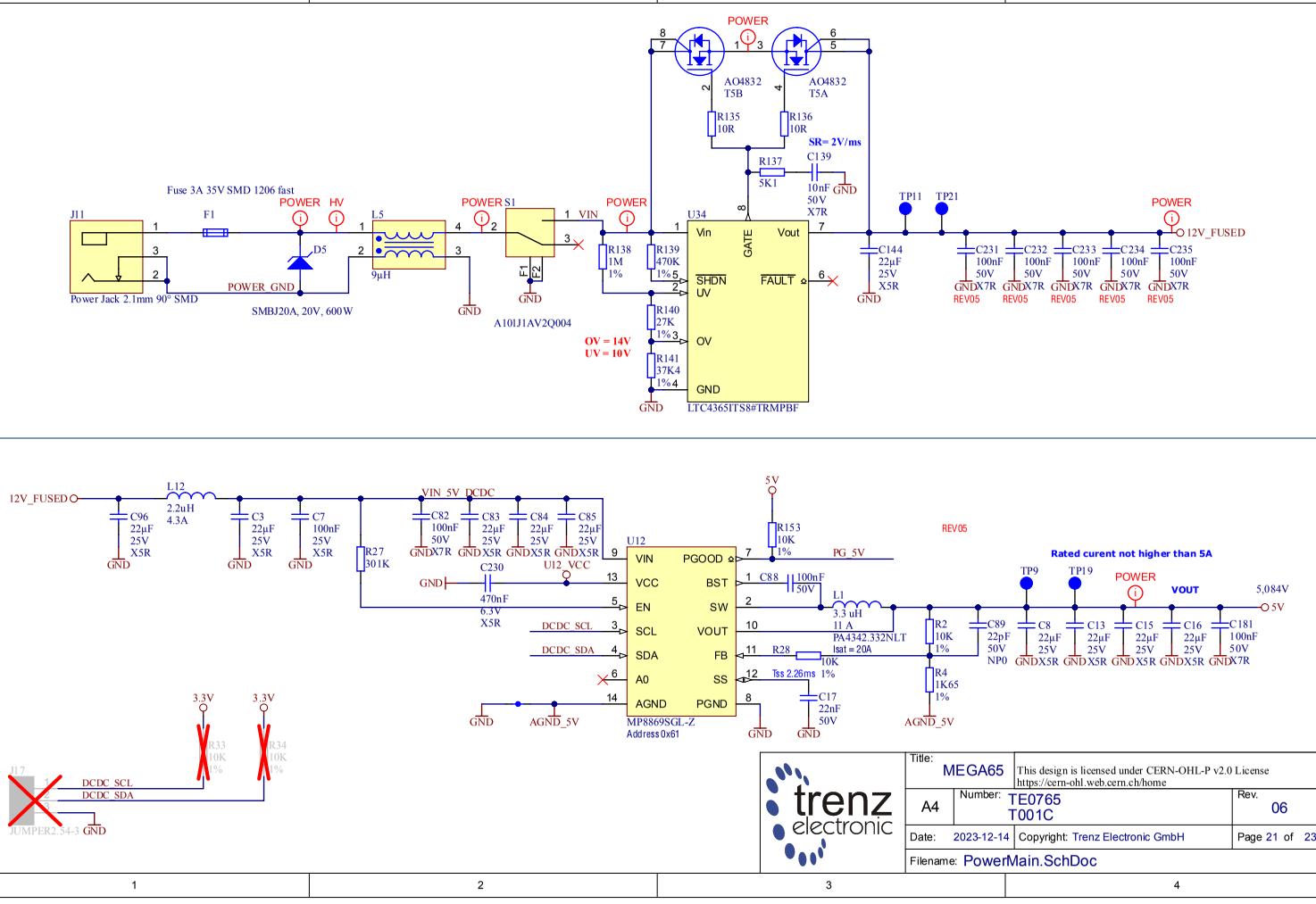
Revision	REV_BIT3	REV_BIT2	REV_BIT1	REV_BIT0
05	0	1	0	1
06	0	1	1	0



SUBREV	SUBREV_BIT3	SUBREV_BIT2	SUBREV_BIT1	SUBREV_BIT0
00	0	0	0	0

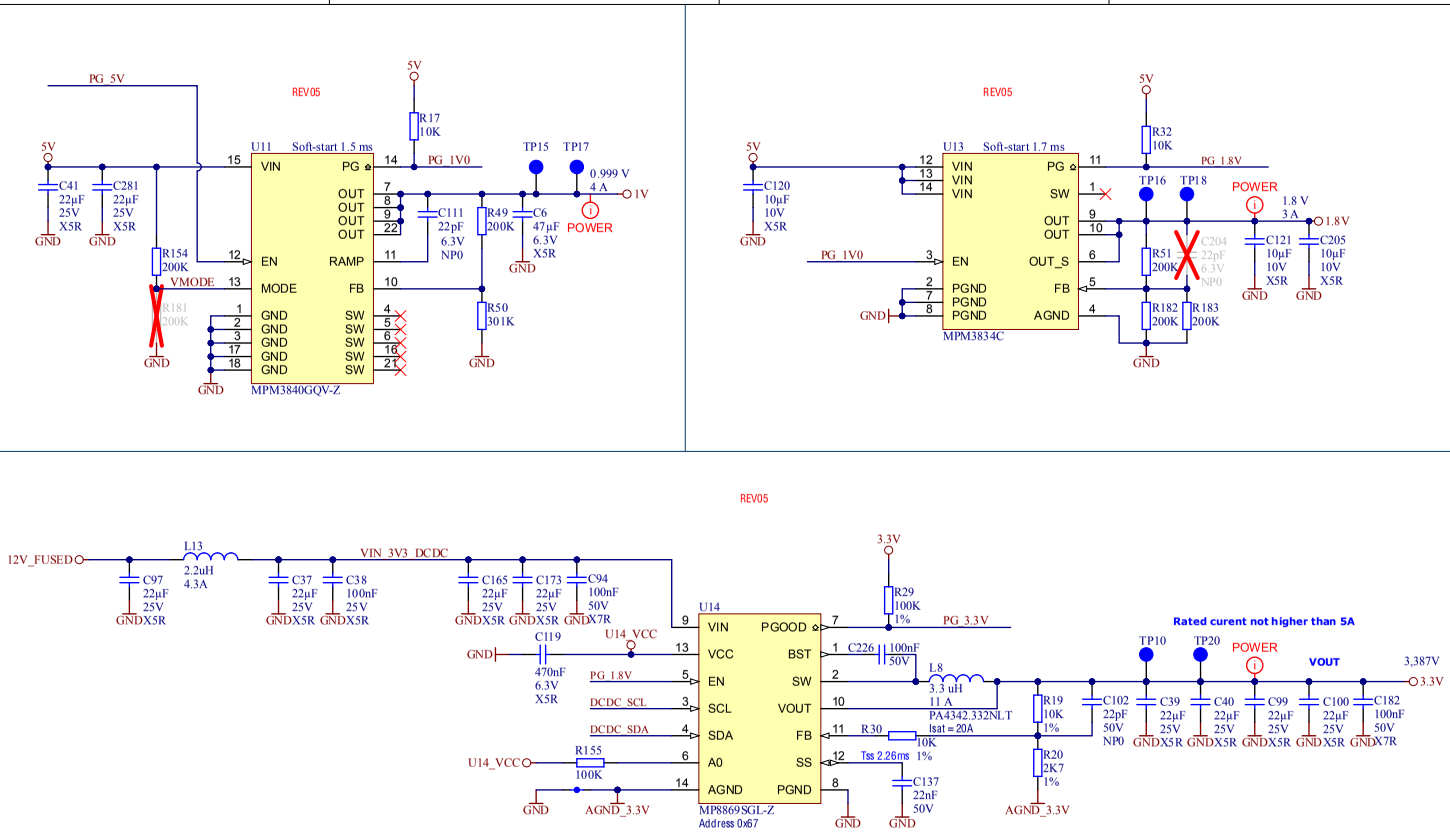
trenz electronic

Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
A4	Number: TE0765 T001C	Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 20 of 23
Filename: LED_SW_BUT.SchDoc		



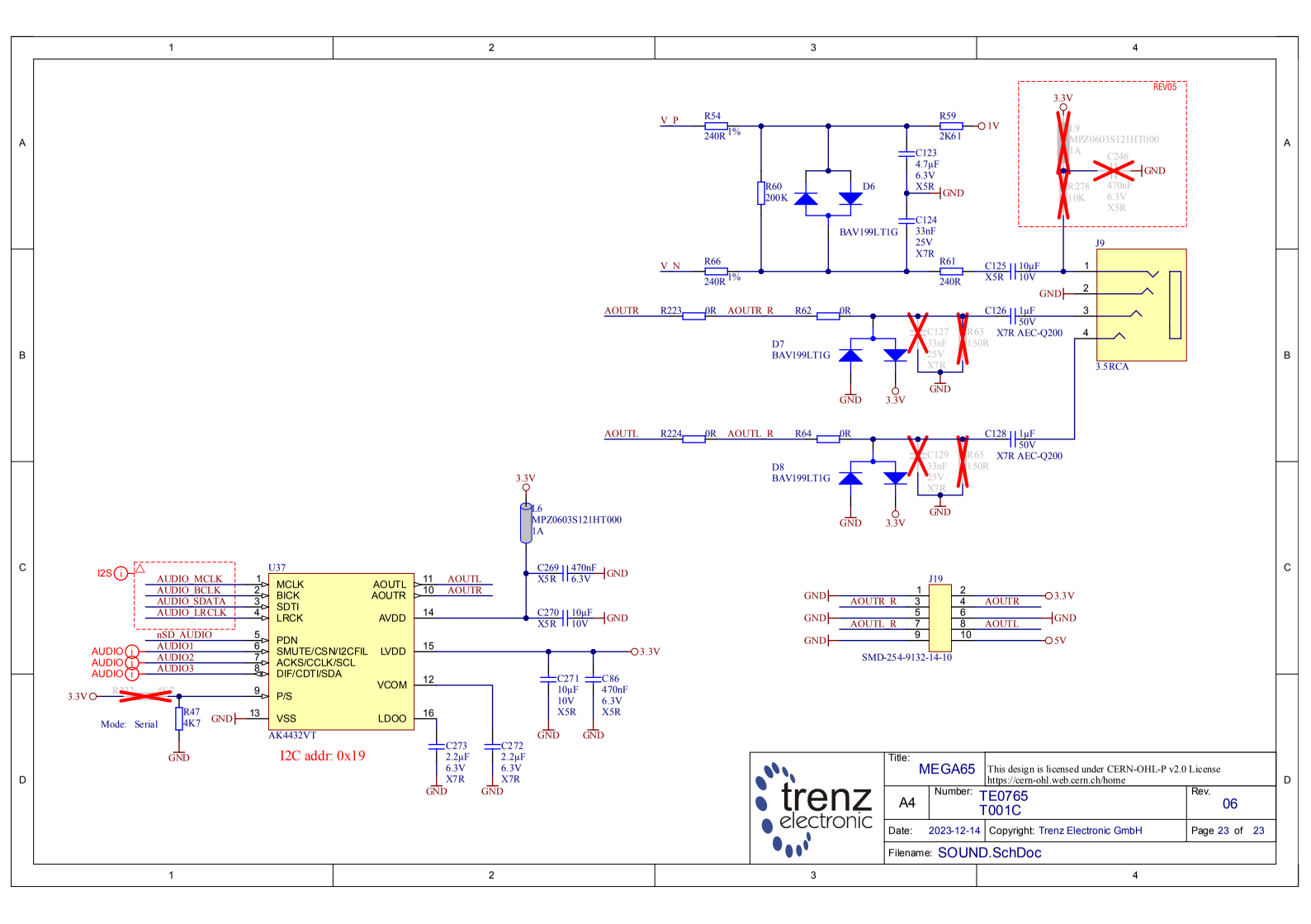
trenz electronic

Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
A4	Number: TE0765 T001C	Rev. 06
Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 21 of 23
Filename: PowerMain.SchDoc		

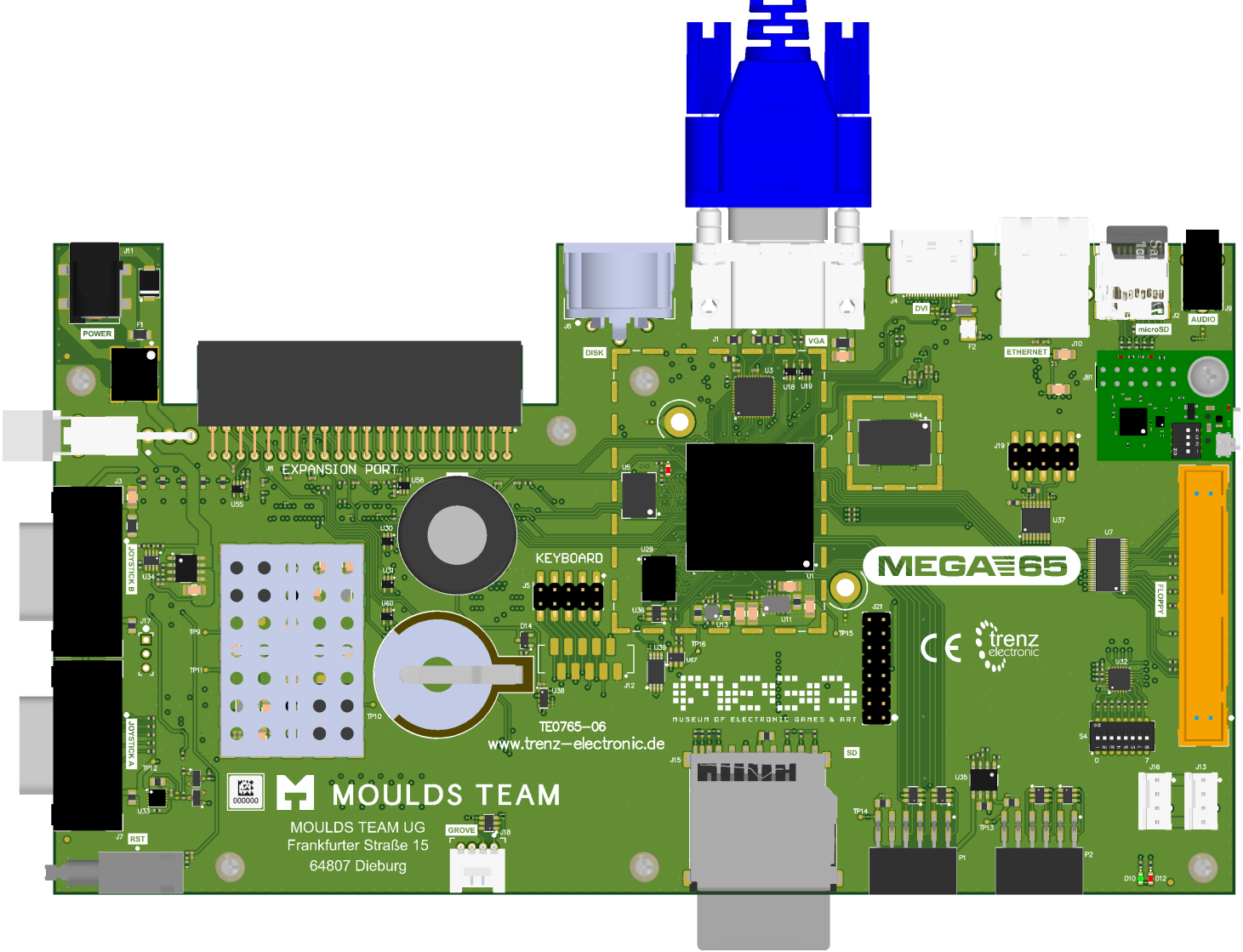


trenz electronic

Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home		
A4	Number: TE0765 T001C	Rev. 06	
Date: 2023-12-14	Copyright: Trenz Electronic GmbH / TT	Page 22 of 23	
Filename: POWER.SchDoc			



trenz electronic	Title: MEGA65	This design is licensed under CERN-OHL-P v2.0 License https://cern-ohl.web.cern.ch/home	
	A4	Number: TE0765 T001C	Rev. 06
	Date: 2023-12-14	Copyright: Trenz Electronic GmbH	Page 23 of 23
	Filename: SOUND.SchDoc		


POWER
DISK
VGA
DVI
ETHERNET
microSD
AUDIO
EXPANSION PORT
KEYBOARD
JOYSTICK B
JOYSTICK A
RST
GROVE
FLOPPY
SD
MEGA65
TE0765-06
www.trenz-electronic.de
MOULDS TEAM
MOULDS TEAM UG
Frankfurter Straße 15
64807 Dieburg
trenz electronic
MUSEUM OF ELECTRONIC GAMES & ART

MEGA65 R3 ELEKTRONISCH SCHEMA

U_HDMI
HDMI.SchDoc

U_Ethernet
Ethernet.SchDoc

U_LED_SW_BUT
LED_SW_BUT.SchDoc

U_SYS_MAX10_CTRL
SYS_MAX10_CTRL.SchDoc

U_SOUND
SOUND.SchDoc

U_B13
B13.SchDoc

U_B14
B14.SchDoc

U_B15
B15.SchDoc

U_B16
B16.SchDoc

U_B34
B34.SchDoc

U_FPGA-CFG
FPGA-CFG.SchDoc

U_FPGA-MGT
FPGA-MGT.SchDoc

U_FPGA-PWR
FPGA-PWR.SchDoc

U_JOY
JOY.SchDoc

U_KEYBOARD
KEYBOARD.SchDoc

U_POWER
POWER.SchDoc

U_PowerMain
PowerMain.SchDoc

U_VGA
VGA.SchDoc

U_EXT_HEADER
EXT_HEADER.SchDoc

U_EXP_Slot
EXP_Slot.SchDoc

U_Floppy
Floppy.SchDoc

PM1 FIDU-DOT - small
PM2 FIDU-DOT - small
PM3 FIDU-DOT - small
PM4 FIDU-DOT - small
PM5 FIDU-DOT - small
PM6 FIDU-DOT - small

Serial1
Serial
Serialnumber 6,3 x 6.3mm

LOGO1
TE Logo PRINT Layer
LOGO PRINT

GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm

Design drawn by:	IG
Checked by:	MR
Assembly variant:	[No Variations]
Created by:	VariantCreatedBy
Modified by:	VariantModifiedBy
Modified at:	VariantDateModification

trenz electronic

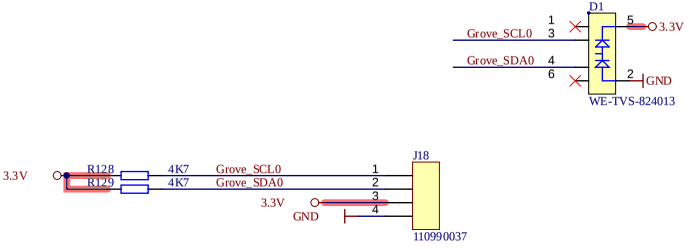
Title: MEGA65

A4 | Number: TE0765 [No Variations] | Rev. 03

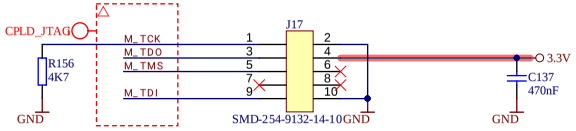
Date: 2020-06-29 | Copyright: Trenz Electronic GmbH | Page 1 of 23

Filename: TE0765.SchDoc

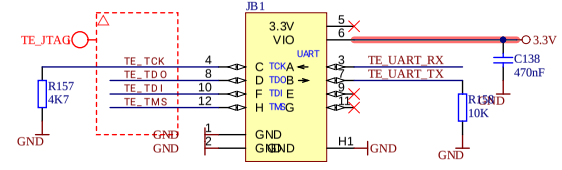
Grove CONN



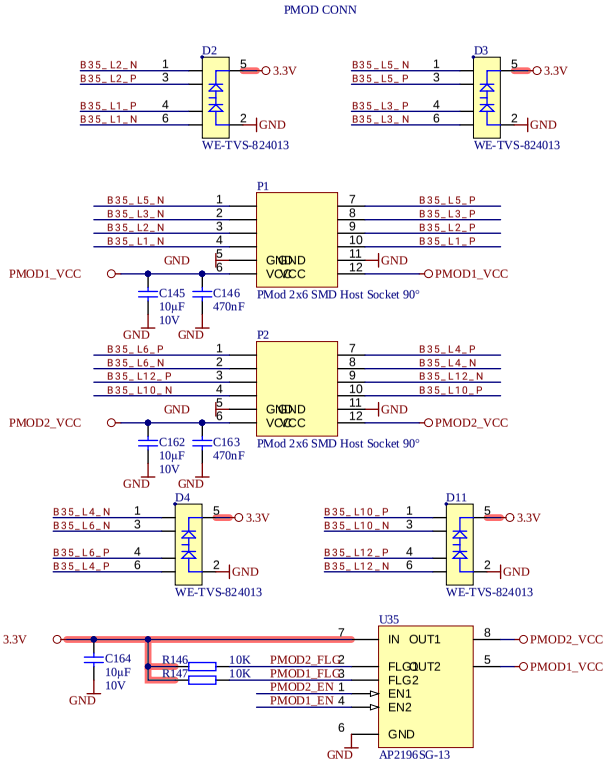
CPLD CONFIG



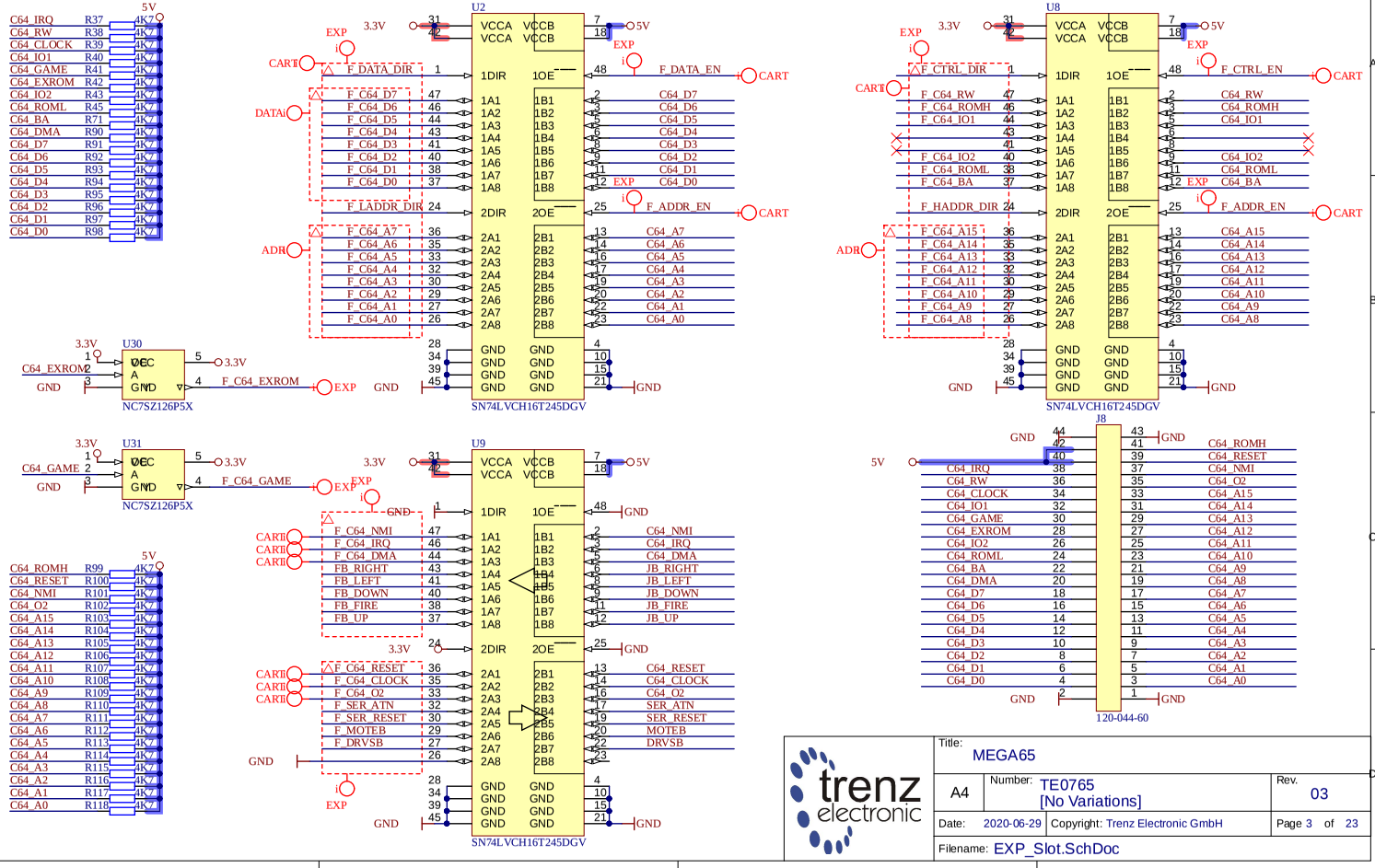
TE0790-Base SMT

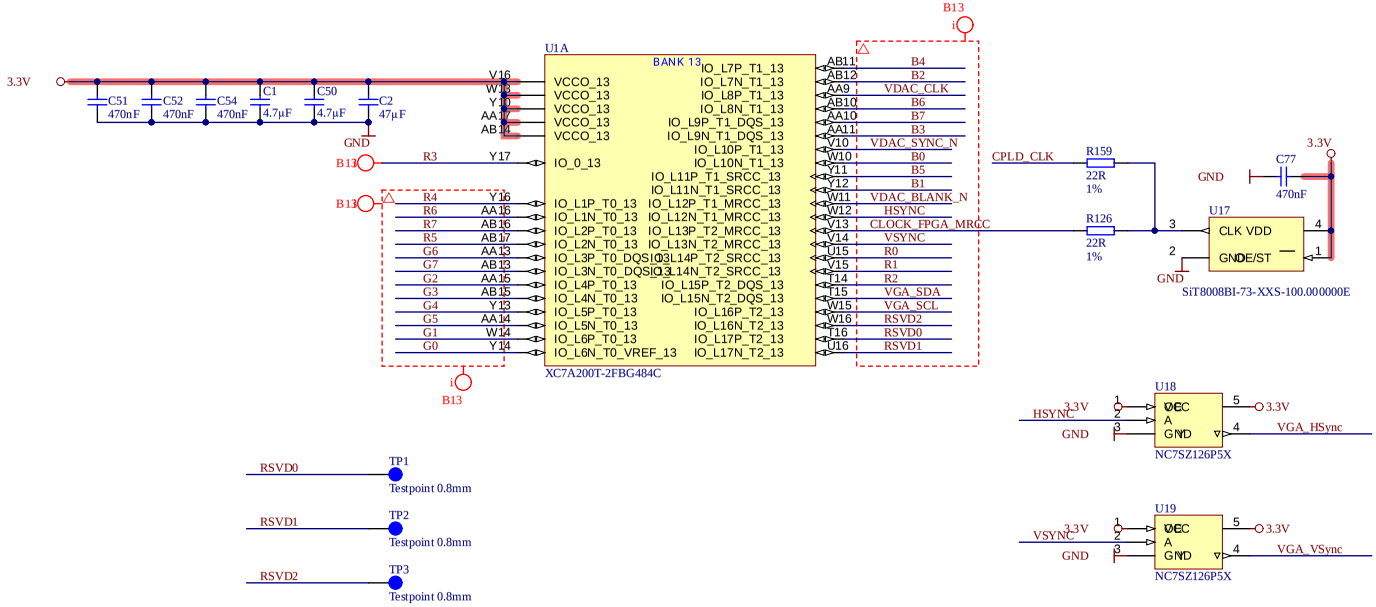


PMOD CONN

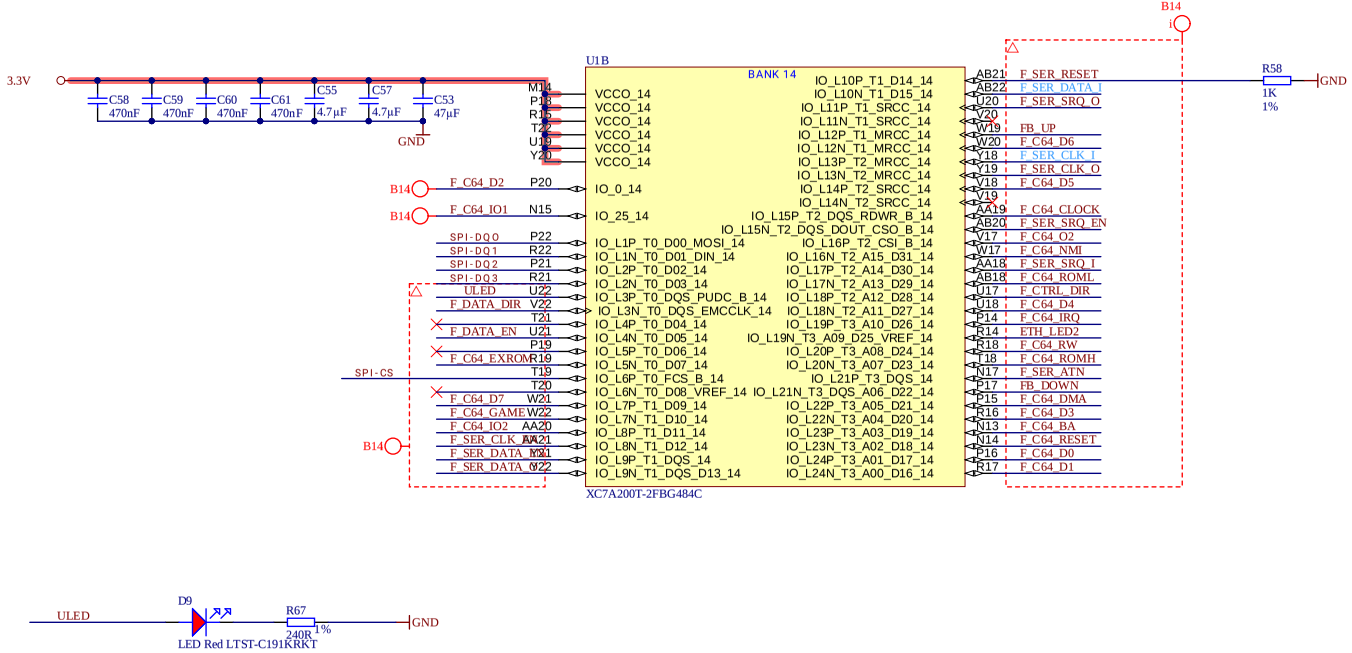


Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 2 of 23
Filename: EXT_HEADER.SchDoc			

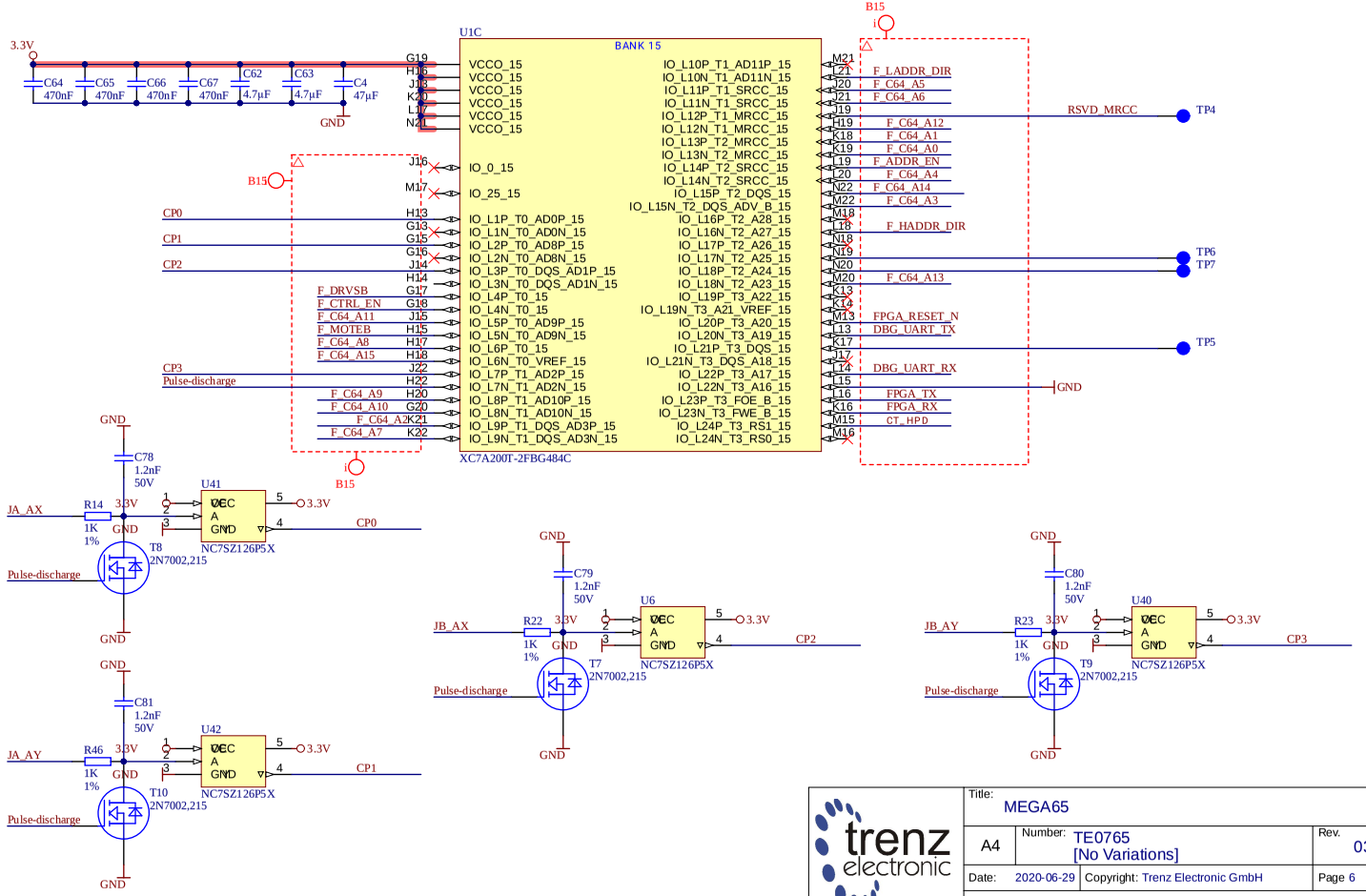

C64_IRQ R37 4K7
C64_RW R38 4K7
C64_CLOCK R39 4K7
C64_IO1 R40 4K7
C64_GAME R41 4K7
C64_EXROM R42 4K7
C64_IO2 R43 4K7
C64_ROML R45 4K7
C64_BA R71 4K7
C64_DMA R90 4K7
C64_D7 R91 4K7
C64_D6 R92 4K7
C64_D5 R93 4K7
C64_D4 R94 4K7
C64_D3 R95 4K7
C64_D2 R96 4K7
C64_D1 R97 4K7
C64_D0 R98 4K7
5V
U2
SN74LVCH16T245DGV
F_DATA_DIR
F_DATA_EN
F_LADDR_DIR
F_ADDR_EN
F_C64_D7
F_C64_D6
F_C64_D5
F_C64_D4
F_C64_D3
F_C64_D2
F_C64_D1
F_C64_D0
F_C64_A7
F_C64_A6
F_C64_A5
F_C64_A4
F_C64_A3
F_C64_A2
F_C64_A1
F_C64_A0
U30
NC7SZ126P5X
C64_EXROM
F_C64_EXROM
U31
C64_GAME
F_C64_GAME
C64_ROMH R99 4K7
C64_RESET R100 4K7
C64_NMI R101 4K7
C64_O2 R102 4K7
C64_A15 R103 4K7
C64_A14 R104 4K7
C64_A13 R105 4K7
C64_A12 R106 4K7
C64_A11 R107 4K7
C64_A10 R108 4K7
C64_A9 R109 4K7
C64_A8 R110 4K7
C64_A7 R111 4K7
C64_A6 R112 4K7
C64_A5 R113 4K7
C64_A4 R114 4K7
C64_A3 R115 4K7
C64_A2 R116 4K7
C64_A1 R117 4K7
C64_A0 R118 4K7
U9
F_C64_NMI
F_C64_IRQ
F_C64_DMA
FB_RIGHT
FB_LEFT
FB_DOWN
FB_FIRE
FB_UP
F_C64_RESET
F_C64_CLOCK
F_C64_O2
F_SER_ATN
F_SER_RESET
F_MOTEB
F_DRVSB
U8
F_CTRL_DIR
F_CTRL_EN
F_HADDR_DIR
F_C64_RW
F_C64_ROMH
F_C64_IO1
F_C64_IO2
F_C64_ROML
F_C64_BA
F_C64_A15
F_C64_A14
F_C64_A13
F_C64_A12
F_C64_A11
F_C64_A10
F_C64_A9
F_C64_A8
J8
120-044-60
Title: MEGA65
A4
Number: TE0765 [No Variations]
Rev. 03
Date: 2020-06-29
Copyright: Trenz Electronic GmbH
Page 3 of 23
Filename: EXP_Slot.SchDoc
trenz electronic



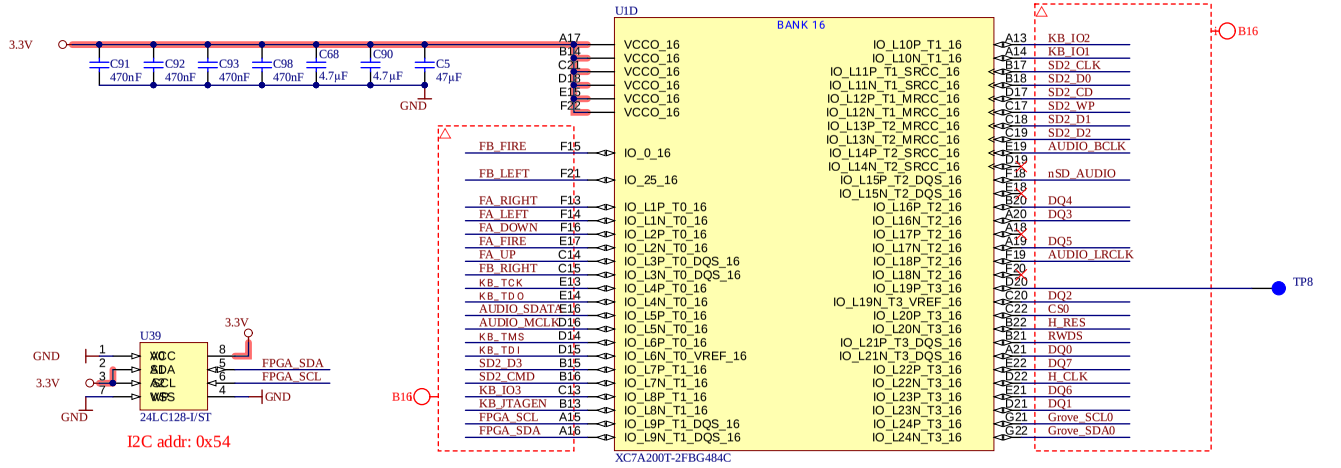
trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 4 of 23
	Filename: B13.SchDoc		



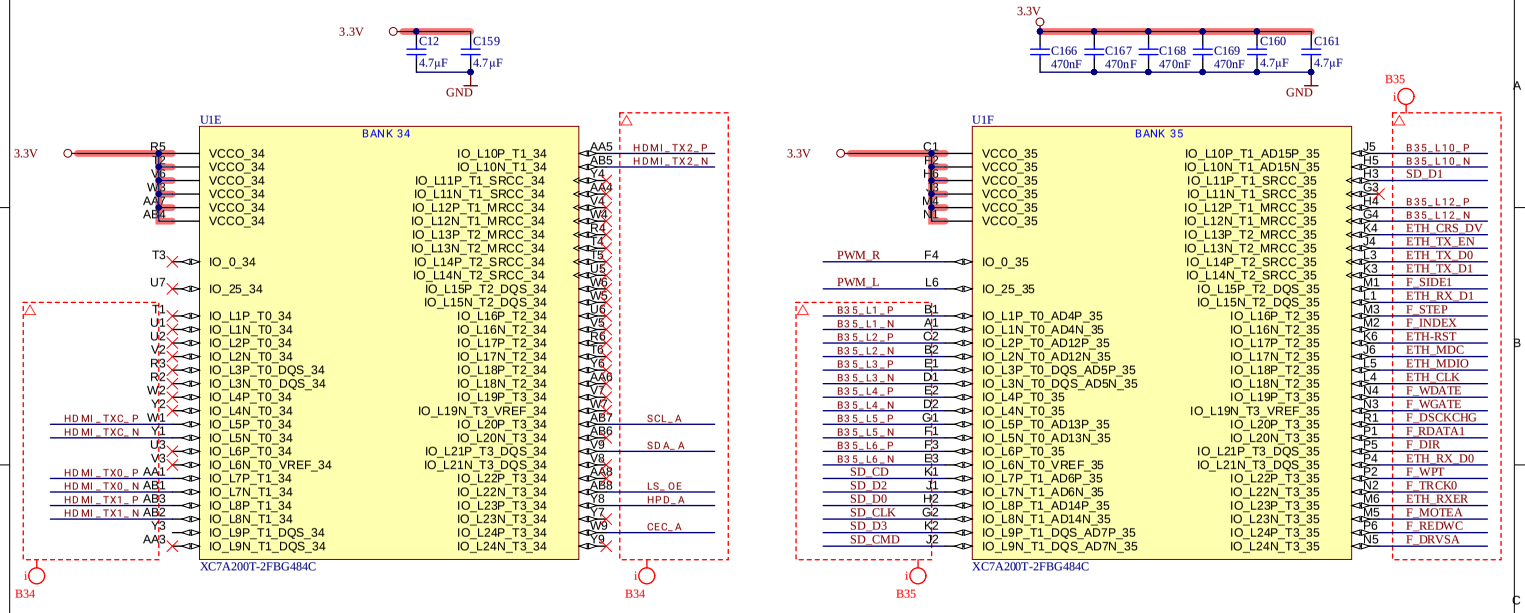
Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 5 of 23
Filename: B14.SchDoc			



Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 6 of 23
Filename: B15.SchDoc			

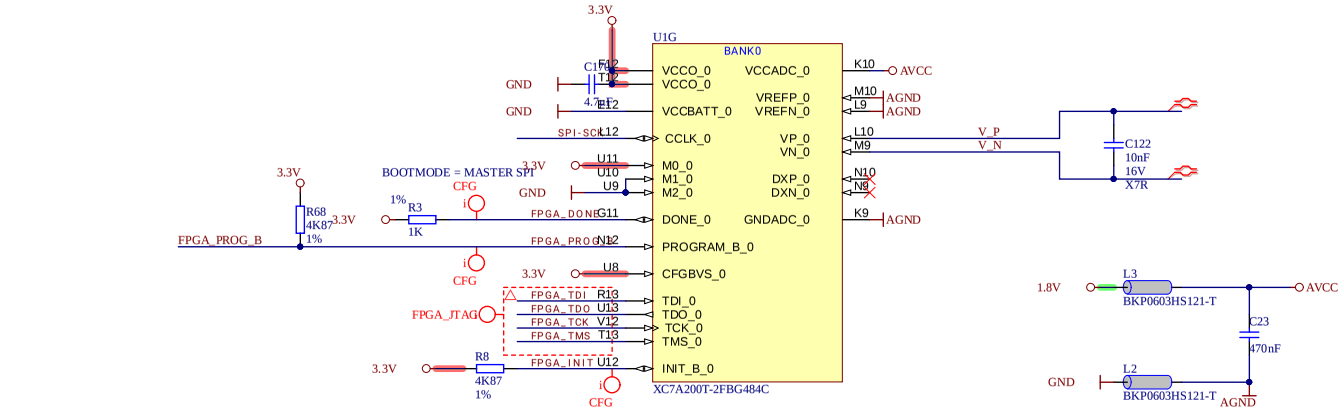


trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 7 of 23
	Filename: B16.SchDoc		

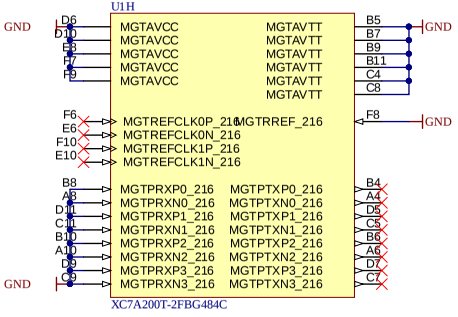


trenz electronic

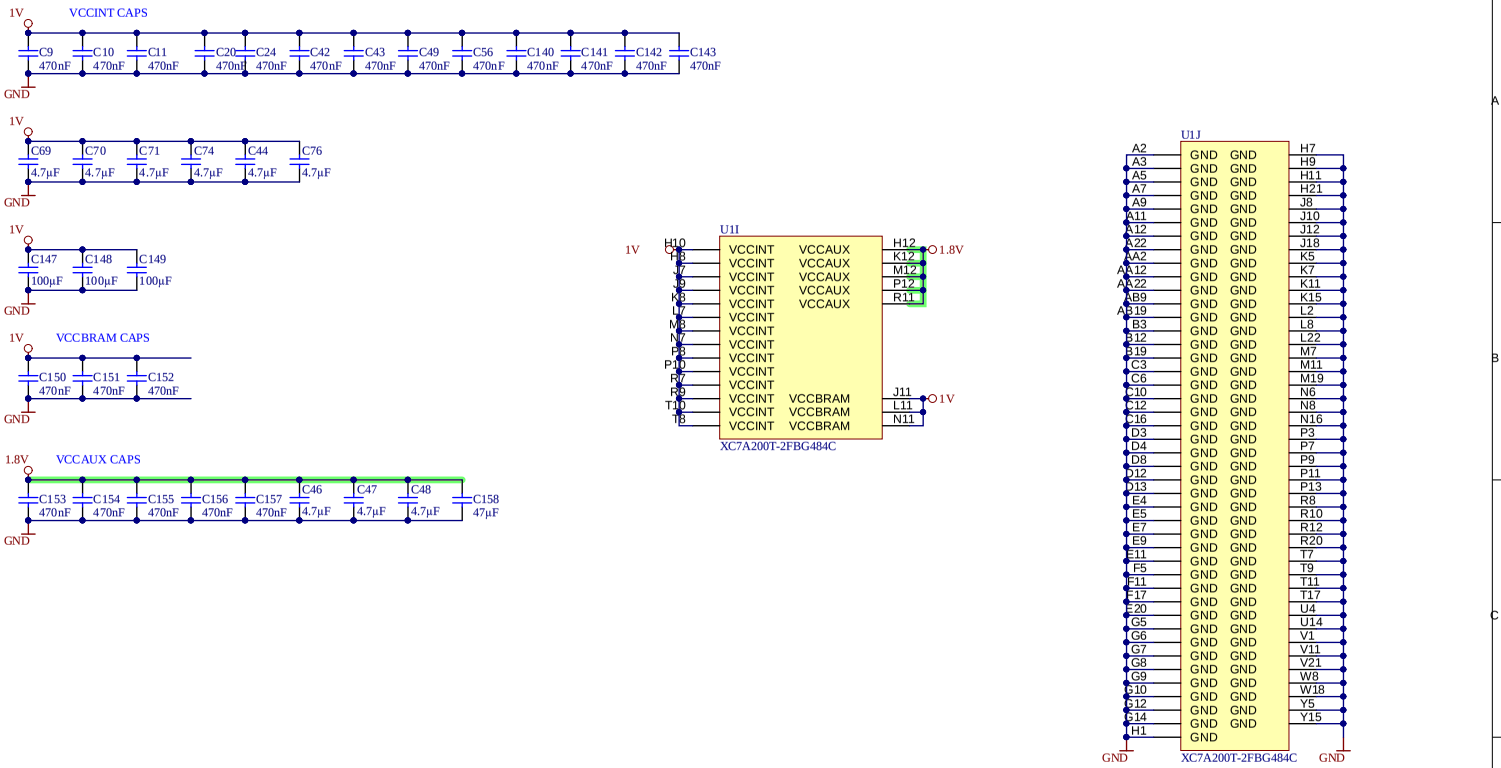
Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 8 of 23
Filename: B34.SchDoc			



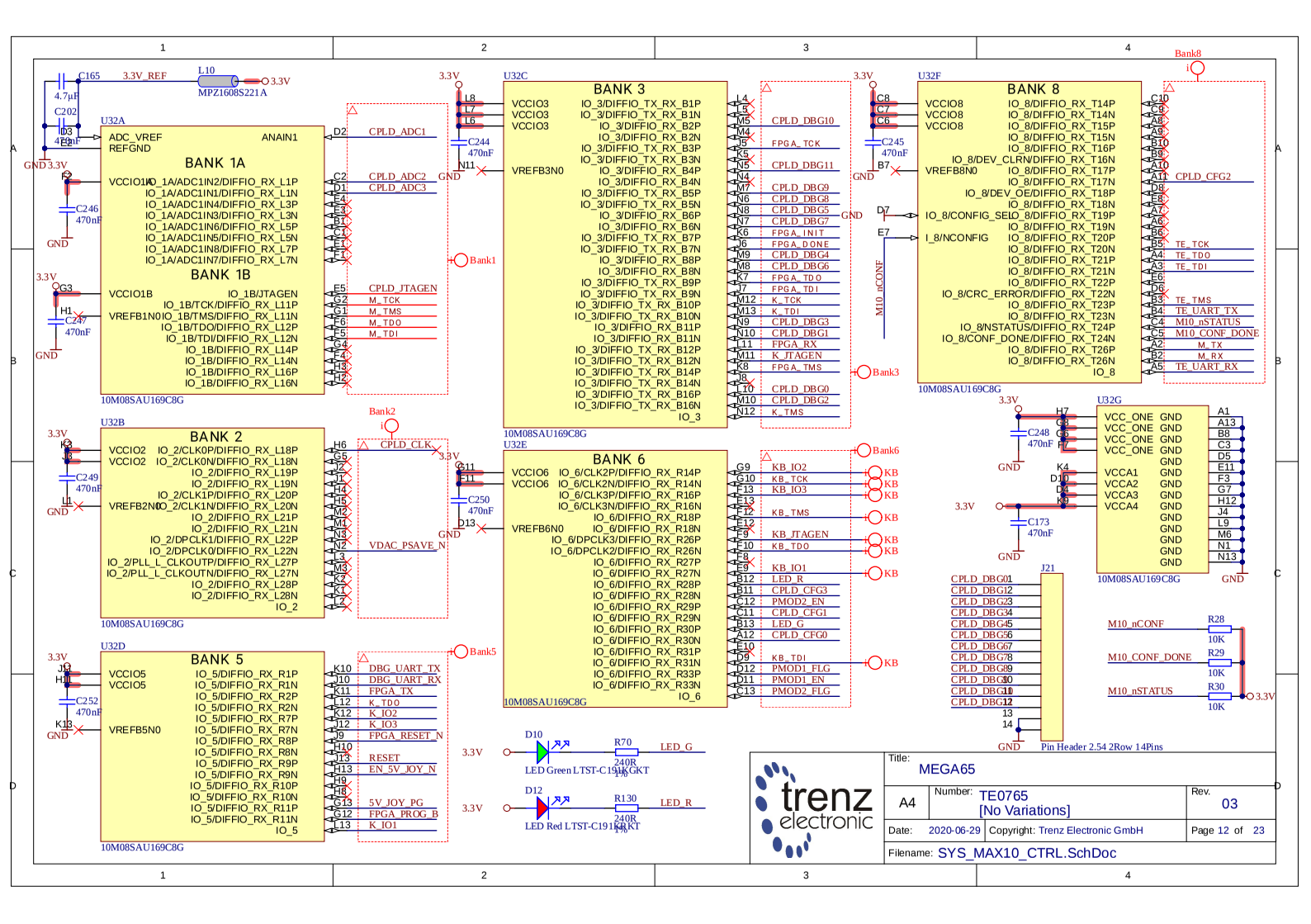
trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 9 of 23
	Filename: FPGA-CFG.SchDoc		

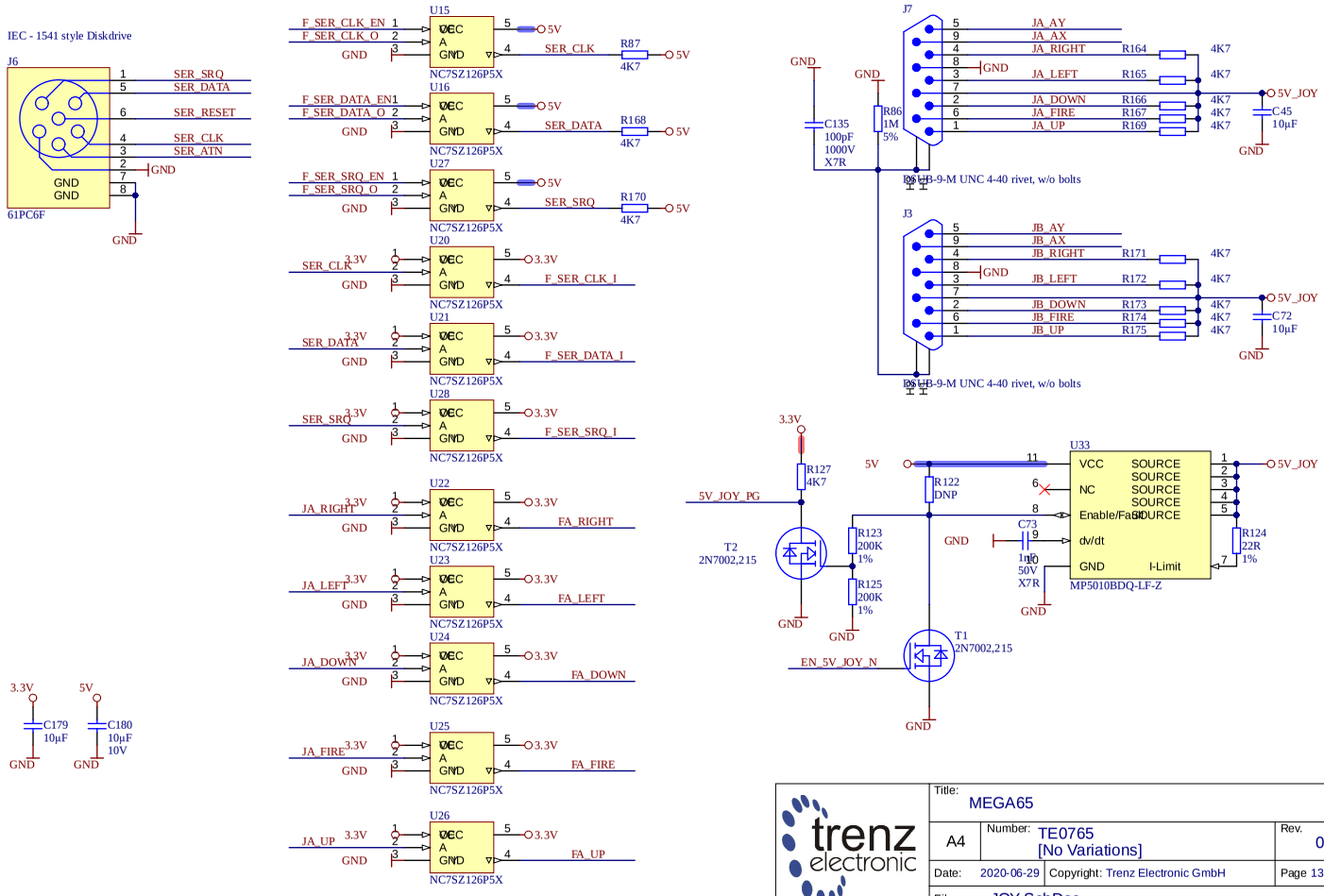


trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 10 of 23
	Filename: FPGA-MGT.SchDoc		

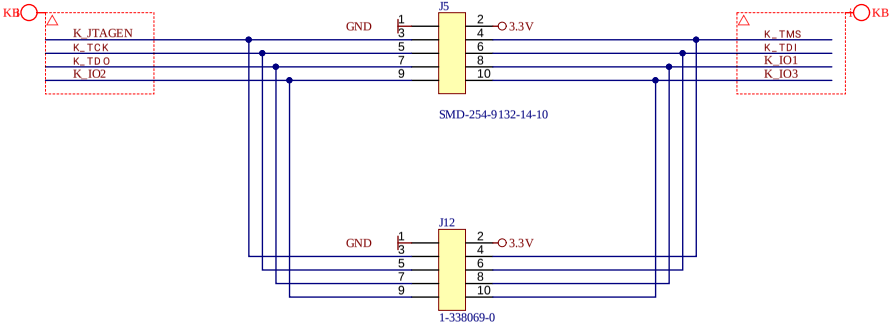


Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 11 of 23
Filename: FPGA-PWR.SchDoc			

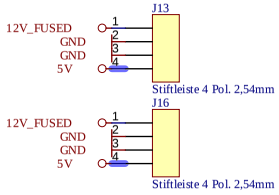

U32A
BANK 1A
BANK 1B
U32B
BANK 2
U32D
BANK 5
U32C
BANK 3
U32E
BANK 6
U32F
BANK 8
U32G
10M08SAU169C8G
3.3V
GND
Bank1
Bank2
Bank3
Bank5
Bank6
Bank8
CPLD_ADC1
CPLD_ADC2
CPLD_ADC3
CPLD_JTAGEN
M_TCK
M_TMS
M_TDO
M_TDI
CPLD_CLK
VDAC_PSAVE_N
DBG_UART_TX
DBG_UART_RX
FPGA_TX
K_TDO
K_IO2
K_IO3
FPGA_RESET_N
RESET
EN_5V_JOY_N
5V_JOY_PG
FPGA_PROG_B
K_IO1
CPLD_DBG10
FPGA_TCK
CPLD_DBG11
CPLD_DBG9
CPLD_DBG8
CPLD_DBG5
CPLD_DBG7
FPGA_INIT
FPGA_DONE
CPLD_DBG4
CPLD_DBG6
FPGA_TDO
FPGA_TDI
K_TCK
K_TDI
CPLD_DBG3
CPLD_DBG1
FPGA_RX
K_JTAGEN
FPGA_TMS
CPLD_DBG0
CPLD_DBG2
K_TMS
KB_IO2
KB_TCK
KB_IO3
KB_TMS
KB_JTAGEN
KB_TDO
KB_IO1
LED_R
CPLD_CFG3
PMOD2_EN
CPLD_CFG1
LED_G
CPLD_CFG0
KB_TDI
PMOD1_FLG
PMOD1_EN
PMOD2_FLG
M10_nCONF
CPLD_CFG2
TE_TCK
TE_TDO
TE_TDI
TE_TMS
TE_UART_TX
M10_nSTATUS
M10_CONF_DONE
M_TX
M_RX
TE_UART_RX
D10
LED Green LTST-C191KGKT
R70
240R
D12
LED Red LTST-C191KRKT
R130
240R
J21
Pin Header 2.54 2Row 14Pins
R28
R29
R30
10K
trenz electronic
Title: MEGA65
A4
Number: TE0765 [No Variations]
Rev. 03
Date: 2020-06-29
Copyright: Trenz Electronic GmbH
Page 12 of 23
Filename: SYS_MAX10_CTRL.SchDoc

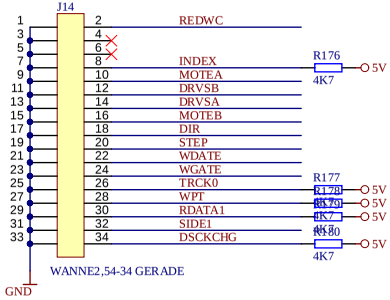


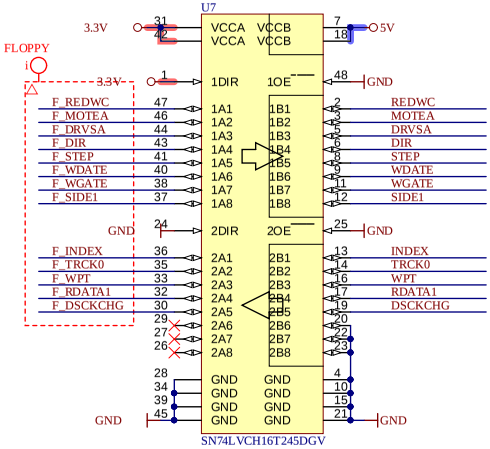
Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 13 of 23
Filename: JOY.SchDoc			



trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 14 of 23
	Filename: KEYBOARD.SchDoc		

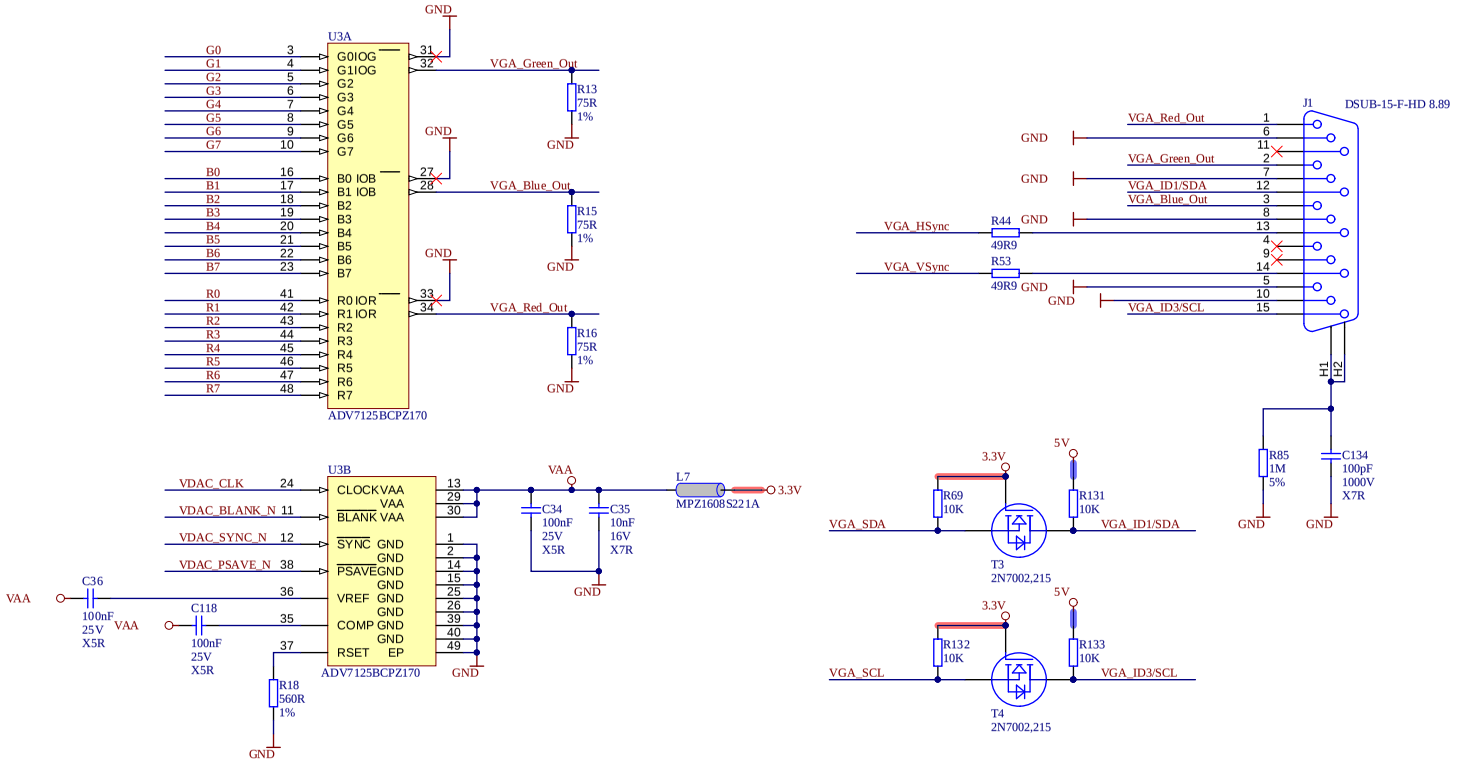




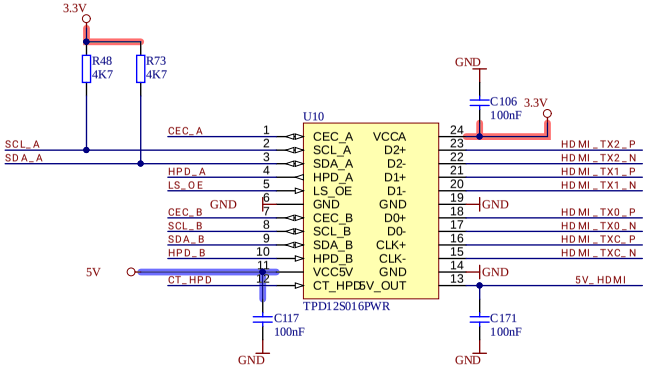


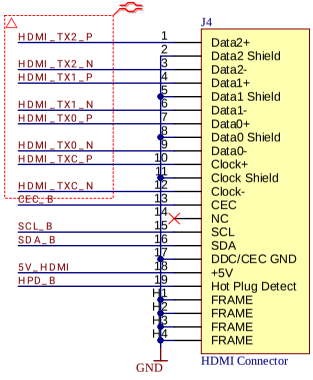


Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 15 of 23
Filename: Floppy.SchDoc			

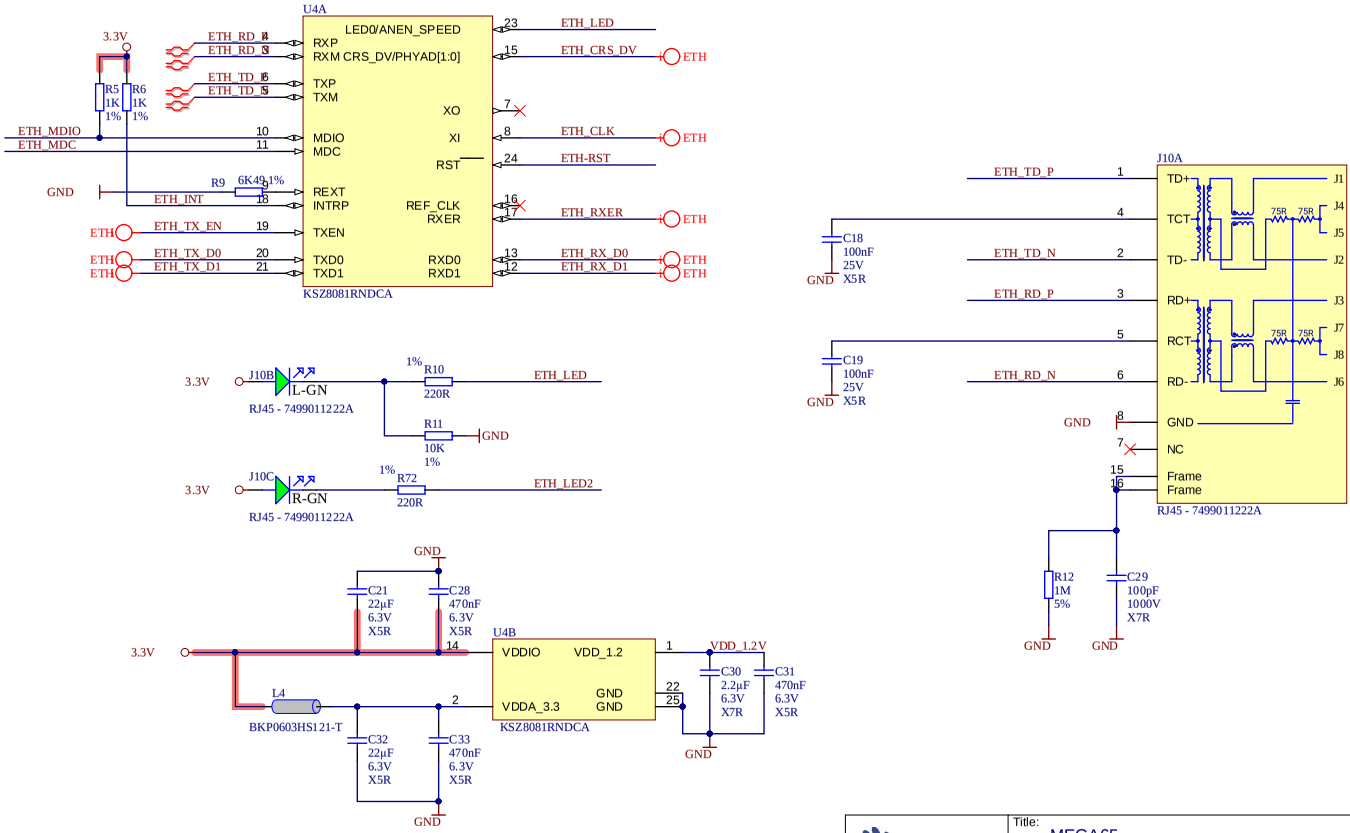


trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 16 of 23
	Filename: VGA.SchDoc		

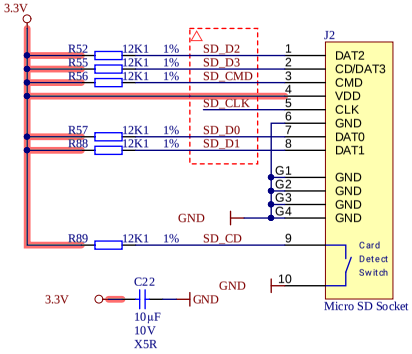


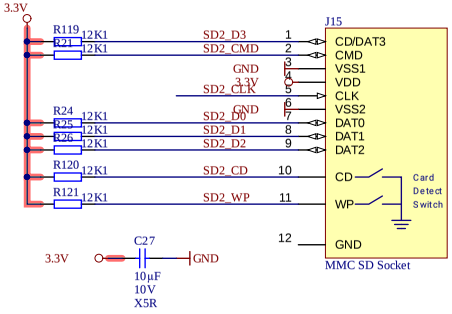


trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 17 of 23
	Filename: HDMI.SchDoc		

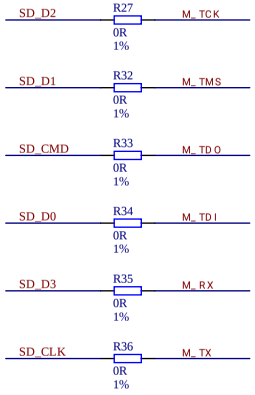


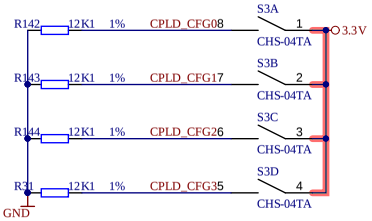
trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 18 of 23
	Filename: Ethernet.SchDoc		





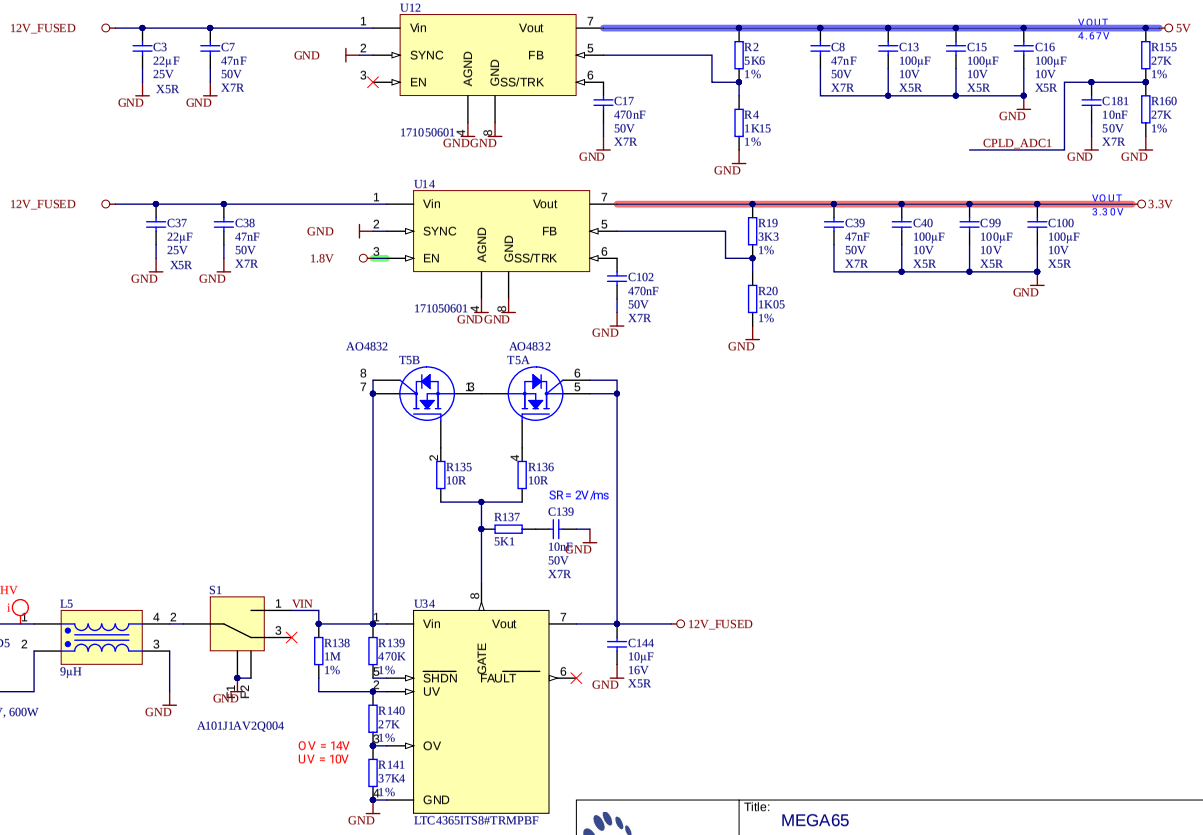
http://linux-sunxi.org/MicroSD_Breakout





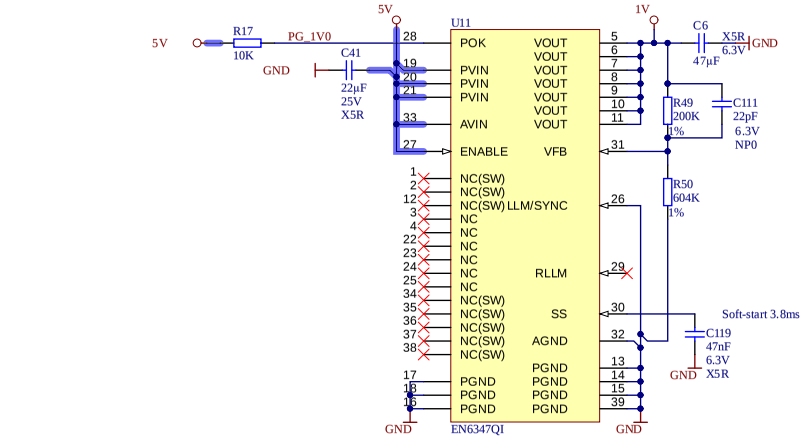
trenz electronic

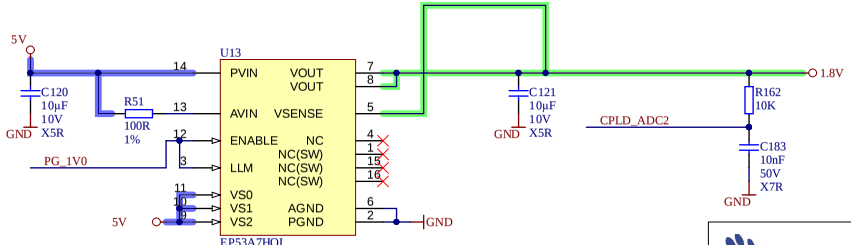
Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 19 of 23
Filename: LED_SW_BUT.SchDoc			



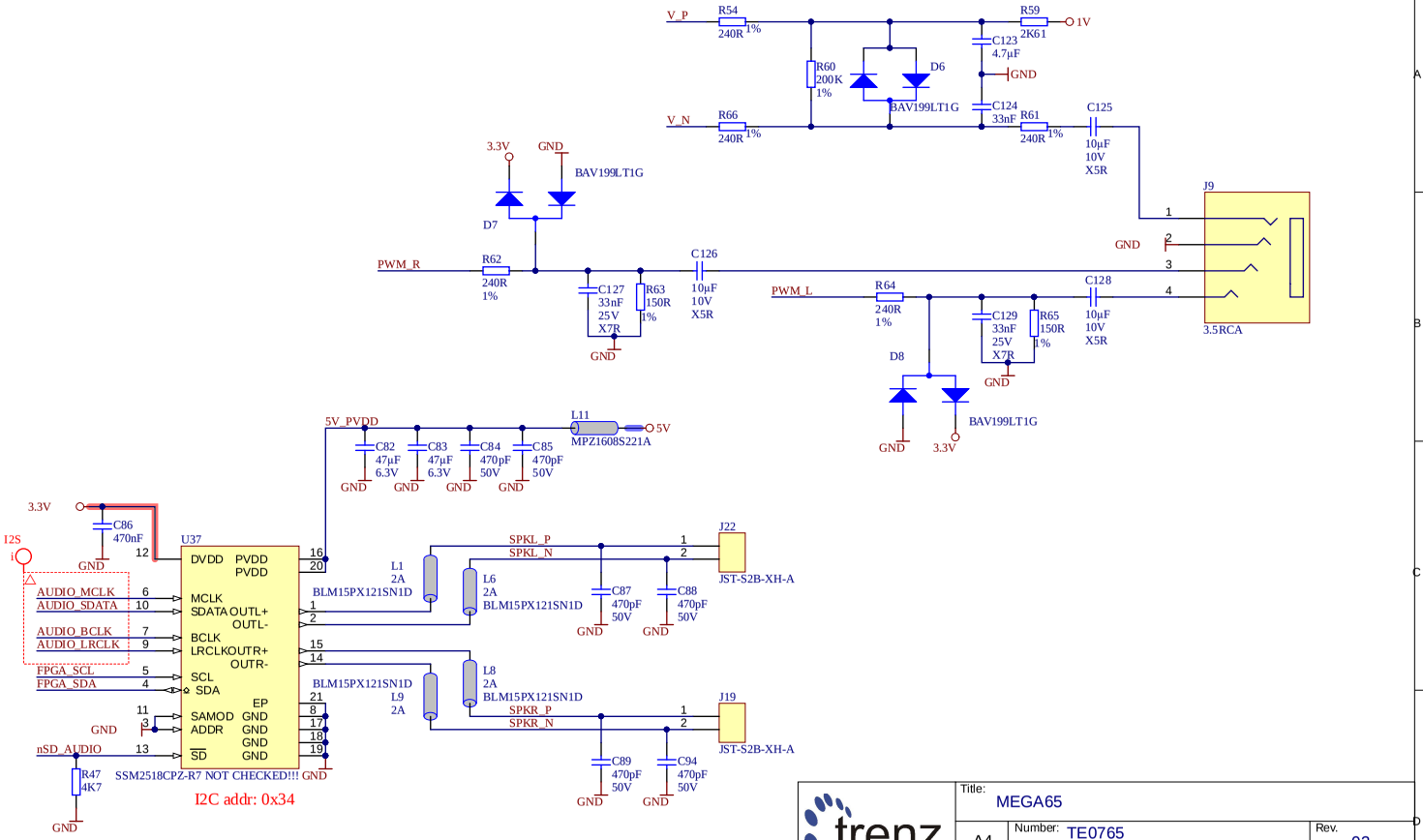
trenz electronic

Title: MEGA65			
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 20 of 23
Filename: PowerMain.SchDoc			





trenz electronic	Title: MEGA65		
	A4	Number: TE0765 [No Variations]	Rev. 03
	Date: 2020-06-29	Copyright: Trenz Electronic GmbH / TT	Page 21 of 23
	Filename: POWER.SchDoc		



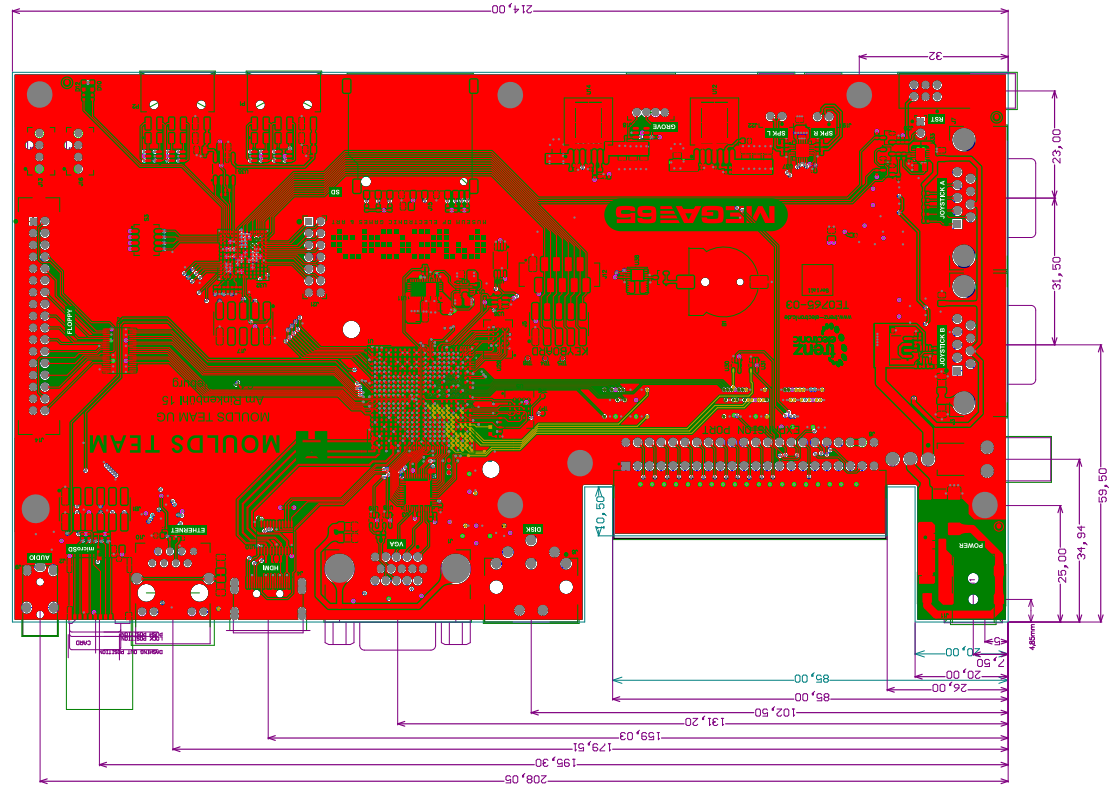
trenz electronic	Title: MEGA65		
A4	Number: TE0765 [No Variations]		Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH		Page 22 of 23
Filename: SOUND.SchDoc			

REVISION HISTORY

REV	Description	
-03	1. Added a VRP resistor on bank 65; 2. LDO U33 is changed on ADP7102ACPZ; 3. Signal FPGA IO0 is connected on AE18 pin of FPGA; 4. Signal DBG_LED3 is connected on AD18 pin of FPGA; 5. Signal MIO13_25 connected to J1 pin 33 instead MIO25. 6. Resistor R84 is removed; 7. LED D1 moved on edge of PCB; 8. Added THT testpoints J4 on CPLD_JTAGEN, R76 was removed; 9. Signals B49_XX_X are renamed in B88_XX_X; 10. C241 is changed on 1nF; 11. Length of CLK signals on RFADC and RFDAC are adjusted; 12. Wrong connection on U8 is fixed (PCB); 13. Wrong connection PGOOD1 pin of U7 is fixed;	IG

trenz electronic

Title: MEGA65		
A4	Number: TE0835 [No Variations]	Rev. 03
Date: 2020-06-29	Copyright: Trenz Electronic GmbH	Page 23 of 23
Filename: Revision_Changes.SchDoc		


214,00
32
23,00
31,50
59,50
34,94
25,00
10,50
7,50
20,00
26,00
85,00
85,00
102,50
131,20
159,03
179,51
195,30
208,05
MEGA65
MOULDS TEAM
MOULDS TEAM UG
An Beckedorf 15
POWER
ETHERNET
HDMI
VGA
DISK
AUDIO
JOYSTICK A
JOYSTICK B
SD
RST

MEGA65 R2 ELEKTRONISCH SCHEMA

U_HDMI
HDMI.SchDoc

U_Ethernet
Ethernet.SchDoc

U_LED_SW_BUT
LED_SW_BUT.SchDoc

U_SYS_MAX10_CTRL
SYS_MAX10_CTRL.SchDoc

U_SOUND
SOUND.SchDoc

U_B13
B13.SchDoc

U_B14
B14.SchDoc

U_B15
B15.SchDoc

U_B16
B16.SchDoc

U_B34
B34.SchDoc

U_FPGA-CFG
FPGA-CFG.SchDoc

U_FPGA-MGT
FPGA-MGT.SchDoc

U_FPGA-PWR
FPGA-PWR.SchDoc

U_JOY
JOY.SchDoc

U_KEYBOARD
KEYBOARD.SchDoc

U_POWER
POWER.SchDoc

U_PowerMain
PowerMain.SchDoc

U_VGA
VGA.SchDoc

U_EXT_HEADER
EXT_HEADER.SchDoc

U_EXP_Slot
EXP_Slot.SchDoc

U_Floppy
Floppy.SchDoc

PM1 FIDU-DOT - small
PM2 FIDU-DOT - small
PM3 FIDU-DOT - small
PM4 FIDU-DOT - small
PM5 FIDU-DOT - small
PM6 FIDU-DOT - small

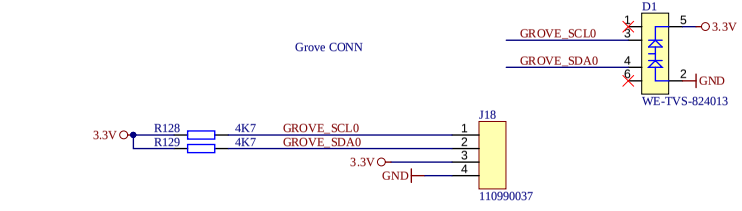
Serial1
Serial
Serialnumber 6,3 x 6.3mm

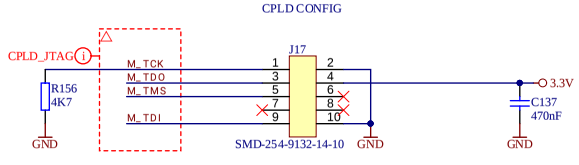
S/N1
1 0R 2
Serialnumber

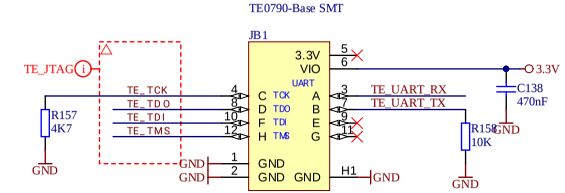
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm
GND Mount.Hole 3.2mm

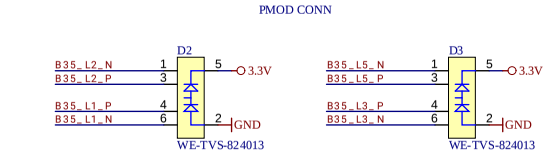
trenz electronic

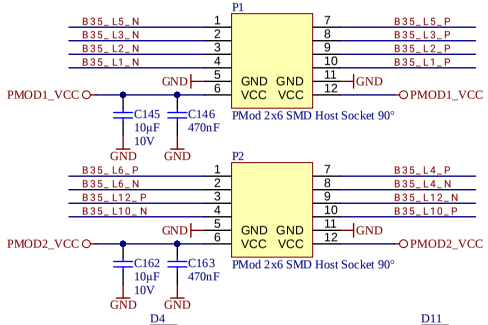
Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page1 of 23
Filename: TE0765.SchDoc			

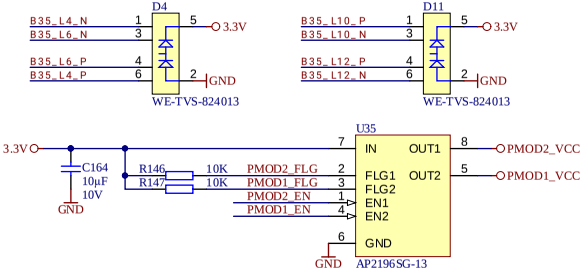






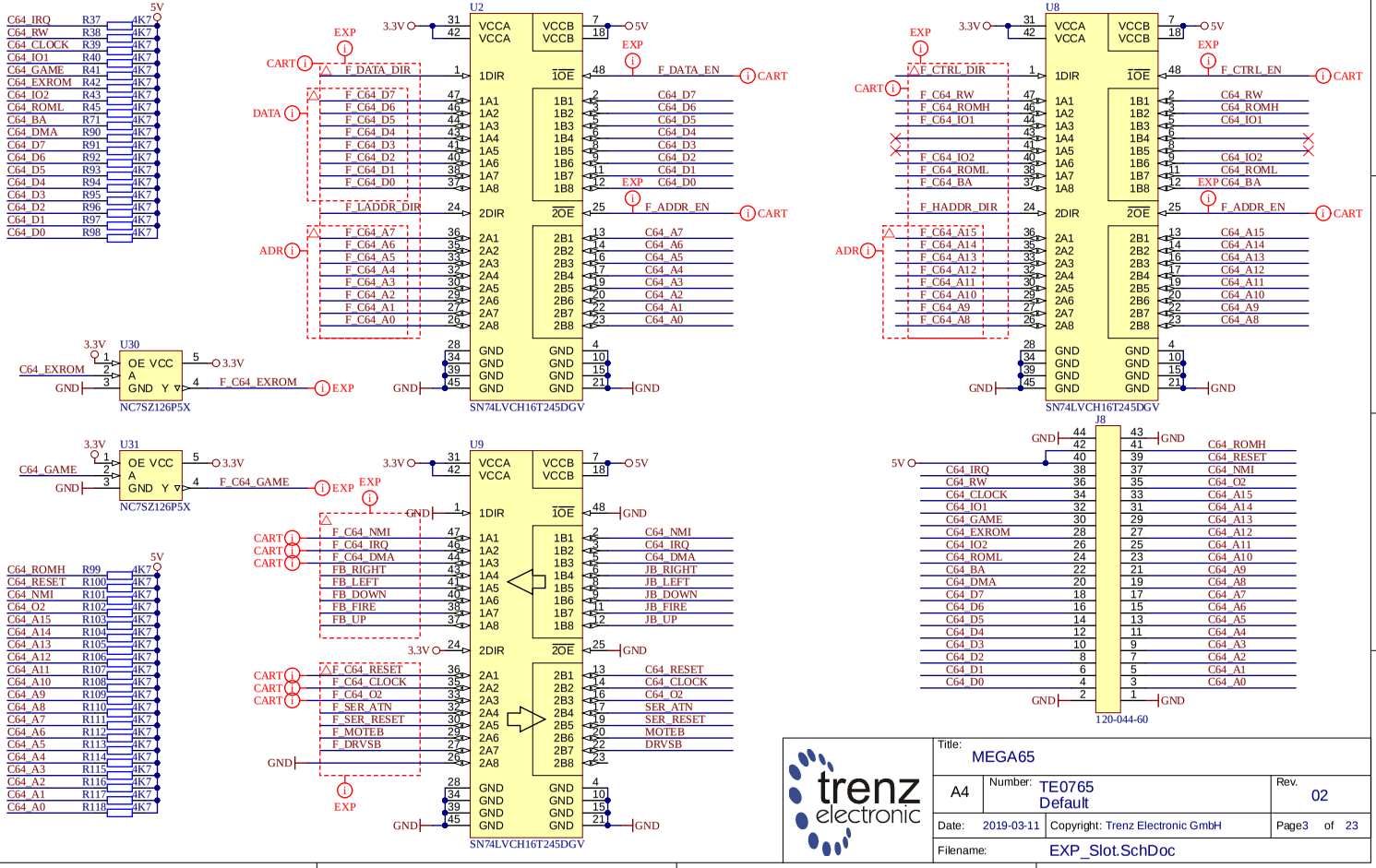




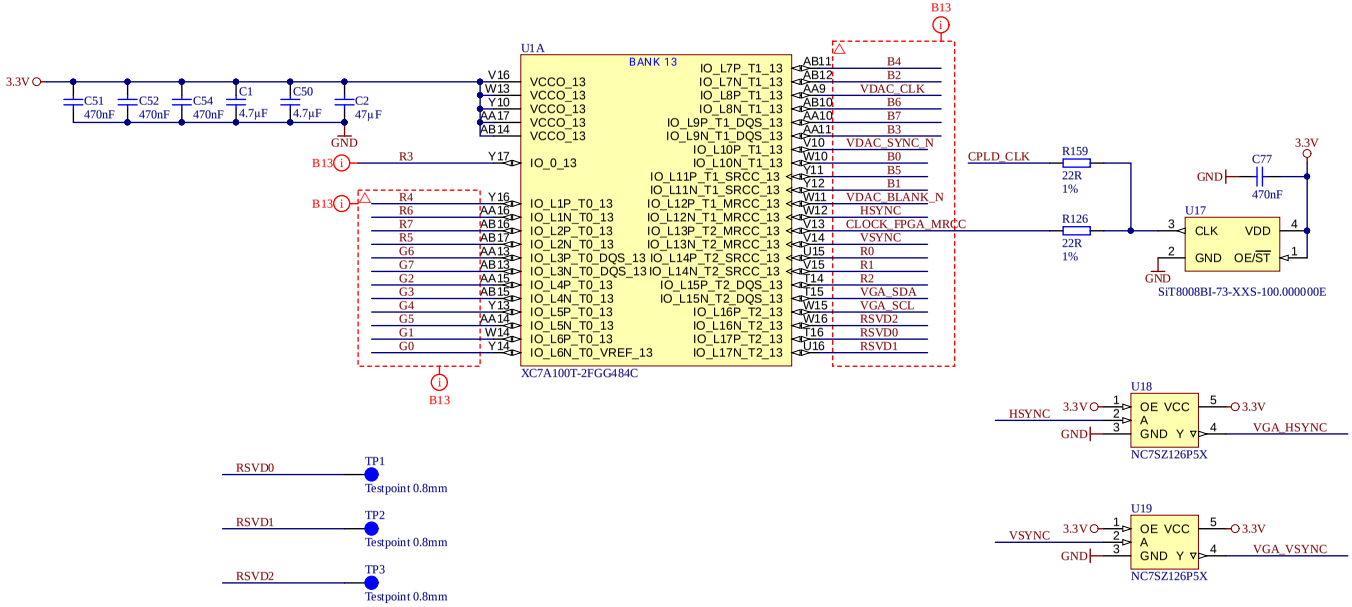


trenz electronic

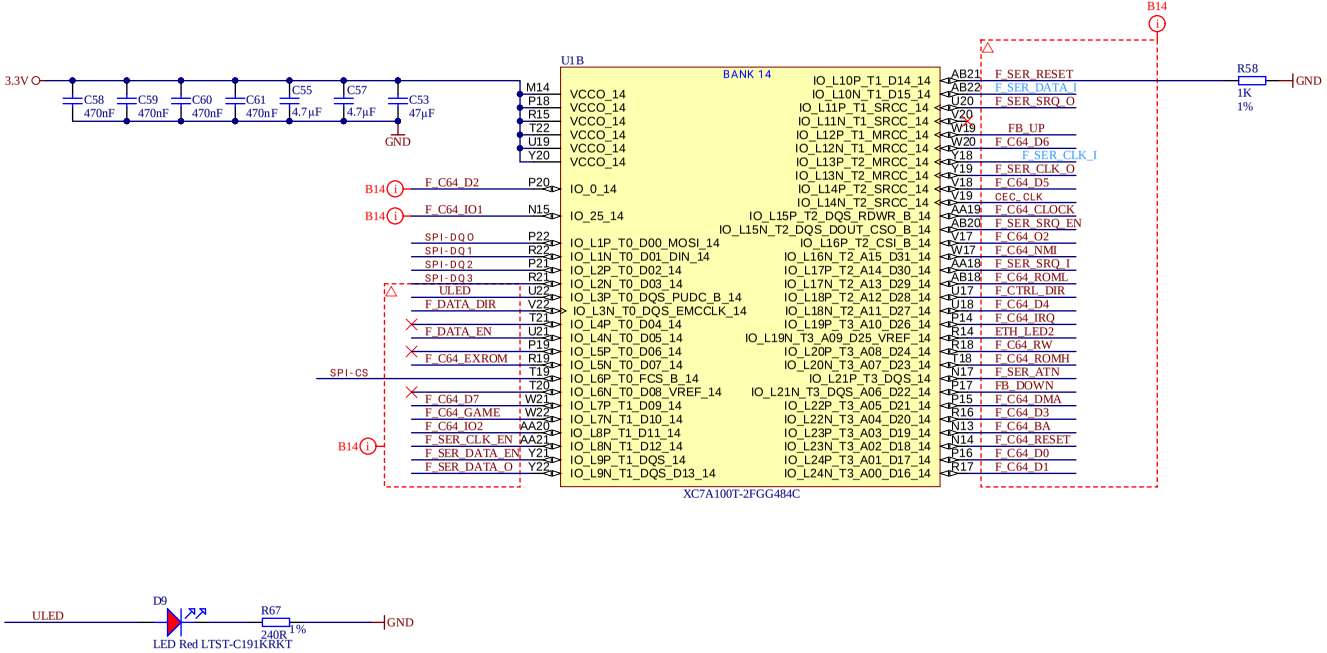
Title: MEGA65		
A4	Number: TE0765 Default	Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH	Page2 of 23
Filename: EXT_HEADER.SchDoc		



Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page3 of 23
Filename: EXP_Slot.SchDoc			

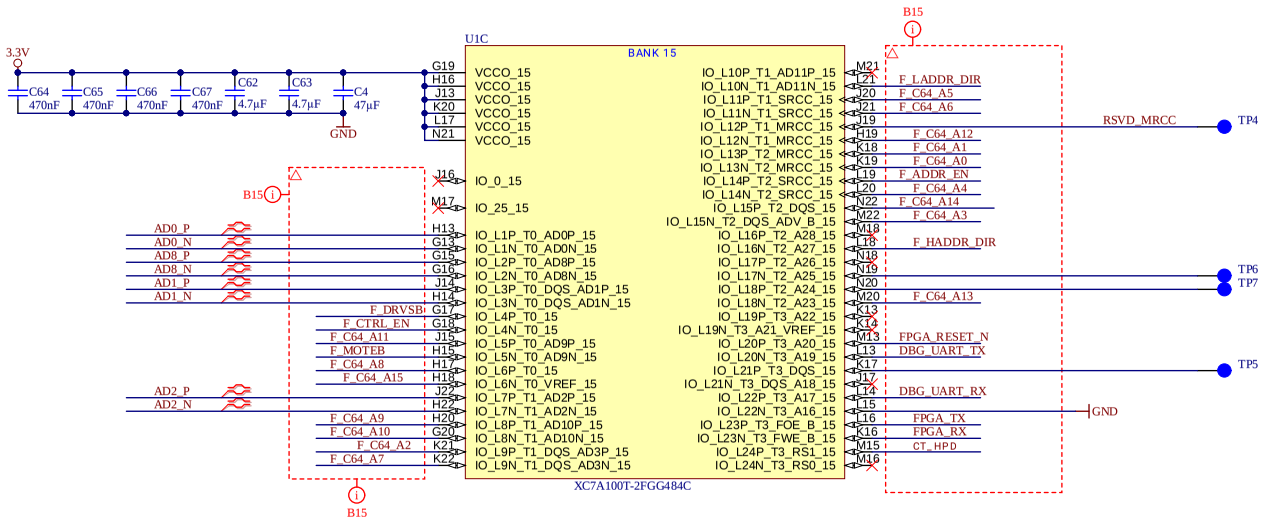


trenz electronic	Title: MEGA65		
	A4	Number: TE0765 Default	Rev. 02
	Date: 2019-03-11	Copyright: Trenz Electronic GmbH	Page4 of 23
	Filename: B13.SchDoc		



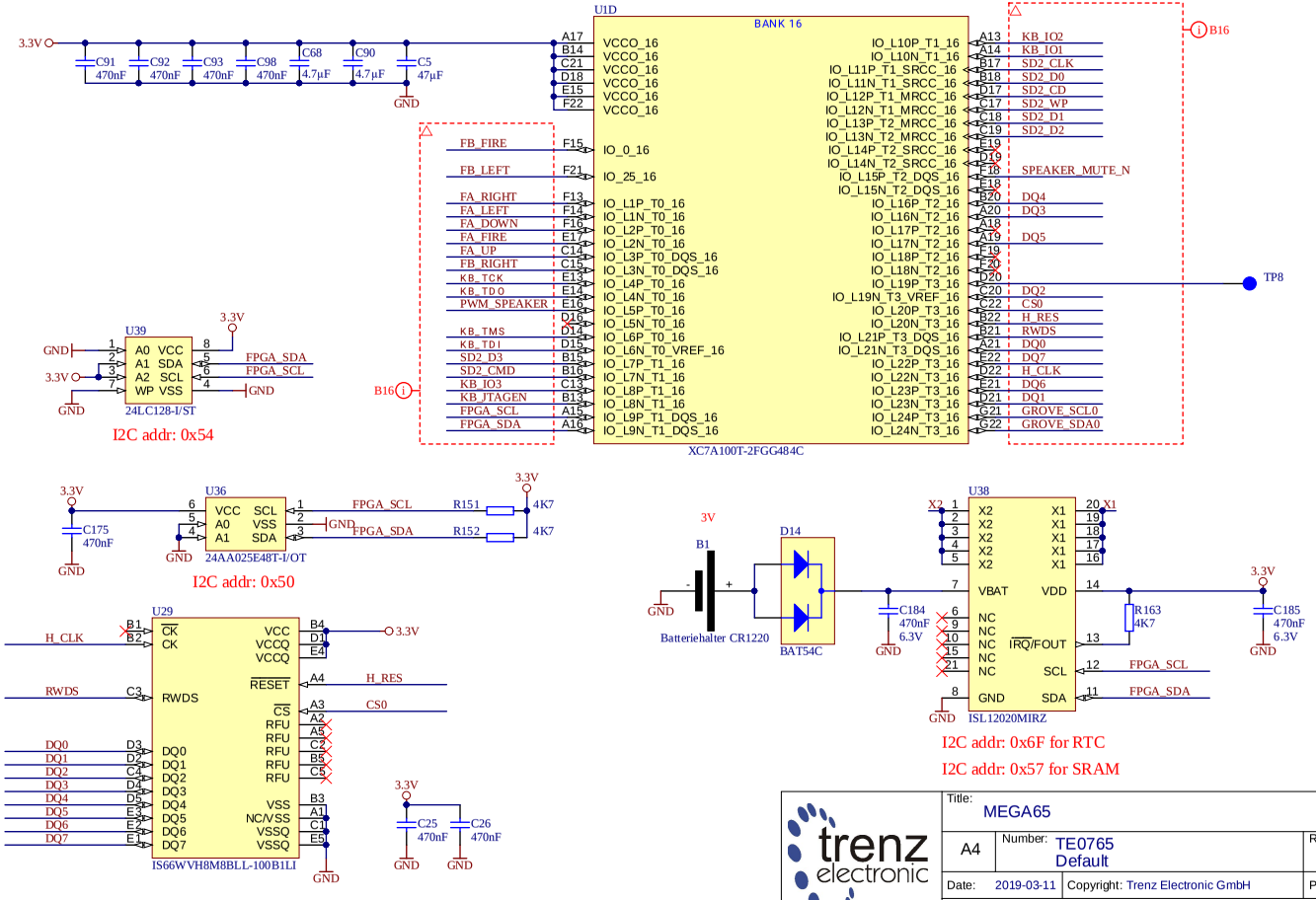
trenz electronic

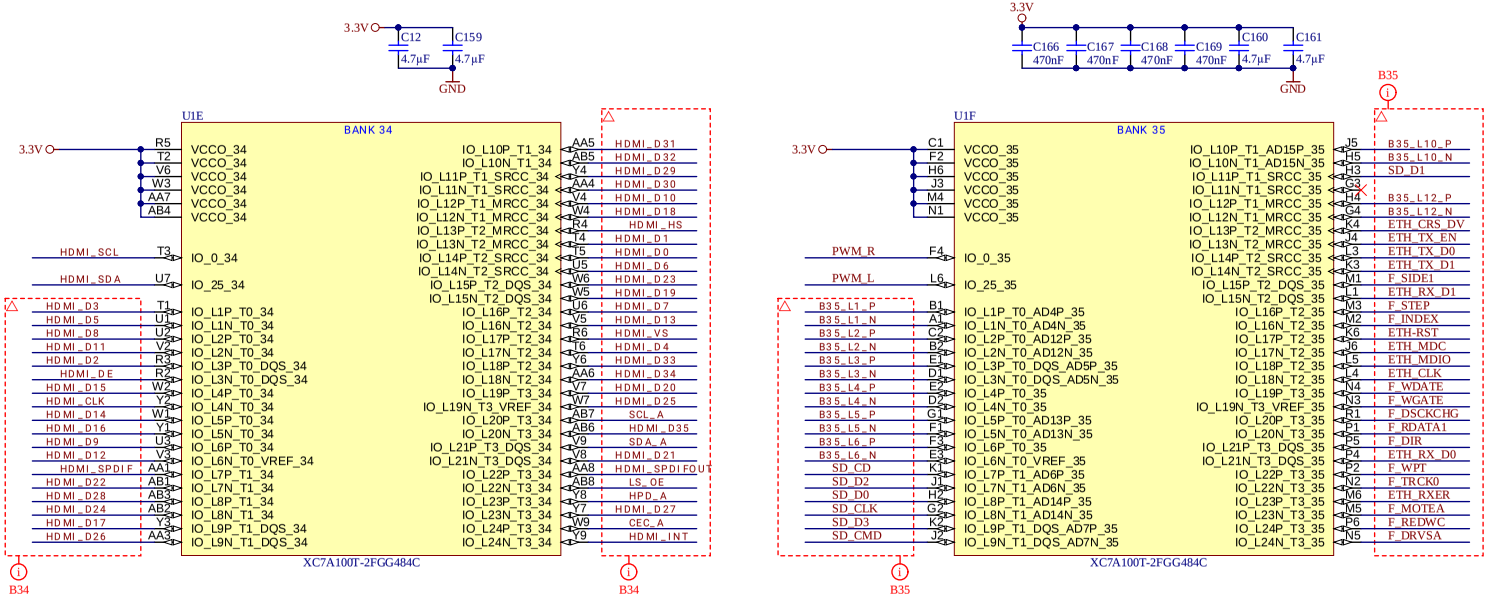
Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page5 of 23
Filename: B14.SchDoc			



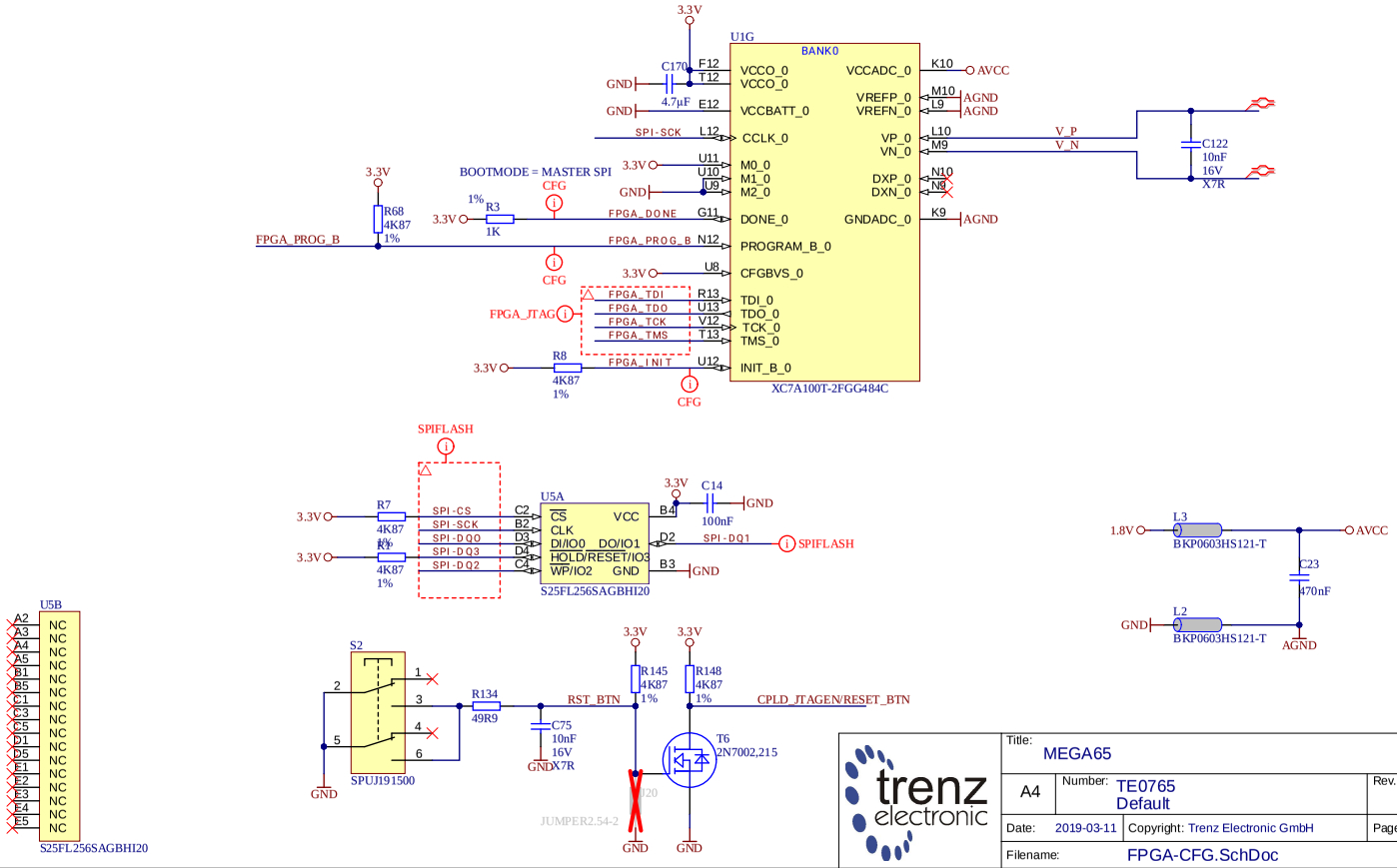
trenz electronic

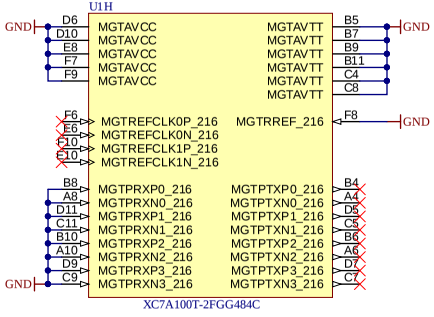
Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page6 of 23
Filename: B15.SchDoc			


U1D
BANK 16
XC7A100T-2FGG484C
3.3V
C91 470nF
C92 470nF
C93 470nF
C98 470nF
C68 4.7µF
C90 4.7µF
C5 47µF
GND
FB_FIRE
FB_LEFT
FA_RIGHT
FA_LEFT
FA_DOWN
FA_FIRE
FA_UP
FB_RIGHT
KB_TCK
KB_TDO
PWM_SPEAKER
KB_TMS
KB_TDI
SD2_D3
SD2_CMD
KB_IO3
KB_JTAGEN
FPGA_SCL
FPGA_SDA
B16
KB_IO2
KB_IO1
SD2_CLK
SD2_D0
SD2_CD
SD2_WP
SD2_D1
SD2_D2
SPEAKER_MUTE_N
DQ4
DQ3
DQ5
DQ2
CS0
H_RES
RWDS
DQ0
DQ7
DQ6
DQ1
TP8
U39
24LC128-I/ST
I2C addr: 0x54
U36
24AA025E48T-I/OT
I2C addr: 0x50
R151 4K7
R152 4K7
C175 470nF
U29
IS66WVH8M8BLL-100B1LI
C25 470nF
C26 470nF
3V
B1
Batteriehalter CR1220
D14
BAT54C
C184 470nF 6.3V
U38
ISL12020MIRZ
R163 4K7
C185 470nF 6.3V
I2C addr: 0x6F for RTC
I2C addr: 0x57 for SRAM
trenz electronic
Title: MEGA65
A4
Number: TE0765 Default
Rev. 02
Date: 2019-03-11
Copyright: Trenz Electronic GmbH
Page7 of 23
Filename: B16.SchDoc

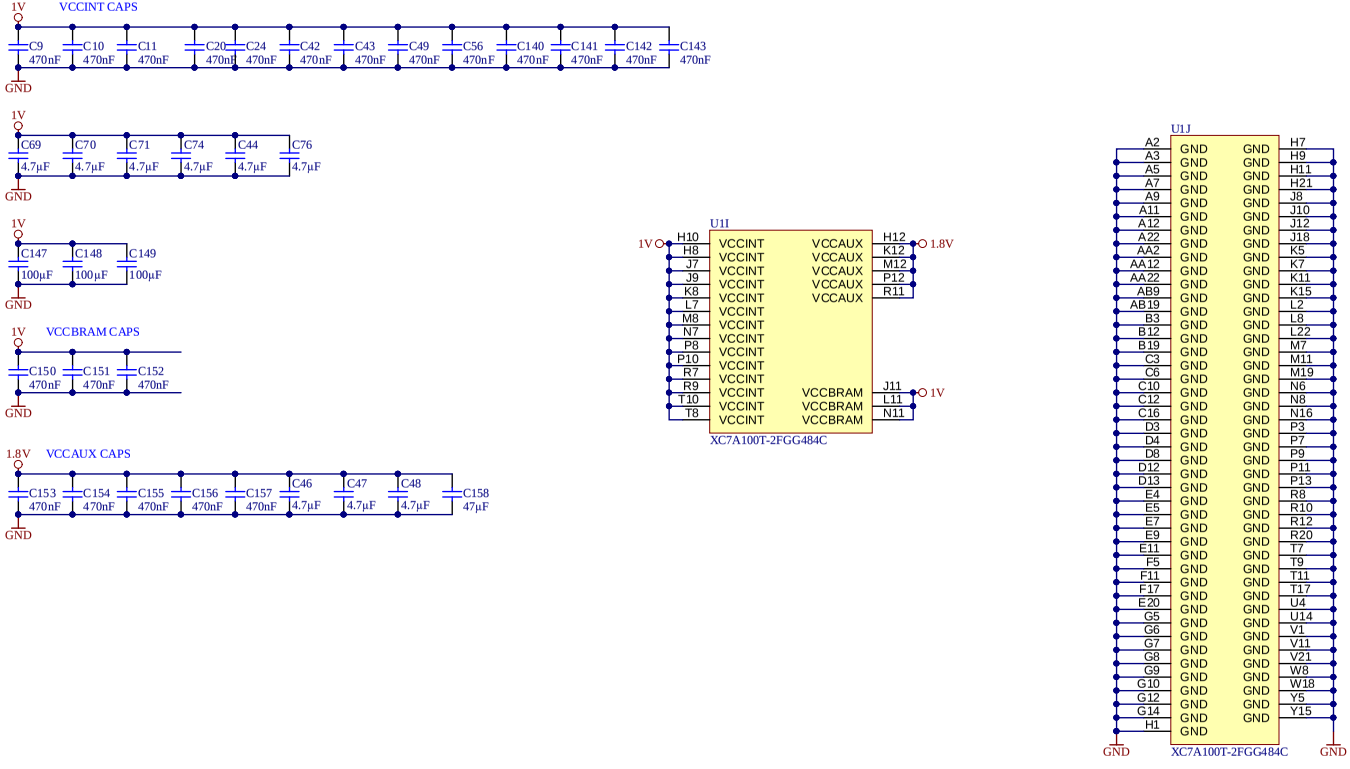


Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page8 of 23
Filename: B34.SchDoc			

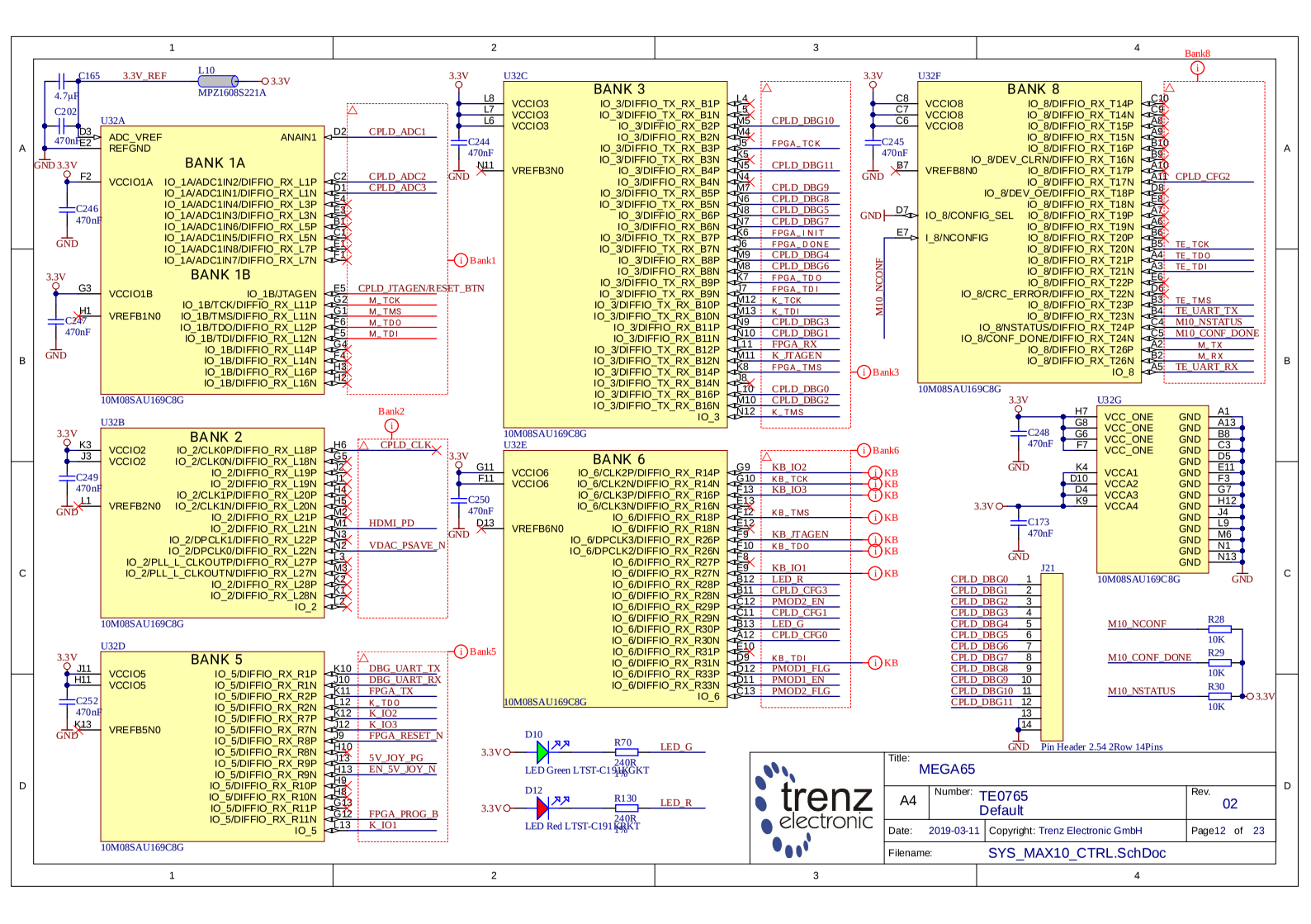

U1G
BANK0
XC7A100T-2FGG484C
VCCO_0
VCCADC_0
VCCBATT_0
VREFP_0
VREFN_0
CCLK_0
VP_0
VN_0
M0_0
M1_0
M2_0
DXP_0
DXN_0
DONE_0
GNDADC_0
PROGRAM_B_0
CFGBVS_0
TDI_0
TDO_0
TCK_0
TMS_0
INIT_B_0
3.3V
AVCC
AGND
GND
SPI-SCK
V_P
V_N
BOOTMODE = MASTER SPI
CFG
FPGA_DONE
FPGA_PROG_B
FPGA_JTAG
FPGA_TDI
FPGA_TDO
FPGA_TCK
FPGA_TMS
FPGA_INIT
C170
4.7µF
C122
10nF
16V
X7R
R68
4K87
1%
R3
1K
R8
SPIFLASH
U5A
S25FL256SAGBHI20
SPI-CS
SPI-DQ0
SPI-DQ1
SPI-DQ2
SPI-DQ3
C14
100nF
R7
L3
L2
BKP0603HS121-T
1.8V
C23
470nF
U5B
S2
SPUJ191500
R134
49R9
C75
10nF
16V
R145
R148
T6
2N7002,215
CPLD_JTAGEN/RESET_BTN
RST_BTN
JUMPER2.54-2
Title:
MEGA65
A4
Number:
TE0765
Default
Rev.
02
Date:
2019-03-11
Copyright: Trenz Electronic GmbH
Page9 of 23
Filename:
FPGA-CFG.SchDoc
trenz electronic

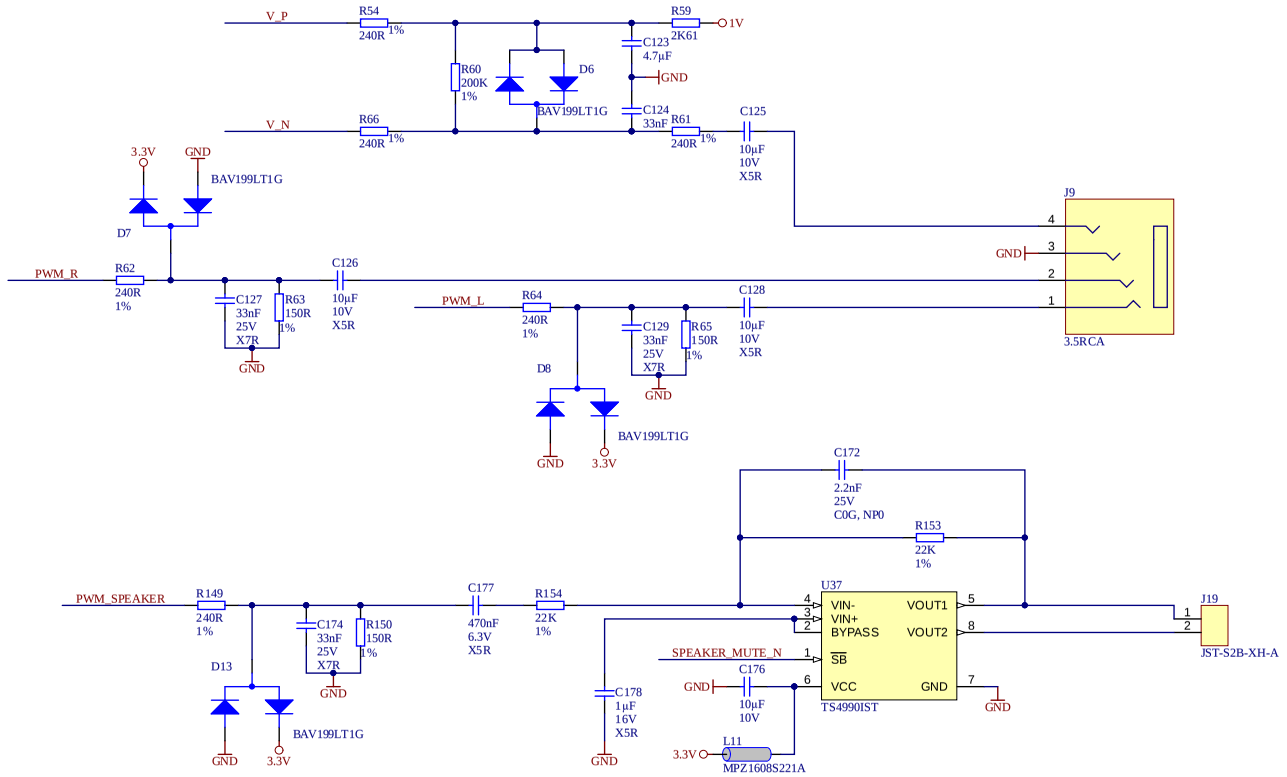


trenz electronic	Title: MEGA65		
	A4	Number: TE0765 Default	Rev. 02
	Date: 2019-03-11	Copyright: Trenz Electronic GmbH	Page10 of 23
	Filename: FPGA-MGT.SchDoc		

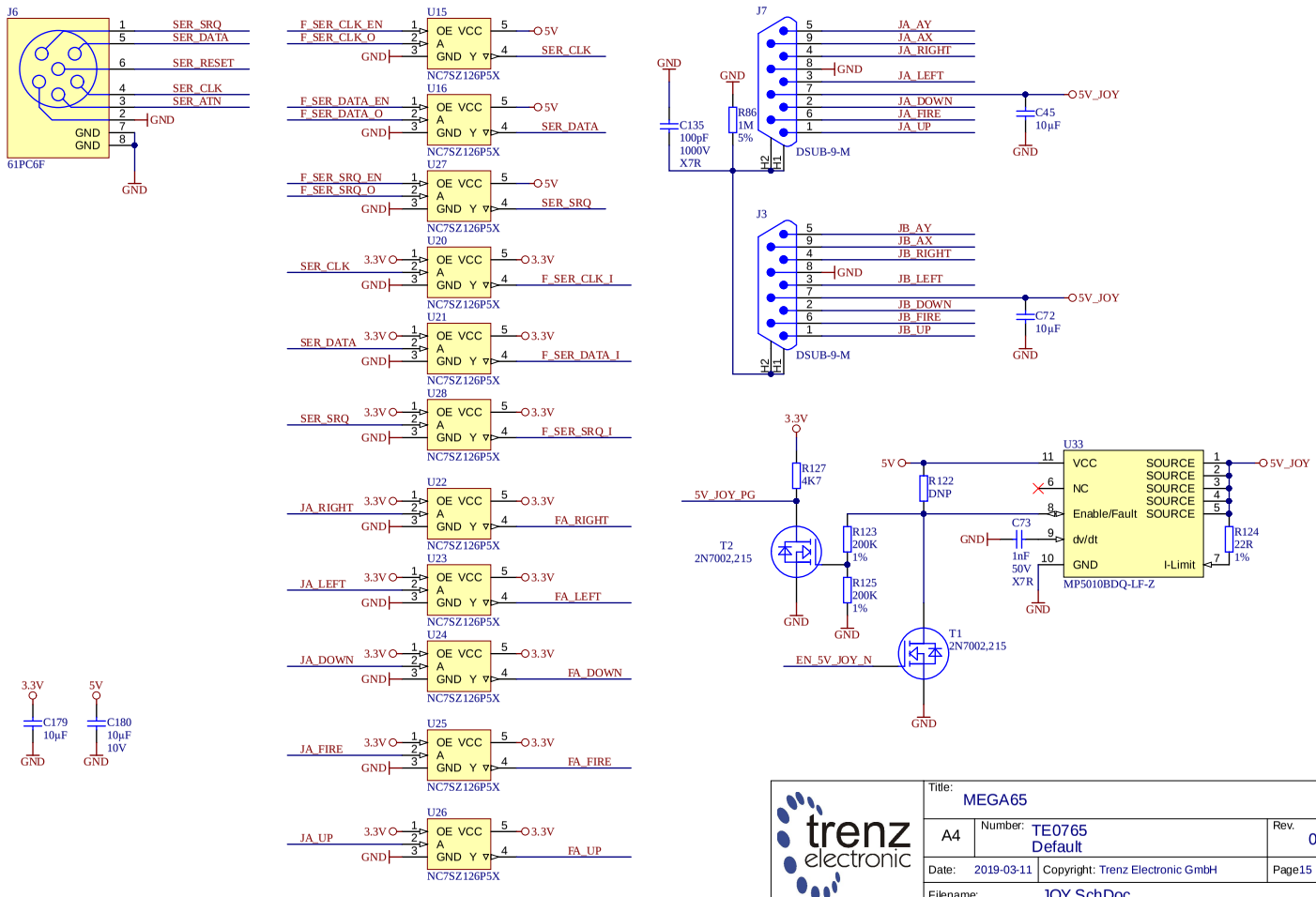


Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page 11 of 23
Filename: FPGA-PWR.SchDoc			

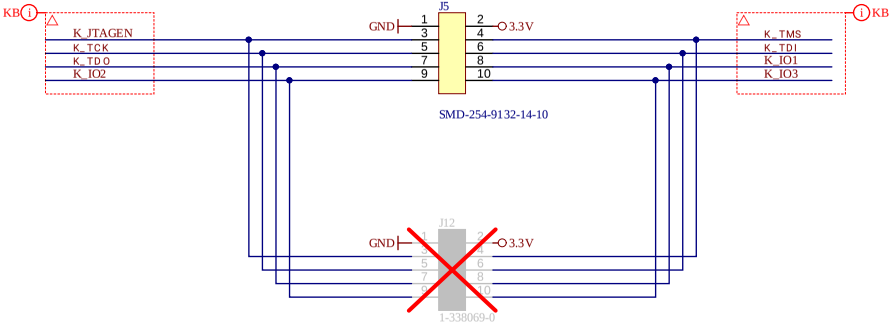

C165
3.3V_REF
L10
MPZ1608S221A
3.3V
4.7µF
C202
470nF
GND
U32A
ADC_VREF
REFGND
ANAIN1
BANK 1A
VCCIO1A
IO_1A/ADC1IN2/DIFFIO_RX_L1P
IO_1A/ADC1IN1/DIFFIO_RX_L1N
IO_1A/ADC1IN4/DIFFIO_RX_L3P
IO_1A/ADC1IN3/DIFFIO_RX_L3N
IO_1A/ADC1IN6/DIFFIO_RX_L5P
IO_1A/ADC1IN5/DIFFIO_RX_L5N
IO_1A/ADC1IN8/DIFFIO_RX_L7P
IO_1A/ADC1IN7/DIFFIO_RX_L7N
BANK 1B
VCCIO1B
IO_1B/JTAGEN
IO_1B/TCK/DIFFIO_RX_L11P
VREFB1N0
IO_1B/TMS/DIFFIO_RX_L11N
IO_1B/TDO/DIFFIO_RX_L12P
IO_1B/TDI/DIFFIO_RX_L12N
IO_1B/DIFFIO_RX_L14P
IO_1B/DIFFIO_RX_L14N
IO_1B/DIFFIO_RX_L16P
IO_1B/DIFFIO_RX_L16N
10M08SAU169C8G
CPLD_ADC1
CPLD_ADC2
CPLD_ADC3
CPLD_JTAGEN/RESET_BTN
M_TCK
M_TMS
M_TDO
M_TDI
Bank1
U32C
BANK 3
VCCIO3
VREFB3N0
IO_3/DIFFIO_TX_RX_B1P
IO_3/DIFFIO_TX_RX_B1N
IO_3/DIFFIO_RX_B2P
IO_3/DIFFIO_RX_B2N
IO_3/DIFFIO_TX_RX_B3P
IO_3/DIFFIO_TX_RX_B3N
IO_3/DIFFIO_RX_B4P
IO_3/DIFFIO_RX_B4N
IO_3/DIFFIO_TX_RX_B5P
IO_3/DIFFIO_TX_RX_B5N
IO_3/DIFFIO_RX_B6P
IO_3/DIFFIO_RX_B6N
IO_3/DIFFIO_TX_RX_B7P
IO_3/DIFFIO_TX_RX_B7N
IO_3/DIFFIO_RX_B8P
IO_3/DIFFIO_RX_B8N
IO_3/DIFFIO_TX_RX_B9P
IO_3/DIFFIO_TX_RX_B9N
IO_3/DIFFIO_TX_RX_B10P
IO_3/DIFFIO_TX_RX_B10N
IO_3/DIFFIO_RX_B11P
IO_3/DIFFIO_RX_B11N
IO_3/DIFFIO_TX_RX_B12P
IO_3/DIFFIO_TX_RX_B12N
IO_3/DIFFIO_TX_RX_B14P
IO_3/DIFFIO_TX_RX_B14N
IO_3/DIFFIO_TX_RX_B16P
IO_3/DIFFIO_TX_RX_B16N
IO_3
10M08SAU169C8G
CPLD_DBG10
FPGA_TCK
CPLD_DBG11
CPLD_DBG9
CPLD_DBG8
CPLD_DBG5
CPLD_DBG7
FPGA_INIT
FPGA_DONE
CPLD_DBG4
CPLD_DBG6
FPGA_TDO
FPGA_TDI
K_TCK
K_TDI
CPLD_DBG3
CPLD_DBG1
FPGA_RX
K_JTAGEN
FPGA_TMS
CPLD_DBG0
CPLD_DBG2
K_TMS
U32F
BANK 8
VCCIO8
VREFB8N0
IO_8/DEV_CLRN/DIFFIO_RX_T16N
IO_8/DEV_OE/DIFFIO_RT18P
IO_8/CONFIG_SEL
I_8/NCONFIG
IO_8/CRC_ERROR/DIFFIO_RX_T22N
IO_8/NSTATUS/DIFFIO_RX_T24P
IO_8/CONF_DONE/DIFFIO_RX_T24N
IO_8
10M08SAU169C8G
Bank8
CPLD_CFG2
TE_TCK
TE_TDO
TE_TDI
TE_TMS
TE_UART_TX
M10_NSTATUS
M10_CONF_DONE
M_TX
M_RX
TE_UART_RX
M10_NCONF
Bank3
U32B
BANK 2
VCCIO2
VREFB2N0
IO_2/CLK0P/DIFFIO_RX_L18P
IO_2/CLK0N/DIFFIO_RX_L18N
IO_2/DIFFIO_RX_L19P
IO_2/DIFFIO_RX_L19N
IO_2/CLK1P/DIFFIO_RX_L20P
IO_2/CLK1N/DIFFIO_RX_L20N
IO_2/DIFFIO_RX_L21P
IO_2/DIFFIO_RX_L21N
IO_2/DPCLK1/DIFFIO_RX_L22P
IO_2/DPCLK0/DIFFIO_RX_L22N
IO_2/PLL_L_CLKOUTP/DIFFIO_RX_L27P
IO_2/PLL_L_CLKOUTN/DIFFIO_RX_L27N
IO_2/DIFFIO_RX_L28P
IO_2/DIFFIO_RX_L28N
IO_2
10M08SAU169C8G
Bank2
CPLD_CLK
HDMI_PD
VDAC_PSAVE_N
U32E
BANK 6
VCCIO6
VREFB6N0
IO_6/CLK2P/DIFFIO_RX_R14P
IO_6/CLK2N/DIFFIO_RX_R14N
IO_6/CLK3P/DIFFIO_RX_R16P
IO_6/CLK3N/DIFFIO_RX_R16N
IO_6/DIFFIO_RX_R18P
IO_6/DIFFIO_RX_R18N
IO_6/DPCLK3/DIFFIO_RX_R26P
IO_6/DPCLK2/DIFFIO_RX_R26N
IO_6/DIFFIO_RX_R27P
IO_6/DIFFIO_RX_R27N
IO_6/DIFFIO_RX_R28P
IO_6/DIFFIO_RX_R28N
IO_6/DIFFIO_RX_R29P
IO_6/DIFFIO_RX_R29N
IO_6/DIFFIO_RX_R30P
IO_6/DIFFIO_RX_R30N
IO_6/DIFFIO_RX_R31P
IO_6/DIFFIO_RX_R31N
IO_6/DIFFIO_RX_R33P
IO_6/DIFFIO_RX_R33N
IO_6
10M08SAU169C8G
KB_IO2
KB_TCK
KB_IO3
KB_TMS
KB_JTAGEN
KB_TDO
KB_IO1
LED_R
CPLD_CFG3
PMOD2_EN
CPLD_CFG1
LED_G
CPLD_CFG0
KB_TDI
PMOD1_FLG
PMOD1_EN
PMOD2_FLG
Bank6
KB
U32D
BANK 5
VCCIO5
VREFB5N0
IO_5/DIFFIO_RX_R1P
IO_5/DIFFIO_RX_R1N
IO_5/DIFFIO_RX_R2P
IO_5/DIFFIO_RX_R2N
IO_5/DIFFIO_RX_R7P
IO_5/DIFFIO_RX_R7N
IO_5/DIFFIO_RX_R8P
IO_5/DIFFIO_RX_R8N
IO_5/DIFFIO_RX_R9P
IO_5/DIFFIO_RX_R9N
IO_5/DIFFIO_RX_R10P
IO_5/DIFFIO_RX_R10N
IO_5/DIFFIO_RX_R11P
IO_5/DIFFIO_RX_R11N
IO_5
10M08SAU169C8G
DBG_UART_TX
DBG_UART_RX
FPGA_TX
K_TDO
K_IO2
K_IO3
FPGA_RESET_N
5V_JOY_PG
EN_5V_JOY_N
FPGA_PROG_B
K_IO1
Bank5
U32G
VCC_ONE
VCCA1
VCCA2
VCCA3
VCCA4
GND
10M08SAU169C8G
J21
Pin Header 2.54 2Row 14Pins
M10_NCONF
M10_CONF_DONE
M10_NSTATUS
R28
R29
R30
10K
D10
LED Green LTST-C191KGKT
D12
LED Red LTST-C191KRKT
R70
R130
240R
LED_G
LED_R
trenz electronic
Title: MEGA65
A4
Number: TE0765 Default
Rev. 02
Date: 2019-03-11
Copyright: Trenz Electronic GmbH
Page12 of 23
Filename: SYS_MAX10_CTRL.SchDoc



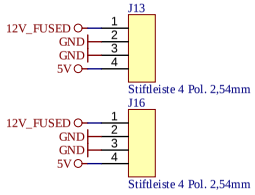
trenz electronic	Title: MEGA65		
	A4	Number: TE0765 Default	Rev. 02
	Date: 2019-03-11	Copyright: Trenz Electronic GmbH	Page14 of 23
	Filename: SOUND.SchDoc		

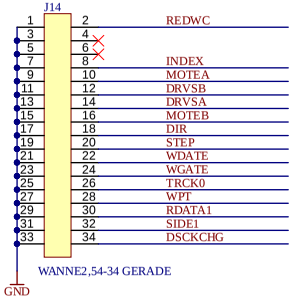


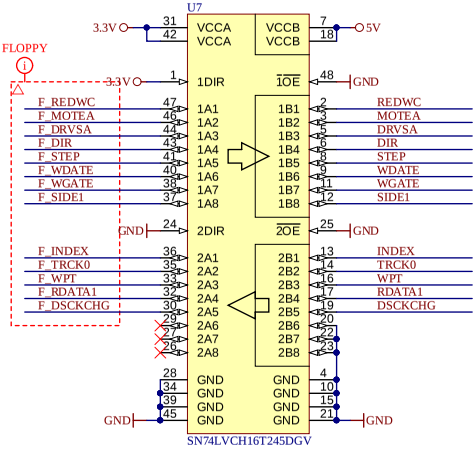
Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page15 of 23
Filename: JOY.SchDoc			



Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page16 of 23
Filename: KEYBOARD.SchDoc			

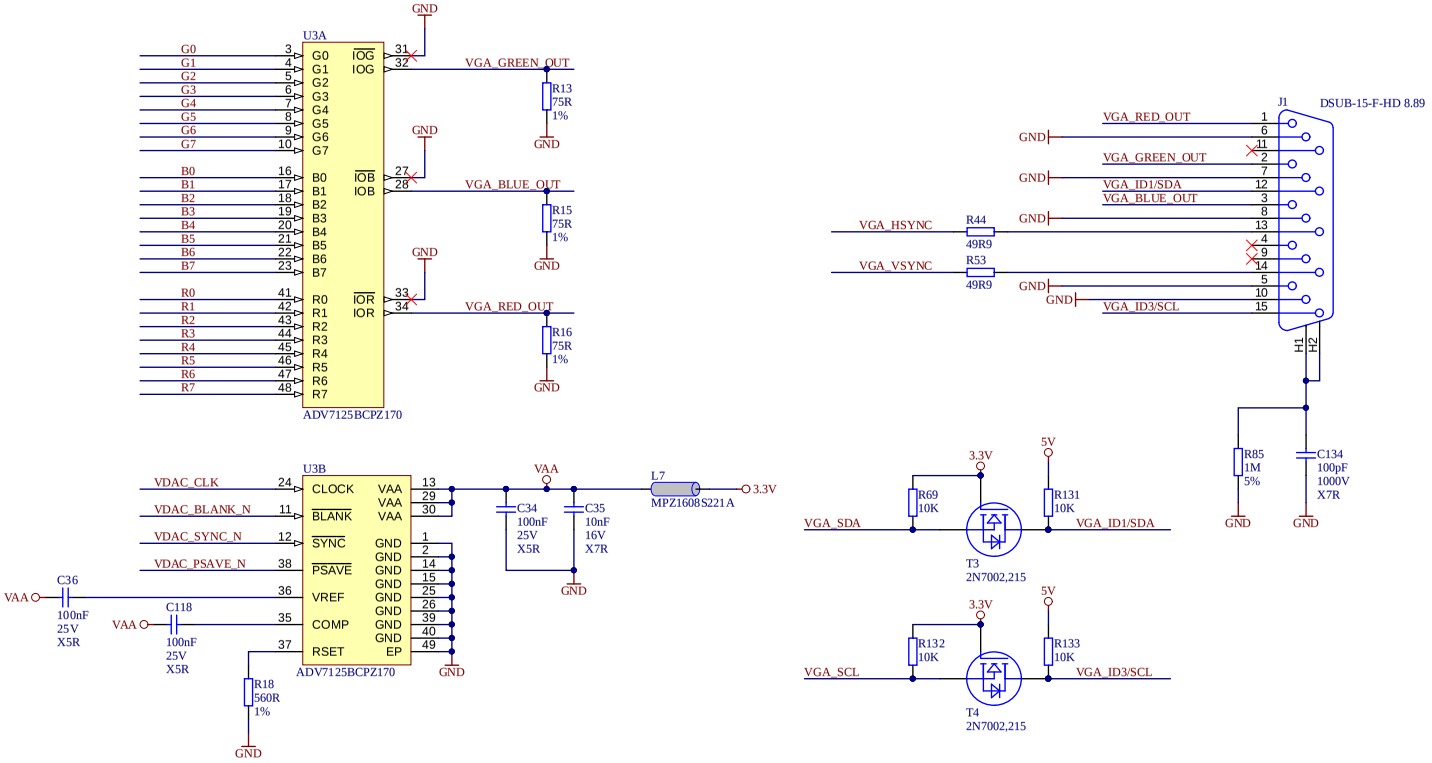




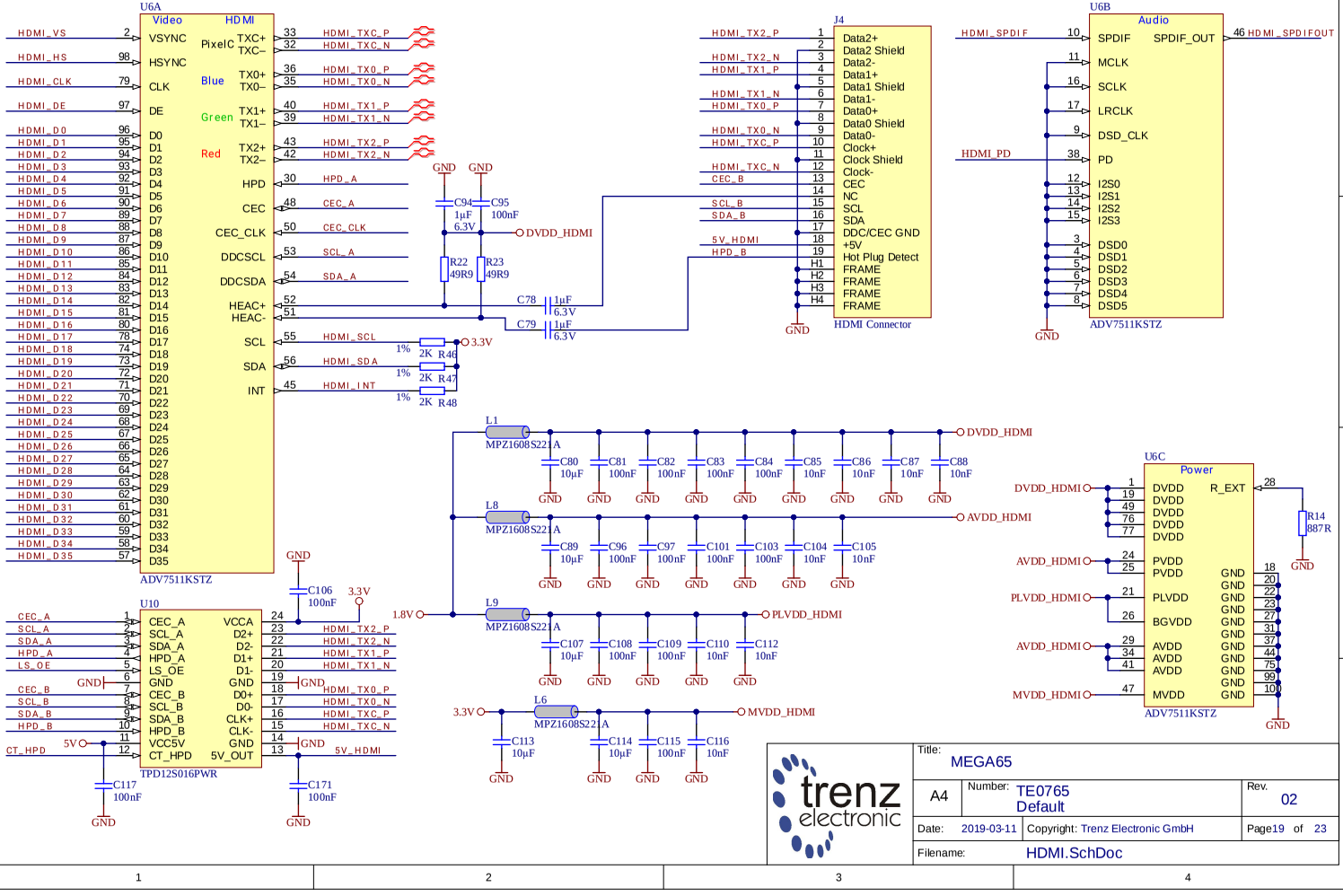


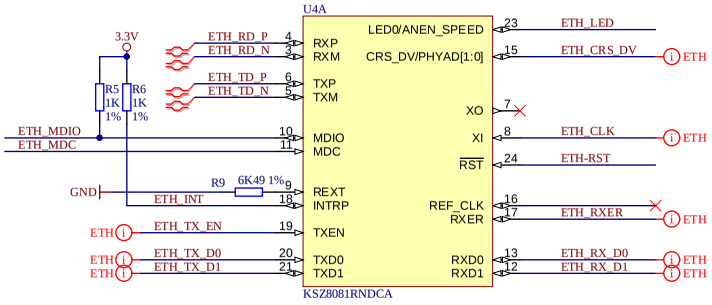
trenz electronic

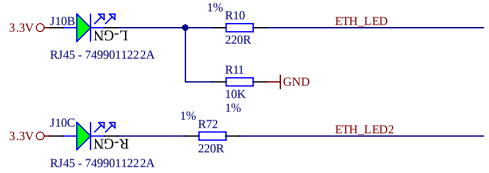
Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page17 of 23
Filename: Floppy.SchDoc			

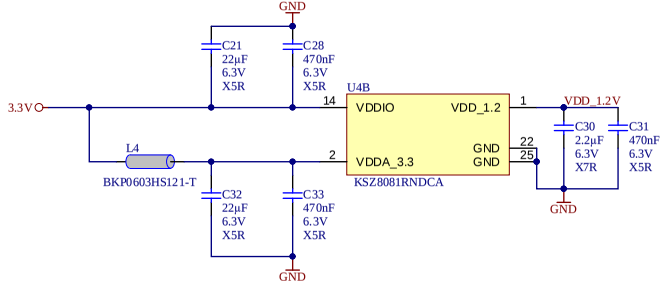


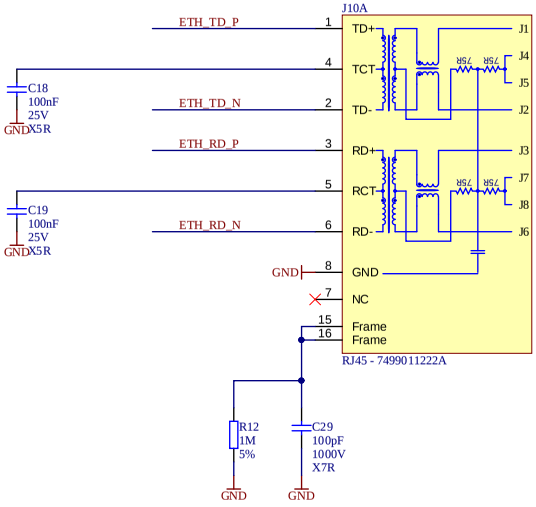
trenz electronic	Title: MEGA65		
	A4	Number: TE0765 Default	Rev. 02
	Date: 2019-03-11	Copyright: Trenz Electronic GmbH	Page18 of 23
	Filename: VGA.SchDoc		


U6A
Video
HDMI
VSYNC
HSYNC
CLK
DE
D0
D1
D2
D3
D4
D5
D6
D7
D8
D9
D10
D11
D12
D13
D14
D15
D16
D17
D18
D19
D20
D21
D22
D23
D24
D25
D26
D27
D28
D29
D30
D31
D32
D33
D34
D35
PixelC
Blue
Green
Red
TXC+
TXC–
TX0+
TX0–
TX1+
TX1–
TX2+
TX2–
HPD
CEC
CEC_CLK
DDCSCL
DDCSDA
HEAC+
HEAC-
SCL
SDA
INT
ADV7511KSTZ
HDMI_VS
HDMI_HS
HDMI_CLK
HDMI_DE
HDMI_D0
HDMI_D1
HDMI_D2
HDMI_D3
HDMI_D4
HDMI_D5
HDMI_D6
HDMI_D7
HDMI_D8
HDMI_D9
HDMI_D10
HDMI_D11
HDMI_D12
HDMI_D13
HDMI_D14
HDMI_D15
HDMI_D16
HDMI_D17
HDMI_D18
HDMI_D19
HDMI_D20
HDMI_D21
HDMI_D22
HDMI_D23
HDMI_D24
HDMI_D25
HDMI_D26
HDMI_D27
HDMI_D28
HDMI_D29
HDMI_D30
HDMI_D31
HDMI_D32
HDMI_D33
HDMI_D34
HDMI_D35
HDMI_TXC_P
HDMI_TXC_N
HDMI_TX0_P
HDMI_TX0_N
HDMI_TX1_P
HDMI_TX1_N
HDMI_TX2_P
HDMI_TX2_N
HPD_A
CEC_A
CEC_CLK
SCL_A
SDA_A
HDMI_SCL
HDMI_SDA
HDMI_INT
GND
C94 1µF 6.3V
C95 100nF
DVDD_HDMI
R22 49R9
R23 49R9
C78 1µF 6.3V
C79 1µF 6.3V
1% 2K R46
1% 2K R47
1% 2K R48
3.3V
J4
Data2+
Data2 Shield
Data2-
Data1+
Data1 Shield
Data1-
Data0+
Data0 Shield
Data0-
Clock+
Clock Shield
Clock-
CEC
NC
SCL
SDA
DDC/CEC GND
+5V
Hot Plug Detect
FRAME
FRAME
FRAME
FRAME
HDMI Connector
HDMI_TX2_P
HDMI_TX2_N
HDMI_TX1_P
HDMI_TX1_N
HDMI_TX0_P
HDMI_TX0_N
HDMI_TXC_P
HDMI_TXC_N
CEC_B
SCL_B
SDA_B
5V_HDMI
HPD_B
U6B
Audio
SPDIF
SPDIF_OUT
MCLK
SCLK
LRCLK
DSD_CLK
PD
I2S0
I2S1
I2S2
I2S3
DSD0
DSD1
DSD2
DSD3
DSD4
DSD5
ADV7511KSTZ
HDMI_SPDIF
HDMI_SPDIFOUT
HDMI_PD
U10
CEC_A
SCL_A
SDA_A
HPD_A
LS_OE
GND
CEC_B
SCL_B
SDA_B
HPD_B
VCC5V
CT_HPD
VCCA
D2+
D2-
D1+
D1-
GND
D0+
D0-
CLK+
CLK-
GND
5V_OUT
TPD12S016PWR
5V
5V_HDMI
C106 100nF
C117 100nF
C171 100nF
L1
MPZ1608S221A
L8
MPZ1608S221A
L9
MPZ1608S221A
L6
MPZ1608S221A
1.8V
C80 10µF
C81 100nF
C82 100nF
C83 100nF
C84 100nF
C85 10nF
C86 10nF
C87 10nF
C88 10nF
C89 10µF
C96 100nF
C97 100nF
C101 100nF
C103 100nF
C104 10nF
C105 10nF
C107 10µF
C108 100nF
C109 100nF
C110 10nF
C112 10nF
C113 10µF
C114 10µF
C115 100nF
C116 10nF
AVDD_HDMI
PLVDD_HDMI
MVDD_HDMI
U6C
Power
DVDD
PVDD
PLVDD
BGVDD
AVDD
MVDD
R_EXT
R14 887R
ADV7511KSTZ
trenz electronic
Title: MEGA65
A4
Number: TE0765 Default
Rev. 02
Date: 2019-03-11
Copyright: Trenz Electronic GmbH
Page19 of 23
Filename: HDMI.SchDoc



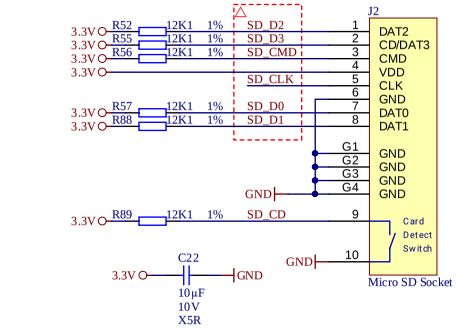


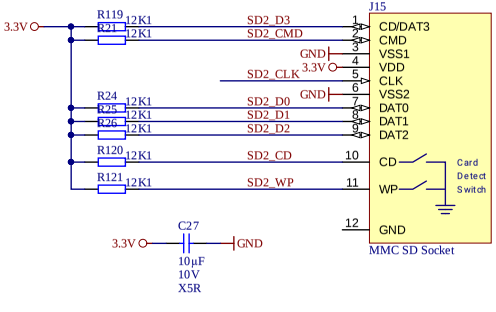




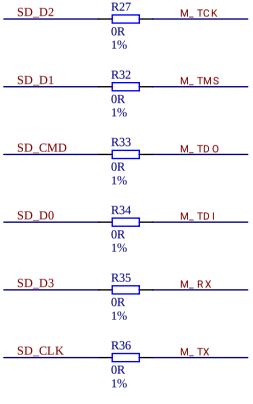
trenz electronic

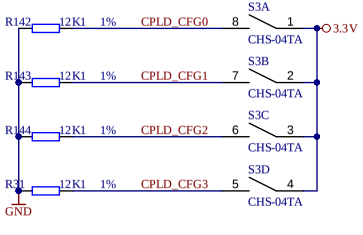
Title: MEGA65			
A4	Nummer: TE0765 Default		Rev. 02
Datum: 2019-03-11	Zeichner: Trenz Electronic GmbH		Blatt 20 von 23
Filename: Ethernet.SchDoc			



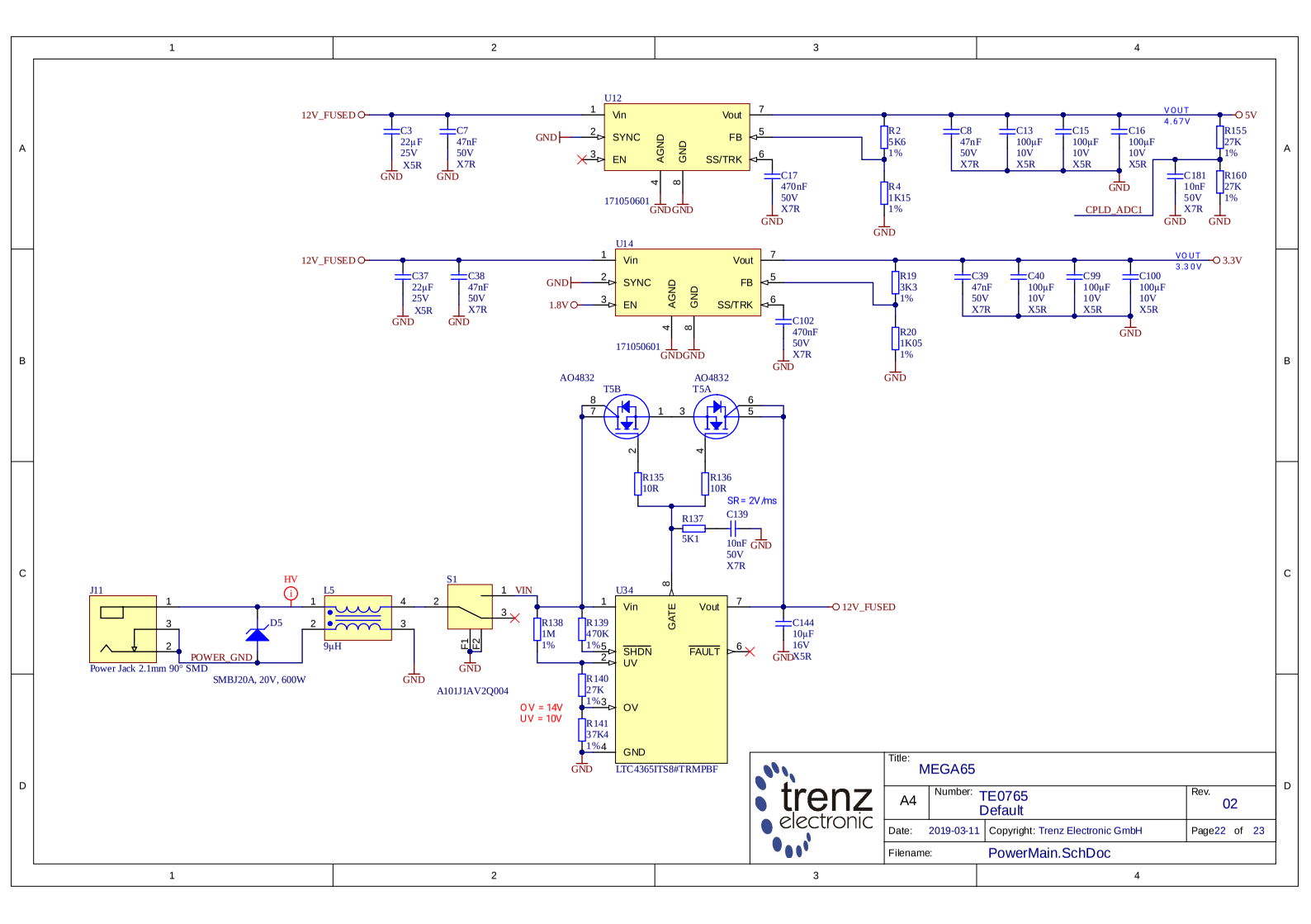


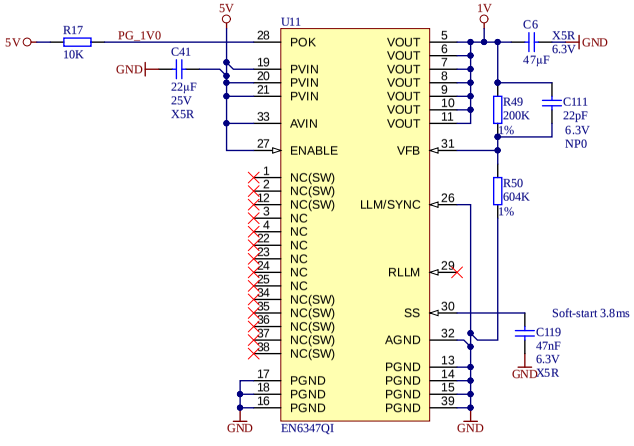
http://linux-sunxi.org/MicroSD_Breakout

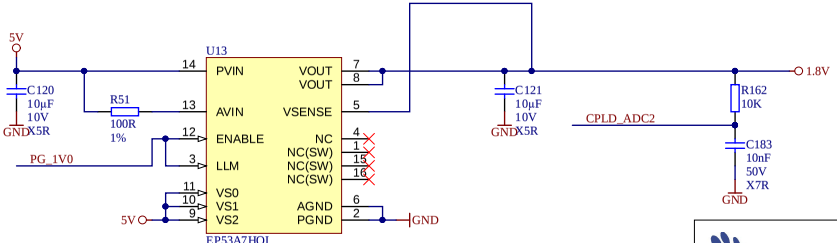


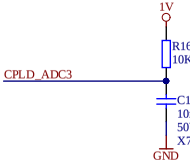


Title: MEGA65			
A4	Number: TE0765 Default		Rev. 02
Date: 2019-03-11	Copyright: Trenz Electronic GmbH		Page21 of 23
Filename: LED_SW_BUT.SchDoc			


12V_FUSED
U12
Vin
Vout
SYNC
FB
EN
AGND
GND
SS/TRK
171050601
C3 22µF 25V X5R
C7 47nF 50V X7R
C17 470nF 50V X7R
R2 5K6 1%
R4 1K15 1%
C8 47nF 50V X7R
C13 100µF 10V X5R
C15 100µF 10V X5R
C16 100µF 10V X5R
VOUT 4.67V
5V
R155 27K 1%
C181 10nF 50V X7R
R160 27K 1%
CPLD_ADC1
U14
1.8V
C37 22µF 25V X5R
C38 47nF 50V X7R
C102 470nF 50V X7R
R19 3K3 1%
R20 1K05 1%
C39 47nF 50V X7R
C40 100µF 10V X5R
C99 100µF 10V X5R
C100 100µF 10V X5R
VOUT 3.30V
3.3V
AO4832
T5B
T5A
R135 10R
R136 10R
SR = 2V/ms
R137 5K1
C139 10nF 50V X7R
J11
Power Jack 2.1mm 90° SMD
HV
D5
SMBJ20A, 20V, 600W
L5
9µH
S1
F1
F2
A101J1AV2Q004
VIN
R138 1M 1%
R139 470K 1%
U34
GATE
SHDN
UV
FAULT
OV
R140 27K 1%
R141 37K4 1%
0 V = 14V
UV = 10V
LTC4365ITS8#TRMPBF
C144 10µF 16V X5R
POWER_GND
Title: MEGA65
A4
Number: TE0765 Default
Rev. 02
Date: 2019-03-11
Copyright: Trenz Electronic GmbH
Page22 of 23
Filename: PowerMain.SchDoc
trenz electronic

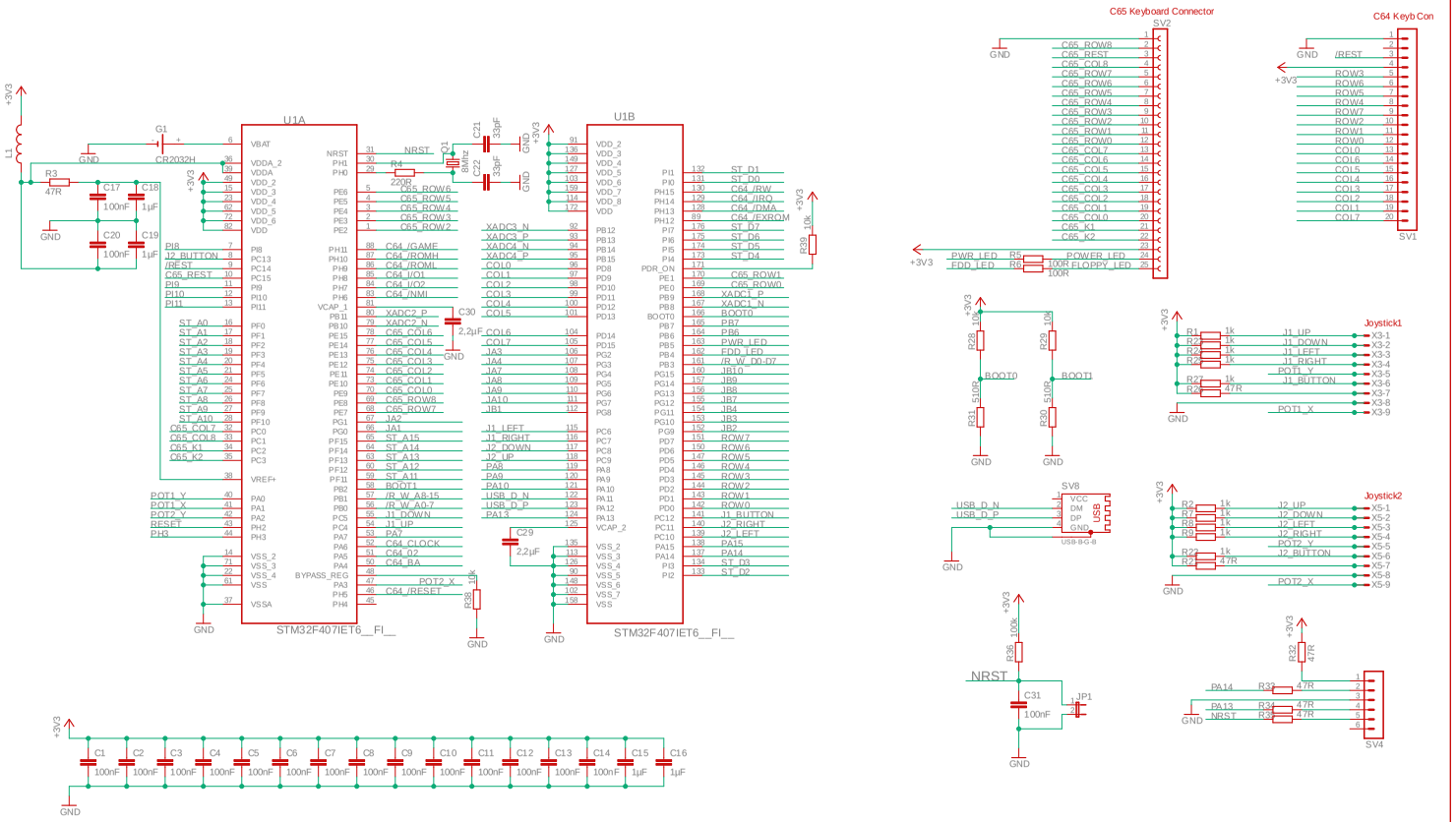




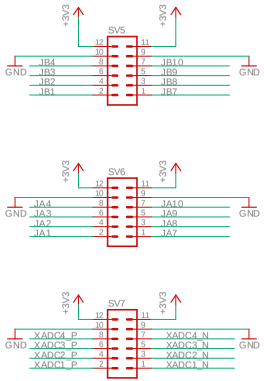


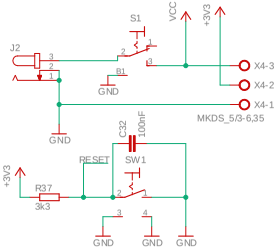
trenz electronic	Title: MEGA65		
	A4	Number: TE0765 Default	Rev. 02
	Date: 2019-03-11	Copyright: Trenz Electronic GmbH / TT	Page23 of 23
	Filename: POWER.SchDoc		

NEXYS WIDGET BOARD ELEKTRONISCH SCHEMA

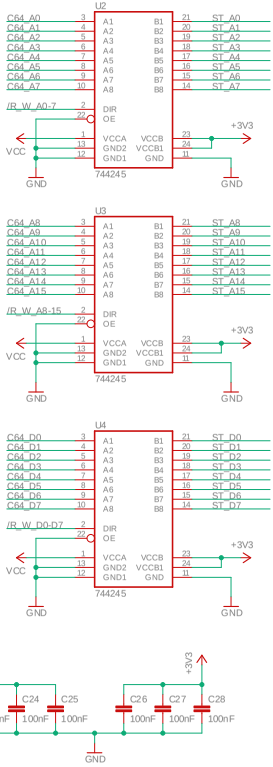

C65 Keyboard Connector
C64 Keyb Con
U1A
U1B
STM32F407IET6__FI__
STM32F407IET6__FI__
SV2
SV1
SV8
SV4
Joystick1
Joystick2
TITLE: c65Keyb
Document Number:
REV:
Date: 6/05/2022 10:40 AM
Sheet: 1/2

Nexys 4 Input

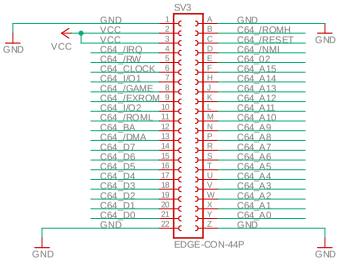


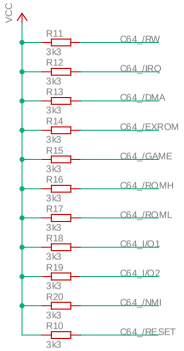


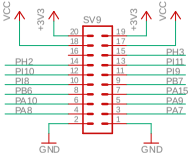
Output Driver



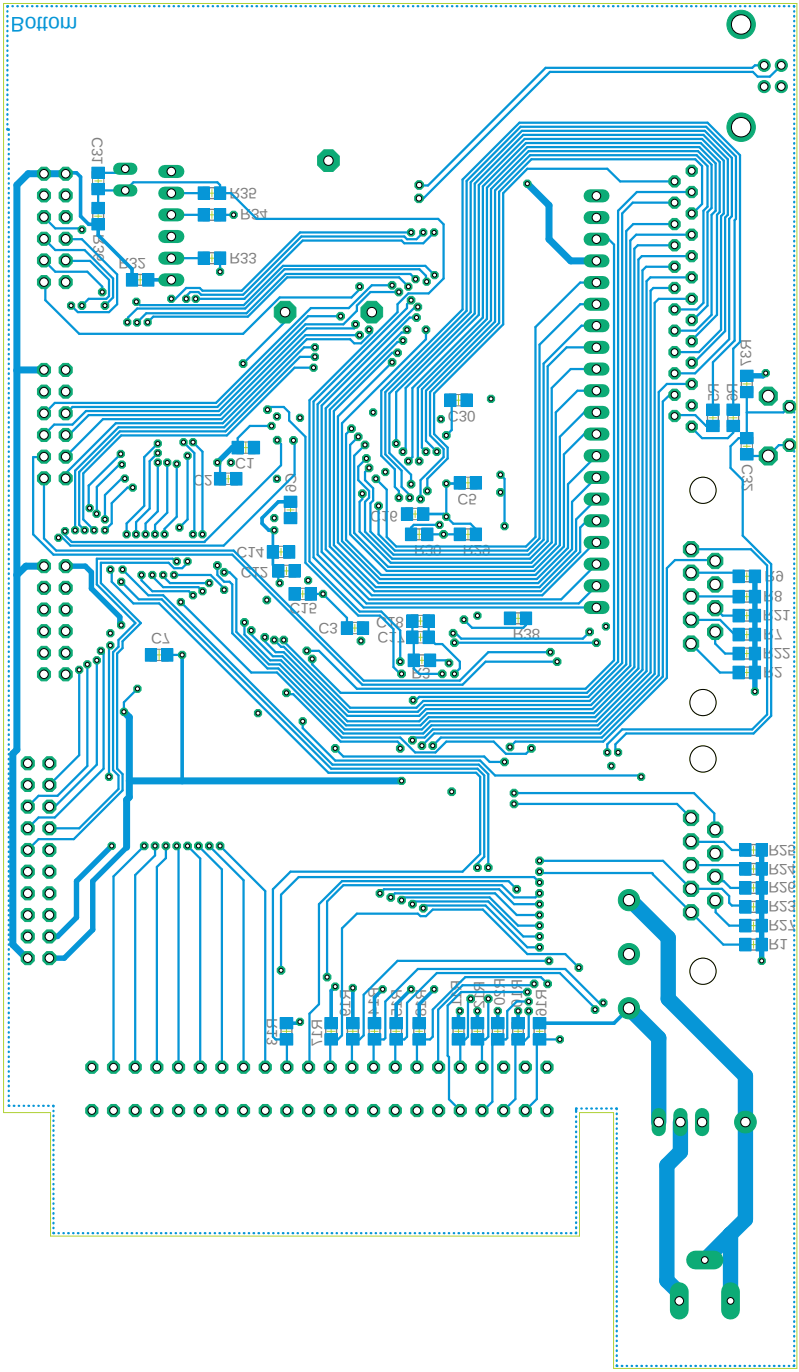
Expansion Slot







TITLE: c65Keyb		
Document Number:		REV:
Date: 6/05/2022 10:40 AM	Sheet: 2/2	


Bottom

BIJLAGE D

Steuners & donateurs

- Organisaties
- Bijdragers
- Steuners

De MEGA65 zou niet tot stand zijn gekomen zonder de gulle steun van veel organisaties en individuen.

Wij stellen deze lijsten nog steeds samen; onze excuses indien u er nog niet in voorkomt. Indien u iemand kent die wij over het hoofd hebben gezien, laat het ons dan weten. Zo kunnen wij de bijdrage erkennen van iedereen die de MEGA65 mogelijk heeft gemaakt en die er het geweldige retrocomputingproject van heeft gemaakt dat de computer inmiddels is.

ORGANISATIES

Het MEGA Museum of Electronic Games & Art e.V. Duitsland
ALLES

Trenz Electronic, Duitsland
MOEDERBORD **FABRICAGE** **VERKOOP**

Hintsteiner, Oostenrijk
BEHUIZING

GMK, Duitsland
TOETSENBORD

KEVAG Telekom, Duitsland
WEBHOSTING

BIJDRAGERS

Andreas Liebeskind
(libi in paradize)
Financieel directeur MEGA eV

Thomas Hertzler
(grumpyninja)
Woordvoerder voor de VS

Russell Peake
(rdpeake)
Bugherder

Alexander Nik Petra
(n0d)
Vroeg ontwerp van de behuizing

Ralph Egas
(0-limits)
Zakelijk adviseur

Lucas Moss
PCB-ontwerp voor de MEGAphone

Daren Klamer
(Impakt)
Proeflezen van de handleiding

Daniël Mantione
(dmantione)
C64-hardwareguru

Dr. Canan Hastik
(indica)
Voorzitter MEGA eV

Simon Jameson
(Shallan)
Platformverbeteringen

Stephan Kleinert
(ubik)
Vernietiger van BASIC 10

Wayne Johnson
(sausage)
Aanvullingen op de handleiding

Raven L. A. Kleiss
(LAK132)
MegaWAT-presentatiesoftware

Maurice van Gils
(Maurice)
BASIC 65-voorbeeldprogramma's

Andrew Owen
(Cheveron)
Toetsenbord, Sinclair-ondersteuning

Adam Barnes
(amb5l)
HDMI-expert en printplaatherziening

Wayne Rittimann, Jr.
(johnwayner)
Bugverplettering op alle niveaus

STEUNERS

3c74ce64
8-Bit Classics
@11110110100
Aaron Smith
Achim Mrotzek
Adolf Nefischer
Adrian Esdaile
Adrien Guichard
Ahmed Kablaoui
Alan Bastian Witkowski
Alan Field
Alastair Paulin-Campbell
Alberto Mercuri
Alexander Haering
Alexander Kaufmann
Alexander Niedermeier
Alexander Soppart
Alfonso Ardire
Amiga On The Lake
André Kudra
André Simeit
André Wösten
Andrea Farolfi
Andrea Minutello
Andreas Behr
Andreas Freier
Andreas Grabski
Andreas Millinger
Andreas Nopper
Andreas Ochs
Andreas Wendel Manufaktur
Andreas Zschunke
Andrew Bingham
Andrew Dixon
Andrew Mondt
Andrzej Hłuchyj
Andrzej Sawiniec
Andrzej Śliwa
Anthony W. Leal
Arkadiusz Bronowicki
Arkadiusz Kwasny
Arnaud Léandre
Arne Drews
Arne Neumann
Arne Richard Tyarks
Axel Klahr
Balaz Ondrej
Barry Thompson
Bartol Filipovic
Benjamin Maas
Bernard Alaiz
Bernhard Zorn
Bieno Marti-Braitmaier
Bigby
Bill LaGrue
Bjoerg Stojalowski
Björn Johannesson
Bjørn Melbøe
Bo Goeran Kvamme
Boerge Noest
Bolko Beutner
Brett Hallen
Brian Gajewski
Brian Green
Brian Juul Nielsen
Brian Reiter
Bryan Pope
Burkhard Franke
Byron Goodman
Cameron Roberton (KONG)
Carl Angervall
Carl Danowski
Carl Stock
Carl Wall
Carlo Pastore
Carlos Silva
Carsten Sørensen
Cenk Miroglu Miroglu
Chang sik Park
Charles A. Hutchins Jr.
Chris Guthrey
Chris Hooper
Chris Stringer
Christian Boettcher
Christian Eick
Christian Gleinser
Christian Gräfe
Christian Heffner
Christian Kersting
Christian Schiller
Christian Streck
Christian Weyer
Christian Wyk
Christoph Haug
Christoph Huck
Christoph Pross
Christopher Christopher
Christopher Kalk
Christopher Kohlert
Christopher Nelson
Christopher Taylor
Christopher Whillock
Claudio Piccinini
Claus Skrepek
Collen Blijenberg
Constantine Lignos
Crnjaninja
Daniel Auger
Daniel Julien
Daniel Lobitz
Daniel O'Connor
Daniel Teicher
Daniel Tootill
Daniel Wedin
Daniele Benetti
Daniele Gaetano Capursi
Dariusz Szczesniak
Darrell Westbury
David Asenjo Raposo
David Dillard
David Gorgon
David Norwood
David Raulo
David Ross
de voughn accooe
Dean Scully
Dennis Jeschke
Dennis Schaffers
Dennis Schierholz

Dennis Schneck
denti
Dick van Ginkel
Diego Barzon
Dierk Schneider
Dietmar Krueger
Dietmar Schinnerl
Dirk Becker
Dirk Wouters
Domingo Fivoli
DonChaos
Donn Lasher
Douglas Johnson
Dr. Leopold Winter
Dusan Sobotka
Earl Woodman
Ed Reilly
Edoardo Auteri
Eduardo Gallardo
Eduardo Luis Arana
Eirik Juliussen Olsen
Emilio Monelli
EP Technical Services
Epic Sound
Erasmus Kuhlmann
ergoGnomik
Eric Hilaire
Eric Hildebrandt
Eric Hill
Eric Jutrzenka
Erwin Reichel
Espen Skog
Evangelos Mpouras
Ewan Curtis
Fabio Zanicotti
Fabrizio Di Dio
Fabrizio Lodi
FARA Gießen GmbH
FeralChild
First Choice Auto's
Florian Rienhardt
Forum64. de
Francesco Baldassarri
Frank Fechner
Frank Glaush
Frank Gulasch
Frank Haaland
Frank Hempel
Frank Koschel
Frank Linhares
Frank Sleeuwaert
Frank Wolf
FranticFreddie
Fredrik Ramsberg
Fridun Nazaradeh
Friedel Kropp
Garrick West
Gary Lake-Schaal
Gary Pearson
Gavin Jones
Geir Sigmund Straume
Gerd Mitlaender
Giampietro Albiero
Giancarlo Valente
Gianluca Girelli
Giovanni Medina
Glen Fraser
Glen R Perye III
Glenn Main
Gordon Rimac
GRANT BYERS
Grant Louth
Gregor Bubek
Gregor Gramlich
Guido Ling
Guido von Gösseln
Guillaume Serge
Gunnar Hemmerling
Günter Hummel
Guy Simmons
Guybrush Threepwood
Hakan Blomqvist
Hans Pronk
Hans-Jörg Nett
Hans-Martin Zedlitz
Harald Dosch
Harri Salokorpi
Harry Culpan
Harry Venema
Heath Gallimore
Heinz Roesner
Heinz Stampfli
Helge Förster
Hendrik Fensch
Henning Harperath
Henri Parfait
Henrik Kühn
Holger Burmester
Holger Sturk
Howard Knibbs
Hubert de Hollain
Huberto Kusters
Hugo Maria Gerardus v.d. Aa
Humberto Castaneda
Ian Cross
IDE64 Staff
Igor Ianov
Igor Kurtes
Immo Beutler
Ingo Katte
Ingo Keck
Insanely Interested Publishing
IT-Dienstleistungen Obsieger
Ivan Elwood
Jaap HUIJSMAN
Jace Courville
Jack Wattenhofer
Jakob Schönpflug
Jakub Tyszko
James Hart
James Marshburn
James McClanahan
James Sutcliffe
Jan Bitruff
Jan Hildebrandt
Jan Iemhoff
Jan Kösters
Jan Peter Borsje
Jan Schulze
Jan Stoltenberg-Lerche
Janne Tompuri
Jannis Schulte
Jari Loukasmäki
Jason Smith
Javier Gonzalez Gonzalez
Jean-Paul Lauque
Jeffrey van der Schilden
Jens Schneider

Jens-Uwe Wessling
Jesse DiSimone
Jett Adams
Johan Arneklev
Johan Berntsson
Johan Svensson
Johannes Fitz
John Cook
John Deane
John Dupuis
John Nagi
John Rorland
John Sargeant
John Traeholt
Jon Sandelin
Jonas Bernemann
Jonathan Prosise
Joost Honig
Jordi Pakey-Rodriguez
Jöre Weber
Jörg Jungermann
Jörg Schaeffer
Jörg Weese
Josef Hesse
Josef Soucek
Josef Stohwasser
Joseph Clifford
Joseph Gerth
Jovan Crnjanin
Juan Pablo Schisano
Juan S. Cardona Iguina
JudgeBeeb
Juliussen Olsen
Juna Luis Fernandez Garcia
Jürgen Endras
Jürgen Herm Stapelberg
Jyrki Laurila
Kai Pernau
Kalle Pöyhönen
Karl Lamford
Karl-Heinz Blum
Karsten Engstler
Karsten Westebbe
katarakt
Keith McComb
Kenneth Dyke
Kenneth Joensson
Kevin Edwards
Kevin Thomasson
Kim Jorgensen
Kim Rene Jensen
Kimmo Hamalainen
Konrad Buryło
Kosmas Einbrodt
Kurt Klemm
Lachlan Glaskin
Large bits collider
Lars Becker
Lars Edelmann
Lars Slivsgaard
Lasse Lambrecht
Lau Olivier
Lee Chatt
Loan Leray
Lorenzo Quadri
Lorenzo Travagli
Lorin Millsap
Lothar James Foss
Lothar Serra Mari
Luca Papinutti
Ludek Smetana
Lukas Burger
Lutz-Peter Buchholz
Luuk Spaetgens
Mad Web Skills
MaDCz
Magnus Wiklander
Maik Diekmann
Malte Mundt
Manfred Wittemann
Manuel Beckmann
Manzano Mérida
Marc "3D-vice" Schmitt
Marc Bartel
Marc Jensen
Marc Schmidt
Marc Theunissen
Marc Tutor
Marc Wink
Marcel Buchtmann
Marcel Kante
Marco Beckers
Marco Cappellari
Marco Rivela
Marco van de Water
Marcus Gerards
Marcus Herbert
Marcus Linkert
Marek Pernicky
Mario Esposito
Mario Fetka
Mario Teschke
Mariusz Tymków
Mark Adams
Mark Anderson
Mark Green
Mark Hucker
Mark Leitiger
Mark Spezzano
Mark Watkin
Marko Rizvic
Markus Bieler
Markus Bonet
Markus Dauberschmidt
Markus Fehr
Markus Fuchs
Markus Guenther-Hirn
Markus Liukka
Markus Merz
Markus Roesgen
Markus Uttenweiler
Martin Bauhuber
Martin Benke
Martin Gendera
Martin Groß
Martin Gutenbrunner
Martin Johansen
Martin Marbach
Martin Sonnleitner
Martin Steffen
Marvin Hardy
Massimo Villani
Mathias Dellacherie
Mathieu Chouinard
Matthew Adams
Matthew Browne
Matthew Carnevale
Matthew Palmer

Matthew Santos
Matthias Barthel
Matthias Dolenc
Matthias Fischer
Matthias Frey
Matthias Grandis
Matthias Guth
Matthias Lampe
Matthias Meier
Matthias Mueller
Matthias Nofer
Matthias Schonder
Maurice Al-Khaliedy
Max Ihlenfeldt
Meeso Kim
Michael Dailly
Michael Dötsch
Michael Dreßel
Michael Fichtner
Michael Fong
Michael Geoffrey Stone
Michael Gertner
Michael Grün
Michael Habel
Michael Härtig
Michael Haynes
Michael J Burkett
Michael Jensen
Michael Jurisch
Michael Kappelgaard
Michael Kleinschmidt
Michael Lorenz
Michael Mayerhofer
Michael Nurney
Michael Rasmussen
Michael Richmond
Michael Sachse
Michael Sarbak
Michael Schneider
Michael Scholz
Michael Timm
Michael Traynor
Michael Whipp
Michal Ursiny
Michele Chiti
Michele Perini
Michele Porcu
Miguel Angel Rodriguez Jodar
Mikael Lund
Mike Betz
Mike Kastrantas
Mike Pikowski
Mikko Hämäläinen
Mikko Suontausta
Mirko Roller
Miroslav Karkus
Morgan Antonsson
Moritz
Morten Nielsen
MUBIQUO APPS,SL
Myles Cameron-Smith
Neil Moore
Nelson
neoman
Nicholas Melnick
Nikolaj Brinch Jørgensen
Nils Andreas
Nils Eilers
Nils Hammerich
Nils77
Norah Smith
Norman King
Normen Zoch
Olaf Grunert
Ole Eitels
Oliver Boerner
Oliver Brüggmann
Oliver Graf
Oliver Smith
Olivier Bori
ONEPSI LLC
oRdYNe
Osaühing Trioflex
OSHA-PROS USA
Padawer
Patrick Becher
Patrick Bürckstümmer
Patrick de Zoete
Patrick Toal
Patrick Vogt
Paul Alexander Warren
Paul Gerhardt (KONG)
Paul Jackson
Paul Johnson
Paul Kuhnast (mindrail)
Paul Massay
Paul Westlake
Paul Woegerer
Pauline Brasch
Paulo Apolonia
Pete Collin
Pete of Retrohax.net
Peter Eliades
Peter Gries
Peter Habura
Peter Herklotz
Peter Huyoff
Peter Knörzer
Peter Leswell
Peter Weile
Petri Alvinen
Philip Marien
Philip Timmermann
Philipp Rudin
Pierre Kressmann
Pieter Labie
Piotr Kmiecik
Power-on.at
Przemysław Safonow
Que Labs
R Welbourn
R-Flux
Rafał Michno
Rainer Kappler
Rainer Kopp
Rainer Weninger
Ralf Griewel
Ralf Pöscha
Ralf Reinhardt
Ralf Schenden
Ralf Smolarek
Ralf Zenker
Ralph Bauer
Ralph Wernecke
Rédl Károly
Reiner Lanowski
Remi Veilleux
Riccardo Bianchi

Richard Englert
Richard Good
Richard Menedetter
Richard Sopuch
Rick Reynolds
Rico Gruninger
Rob Dean
Robert Bernardo
Robert Eaglestone
Robert Grasböck
Robert Miles
Robert Schwan
Robert Shively
Robert Tangmar
Robert Trangmar
Rodney Xerri
Roger Olsen
Roger Pugh
Roland Attila Kett
Roland Evers
Roland Schatz
Rolf Hass
Ronald Cooper
Ronald Hunn
Ronny Hamida
Ronny Preiß
Roy van Zundert
Rüdiger Wohlfromm
Ruediger Schlenter
Rutger WIllemsen
Sampo Peltonen
Sarmad Gilani
SAS74
Sascha Hesse
Scott Halman
Scott Hollier
Scott Robison
Sebastian Baranski
Sebastian Bölling
Sebastian Felzmann
Sebastian Lipp
Sebastian Rakel
Şemseddin Moldibi
Šeth Morabito
Shawn McKee
Siegfried Hartmann
Zytex Online Store
Sigurbjorn Larusson
Sigurdur Finnsson
Simon Lawrence
Simon Wolf
spreen.digital
Stefan Haberl
Stefan Kramperth
Stefan Richter
Stefan Schultze
Stefan Sonnek
Stefan Theil
Stefan Vrampe
Stefano Canali
Stefano Mozzi
Steffen Reiersen
Stephan Bielmann
Stephen Jones
Stephen Kew
Steve Gray
Steve Kurlin
Steve Lemieux
Steven Combs
Stewart Dunn
Stuart Marsh
Sven Neumann
Sven Stache
Sven Sternberger
Sven Wiegand
Szabolcs Bence
Tantrumedia Limited
Techvana Operations Ltd.
Teddy Turmeaux
Teemu Korvenpää
The Games Foundation
Thierry Supplisson
Thieu-Duy Thai
Thomas Bierschenk
Thomas Edmister
Thomas Frauenknecht
Thomas Gitzen
Thomas Gruber
Thomas Haidler
Thomas Jager
Thomas Karlsen
Thomas Laskowski
Thomas Marschall
Thomas Niemann
Thomas Scheelen
Thomas Schilling
Thomas Tahsin-Bey
Thomas Walter
Thomas Wirtzmann
Thorsten Knoll
Thorsten Nolte
Tim Krome
Tim Waite
Timo Weirich
Timothy Blanks
Timothy Henson
Timothy Prater
Tobias Butter
Tobias Heim
Tobias Köck
Tobias Lüthi
Tommi Vasarainen
Toni Ammer
Tore Olsen
Torleif Strand
Torsten Schröder
Tuan Nguyen
Uffe Jakobsen
Ulrich Hintermeier
Ulrich Nieland
Ulrik Kruse
Urban Lindeskog
Ursula Förstle
Uwe Anfang
Uwe Boschanski
Vedran Vrbanc
Verm Project
Wayne Rittimann
Wayne Sander
Wayne Steele
Who Knows
Winfried Falkenhahn
Wolfgang Becker
Wolfgang Stabla
Worblehat
www.patop69.net
Yan B
Zoltan Markus
Zsolt Zsila

Bibliografie

[1] L. Soares en M. Stumm, "Flexsc: Flexible system call scheduling with exception-less system calls." in *Osdi*, dl. 10, 2010, blz. 1–8.

[2] N. Montfort, P. Baudoin, J. Bell, I. Bogost, J. Douglass, M. C. Marino, M. Mateas, C. Reas, M. Sample, en N. Vawter, *10 PRINT CHR $(205.5+ RND (1));: GOTO 10*. MIT Press, 2012.

[3] Actraiser, "Vic-ii for beginners: Screen modes, cheaper by the dozen," 2013. [Online]. Beschikbaar: http://dustlayer.com/vic-ii/2013/4/26/vic-ii-for-beginners-screen-modes-cheaper-by-the-dozen

Index

altpal, 50
Apple, 20

bgcolor, 49
blink, 50
bordercolor, 49
box, 53

cellcolor, 51
cgetc, 61
cinput, 62
clearattr, 51
clrscr, 47
conionit, 45
cprintf, 60
cputc, 58
cputcxy, 57, 60
cputdec, 59
cputhex, 58
cputnc, 58
cputncxy, 60
cputs, 59
cputsxy, 59

emulator, 19

fillrect, 53
Flashmenu, 87
flushkeybuf, 62
Freezer-menu, 87

getcharsetaddr, 47
getcolramoffset, 46
getkeymodstate, 62
getmapedpal, 52
getpalbank, 52
getpalbanka, 52
getscreenaddr, 46
getscreensize, 47
gohome, 55
gotox, 55
gotoxy, 55
gotoy, 55

highlight, 50
hline, 54
Hoofdmap, 109
Hulpprogrammamenu, 87
Hyppo-diensten, 95
    $D640 $00, 105
    $D640 $02, 121
    $D640 $04, 119
    $D640 $06, 133
    $D640 $08, 122
    $D640 $0A, 120
    $D640 $0C, 107
    $D640 $0E, 125
    $D640 $10, 132
    $D640 $12, 126
    $D640 $14, 128
    $D640 $16, 111
    $D640 $18, 127
    $D640 $1A, 130
    $D640 $1C, 135
    $D640 $1E, 125
    $D640 $20, 112
    $D640 $22, 110
    $D640 $24, 132
    $D640 $26, 132
    $D640 $28, 118
    $D640 $2A, 132
    $D640 $2C, 113
    $D640 $2E, 134
    $D640 $30, 115
    $D640 $32, 116
    $D640 $34, 114
    $D640 $36, 123
    $D640 $38, 103
    $D640 $3A, 106
    $D640 $3C, 109
    $D640 $3E, 124
    $D640 $40, 138
    $D640 $42, 139
    $D640 $44, 139
    $D640 $46, 138
    $D640 $48, 142

$D640 $4A, 137
$D640 $50, 144
$D640 $52, 148
$D640 $54, 144
$D640 $56, 155
$D640 $58, 144
$D640 $60, 152
$D640 $62, 140
$D640 $64, 144
$D640 $66, 140
$D640 $68, 140
$D640 $6A, 140
$D640 $6C, 152
$D640 $6E, 140
$D640 $70, 154
$D640 $72, 153
$D640 $74, 141
$D640 $76, 151
$D640 $7C, 150
$D640 $7E, 145
$D641 $00, 147
$D641 $02, 146
$D642 $00, 156
$D642 $02, 156
$D642 $04, 156
$D642 $06, 156
$D642 $10, 157
$D642 $12, 157
$D642 $14, 157
$D642 $16, 157
$D643 $xx, 149
$D67F $xx, 157
Hyppo-foutcodes, 97, 103
$01, 103
$02, 103
$03, 103
$04, 103
$05, 103
$06, 103
$07, 103
$08, 103, 128
$10, 103, 106, 128, 134, 141, 143, 151
$11, 103
$20, 104
$21, 104
$80, 104, 133
$81, 104, 134, 137
$82, 104
$83, 104
$84, 104, 123, 124, 126, 127
$85, 104, 128
$86, 104, 127
$87, 104, 107, 126
$88, 104, 114–116, 123, 124, 137
$89, 104, 130, 135
$8A, 104, 137
$8B, 104, 137
$8C, 104
$8D, 104, 125, 132
$8E, 104
$8F, 104, 137
$FF, 104
Hyppo: verplaatsen naar hoofdmap, 109

kbhit, 61

macOS, 20
MEGA Flash, 87
movedown, 56
moveleft, 56
moveright, 56
moveup, 56

OpenROMs, 22

pcprintf, 61
pcputc, 57
pcputs, 58
pcputsxy, 57

Registers

$D640, 95
$D641, 95
$D642, 95
$D643, 95
$D644, 95
$D645, 95
$D646, 95
$D647, 95
$D648, 95
$D649, 95
$D64A, 95
$D64B, 95
$D64C, 95
$D64D, 95
$D64E, 95
$D64F, 95
$D650, 95
$D651, 95
$D652, 95
$D653, 95
$D654, 95
$D655, 95
$D656, 95
$D657, 95
$D658, 95
$D659, 95
$D65A, 95
$D65B, 95
$D65C, 95
$D65D, 95
$D65E, 95
$D65F, 95
$D660, 95
$D661, 95
$D662, 95
$D663, 95
$D664, 95
$D665, 95
$D666, 95
$D667, 95
$D668, 95
$D669, 95
$D66A, 95
$D66B, 95
$D66C, 95
$D66D, 95
$D66E, 95
$D66F, 95
$D670, 95
$D671, 95
$D672, 95
$D673, 95
$D674, 95
$D675, 95
$D676, 95
$D677, 95
$D678, 95
$D679, 95
$D67A, 95
$D67B, 95
$D67C, 95
$D67D, 95
$D67E, 95
$D67F, 95
revers, 49

SD-kaart, 21
set16bitcharmode, 48
setcharsetaddr, 46
setcolramoffset, 46
setextendedattrib, 48
sethotregs, 48
setmapedpal, 52
setpalbank, 51
setpalbanka, 51
setpalentry, 53
setscreenaddr, 45
setscreensize, 47

textcolor, 49
toetsenbord, 19
togglecase, 48, 49

underline, 50

vline, 54

wherex, 57
wherey, 57