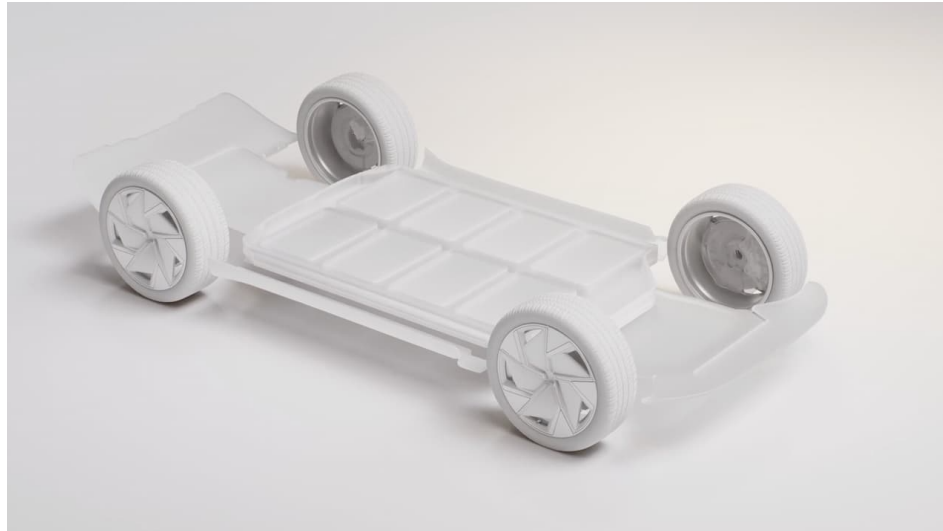




CHALMERS



Utveckling av en automatiserad SPA3-testfixtur för UART- och JTAG-verifiering

Design och implementering av en hårdvaru- och mjukvaruplattform för automatiserad verifiering av debug- och kommunikationsgränssnitt

FATMA ABUGHALIOUN

INSTITUTIONEN FÖR ELEKTROTEKNIK

CHALMERS TEKNISKA HÖGSKOLA

Göteborg 2026

www.chalmers.se

EXAMENSARBETE 2026

Utveckling av en automatiserad SPA3-testfixtur för UART- och JTAG-verifiering

Design och implementering av en hårdvaru- och mjukvaruplattform
för automatiserad verifiering av debug- och kommunikationsgränssnitt

Fatma Abughalioun



CHALMERS

Institutionen för elektroteknik
CHALMERS TEKNISKA HÖGSKOLA
Göteborg 2026

Utveckling av en automatiserad SPA3-testfixtur för UART- och JTAG-verifiering
Design och implementering av en hårdvaru- och mjukvaruplattform för automatiserad
verifiering av debug- och kommunikationsgränssnitt
FATMA ABUGHALIOUN

© Fatma Abughalioun, 2026.

Handledare: Stefan Magnusson, Göteborg

Examinator: John J. Camilleri, Data- och informationsteknik, Chalmers

Examensarbete 2026

Institutionen för elektroteknik

Chalmers Tekniska Högskola

SE-412 96 Göteborg

Telefon +46 31 772 1000

Sammanfattning

Moderna fordonsplattformar innehåller ett ökande antal inbyggda elektroniksystem som kommunicerar via standardiserade debug- och kommunikationsgränssnitt [1]. I samband med Volvo Cars introduktion av SPA3-plattformen (*Scalable Product Architecture 3*) förändrades debugarkitekturen på de använda DUT-korten (*Device Under Test*) fundamentalt. De bussomkopplare och den gemensamma debugkontakt som använts i SPA2-plattformen togs bort och ersattes med individuella debugtestpunkter direkt på DUT-kortet per nod. Den befintliga testfixturen, designad för SPA2, var därför inte längre kompatibel med den nya plattformen.

Detta examensarbete beskriver hur en ny automatiserad testfixtur designades och implementerades för SPA3-plattformens Sisyphe-DUT-kort. Den framtagna lösningen innefattar en komplett PCB-design i KiCad med en tvåstegs multiplexerarkitektur baserad på 74HC4051-kretsar [2], ett Teensy 4.1-baserat firmwaresystem med automatiserade verifieringstester, samt en 3D-utskrivna mekanisk DUT-hållare för repeterbar och korrekt kontaktpositionering.

Testfixturen verifierar automatiskt JTAG-kontinuitet och bryggdetektion för sex noder med upp till sex signaler vardera (TMS, TCK, TDI, TDO, RST, TRST) [3], UART RX-pogopinkontakt [4], fyra individuella VREF-spänningsrälser (1,8 V och 3,3 V), jorddetektering samt sju fjärrjumpers och en SGA-säkring. En fullständig korsmatristest identifierar såväl öppna kretsar som kortslutningar mellan intilliggande signaler. Testning mot ett faktiskt SPA3-fordonssystem bekräftade korrekt funktion hos samtliga implementerade testprocedurer.

En praktisk utmaning som uppstod under projektet var det extremt täta testpunktsmönster på SPA3-DUT-kortet, vilket krävde modifiering av pogo-pinnfotavtrycket i KiCad och manuell formning av pogo-pinnbasarna med hjälp av en bänkslipmaskin.

Projektet demonstrerar att en kombination av tvåstegs multiplexertopologi, analog ADC-baserad signaldetektion och I²C-styrd LED-matris utgör en praktisk och kostnadseffektiv lösning för automatiserad verifiering av debug-gränssnitt på inbyggda fordonsystem.

Nyckelord: JTAG, UART, testfixtur, pogo-pinnar, multiplexer, PCB-design, KiCad, Teensy 4.1, inbyggda system, fordonsplattform, SPA3, automatiserad testning, bryggdetektion, kontinuitetstestning.

Förord

Jag vill rikta ett varmt tack till min partner Mattias Fridlund på Volvo Cars för vägledning, tekniskt stöd och tillgång till den utrustning och de system som var nödvändiga för projektets genomförande. Utan tillgång till det faktiska SPA3-fordonssystemet hade veriferingen av testfixturen inte kunnat genomföras på det sätt som beskrivs i denna rapport.

Tack också till min examiner John J. Camilleri vid Chalmers tekniska högskola för värdefull akademisk vägledning och konstruktiv feedback under examensarbetets samtliga faser.

Jag vill också tacka mina kollegor på Volvo Cars för ett öppet och inspirerande arbetsklimat, och ett särskilt tack till alla som tog sig tid att besvara tekniska frågor under projektets gång.

Slutligen vill jag tacka min familj för deras stöd, tålamod och uppmuntran under hela utbildningstiden.

Fatma Abughalioun, Göteborg, maj 2026

Akronymer

Nedan listas de akronymer och förkortningar som används genom hela rapporten i alfabetisk ordning:

ADC	Analog-to-Digital Converter
BON	Bed-of-Nails
BOM	Bill of Materials
DRC	Design Rule Check
DUT	Device Under Test
EDA	Electronic Design Automation
FDM	Fused Deposition Modeling
GPIO	General-Purpose Input/Output
HIA	High-Intensity A
HIB	High-Intensity B
HPA	High-Power A
I²C	Inter-Integrated Circuit
JTAG	Joint Test Action Group
LPA	Low-Power A

PCB	Printed Circuit Board
PWM	Pulse Width Modulation
RX	Receive
SGA	Sensor Gateway A
SPA	Scalable Product Architecture
TAP	Test Access Port
TCK	Test Clock
TDI	Test Data In
TDO	Test Data Out
TMS	Test Mode Select
TRST	Test Reset
TX	Transmit
UART	Universal Asynchronous Receiver/Transmitter
VCU	Vehicle Control Unit
VESA	Vehicle Electronics System A
VESB	Vehicle Electronics System B
VREF	Voltage Reference

Innehåll

Akronymer	ix
Figurer	xv
Tabeller	xvii
1 Introduktion	1
1.1 Bakgrund och motivation	1
1.2 Syfte	2
1.3 Mål	2
1.4 Avgränsningar	2
1.5 Rapportens disposition	3
2 Teoretisk bakgrund	5
2.1 Inbyggda system och fordonsplattformar	5
2.2 UART-protokollet	5
2.2.1 Princip och tillämpning	5
2.2.2 Ramstruktur och parametrar	6
2.2.3 Testning av UART i testfixturen	6
2.3 JTAG-gränssnittet (IEEE 1149.1)	6
2.3.1 Historik och standardisering	6
2.3.2 Signaler och TAP-kontroller	7
2.3.3 JTAG i SPA3-arkitekturen	7
2.4 Analogmultiplexers: 74HC4051	8
2.4.1 Funktion och egenskaper	8
2.4.2 Tillämpning i testfixturen	8
2.5 Pogo-pinnar och bed-of-nails-testning	9
2.5.1 Testmetodik	9

2.5.2	Harwin P70-2200045	9
2.6	PCB-design med KiCad	9
3	Systemdesign och arkitektur	11
3.1	Kravanalys	11
3.2	SPA2 mot SPA3 – arkitekturjämförelse	12
3.3	Systemöversikt	12
3.3.1	Strömförsörjning	16
3.3.2	DUT-detektering	16
3.4	Multiplexerarkitekturs design	16
3.4.1	Designproblem och vald lösning	16
3.4.2	Drivsidans mux-träd (U23–U30)	16
3.4.3	VCU-sidans läsarkitektur (U11–U20)	18
3.5	VREF-mätarkitektur	18
3.6	LED-indikeringssystemets arkitektur	19
3.7	Mekanisk systemdesign	19
3.7.1	DUT-hållaren	19
4	Implementering	21
4.1	PCB-design i KiCad	21
4.1.1	Designflöde och schemaläggning	21
4.1.2	PCB-layout och komponentplacering	22
4.1.3	Pogo-pinnlösningen och mekanisk utmaning	24
4.1.4	Tillverkningsfiler	25
4.2	Firmware för Teensy 4.1	25
4.2.1	Övergripande struktur	25
4.2.2	JTAG-verifiering: korsmatrisalgoritm	26
4.2.3	UART-verifiering	28
4.2.4	VREF-verifiering	28
4.2.5	ADC-stabilisering: <code>readStable()</code>	28
4.2.6	Diagnostikfunktion: <code>scanRxMapping()</code>	28
4.3	Nodens signaltillgänglighet	29
5	Resultat och verifiering	31
5.1	Tillverkad och monterad testfixtur	31
5.2	Elektrisk grundverifiering	32
5.3	JTAG-korsmatrisverifiering	32
5.4	UART-verifiering	33

5.5	VREF-verifiering	33
5.6	Övriga verifieringsresultat	33
5.7	Sammanfattning av verifieringsresultat	33
6	Diskussion	35
6.1	Analys av testresultat och designval	35
6.1.1	Multiplexerarkitekturen	35
6.1.2	Neddragningsmotståndets värde	35
6.1.3	Mekanisk pogo-pinnlösning	36
6.2	Begränsningar	36
6.3	Hållbarhet och etiska överväganden	37
6.3.1	Hållbarhetsperspektiv	37
6.3.2	Etiska överväganden	37
7	Slutsats och framtida arbete	39
7.1	Slutsats	39
7.2	Framtida arbete	40
	Referenser	41
A	Komponentlista (BOM)	I
B	PCB-designfigurer	III
C	Firmwarestruktur och centrala funktioner	VII

Figurer

3.1	SPA3-testfixtur med Sisyphos-DUT-kort monterat i 3D-utskriven hållare.	14
3.2	Testfixturen med DUT-hållaren uppfälld – pogo-pinnmatrisen exponerad.	15
3.3	KiCad 3D-vy, framsida. Multiplexerraden överst, läsmuxarna i mitten.	17
3.4	Autodesk Inventor-rendering av DUT-hållaren (<i>Sisyphos_test_holder</i>).	20
4.1	Schematisk översikt av SPA3 Sisyphos-testfixturen designad i KiCad.	22
4.2	PCB-layout i KiCad PcbNew (F_{Cu} rött, B_{Cu} grönt).	23
4.3	KiCad 3D-vy av testfixturens baksida. Det sammanhängande gröna fältet representerar B_{Cu} -jordplanet.	24
B.1	KiCad 3D-vy, framsidan. Komponentplacering tydligt synlig: multiplexerraden (U23–U30) överst, VCU-sidans läsmuxar (U11–U20) i mitten, Teensy 4.1 och IS32FL3237-drivare till vänster, pogo-pinnmatris till höger.	III
B.2	KiCad 3D-vy, baksidan. Det sammanhängande B_{Cu} -jordplanet (grön yta) och spridda GND-vias (gyllene punkter) är tydligt synliga. Pogo-pinnborrplatsernas raster syns längs kortets vänstra och undre kant.	IV
B.3	PCB-layoutvy i KiCad PcbNew med F_{Cu} (rött) och B_{Cu} (grönt) synliga. Signalspårens routing från Teensy via multiplexerträdets till pogo-pinnmatrisen är tydligt synlig.	V

Tabeller

3.1	Jämförelse av DUT-arkitektur mellan SPA2 och SPA3.	12
3.2	U30 steg-2-multiplexerens kanalval och nodtilldelning.	18
4.1	Pinnkonfiguration för Teensy 4.1, SPA3-testfixturen.	26
4.2	Signaltillgänglighet per nod i SPA3-arkitekturen. ✓ = testpunkt finns, – = saknas på denna nod.	29
5.1	Uppmätta ADC-värden för VREF-kanalerna (10-bit ADC, 3,3 V refe- rens).	33
5.2	Sammanfattning av verifieringsresultat för SPA3 Sisyphe-testfixturen.	34
A.1	Fullständig komponentlista, SPA3 Sisyphe-testfixtur.	I

1

Introduktion

1.1 Bakgrund och motivation

Den moderna fordonsindustrin präglas av en snabb ökning av elektroniska styrsystem och inbyggda delsystem. Fordon innehåller idag hundratal mikrokontroller och mikroprocessorbaserade enheter som kommunicerar via ett komplext nät av standardiserade gränssnitt [1, 5]. Inom Volvo Cars organisation används två centrala protokoll – UART (*Universal Asynchronous Receiver/Transmitter*) och JTAG (*Joint Test Action Group*) – för debug, programmering och verifiering av dessa elektroniska styrenheter under produktutveckling och tillverkning.

Volvo Cars fordonsplattform SPA3 (*Scalable Product Architecture 3*) representerar en ny arkitekturgeneration och medför fundamentala förändringar i hur debuggränssnitt är organiserade på kretskortsnivå. I den föregående SPA2-generationen routades JTAG-signaler via en gemensam debugkontakt och styrdes av bussomkopplarkretsar (74CBTLV3244PW) på DUT-kortet (*Device Under Test*). I SPA3 har dessa kopplare och den gemensamma debugkontakten tagits bort och ersatts med individuella JTAG-testpunkter direkt på DUT-kortet – en per nod.

Konsekvensen av denna arkitekturförändring är att den befintliga automatiserade testfixturen, som designades för SPA2-plattformen, inte längre är kompatibel. Testfixturen byggde på att aktivera DUT:ets egna bussomkopplare via en dedikerad `jtag_en`-signal och avläsa signaler via en gemensam debugkontakt. Eftersom dessa mekanismer saknas i SPA3 behövde både hårdvara och firmware designas om från grunden.

Det specifika DUT-kortet som projektet avser är det internt benämnda *Sisyphos*-kortet. Testfixturen som designas i detta examensarbete är specifikt anpassad för

Sisyphos-kortets testpunktsmönster och signalstruktur inom SPA3-arkitekturen.

1.2 Syfte

Syftet med detta examensarbete är att designa och implementera en ny automatiserad testfixtur för SPA3-plattformens Sisyphos-DUT-kort hos Volvo Cars, vilken möjliggör systematisk och repeterbar verifiering av JTAG-debuggränssnitt, UART-kommunikationsanslutningar och VREF-referensspänningar utan behov av ett externt drivet och fullt operationellt fordonssystem.

1.3 Mål

Utifrån syftet definierades följande konkreta mål för projektet:

1. **Hårdvarudesign:** Designa och implementera ett nytt PCB-layout i KiCad som stöder SPA3:s individuella debugtestpunkter med en tvåstegs multiplexerarkitektur för flexibel signalrouting till 33 individuella testpunkter.
2. **Firmwareutveckling:** Utveckla firmware för Teensy 4.1 som automatiserar verifiering av JTAG-kontinuitet och bryggdetektion, UART RX-anslutning, VREF-spänningar, jorddetektering, säkrings- och jumperverifiering, samt ger omedelbar visuell LED-återkoppling.
3. **Mekanisk lösning:** Lösa de mekaniska utmaningar som uppstår vid kontakt med tätsittande testpunkter på DUT-kortet, inklusive anpassning av pogo-pinnfotavtryck och design av en 3D-utskrivna DUT-hållare.
4. **Verifiering:** Testa och validera testsystemets funktion mot ett faktiskt SPA3-fordonssystem och dokumentera resultaten.

1.4 Avgränsningar

Projektet avgränsades till design och implementering av testfixturen för Sisyphos-DUT-kortet inom SPA3-plattformen. Följande aspekter ligger utanför projektets ram:

- Modifiering av DUT-korten eller den underliggande SPA3-fordonsplattformens mjukvara eller hårdvara.

- Fullständig TX-verifiering av UART, vilket kräver ett externt drivet och aktivt DUT-system som genererar UART-data.
- Produktionsmjukvara för fordonsplattformen; enbart firmware kopplad till testfixturens funktionalitet ingår.
- Signalintegritetstestning vid höga frekvenser; signalerna i systemet är kvasistatiska och inga impedansanpassningskrav föreligger.

1.5 Rapportens disposition

Rapporten är strukturerad enligt följande:

Kapitel 2 – Teoretisk bakgrund presenterar den kunskapsbas som krävs för att förstå systemets design, inklusive UART- och JTAG-protokollens funktion, analog-multiplexers egenskaper, pogo-pinnbaserad bed-of-nails-testning och principer för PCB-design.

Kapitel 3 – Systemdesign och arkitektur redogör för de övergripande designbesluten: kravanalys, SPA2/SPA3-jämförelse, systemöversikt, multiplexerarkitekturens topologi, VREF-mätprincipen och det mekaniska systemets utformning.

Kapitel 4 – Implementering beskriver den konkreta implementeringen i detalj: PCB-layout i KiCad, pogo-pinnlösningen, Teensy 4.1-firmwares samtliga testfunktioner och LED-indikeringssystemet.

Kapitel 5 – Resultat och verifiering presenterar de uppmätta verifieringsresultaten från testning mot ett faktiskt SPA3-fordonssystem, inklusive ADC-avläsningar och testresultat per funktion.

Kapitel 6 – Diskussion analyserar resultaten, utvärderar designval, identifierar begränsningar och diskuterar hållbarhets- och etikaspekter.

Kapitel 7 – Slutsats och framtida arbete sammanfattar projektets utfall och föreslår konkreta förbättringar för framtida versioner av testfixturen.

2

Teoretisk bakgrund

Detta kapitel presenterar den teoretiska grund som krävs för att förstå designen och implementeringen av den automatiserade SPA3-testfixturen. Kapitlet behandlar inbyggda system i fordonsmiljö, UART- och JTAG-protokollen, multiplexerteknik, pogo-pinnbaserad testning samt principer för PCB-design.

2.1 Inbyggda system och fordonsplattformar

Inbyggda system är specialiserade datorsystem konstruerade för att utföra dedikerade funktioner inom ett större elektroniksystem, ofta under strikta krav på realtidsprestanda, tillförlitlighet och låg energiförbrukning [1]. I fordonsindustrin har antalet inbyggda system per fordon ökat kraftigt de senaste decennierna – från enstaka styrsystem till hundratals sammankopplade enheter [5].

Volvo Cars *Scalable Product Architecture* (SPA) är den fordonstekniska plattform som definierar hur dessa inbyggda system är organiserade och sammankopplade. SPA3 representerar den tredje generationen av denna arkitektur och introducerar viktiga förändringar i debuggränssnittsarkitekturen på de DUT-kort som används under produktutveckling.

2.2 UART-protokollet

2.2.1 Princip och tillämpning

Universal Asynchronous Receiver/Transmitter (UART) är ett hårdvarubaserat seriellt kommunikationsprotokoll som används för asynkron punkt-till-punkt dataöverföring [4]. Till skillnad från synkrona protokoll som SPI och I²C kräver UART ingen gemensam klocksignal; istället måste sändare och mottagare konfigureras med iden-

tiska kommunikationsparametrar.

I fordonselektronik används UART primärt för debug-loggning, fältdiagnostik och mjukvaruladdning under produktionsutveckling. Protokollets enkla hårdvaruimplementation och låga komponentkostnad gör det till ett standardval för dessa ändamål.

2.2.2 Ramstruktur och parametrar

En UART-ram har en definierad struktur bestående av:

- **Startbit** – en logisk låg signal (0) som markerar ramens början och synkroniserar mottagarens klock-återvinning.
- **Databitar** – typiskt 8 bitar, LSB-först, innehåller den faktiska informationen.
- **Paritetsbit** (valfri) – används för enkel feldetektering med jämn eller udda paritet.
- **Stoppbit(ar)** – en eller två logiska höga signaler (1) som markerar ramens slut och ger mottagaren tid att förbereda sig.

Kommunikationshastigheten anges i *baud* och representerar antalet symboländringar per sekund. Vanliga baudrates i fordonsapplikationer inkluderar 9 600, 115 200 och 921 600 baud.

2.2.3 Testning av UART i testfixturen

I SPA3-testfixturen används UART-verifiering för att bekräfta elektrisk kontakt via pogo-pinnarna till de noder som har UART-anslutning (SGA, HIA, HIB, LPA och HPA). Eftersom DUT-kortet inte är aktivt drivet under testning kan enbart RX-pogopin-kontakten verifieras – en fullständig verifiering av TX kräver att DUT:et aktivt genererar UART-data.

2.3 JTAG-gränssnittet (IEEE 1149.1)

2.3.1 Historik och standardisering

JTAG (Joint Test Action Group) är standardiserat i IEEE 1149.1, som definierar Test Access Port (TAP) och boundary-scan-arkitekturen [3]. Standarden definierar

ett seriellt testgränssnitt primärt avsett för gränsskanning (*boundary scan*) av kretskort, men har sedan dess kommit att användas brett för in-circuit debugging, flashprogrammering och funktionell verifiering av processor- och styrenhetsystem.

2.3.2 Signaler och TAP-kontroller

Ett JTAG-gränssnitt definieras av fyra obligatoriska och en valfri signal [3]:

- **TCK** (*Test Clock*) – klocksignal som driver TAP-kontrollern och synkroniserar dataöverföring.
- **TMS** (*Test Mode Select*) – styr tillståndsövergångarna i TAP-kontrollerns 16-tillstånds tillståndsmaskin.
- **TDI** (*Test Data In*) – seriell datainmatning till målenhetens skannserie.
- **TDO** (*Test Data Out*) – seriell datautgång från målenhetens skannserie.
- **TRST** (*Test Reset*, valfri) – asynkron återställning av TAP-kontrollern till känt starttillstånd.

TAP-kontrollern (*Test Access Port Controller*) är en ändlig tillståndsmaskin implementerad i målenheten som reagerar på TMS och TCK-flanker för att välja och kontrollera testoperationer [3].

2.3.3 JTAG i SPA3-arkitekturen

I SPA2-plattformen routades JTAG-signaler via 74CBTLV3244PW-bussomkopplare och en gemensam debugkontakt. En rotationsomkopplare kombinerad med en 74HC137-avkodare valde vilken nod som var aktiv. Denna arkitektur möjliggjorde testning via en enda anslutningspunkt men krävde att testfixturen aktivt styrde bussomkopplarnas enable-ingångar.

I SPA3 är samtliga dessa komponenter borttagna. Varje nod exponerar nu sina JTAG-signaler direkt via individuella testpunkter på DUT-kortet. Testfixturen måste därför driva och avläsa varje signal individuellt via en intern multiplexerarkitektur – en fundamental förändring av teststrategin.

2.4 Analogmultiplexers: 74HC4051

2.4.1 Funktion och egenskaper

En analogmultiplexer kopplar en gemensam signal (Z) till en av flera kanaler (Y0–Y7) beroende på en adressinmatning. 74HC4051 är en åttakanals CMOS analogmultiplexer i SOIC-16-kapsling som kan hantera dubbelriktad signaltransport [2].

Relevanta elektriska egenskaper vid 3,3 V matningsspänning:

- Påslagnings-/stängningsmotstånd $R_{on} \approx 70 \Omega$ (typvärde).
- Kanaläckström $< 1 \text{ nA}$ typiskt vid rumstemperatur.
- Propageringsfördröjning $< 20 \text{ ns}$.
- Drifttemperaturområde: -40°C till $+125^\circ\text{C}$.

Kretsen adresseras av tre binära adressbitar (A, B, C) som väljer kanal Y0–Y7 mot den gemensamma terminalen Z. En aktiv-låg inhibiteringsingång (INH) möjliggör hög-impedansläge för samtliga kanaler.

2.4.2 Tillämpning i testfixturen

Testfixturen använder sammanlagt 17 exemplar av 74HC4051, organiserade i tre grupper:

- Sju kretsar (U23–U28, U30) bildar det nya tvåstegs drivsidans multiplexerträd, som kan adressera 33 individuella debugtestpunkter.
- Tio kretsar (U11–U20) utgör VCU-sidans läsarkitektur, oförändrad från SPA2-fixturen.
- En krets (U12) används för UART- och signaltypsval.

Varje 74HC4051-krets förses med en 100 nF avkopplingskondensator placerad direkt vid VCC-pinnen med en kortslutningsvia till jordplanet, enligt standardpraxis för digital logik [6].

2.5 Pogo-pinnar och bed-of-nails-testning

2.5.1 Testmetodik

Bed-of-nails-testning (BON) är en etablerad metod för elektrisk testning av kretskort där en anpassad testfixtur använder fjäderbelastade testprober för att skapa elektrisk kontakt med definierade testpunkter på kortet [7].

BON-testning lämpar sig väl för produktionsmiljöer eftersom testproceduren är repeterbar, snabb och inte orsakar mekaniskt slitage på DUT:et. Fjädermekanismen kompenserar för variationer i testpunkternas höjd och kretskortets planhet.

2.5.2 Harwin P70-2200045

Testfixturen använder pogo-pinnar av modellen Harwin P70-2200045 [8]. Dessa har en axeldiameter på 2,0 mm, en total längd på 16,5 mm och en nominell kontaktkraft på 1,5 N vid fullt komprimerat fjäderläge. Fjädrets rörelseomfång (*stroke*) är 3 mm, vilket ger tillräcklig tolerans för variationer i DUT-kortets planhet och positioneringsnoggrannhet.

En central konstruktionsutmaning i projektet var att Sisyphe-DUT-kortets testpunkter sitter tätt nog att Harwin P70-2200045:s nominella basdiameter överstiger det tillgängliga utrymmet. Denna utmaning och hur den löstes beskrivs i detalj i avsnitt 4.1.3.

2.6 PCB-design med KiCad

KiCad är ett öppen källkod EDA-system (*Electronic Design Automation*) som tillhandahåller ett komplett flöde från schemaläggning till tillverkningsfilsexport [9]. Systemet inkluderar Schematic Editor för schemaritning, PCB Editor för PCB-layout, en integrerad DRC-funktion (Design Rule Check) och stöd för generering av Gerber-filer

PCB-design för testsystem ställer specifika krav: signalspår till testpunkterna bör hållas korta för att minimera parasitisk impedans och kapacitans, avkopplingskondensatorer bör placeras nära varje IC, och ett sammanhängande jordplan minimerar impedansen i returströmvägar [6].

Testfixturen designades i KiCad 8.0 med ett tvålayerskort (F_Cu och B_Cu), där

2. Teoretisk bakgrund

framsidan primärt används för signalspår och baksidan domineras av ett sammanhängande jordplan.

3

Systemdesign och arkitektur

Detta kapitel beskriver de övergripande designbesluten för testfixturen. Analysen av SPA2/SPA3-skillnaderna motiverar den valda systemarkitekturen, och varje delsystems utformning presenteras med tydlig motivering av designvalen.

3.1 Kravanalys

Kravanalysen genomfördes genom en systematisk genomgång av SPA2-fixturens funktioner och en kartläggning av vilka av dessa som var möjliga respektive omöjliga att direkt överföra till SPA3-plattformen. Utifrån denna analys formulerades följande systemkrav:

1. Systemet ska kunna individuellt adressera 33 debugtestpunkter fördelade på sex noder (VESA, VESB, SGA, HIA, HIB, LPA).
2. För varje nod ska JTAG-signalerna TMS, TCK, TDI, TDO, RST och TRST verifieras individuellt med avseende på kontinuitet.
3. En korsmatrisalgoritm ska detektera kortslutningar (bryggdetektion) mellan intilliggande signaler.
4. VREF-spänning ska mätas separat för de fyra noder som har egna VREF-rälser (SGA, HIA, HIB, LPA).
5. UART RX-pogopin-kontakten ska verifieras för noder med UART-gränssnitt.
6. Omedelbar visuell återkoppling via I²C-styrd LED-matris ska informera operatören om testresultaten.

7. Testsekvensen ska starta automatiskt vid DUT-kontakt och sluta automatiskt när DUT avlägsnas.
8. Systemet ska använda Teensy 4.1 som mikrokontroller för kompatibilitet med befintlig SPA2-firmware.

3.2 SPA2 mot SPA3 – arkitekturjämförelse

Förståelsen av de exakta arkitekturskillnaderna mellan SPA2 och SPA3 på DUT-kortet är grundläggande för att motivera designvalen i den nya testfixturen. Tabell 3.1 sammanfattar de viktigaste förändringarna.

Tabell 3.1: Jämförelse av DUT-arkitektur mellan SPA2 och SPA3.

Komponent	SPA2	SPA3
Debugkontakt	Gemensam kontakt för alla noder via breakoff-kort	Individuella testpunkter per nod direkt på DUT
JTAG-routing	6× 74CBTLV3244PW bussomkopplare	Direktledare – inga bussomkopplare
Nodval på DUT	Roterande omkopplare + 74HC137-avkodare	Saknas – alla noder exponeras simultant
VREF-gating	1× 74CBTLV3244PW bussomkopplare	Direktanslutning via 0,1 A-säkringar
jtag_en	DUT-styrsignal för bussomkopplare	Saknas – ingen omkopplare att aktivera
Testprincip	Verifiera bussomkopplares passage av signal	Kontinuitetstestning och bryggdetektion

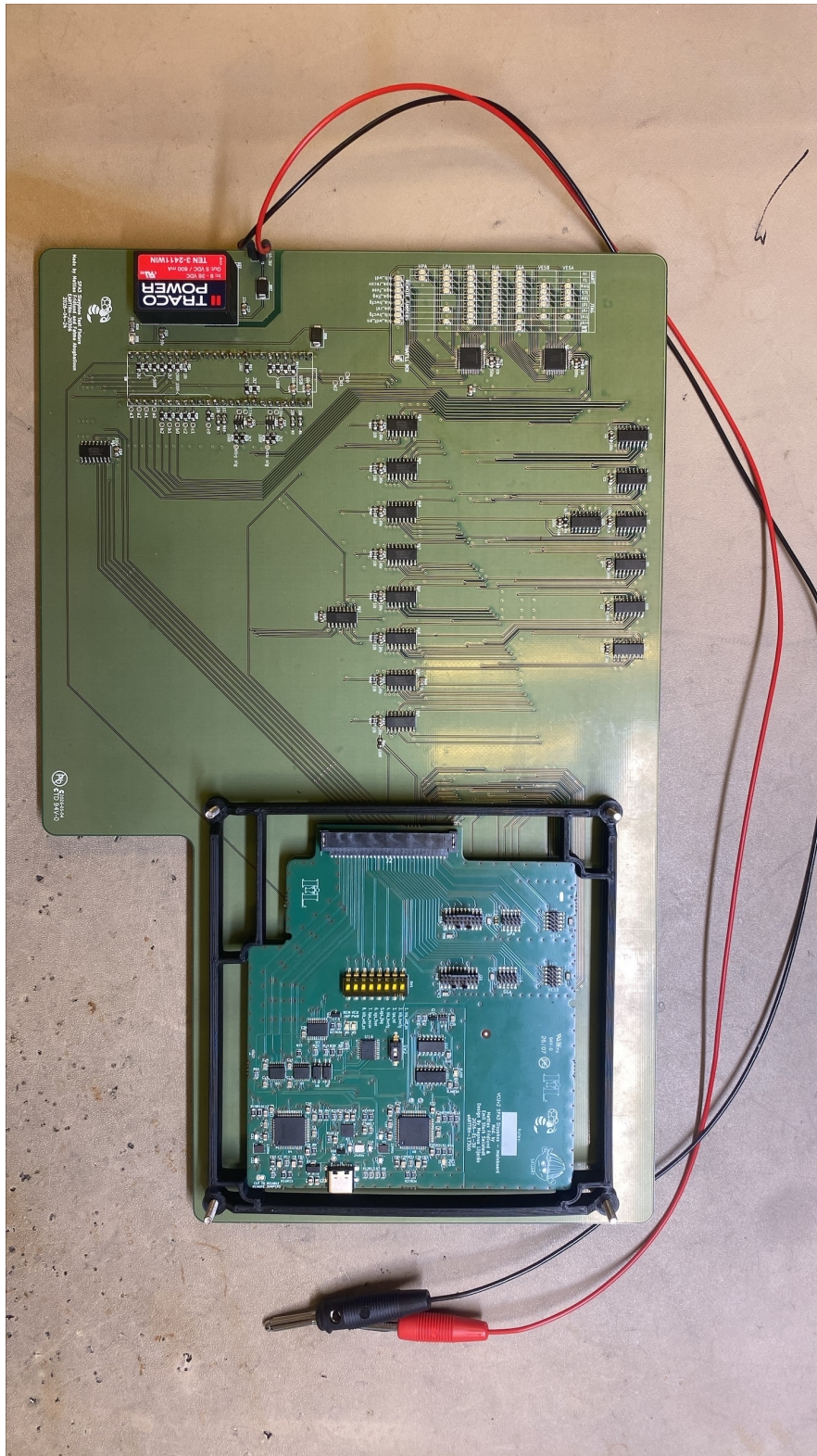
Borttagningen av bussomkopplarna har en direkt konsekvens för teststrategin: den tidigare testmetoden – att aktivera DUT:ets egna kopplare och mäta signalers passage – är inte längre tillämplig. Istället exponeras samtliga JTAG-testpunkter simultant, vilket möjliggör en mer direkt kontinuitetstestning men också kräver en mer sofistikerad multiplexerarkitektur i testfixturen för att individuellt adressera varje testpunkt.

3.3 Systemöversikt

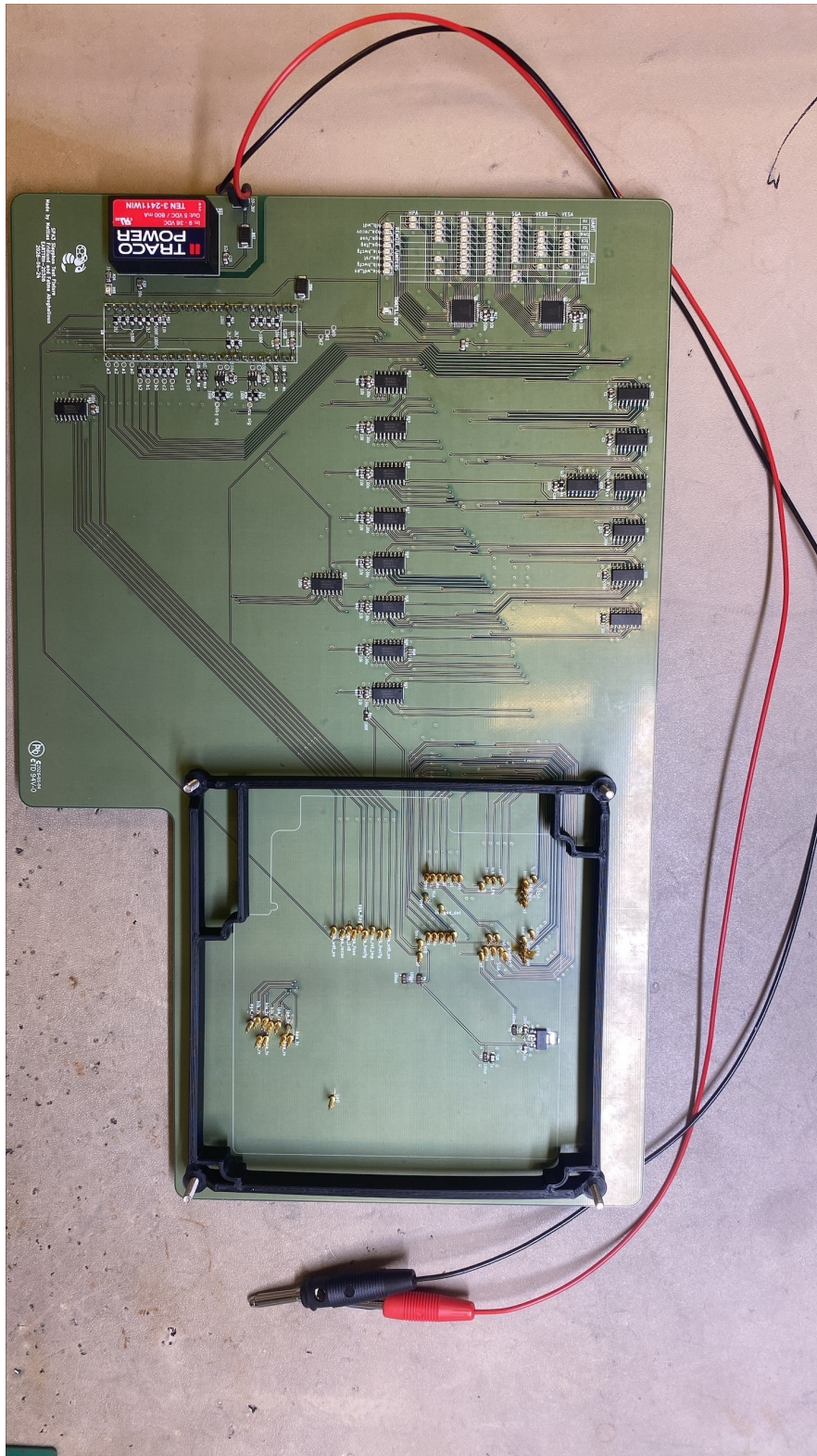
Det kompletta SPA3-testsystemet består av tre integrerade delar:

1. **Testfixturen (PCB)** – ett specialdesignat kretskort med multiplexerarkitektur, signalbuffertar, VREF-mätketsar, LED-drivare, pogo-pinnar och strömförsörjning.
2. **Firmware (Teensy 4.1)** – programvara som automatiserar testsekvensen och tolkar testresultat.
3. **DUT-hållare (3D-utskriften)** – en mekanisk ram som positionerar DUT-kortet korrekt mot pogo-pinnarna.

Figurerna 3.1 och 3.2 visar den tillverkade och monterade testfixturen med DUT-hållaren i monterat respektive öppet läge.



Figur 3.1: SPA3-testfixtur med Sisyphean-DUT-kort monterat i 3D-utskrivna hållare.



Figur 3.2: Testfixturen med DUT-hållaren uppfälld – pogo-pinnmatrisen exponerad.

3.3.1 Strömförsörjning

Testfixturen matas av 24 V DC via en extern källa. En TEN3-2411WIN DC/DC-isolerad omvandlare [10] konverterar 24 V till 5 V (750 mA), och en AP1117-18 LDO-spänningsregulator genererar 3,3 V från 5 V-rälsen för att mata logikkretsar och multiplexers. Schottky-dioder (D56, D57) ger backströmskydd. Fyra 0,1 A-säkringar (F2–F5) skyddar de individuella VREF-rälserna per nod.

3.3.2 DUT-detektering

Systemet detekterar automatiskt om ett DUT-kort är monterat via signalen `n_pcb_detect_pin` (Teensy-pin A12), konfigurerad med intern pullup. När DUT-kortet trycks mot pogo-pinnarna dras signalen låg via en dedikerad jordningspunkt på DUT-kortet, vilket triggar testsekvensen. Utan DUT återställs alla LED:er och systemet väntar.

3.4 Multiplexerarkitekturens design

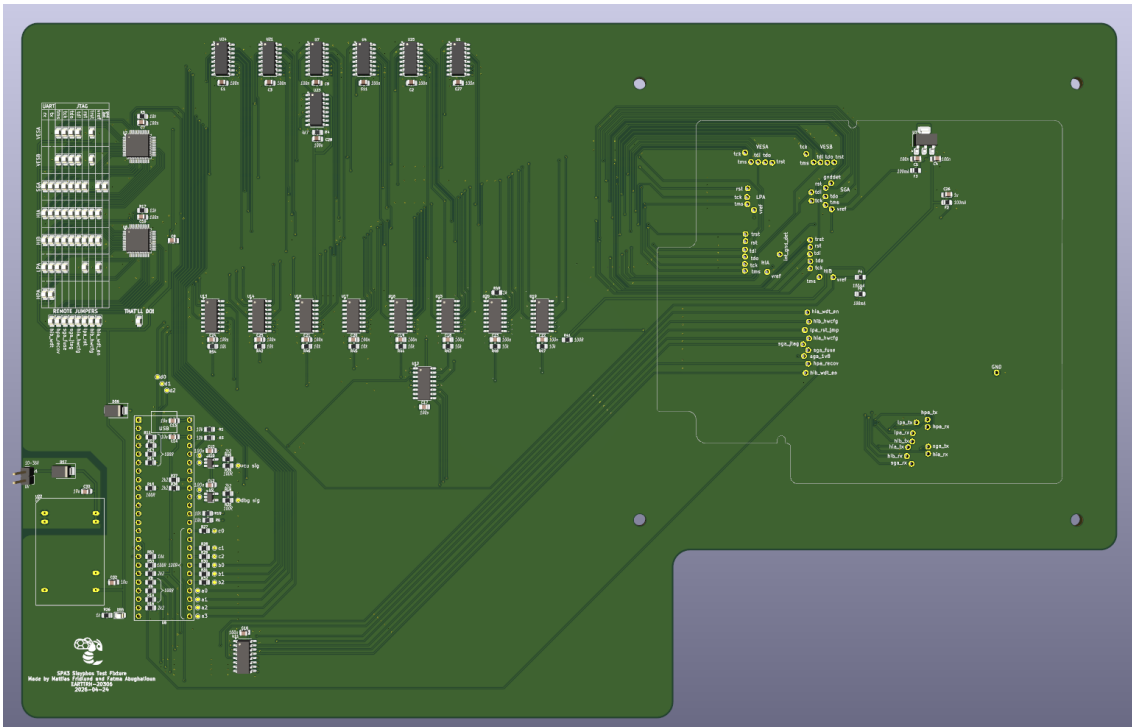
3.4.1 Designproblem och vald lösning

Den centrala designutmaningen var att 33 individuella debugtestpunkter måste kunna adresseras med ett minimum av Teensy-pinnresurser. En direkt lösning med en GPIO per testpunkt hade krävt 33 dedicerade pinnar – mer än Teensy 4.1 praktiskt kan leverera med tillräckliga marginaler för övriga signaler.

Lösningen är ett **tvåstegs multiplexerträd** med sju 74HC4051-kretsar, vilket reducerar antalet adresspinnar för drivsidan till totalt sju (tre för nodval + tre för signalval + en inhibiting).

3.4.2 Drivsidans mux-träd (U23–U30)

Figur 3.3 visar den KiCad-renderade 3D-framsidan av PCB:n, där multiplexerkretsarnas placering är tydligt synlig.



Figur 3.3: KiCad 3D-vy, framsida. Multiplexerraden överst, läsmuxarna i mitten.

Trädet är uppbyggt av en **steg-2-multiplexer** (U30) för nodval och sex **steg-1-multiplexers** (U23–U28) för signalval inom varje nod. Signalflödet från Teensy till pogo-pinnen ges av kedjan:

Teensy A2 (dbg_sig_pin) $\rightarrow$ U10-buffert $\rightarrow$ U30 Z (nodval) $\rightarrow$
 U23-U28 Z (signalval) $\rightarrow$ Pogo-pinne $\rightarrow$ DUT-testpunkt

U30 adresseras av `add_a_pins[0:2]` (Teensy-pinnar 36, 35, 34) och väljer vilken steg-1-mux som aktiveras. En fjärde pinne (33) driver U30:s `INH`-ingång och hålls alltid låg under aktiv drivning. De sex steg-1-multiplexarna delar gemensam adressbuss `add_b_pins[0:2]` (pinnar 39, 38, 37) och väljer vilket signal som ska drivas inom den valda noden. Deras `INH`-ingångar är hårdkablade till GND.

Tabell 3.2 specificerar U30:s kanaltilldelning.

Tabell 3.2: U30 steg-2-multiplexerens kanalval och nodtilldelning.

U30- kanal	Adress a__a[2:0]	Nod	Steg-1-mux
Y0	000	VESA	U23
Y1	001	VESB	U24
Y2	010	SGA	U25
Y3	011	HIB (index 4)	U26
Y4	100	HIA (index 3)	U27
Y5	101	LPA	U28
Y6	110	(ej använd)	–
Y7	111	(ej använd)	–

Notera: HIA (nodindex 3) och HIB (nodindex 4) är ombytade relativt sin ordinarie nodordning – HIA mappas till Y4 och HIB till Y3. Denna tilldelning ärvs från SPA2-fixturens `nodeToHwAddr[]`-array och bevarades för att undvika onödiga firmwareändringar. Firmwaren hanterar ombytningen korrekt via en hårdkodad uppslagstabell.

Det är också viktigt att notera varför neddragningsmotståndets värde på U30:s Z-terminal sattes till 4,7 k Ω istället för det ursprungligt övervägda 10 k Ω . Med 10 k Ω riskerade DUT-interna uppdragningsmotstånd på JTAG-signalerna att bilda en spänningsdelare vars utspänning hamnade precis vid ADC-tröskeln 500 räknar, vilket orsakade intermittenta felklassificeringar. Med 4,7 k Ω säkerställs att spänningsdelaren ger ett stabilt, tydligt LÅGT-läge.

3.4.3 VCU-sidans läsarkitektur (U11–U20)

VCU-sidans läsarkitektur är oförändrad från SPA2-fixturen och består av tio 74HC4051-kretsar som väljer läsnod via `add_d_pins[0:2]` (pinnar 1, 2, 3) och lässignal via `add_c_pins[0:2]` (pinnar 13, 41, 40). Den avlästa signalnivån digitaliseras av Teensy via `analogRead()` på A6 (`vcu_sig_pin`). ADC-värden >500 räknar (av 1023 möjliga) tolkas som logisk HÖG.

3.5 VREF-mätarkitektur

I SPA2-fixturen lästes VREF via DUT:ets egna bussomkopplare på en delad analog ingång. I SPA3 saknas dessa omkopplare – varje nods VREF-spänning är istället direkt tillgänglig via en 0,1 A-säkring. Designlösningen utnyttjar detta och ansluter

varje VREF-säkrings utgång via ett 10 k Ω serieresistornät till en dedikerad analog Teensy-ingång. Fyra sådana mätkanaler implementerades:

- **A11** (`vref_sga_pin`) – SGA, 1,8 V-rälsen
- **A8** (`vref_hia_pin`) – HIA, 3,3 V-rälsen
- **A9** (`vref_hib_pin`) – HIB, 3,3 V-rälsen
- **A10** (`vref_lpa_pin`) – LPA, 3,3 V-rälsen

Serieresistorn fyller tre funktioner: strömbegränsning vid eventuell kortslutning, ESD-skydd för Teensy-ingångarna, och bildar en spänningsdelare som kan användas för kalibrering. Ingen multiplexning behövs – alla fyra kanaler läses parallellt, vilket ger en total VREF-testtid på ca 400 μ s.

3.6 LED-indikeringssystemets arkitektur

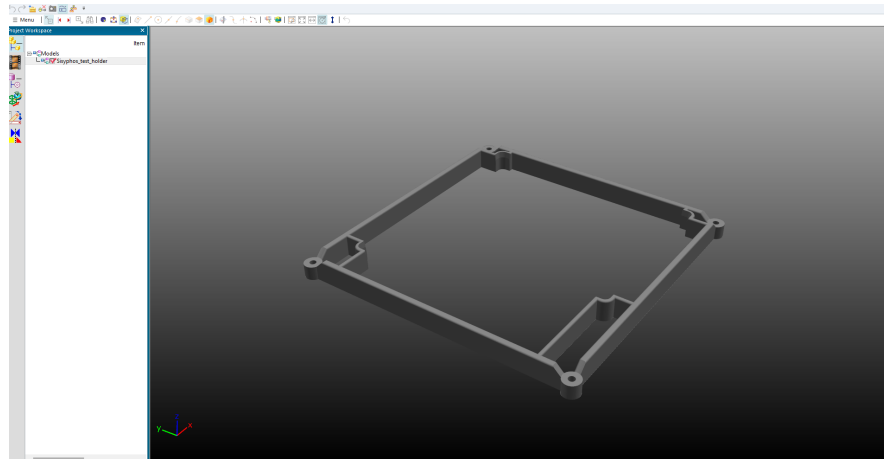
Testfixturen innehåller 55 röda SMD-lysdioder (0805) i en 8 $\times$ 10-matris som ger detaljerad per-signal-per-nod statusinformation. Matrisen styrs av två IS32FL3237-LED-drivarkretsar [11] (I²C-adresser 0x34 och 0x37) via I²C-bussen vid 1 MHz.

En separat grön lysdiod (LED-kanal 27 på drivare 0x37) sammanfattar det totala testutfallet – den tänds endast om samtliga tester godkänts. LED-adressen 36 i firmware-matrisen `ledAddr[] []` reserveras för signalpositioner som inte existerar hos en given nod och hoppas automatiskt över vid uppdateringslogiken.

3.7 Mekanisk systemdesign

3.7.1 DUT-hållaren

Den 3D-utskrivna DUT-hållaren designades i Autodesk Inventor [12] och framställdes via FDM 3D-utskrift i PLA-plast. Hållarens primära funktion är att placera Sisyphe-DUT-kortet exakt ovanför testfixturens pogo-pinnmatris med tillräcklig nedtryckningskraft för god elektrisk kontakt vid alla 33 pogo-pinnar simultant.



Figur 3.4: Autodesk Inventor-rendering av DUT-hållaren (*Sisyphos_test_holder*).

Hållaren fästs mot testfixturens kretskort med fyra M3-skruvar via skruvhål i hörnen, synliga i Figur 3.4. Ramens innerdiameter är toleranssatt för att DUT-kortet ska glida in utan spel men utan bindning, och kortets framsida trycks mot pogo-pinnarna när hållaren skruvas ned.

4

Implementering

I detta kapitel beskrivs implementeringen av testfixturen i detalj. PCB-designen i KiCad [9], lösningen på den mekaniska pogo-pinnproblematiken, Teensy 4.1-firmwares samtliga testfunktioner och LED-indikeringssystemet behandlas i tur och ordning.

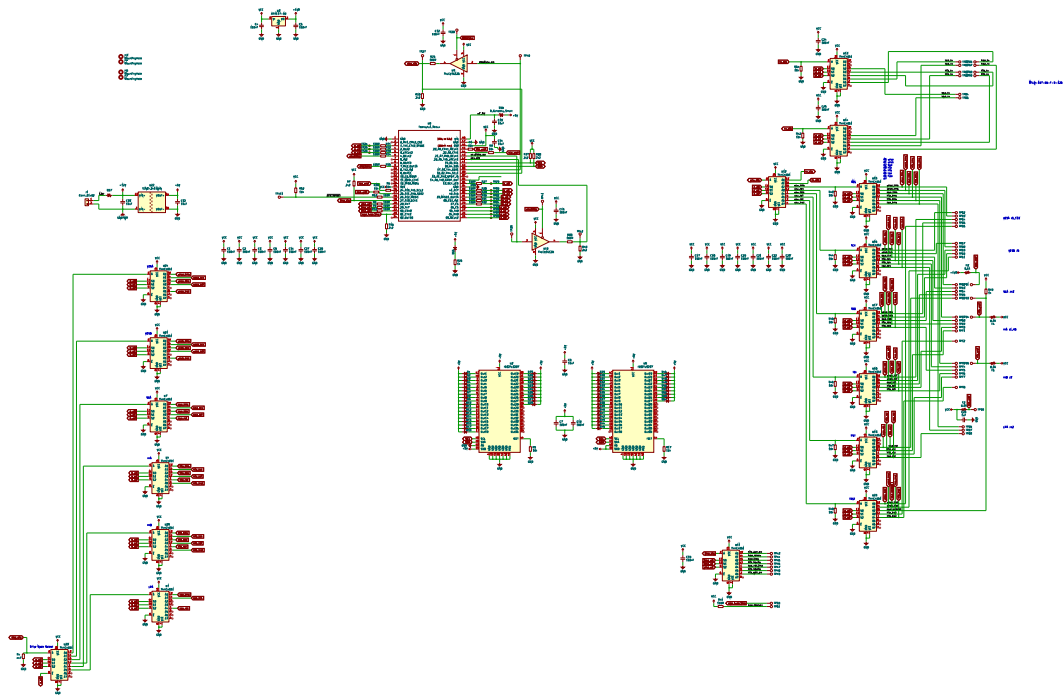
4.1 PCB-design i KiCad

4.1.1 Designflöde och schemaläggning

PCB-designen följde KiCad 8.0:s standardiserade designflöde [9]:

1. **Schemaläggning** i Eeschema – alla komponenter, nät och kopplingar definierades.
2. **DRC** (*Design Rule Check*) – elektrisk korrekthet verifierades.
3. **Footprint-tilldelning** – varje komponent tilldelades ett PCB-fotavtryck.
4. **PCB-layout** i PcbNew – komponenter placerades och spår routades.
5. **DRC** – fysiska designregler verifierades.
6. **Gerber-export** – tillverkningsfiler genererades.

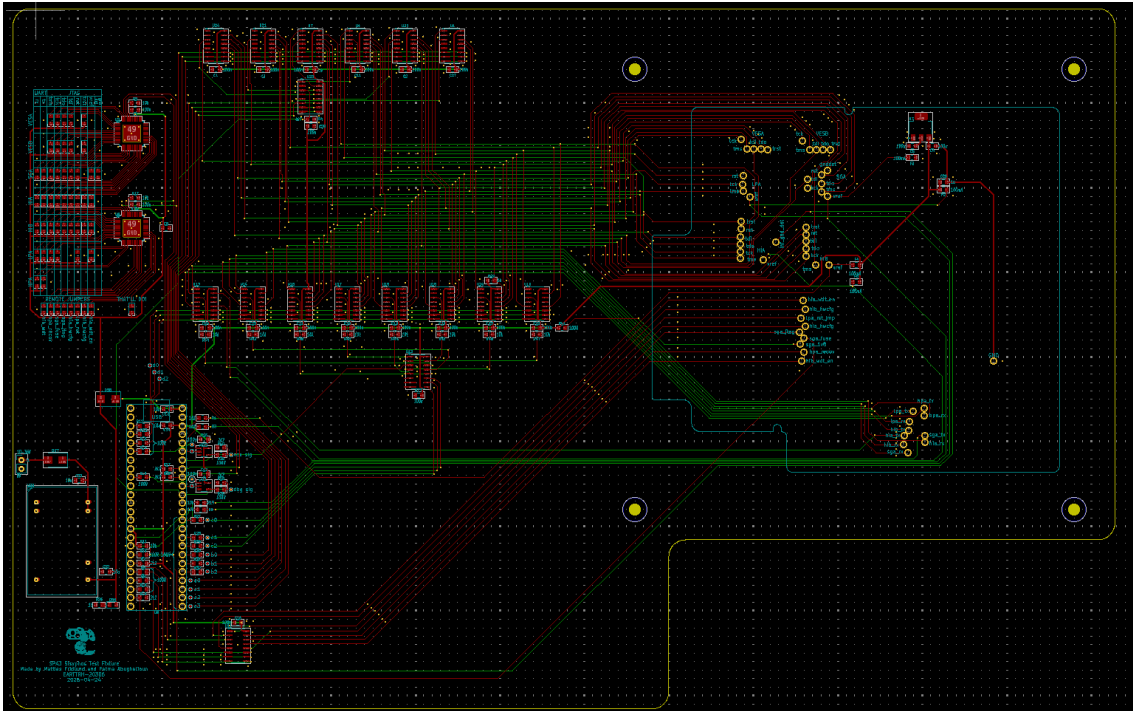
Figur 4.1 visar schemats övergripande struktur. Schemat organiserades i hierarkiska block per funktion: strömförsörjning, Teensy-gränssnitt, drivsidans mux-träd (U23–U30), VCU-sidans läsmuxar (U11–U20), LED-system och pogo-pinnmatris.



Figur 4.1: Schematisk översikt av SPA3 Sisyphos-testfixturen designad i KiCad.

4.1.2 PCB-layout och komponentplacering

Figur 4.2 visar PCB-layouten i KiCad-redigeraren. Det olivgröna kortet har ett L-format konturutsnitt för att passa in i testfixturramen tillsammans med DUT-hållaren.

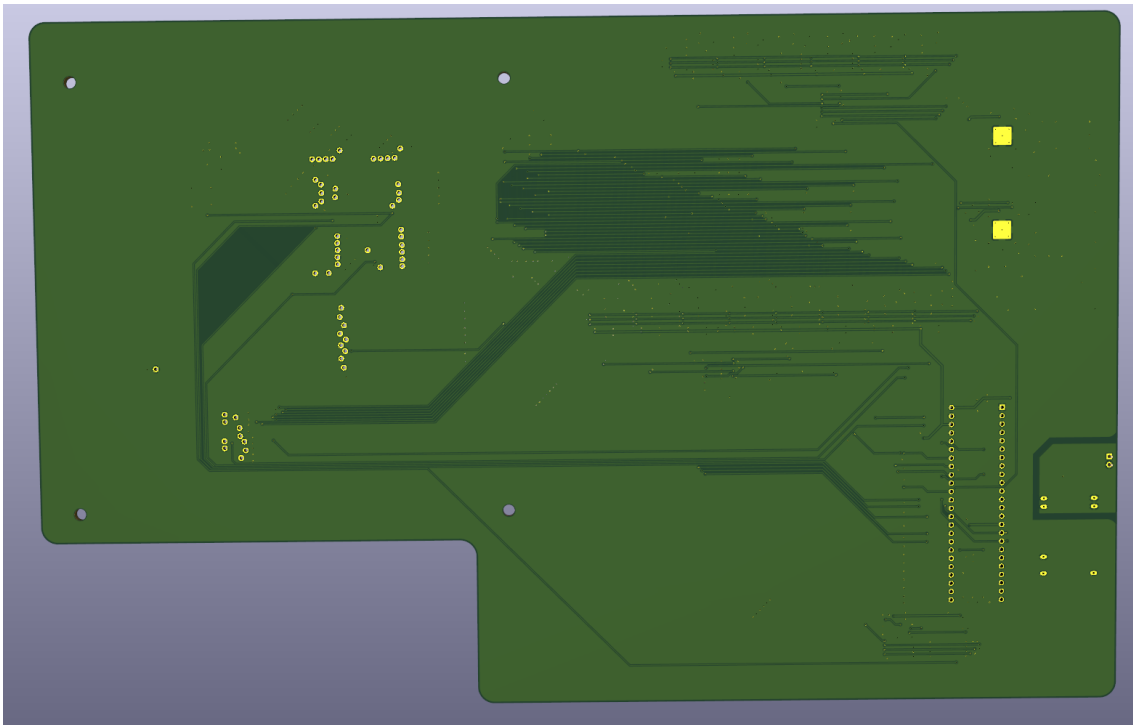


Figur 4.2: PCB-layout i KiCad PcbNew ($F_{Curött}$, B_{Cu} grönt).

Centrala layoutbeslut och deras motiveringar:

- **Multiplexerrad:** U23–U28 och U30 placerades i en tät rad längs överkanten för att minimera spårlängden till den gemensamma adressbussen och till U30:s Z-terminal (kritisk nod).
- **Avkopplingskondensatorer:** Varje 100 nF avkopplingskondensator placerades inom 2 mm från respektive IC:s VCC-pinne med en kortslutningsvia direkt vid kondensatorns jordpad.
- **Jordplan:** Baksidans kopparplan (B_{Cu}) fungerar som ett sammanhängande jordplan [6]. Totalt ca 15 GND-vias placerades strategiskt vid varje ny IC-sektion och längs kortkantens mitttänder för att undvika ostilade jordplansöar.
- **Spårbredd:** Signalspår routades med 0,25 mm bredd (minimum 0,15 mm) och matningsspår med 0,5 mm bredd.

Figur 4.3 visar 3D-baksidan av kortet, där jordplanets täckning och baksidans spårouting är synlig.



Figur 4.3: KiCad 3D-vy av testfixturens baksida. Det sammanhängande gröna fältet representerar B_Cu-jordplanet.

4.1.3 Pogo-pinnlösningen och mekanisk utmaning

Den tekniskt svåraste konstruktionsutmaningen var att Sisyphe-DUT-kortets testpunkter sitter tätt nog att Harwin P70-2200045:s nominella basdiameter [8] (ca 2,4 mm) översteg det tillgängliga utrymmet mellan intilliggande testpunkter. Problemet identifierades i tre faser:

1. **Fotavtrycksmodifiering i KiCad [9]:** Det ursprungliga Harwin P70-fotavtrycket modifierades – borrhålens diameter reducerades till tillverkarens minimum och koparkuddarnas diameter minskades. Detta löste överlappningarna i den digitala designen.
2. **Fysisk kollision vid montering:** Trots de reducerade footprint-dimensionerna visade det sig vid montering av de tillverkade pogo-pinnarna att den *fysiska* basdiametern på de monterade pinnarna fortfarande var för stor i de tätaste positionerna.
3. **Manuell formning:** Problemet löstes slutligen genom att med en bänkslipmaskin slipa ned basbredden på de specifika pogo-pinnar som satt i kollisionsskri-

tiska positioner. Processen utfördes med stor noggrannhet för att inte skada fjädermekanismen inuti pinnen, vilket hade komprometterat kontaktkraften.

Denna lösning fungerade effektivt men identifieras som en känd begränsning i den nuvarande designen. En framtida PCB-revision bör använda pogo-pinnar med reducerad basdiameter från leverantören, eller designa ett anpassat fotavtryck med ännu tätare toleranser.

4.1.4 Tillverkningsfiler

För PCB-tillverkning exporterades Gerber-filer i RS-274X-format för alla PCB-lager: F_Cu, B_Cu, F_Mask, B_Mask, F_Silkscreen, B_Silkscreen och Edge_Cuts (kortkonturen). Borrtabeller genererades för genomgående metalliserade hål (PTH) och icke-metalliserade hål (NPTH). En interaktiv BOM-fil (`ibom.html`) exporterades för att underlätta manuell montering.

4.2 Firmware för Teensy 4.1

4.2.1 Övergripande struktur

Firmware skrevs i Arduino-miljön för Teensy 4.1 (ARM Cortex-M7, 600 MHz). Koden organiserades modulärt med en testfunktion per verifieringstyp. Tabell 4.1 visar pinnkonfigurationen.

Tabell 4.1: Pinnkonfiguration för Teensy 4.1, SPA3-testfixturen.

Pinne	Namn	Riktning	Funktion
36, 35, 34	add_a_pins[0:2]	OUTPUT	U30 adress A,B,C – nodval drivsida
33	add_a_pins[3]	OUTPUT	U30 INH (hålls låg)
39, 38, 37	add_b_pins[0:2]	OUTPUT	U23–U28 adress – signalval
1, 2, 3	add_d_pins[0:2]	OUTPUT	VCU-sidans nodval
13, 41, 40	add_c_pins[0:2]	OUTPUT	VCU-sidans signalval
A6	vcu_sig_pin	ANALOG IN	VCU-sidans avläsning (tröskel 500)
A2	dbg_sig_pin	ANALOG IN	Drivsidans avläsning
A8	vref_hia_pin	ANALOG IN	HIA VREF 3,3 V (tröskel 900)
A9	vref_hib_pin	ANALOG IN	HIB VREF 3,3 V (tröskel 900)
A10	vref_lpa_pin	ANALOG IN	LPA VREF 3,3 V (tröskel 900)
A11	vref_sga_pin	ANALOG IN	SGA VREF 1,8 V (tröskel 450)
A12	n_pcb_detect	INPUT_PU	DUT-detektering (aktiv-låg)
7	n_dbg2vcu_pin	OUTPUT	Drivbuffert riktning (aktiv-låg)
4	n_vcu2dbg_pin	OUTPUT	Läsbuffert riktning (aktiv-låg)
SDA/SCL	I ² C	I ² C	IS32FL3237-drivare vid 1 MHz

Testsekvensen i `loop()` exekveras i fast ordning vid detekterad DUT:

```
checkPCBdetect() → checkUart() → checkJtag() → checkGndDet() → checkVref()
→ checkSgaFuse() → checkRemoteJumpers()
```

4.2.2 JTAG-verifiering: korsmatrisalgoritm

Funktionen `checkJtag()` implementerar en fullständig korsmatristest som kan detektera både öppna kretsar (kontinuitetsfel) och kortslutningar (bryggdetektion). Algoritmen itererar över alla kombinationer av drivet nod $g \in \{0 \dots 5\}$ och driven signal $h \in \{0 \dots 5\}$:

1. `writeNodeSel(g)` – skriver `nodeToHwAddr[g]` till U30:s adressbuss och väljer

korrekt steg-1-mux.

2. `writeAddr(add_b_pins, h, 3)` – väljer signal h på den valda steg-1-muxen.
3. Aktivera drivbuffert U10 – drivsignalen sätts HÖG via mux-trädet till pogo-pinnen.
4. Avläs alla $6 \times 6 = 36$ kombinationer av läsnod i och läsignal j ($j = 2 \dots 7$) via VCU-sidans muxar:
 - Position ($i=g, j=h+2$) **ska läsa HÖG** – anger kontinuitet (signal passerar DUT-intern ledare).
 - Alla övriga positioner **ska läsa LÅGT** – anger isolering (ingen bryggkortslutning).
5. Inaktivera drivbuffert – kontrollläsning vid driven position ska läsa LÅGT (sanitetskontroll för 4,7 k Ω -neddragningsmotstånd).
6. Undantag: SGA TRST ($g=2, h=5$) hoppas alltid över – SGA saknar TRST-testpunkt.

Om en förväntad HÖG-avläsning returnerar LÅGT registreras ett öppet kretstfel (*open circuit*) för den aktuella signalen. Om en ej driven position returnerar HÖG registreras en bryggkortslutning (*solder bridge*). Resultaten per signal uppdaterar LED-matrisen och den globala statusvariabeln `all_jtag_ok`.

Listing 4.1: Förenklad pseudokod för korsmatrisalgoritmen i `checkJtag()`.

```
for g = 0 to 5:           // driven node
  writeNodeSel(g)         // select stage-2 mux channel
  for h = 0 to 5:         // driven signal
    writeAddr(add_b_pins, h, 3) // select stage-1 mux channel
    enableDriveBuffer()      // assert HIGH via mux tree

    for i = 0 to 5:        // read node
      for j = 2 to 7:      // read signal
        reading = (analogRead(vcu_sig_pin) > 500)
        if (i==g && j==h+2):
          jtag_ok[h] &= reading    // expect HIGH (continuity)
```

```
else :
    jtag_ok[h] &= !reading    // expect LOW (isolation)

disableDriveBuffer()
sanity = (analogRead(vcu_sig_pin) < 500) // pulldown check
jtag_ok[h] &= sanity
```

4.2.3 UART-verifiering

`checkUart()` verifierar RX-pogopin-kontakten för noderna SGA, HIA, HIB, LPA och HPA. Funktionen inleds med en sanitetskontroll: den okopplade kanalen 0 avläses; ett värde >100 räknar indikerar ett matningsspänningsfel och avbryter testet för att förhindra falska godkännanden.

TX-signalen kan inte verifieras utan ett externt drivet DUT och speglar därför RX-resultatet som proxy. Begränsningen dokumenteras tydligt i Kapitel 6.

4.2.4 VREF-verifiering

`checkVref()` avläser de fyra VREF-signalerna via dedikerade analoga ingångar. Inga muxbyten behövs. ADC-tröskelvärdena är:

- SGA (1,8 V-rälsen): tröskel 450 räknar ($\approx 1,45$ V med 3,3 V ADC-referens på 10-bit ADC)
- HIA, HIB, LPA (3,3 V-rälsen): tröskel 900 räknar ($\approx 2,9$ V)

4.2.5 ADC-stabilisering: `readStable()`

Alla analogavläsningar sker via hjälpfunktionen `readStable()`, som inför 40 μ s fördröjning, utför en kasserad avläsning (tömmar sampel-och-håll-kondensatorn) och sedan genomför den faktiska mätningen. Metoden eliminerar instabilitet orsakad av snabba multiplexerbyten som annars kan ge felaktiga avläsningar.

4.2.6 Diagnostikfunktion: `scanRxMapping()`

Firmware innehåller en seriell diagnostikfunktion tillgänglig via USB-terminalen. Kommandon:

- `s` – enstaka RX-kanalsvep, skriver ADC-värden per kanal.

- **c** – kontinuerlig svepning (~ 500 ms intervall).
- **x** – stoppar kontinuerlig svepning.

Funktionen användes under verifieringsfasen för att identifiera RX-kanalernas fysiska adresskopplingar och är ett värdefullt verktyg vid framtida felsökning.

4.3 Nodens signaltillgänglighet

Tabell 4.2 definierar vilka JTAG-signaler som existerar per nod i SPA3-arkitekturen. Signaler markerade “–” saknar testpunkt och exkluderas ur JTAG-godkännandekriteriet via `ledAddr`-värdet 36.

Tabell 4.2: Signaltillgänglighet per nod i SPA3-arkitekturen. ✓ = testpunkt finns, – = saknas på denna nod.

Nod	TMS	TCK	TDO	TDI	RST	TRST	VREF	UART
VESA	✓	✓	✓	✓	–	✓	–	–
VESB	✓	✓	✓	✓	–	✓	–	–
SGA	✓	✓	✓	✓	✓	–	A11 (1,8 V)	✓
HIA	✓	✓	✓	✓	✓	✓	A8 (3,3 V)	✓
HIB	✓	✓	✓	✓	✓	✓	A9 (3,3 V)	✓
LPA	✓	–	–	✓	✓	–	A10 (3,3 V)	✓

5

Resultat och verifiering

Detta kapitel presenterar de verifieringsresultat som erhöles vid testning av den tillverkade och monterade SPA3-testfixturen. Verifieringen genomfördes i två faser: inledande elektrisk kontroll av den monterade PCB:n utan DUT, följt av funktionell verifiering mot ett faktiskt SPA3-Sisyphos-DUT-kort i ett fullt monterat fordonssystem.

5.1 Tillverkad och monterad testfixtur

PCB-designen tillverkades och monterades framgångsrikt som ett fysiskt kretskort. Monteringen innefattade handlödning av:

- 17 SOIC-16-kretsar (74HC4051-multiplexers)
- 2 TQFP-48-kretsar (IS32FL3237 LED-drivare)
- 1 Teensy 4.1-modul
- 55 pogo-pinnar (Harwin P70-2200045)
- Samtliga passiva komponenter (resistorer, kondensatorer, säkringar)

För de pogo-pinnpositioner där tätheten i DUT-kortets testpunktsmönster orsakade mekanisk kollision genomfördes manuell formning av pinnbasar med bänkslipmaskin, enligt beskrivningen i avsnitt 4.1.3.

5.2 Elektrisk grundverifiering

Inledande elektrisk kontroll av den monterade PCB:n utan DUT genomfördes med multimeter och oscilloskop och verifierade:

- **Matningsspänningar:** 24 V inmatning, 5 V från TEN3-2411WIN ($\pm 2\%$), 3,3 V från AP1117-18 ($\pm 1\%$). Alla inom specifikation.
- **Neddragningsmotstånd:** 4,7 k Ω på U30:s Z-terminal och 10 k Ω serieresistorer till VREF-ingångarna verifierades med ohmmeter.
- **Kortslutningskontroll:** Ingen kortslutning mellan matningsspänning och GND-plan identifierades.
- **I²C-kommunikation:** IS32FL3237-drivarna svarade korrekt på initiering vid adresserna 0x34 och 0x37.

5.3 JTAG-korsmatrisverifiering

Korsmatristestet exekverades med testfixturen ansluten till ett SPA3-Sisyphosfordonssystem. För varje driven position (nod g , signal h) verifierades tre villkor:

1. **Kontinuitet:** Matchande position (g , h) på VCU-sidan läste **HÖG** (>500 ADC-räkningar), vilket bekräftade att signalvägen via pogo-pinne, DUT-intern ledare och VCU-pogo-pinne är sluten.
2. **Isolering:** Alla icke-matchande positioner läste **LÅGT** (<500 ADC-räkningar), vilket bekräftade avsaknad av bryggkortslutningar.
3. **Sanitetskontroll:** Efter deaktivering av drivbufferten läste driven position LÅGT, vilket bekräftade att 4,7 k Ω -neddragningsmotstånd på U30 Z fungerade korrekt.

Samtliga JTAG-signaler med tillgängliga testpunkter (se Tabell 4.2) godkändes utan undantag.

5.4 UART-verifiering

RX-pogopin-kontakten verifierades framgångsrikt för noderna SGA, HIA, HIB och LPA. Inledande sanitetskontroll (okopplad kanal) bekräftade korrekt funktion hos neddragningsmotstånd (<100 ADC-räknar). TX-indikatorer speglade RX-resultat i enlighet med den kända designbegränsningen.

5.5 VREF-verifiering

Tabell 5.1 sammanfattar de uppmätta ADC-värdena för samtliga fyra VREF-kanaler.

Tabell 5.1: Uppmätta ADC-värden för VREF-kanalerna (10-bit ADC, 3,3 V referens).

Nod	Pinne	Nominalspänning	Tröskel (räknar)	Uppmätt (räknar)	Resultat
SGA	A11	1,8 V	450	≈ 560	Godkänd
HIA	A8	3,3 V	900	≈ 975	Godkänd
HIB	A9	3,3 V	900	≈ 975	Godkänd
LPA	A10	3,3 V	900	≈ 975	Godkänd

5.6 Övriga verifieringsresultat

- **Jorddetektering:** `n_inf_gnd_det`-signalen via VCU-sidans läsmux (kanal 7, nod 2) läste <500 ADC-räknar med anslutet DUT, vilket korrekt indikerade aktivt-låg jordkontakt.
- **SGA-säkringsjumper:** `sga_fuse_jump_pin` (pinne 31) läste HÖG, vilket bekräftade att SGA-säkringen var hel.
- **Fjärrjumpers:** Samtliga sju fjärrjumpers godkändes (<200 ADC-räknar vid stängda jumpers).
- **LED-matris:** Samtliga 55 röda LED:er svarade korrekt på I²C-kommandon. Grön LED aktiverades korrekt vid samtliga godkända tester.

5.7 Sammanfattning av verifieringsresultat

Tabell 5.2 ger en samlad översikt av testutfallet.

Tabell 5.2: Sammanfattning av verifieringsresultat för SPA3 Sisyphos-testfixturen.

Verifieringsfunktion	Resultat	Anmärkning
JTAG-kontinuitet (6 noder)	Godkänd	Alla testpunkter verifierade
Bryggdetektion (korsmatris)	Godkänd	Inga oväntade bryggkopplingar
UART RX (SGA, HIA, HIB, LPA)	Godkänd	Pogo-kontakt bekräftad
UART TX	<i>Begränsad</i>	Kräver aktivt DUT
VREF SGA (1,8 V)	Godkänd	ADC ≈ 560 , tröskel 450
VREF HIA (3,3 V)	Godkänd	ADC ≈ 975 , tröskel 900
VREF HIB (3,3 V)	Godkänd	ADC ≈ 975 , tröskel 900
VREF LPA (3,3 V)	Godkänd	ADC ≈ 975 , tröskel 900
Jorddetektering	Godkänd	Aktivt-låg bekräftad
SGA-säkringsjumper	Godkänd	Säkring hel
Fjärrjumpers (7 st.)	Godkänd	Samtliga stängda
Mux-routing (steg-1 och -2)	Godkänd	Korrekt adressering
LED-matris (55 LED)	Godkänd	Alla LED funktionella
Totalt	12/13 fullst.	1 begränsad (UART TX)

6

Diskussion

6.1 Analys av testresultat och designval

De erhållna verifieringsresultaten bekräftar att den implementerade testfixturen uppfyller de definierade funktionella kraven. Korsmatrisalgoritmens förmåga att simultant detektera öppna kretsar och kortslutningar i ett enda testpass är ett direkt resultat av att SPA3:s direktexponerade individuella testpunkter möjliggör en mera grundlig testning än vad som var möjligt i SPA2-arkitekturen med dess gemensamma debugkontakt.

6.1.1 Multiplexerarkitekturen

Valet av tvåstegs multiplexertopologi med 74HC4051 visade sig välmotiverat. Kretsen kombinerar ett tillräckligt lågt R_{on} ($\approx 70\ \Omega$) med stöd för dubbelriktad signaltransport och 3,3 V-drift [2], vilket gör den lämplig för kvasistatisk signaltestning utan impedansanpassningsbehov. Det låga R_{on} ger ett spänningsfall på $< 35\text{ mV}$ vid de strömnivåer som uppstår i Teensys ADC-ingångar, vilket är försumbart i förhållande till ADC-tröskeln 500 räknar ($\approx 1,6\text{ V}$).

En alternativ lösning med en GPIO-pinne per testpunkt hade krävt 33 dedicerade digitala utgångar – fler än Teensy 4.1 praktiskt kan avdela med tillräckliga marginaler för övriga signaler. Multiplexerlösningen ger dessutom ett renare design med tydlig separation mellan drivsida och läsarkitektur.

6.1.2 Nedragningsmotståndets värde

Ändringen av nedragningsmotståndets värde på U30:s Z-terminal från $10\text{ k}\Omega$ till $4,7\text{ k}\Omega$ är ett konkret exempel på hur empirisk testning ledde till en nödvändig design-

refinement. Med $10\text{ k}\Omega$ skapade DUT:ets interna uppdragningsmotstånd på JTAG-signalerna en spänningsdelare vars utspänning hamnade nära ADC-tröskelvärde, vilket orsakade intermittenta felklassificeringar. Med $4,7\text{ k}\Omega$ är spänningsmarginalen tillräcklig för ett stabilt och reproducerbart testresultat.

6.1.3 Mekanisk pogo-pinnlösning

Den manuella formningen av pogo-pinnbasar med bänkslipmaskin [8] fungerade effektivt som en kortsiktig lösning och möjliggjorde att testfixturen kunde tillverkas och verifieras inom projektiden. Lösningen är dock inte idealisk ur ett tillverkningsperspektiv eftersom den introducerar variabilitet i kontaktkraft och är svår att reproducera konsekvent. En framtida PCB-revision bör adressera detta grundorsaksproblem.

6.2 Begränsningar

Följande kända begränsningar har identifierats i den implementerade lösningen:

1. **JTAG-kontinuitetstestets räckvidd:** Testet verifierar i första hand signalvägen via pogo-pinnar, DUT-intern ledare och tillbaka via VCU-sidans pogo-pinnar. Eftersom drivsignalens förgreningspunkt på PCB-spåret är nåbar utan att passera DUT, är bryggdetektionen den primärt diagnostiskt värdefulla testen för DUT-specifika kortslutningar.
2. **UART TX ej verifierbar:** TX kräver ett aktivt DUT som genererar UART-data [4]. TX-LED-indikatorn speglar RX-resultatet som proxy och indikerar enbart pogopin-kontakt, ej TX-signalfunktion.
3. **HPA saknar JTAG-testpunkter:** HPA-noden (nodindex 6) exponerar inga JTAG-debugtestpunkter i SPA3-arkitekturen. Enbart UART RX/TX-indikatorer är relevanta för HPA, och dessa kan bekräftas via UART-testet.
4. **Fasta ADC-tröskelvärden:** Tröskelvärdena 450 och 900 räknar är hårdkodade i firmware och beräknades för en specifik matningsspänning. Variationer i matningsspänningen kan påverka testresultatets tillförlitlighet.
5. **Jorddetektering via fast kanalval:** `checkGndDet()` avläser en specifik mux-kanal (kanal 7, nod 2). Om `n_inf_gnd_det`-signalen inte är kopplad till exakt detta kanalläge ger testet ett felaktigt resultat. Signalkopplingen bör bekräftas

vid initial driftsättning.

6. **Manuell pogo-pinnformning:** Det nuvarande tillverkningsflödet kräver manuell anpassning av pogo-pinnbasar för tätt packade positioner, vilket introducerar variabilitet och är tidskrävande.

6.3 Hållbarhet och etiska överväganden

6.3.1 Hållbarhetsperspektiv

Den automatiserade testfixturen bidrar positivt till hållbar tillverkning på flera sätt. Tidig och tillförlitlig identifiering av tillverkningsfel – öppna kretsar, kortslutningar och felaktiga anslutningar – innebär att defekta DUT-kort kan sorteras bort tidigt i produktionsflödet. Detta minskar resursåtgången för vidare bearbetning, testning och eventuell montering av felaktiga enheter i fordon.

Automatisering av testprocessen minskar dessutom beroendet av manuell testning, vilket reducerar risken för mänskliga fel och ökar reproducerbarheten. Mer tillförlitlig testning av inbyggda fordonssystem bidrar till längre produktlivslängd och minskad risk för fältproblem som kräver kostsamma återkallningar.

6.3.2 Etiska överväganden

Projektet avser att förbättra verifieringen av säkerhetskritisk elektronik i fordon. Tillförlitlig testning av JTAG- och UART-kommunikationsgränssnitt minskar risken att felaktiga styrenheter når slutanvändaren, vilket har direkt relevans för fordons-säkerhet. Designen är icke-konfidentiell och kan redovisas akademiskt i sin helhet, vilket underlättar transparens och öppen granskning.

En etisk reflektion är att fast ADC-tröskelvärden utan kalibreringsfunktion kan ge falskt godkända resultat vid spänningsdrift. Framtida versioner bör implementera adaptiva trösklar för att minimera risken för att felaktiga DUT-kort klassificeras som godkända.

7

Slutsats och framtida arbete

7.1 Slutsats

Detta examensarbete har resulterat i en fullt fungerande automatiserad testfixtur för SPA3-plattformens Sisyphos-DUT-kort hos Volvo Cars. Projektet demonstrerar att en kombination av tvåstegs 74HC4051-multiplexertopologi [2], analog ADC-baserad signaldetektion [2] och I²C-styrd LED-matrisåterkoppling [2] utgör en praktisk, kostnadseffektiv och tekniskt välmotiverad lösning för automatiserad verifiering av debug-gränssnitt på inbyggda fordonssystem.

Den implementerade lösningen uppfyller samtliga definierade systemkrav. Korsmatristestets förmåga att simultant verifiera kontinuitet för 33 individuella debugtestpunkter och detektera potentiella bryggkortslutningar representerar en fundamental förbättring jämfört med den tidigare SPA2-baserade teststrategin. Verifiering mot ett faktiskt SPA3-fordonssystem bekräftade korrekt funktion hos samtliga 12 fullt implementerade testfunktioner.

En central lärdom från projektet är att tätheten hos testpunkterna på SPA3-DUT-kortet ställer krav på noggrann mekanisk planering redan i PCB-designfasen. Det är prefererbart att lösa mekaniska kompatibilitetsproblem digitalt i KiCad snarare än att förlita sig på manuell post-tillverkningsmodifiering. Den iterativa designprocessen – från initial arkitekturanalys via PCB-layout till fysisk verifiering mot ett fordonssystem – visade sig ändamålsenlig för ett inbyggt hårdvaruprojekt av denna komplexitet.

7.2 Framtida arbete

Baserat på de identifierade begränsningarna och erfarenheterna från projektet föreslås följande förbättringar och utvidgningar för framtida versioner av testfixturen:

1. **Konfigurerbara ADC-tröskelvärden:** Implementation av en kalibreringsrutin som mäter den faktiska matningsspänningen vid uppstart och beräknar adaptiva tröskelvärden, vilket robustifierar systemet mot spänningsvariationer och eliminerar risken för falskt godkända testresultat.
2. **PCB-revision med förbättrade pogo-pinnfotavtryck:** Design av ett anpassat pogo-pinnfotavtryck för pinnar med reducerad basdiameter [8], alternativt val av en alternativ pinnmodell med smalare bas som passar DUT-kortets testpunktsavstånd utan manuell anpassning.
3. **UART TX-verifiering:** Undersökning av en loopback-konfiguration eller en extern seriell signalgenerator för att möjliggöra fullständig TX-verifiering utan aktivt DUT, exempelvis via ett separat testkort som genererar en UART-testsekvens.
4. **Strukturerad testloggning:** Tillägg av USB-seriell loggning i strukturerat format (exempelvis JSON) för integration med ett överordnat produktionsdatasystem, vilket möjliggör automatiserad felstatistik, spårbarhet och trendanalys.
5. **Utökning till HPA-noden:** Undersökning av om HPA-noden kan verifieras via JTAG-testning genom anpassning av pogo-pinnpositioner till eventuella framtida HPA JTAG-testpunkter i kommande DUT-revisioner.
6. **Produktionssystemintegration:** Gränssnitt mot ett överordnat testhanteringssystem för batchverifiering, automatisk godkänd/underkänd-spårning och integration med produktionsdatabasen.

Litteraturförteckning

- [1] P. Marwedel, *Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things*, 3rd ed. Cham, Switzerland: Springer, 2018. [Online]. Available: <https://doi.org/10.1007/978-3-319-56045-8>
- [2] Nexperia B.V., “74HC4051; 74HCT4051—8-channel analog multiplexer/demultiplexer,” Rev. 8, 2023. [Online]. Available: https://assets.nexperia.com/documents/data-sheet/74HC_HCT4051.pdf
- [3] IEEE, *IEEE Standard for Test Access Port and Boundary-Scan Architecture*, IEEE Std 1149.1-2013, 2013. [Online]. Available: <https://standards.ieee.org/ieee/1149.1/4484/>
- [4] Analog Devices, “Introduction to UART: A Hardware Communication Protocol,” Analog Devices, 2023. [Online]. Available: <https://www.analog.com/en/resources/analog-dialogue/articles/uart-a-hardware-communication-protocol.html>
- [5] M. Wolf, A. Weimerskirch, and T. Wollinger, “State of the art: Embedding security in vehicles,” *EURASIP Journal on Embedded Systems*, vol. 2007, no. 1, Art. no. 74706, 2007. [Online]. Available: https://www.researchgate.net/publication/26483132_State_of_the_Art_Embedding_Security_in_Vehicles
- [6] H. W. Ott, *Electromagnetic Compatibility Engineering*. Hoboken, NJ, USA: John Wiley & Sons, 2009. [Online]. Available: <https://doi.org/10.1002/9780470508510>

- [7] J. Buckroyd, *In-Circuit Testing*. Oxford, U.K.: Butterworth-Heinemann, 1994. [Online]. Available: <https://www.sciencedirect.com/book/9780750609302/in-circuit-testing>
- [8] Harwin Plc, “P70-2200045—Spring Loaded Contact,” product datasheet, Hampshire, U.K., 2024. [Online]. Available: <https://www.harwin.com/products/P70-2200045>
- [9] KiCad Developers Team, *Getting Started in KiCad*, Version 8.0, KiCad Documentation, 2024. [Online]. Available: https://docs.kicad.org/8.0/en/getting_started_in_kicad/getting_started_in_kicad.html
- [10] TRACO Electronic AG, “TEN 3—3 Watt DC/DC Converters,” Rev. 13, product datasheet, Zurich, Switzerland, 2023. [Online]. Available: <https://www.tracopower.com/model/ten-3-2411win>
- [11] Lumissil Microsystems, “IS32FL3237—36-Channel LED Driver,” Rev. B, Jan. 2020. [Online]. Available: https://www.lumissil.com/assets/pdf/core/IS32FL3237_DS.pdf
- [12] Autodesk Inc., “Autodesk Inventor—3D CAD Software for Product Design,” Autodesk, 2024. [Online]. Available: <https://www.autodesk.com/products/inventor/overview>
- [13] Texas Instruments Inc., “CD74HC4051, CD74HCT4051 High-Speed CMOS Logic 8-Channel Analog Multiplexer/Demultiplexer,” Texas Instruments, Dallas, TX, USA, 2022. [Online]. Available: <https://www.ti.com/product/CD74HC4051>

A

Komponentlista (BOM)

Tabell A.1 presenterar den fullständiga komponentlistan för den tillverkade SPA3 Sisyphe-testfixturen, baserad på KiCad-projektets BOM-export.

Tabell A.1: Fullständig komponentlista, SPA3 Sisyphe-testfixtur.

Id	Beteckning	Kapsel	Ant.	Typ
1	C1–C5, C7, C9–C12, C15–C25, C27, C28	0805	23	100 nF kondensator
2	U1, U4, U7, U11–U21, U23–U28, U30	SOIC-16	17	74HC4051 mux
3	D1–D55	0805	55	Röd LED
4	R1, R3, R5, R6, R17, R19, R42–R48, R52, R54	0805	15	10 k Ω
5	R8–R16, R25, R27–R32, R39, R41	0805	18	100 Ω
6	TP43–TP111	Harwin P70-2200045	55	Pogo-pinne
7	H3, H4, H6, H7	3,5 mm	4	Skruvhål
8	C8, C13, C14, C32, C33	0805	5	10 μ F kondensator
9	R26, R49	0805	2	1 k Ω
10	R7, R16, R18, R37, R38, R40	0805	6	2,2 k Ω
11	U8	Teensy41_Extra	1	Teensy 4.1
12	U9, U10	SOT-23-5	2	74LVC1G125 buffert

(Tabell A.1 fortsätter)

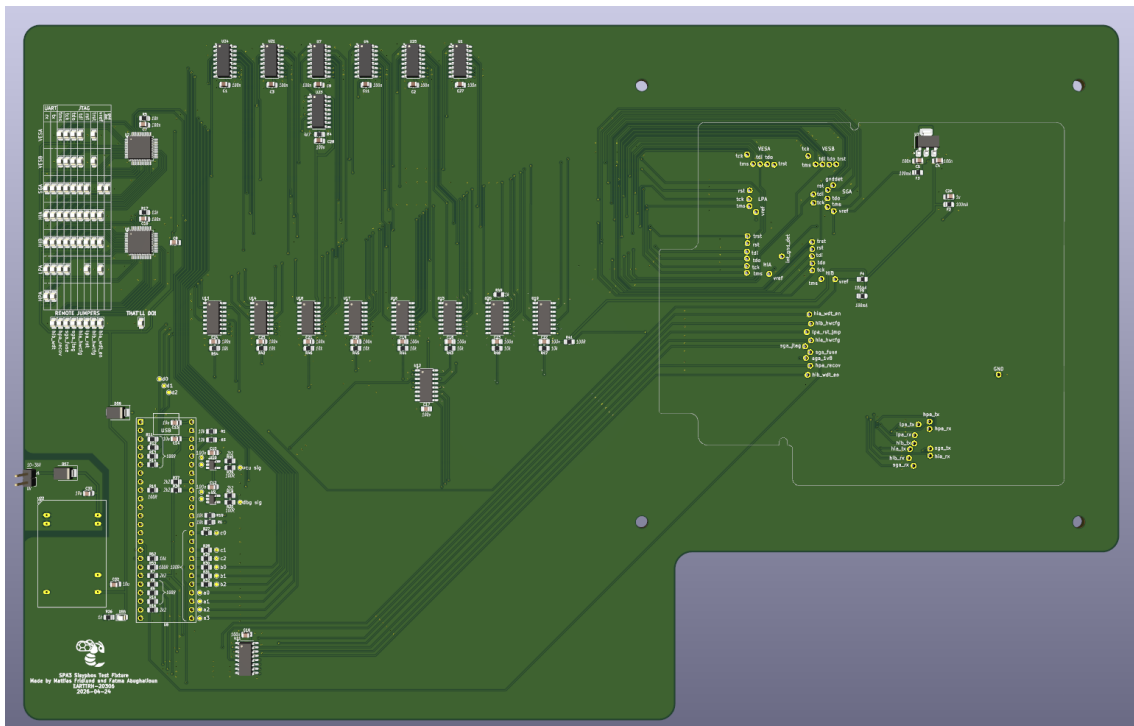
Id	Beteckning	Kapsel	Ant.	Typ
13	U22	THT (9,1×24 mm)	1	TEN3-2411WIN DC/DC
14	D56, D57	SMB	2	Schottky-diod
15	C26	0805	1	1 µF kondensator
16	R4	0805	1	4,7 kΩ
17	U2, U6	TQFP-48	2	IS32FL3237 LED-drivare
18	F2, F3, F4, F5	0805	4	0,1 A säkring
19	U3	SOT-223-3	1	AP1117-18 spänningsreg.
20	J1	2-pin, 2,54 mm	1	Kontaktdon

B

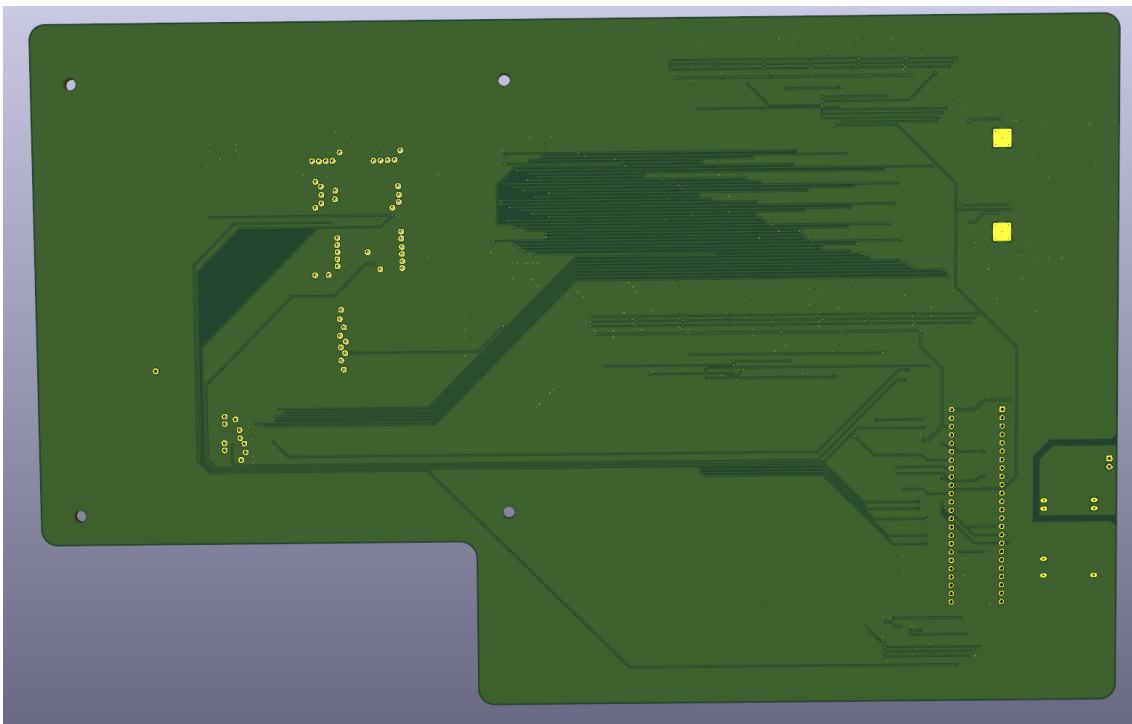
PCB-designfigurer

Denna bilaga visar ytterligare illustrationer av PCB-designen från KiCad.

3D-vyer

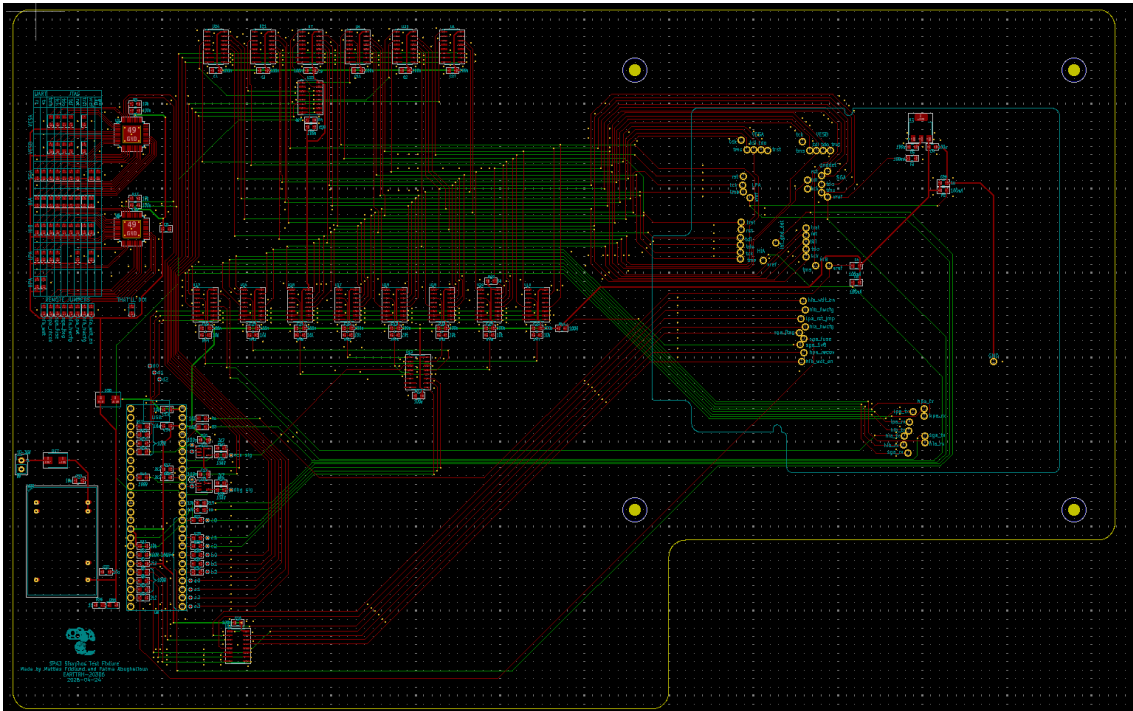


Figur B.1: KiCad 3D-vy, framsidan. Komponentplacering tydligt synlig: multiplexerraden (U23–U30) överst, VCU-sidans läsmuxar (U11–U20) i mitten, Teensy 4.1 och IS32FL3237-drivare till vänster, pogo-pinnmatris till höger.



Figur B.2: KiCad 3D-vy, baksidan. Det sammanhängande B_Cu-jordplanet (grön yta) och spridda GND-vias (gyllene punkter) är tydligt synliga. Pogo-pinnborr-platsernas raster syns längs kortets vänstra och undre kant.

PCB-layoutvy



Figur B.3: PCB-layoutvy i KiCad PcbNew med F_Cu (rött) och B_Cu (grönt) synliga. Signalspårens routing från Teensy via multiplexerträdets till pogo-pinnmatrisen är tydligt synlig.

C

Firmwarestruktur och centrala funktioner

Denna bilaga presenterar nyckeldelar av Teensy 4.1-firmwares struktur. Den fullständiga källkoden finns i projektets KiCad-projektmap (Sisyphos_TestFixture_Arduino_Code_vscode).

Pinnkonfiguration och globala variabler

Listing C.1: Pinnkonfiguration och LED-adressmatris (urval).

```
// Address buses
const uint8_t add_a_pins[4] = {36,35,34,33}; // Stage-2 mux (node select)
const uint8_t add_b_pins[3] = {39,38,37};    // Stage-1 mux (signal select)
const uint8_t add_c_pins[3] = {13,41,40};    // VCU-side signal select
const uint8_t add_d_pins[3] = {1,2,3};       // VCU-side node select

// VREF analog inputs (new in SPA3)
const uint8_t vref_sga_pin = A11; // SGA 1.8V, threshold 450
const uint8_t vref_hia_pin = A8;  // HIA 3.3V, threshold 900
const uint8_t vref_hib_pin = A9;  // HIB 3.3V, threshold 900
const uint8_t vref_lpa_pin = A10; // LPA 3.3V, threshold 900

// Node-to-hardware address lookup (HIA/HIB intentionally swapped)
const uint8_t nodeToHwAddr[] = {
    0b000, // VESA -> Y0
    0b001, // VESB -> Y1
    0b010, // SGA  -> Y2
    0b100, // HIA  -> Y4 (swapped)
    0b011, // HIB  -> Y3 (swapped)
```

```
0b101, // LPA -> Y5
0b110, // HPA -> Y6 (unused)
};
```

JTAG-korsmatrisalgoritm

Listing C.2: checkJtag() – korsmatrisalgoritm för kontinuitet och bryggdetektion.

```
void checkJtag() {
    all_jtag_ok = true;

    for (int g = 0; g < 6; g++) {                // driven node (VESA..LPA)
        writeNodeSel(g);                          // select stage-2 mux
        bool tmp_ok[6] = {1,1,1,1,1,1};

        for (int h = 0; h < 6; h++) {            // driven signal (TMS..TRST)
            writeAddr(add_b_pins, h, 3);          // select stage-1 mux

            // Drive HIGH
            vcu2dbg = false; com_en = true;
            setEnables();
            delayMicroseconds(10);

            // Read all 6x6 positions
            for (int i = 0; i < 6; i++) {
                writeAddr(add_d_pins, i, 3);
                for (int j = 2; j < 8; j++) {
                    writeAddr(add_c_pins, j, 3);
                    delayMicroseconds(5);
                    bool r = (analogRead(vcu_sig_pin) > 500);
                    if (i==g && j==h+2)
                        tmp_ok[h] &= r;          // expect HIGH (continuity)
                    else
                        tmp_ok[h] &= !r;          // expect LOW (isolation)
                }
            }
        }

        // Disable and sanity check
    }
```

```
com_en = false; setEnables();
delayMicroseconds(10);
writeAddr(add_d_pins, g, 3);
writeAddr(add_c_pins, h+2, 3);
tmp_ok[h] &= (analogRead(vcu_sig_pin) < 500);
}

// Update LEDs (skip SGA TRST: g==2 && h==5)
for (int h = 0; h < 6; h++) {
    if (!(g==2 && h==5)) {
        sendToDriver((g>3), tmp_ok[h]*255, h+2, g, true);
        if (ledAddr[g][h+2] < 36) all_jtag_ok &= tmp_ok[h];
    }
}
}
```

VREF-verifiering

Listing C.3: checkVref() – VREF-mätning via dedikerade analoga ingångar.

```
void checkVref() {
    all_vref_ok = true;
    const uint8_t pins[] = {vref_sga_pin, vref_hia_pin,
                             vref_hib_pin, vref_lpa_pin};
    const uint8_t nodes[] = {2, 3, 4, 5};
    const uint16_t tholds[] = {450, 900, 900, 900};

    for (int i = 0; i < 4; i++) {
        delayMicroseconds(100);
        int reading = analogRead(pins[i]);
        bool vref_ok = (reading > tholds[i]);
        bool drv2 = (nodes[i] > 3);
        sendToDriver(drv2, vref_ok * 255, 8, nodes[i]);
        all_vref_ok &= vref_ok;
    }
}
```

INSTITUTIONEN FÖR ELEKTROTEKNIK

CHALMERS TEKNISKA HÖGSKOLA

Göteborg, Sverige

www.chalmers.se



CHALMERS