



CroG 2

Endbericht | Call 13 | Projekt ID 3084

Lizenz CC-BY-SA

## Inhalt

1	Einleitung	3
2	Projektbeschreibung	3
3	Verlauf der Arbeitspakete	3
3.1	Arbeitspaket 1 - <i>Projektmanagement</i>	3
3.2	Arbeitspaket 2 - <i>Hardware</i>	4
3.3	Arbeitspaket 3 - <i>Software</i>	4
3.4	Arbeitspaket 3 - <i>Vermarktung</i>	4
4	Umsetzung Förderauflagen	4
5	Liste Projekt Endergebnisse	5
6	Verwertung der Projektergebnisse in der Praxis	5
7	Öffentlichkeitsarbeit/ Vernetzung	6
8	Eigene Projektwebsite	6
9	Geplante Aktivitäten nach netidee-Projektende	6
10	Anregungen für Weiterentwicklungen durch Dritte	6

## 1 Einleitung

Das CroG2 Projekt ist eine Folgeprojekt, welches die Entwicklung von open source FPGA Werkzeugen weiterführt. Das ursprüngliche Projekt hat bereits über 10000 Nutzer weltweit und ist weltweit für eine Renaissance des FPGA Designs verantwortlich.

## 2 Projektbeschreibung

*Beschreibung der Projektziele / Zielgruppe und inhaltlicher Überblick über das Projektergebnis (max. 5 Seiten)*

Ziel des Projektes ist es, der Community von Elektronik Makern, Studenten, Firmengründern und Forschern ein quelloffenes und flexibles Werkzeug zum Erstellen digitaler Designs unter Verwendung von frei programmierbaren digital Bausteinen (=FPGA) zur Verfügung zu stellen.

FPGAs als Digitale Bausteine sind das “Lego” der Schaltungsdesigner. Sie sind super flexibel, schnell einsetzbar, erlauben die Realisierung schneller Schaltungen und sind viel preiswerter als der Entwurf und die Produktion eines eigenen Chip. Vor allem bei Geräten mit einer Stückzahl von unter 1000 sind FPGAs unschlagbar.

Die in diesem Dokument angeführten Beispiele zeigen, dass uns dies in vollem Umfang gelungen ist. Wir haben eine weltweite Gemeinschaft, die die in Rahmen dieses Projektes entwickelten Werkzeuge nutzen und weiterentwickeln.

## 3 Verlauf der Arbeitspakete

### 3.1 **Arbeitspaket 1 - Projektmanagement**

*Kurzbeschreibung der Haupttätigkeiten*

*Das Projektmanagement kümmert sich vor allem auf die Koordination der Mitarbeiter der einzelnen Arbeitspakete, den Erwerb von Hardware, und die Dokumentation und Abrechnung des Projektes.*

*Besondere Erfolge/ Probleme*

*Gab es große Abweichungen zum Plan? Warum?*

*Nachdem David Shah nicht in ausreichendem Maß zur Abwicklung des gesamten Vorhabens verfügbar war, wurde das Team um Clifford Wolf ergänzt. Clifford Wolf war der Software Hauptentwickler im ersten CroG Projekt und es brauchte daher kaum Einarbeitung und Clifford Wolf besaß auch die notwendige Qualifikation.*

*Wie in der Vergangenheit bereits angeführt, wurde ein erheblicher Teil von Projektkosten von Symbiotic GmbH übernommen. Eine Verrechnung der Kosten der projektdurchführenden Onsite Broadcast eU an die Symbiotic GmbH hätte mit anschließender Zahlung der Kosten als Sponsorship hätte das Projekt erheblich in seiner Abwicklung verkompliziert. Deshalb wurde manche Kosten des Projektes direkt von Symbiotic GmbH übernommen, wie z.B. der Erwerb des Messgerätes, die Produktion der FPGA Boards, Flug- und Hotelkosten. Damit entfallen diese ursprünglich für dieses Projekt vorgesehenen Kosten und sind nicht in der Abrechnung der Kosten enthalten.*

*Die Abrechnung des Projektes hat sich stark verzögert durch die Corona Krise, die zu Herausforderungen in einem ebenfalls von Edmund Humenberger geführten Unternehmen Symbiotic GmbH geführt hat. Es wurde keine Zwischenabrechnung durchgeführt, da kein Liquiditätsbedarf für das Unternehmen bestand.*

*Die Symbiotic GmbH wurde auf Basis des CroG Projektes 2018 gegründet.*

*Erkenntnisse zur Vorgangsweise*

Die Abwicklung des Projektes hat massiv von der Qualifikation der teilnehmenden Mitarbeiter und der Erfahrung profitiert, die die Mitarbeiter miteinander schon vor dem Projekt gemacht haben. Leider hat Davis Shah nach dem Ende des Projektes das Unternehmen verlassen.

### **3.2 Arbeitspaket 2 - Hardware**

*Kurzbeschreibung der Haupttätigkeiten*

*Zur Entwicklung der Software wurde eine Hardware Plattform benötigt, die die Entwicklung und das Testen aller Eigenschaften des FPGAs ermöglicht. Eine geeignete Hardware war käuflich nicht zu erwerben. Deshalb wurde eine*

*eigene FPGA Hardware entwickelt. Das Design wurde mit KiCad entwickelt, welche sich als ausreichend funktional dafür herausgestellt hat. Die bei einem Chinesischen Auftragsfertiger bestellte Hardware war preiswert und hat auf Anhieb funktioniert.*

*Gab es große Abweichungen zum Plan? Warum?*

Es gab keine Abweichungen zum Plan. Die Implementierung der Hardware war leichter als erwartet.

*Erkenntnisse zur Vorgangsweise*

Die Produktion der Prototypenhardware durch einen chinesischen Dienstleister hat sich als unproblematisch herausgestellt. Glücklicherweise funktionierte bereits die erste Version der Hardware ohne Probleme.

Der veröffentlichte Bauplan der Hardware wurde von anderen genutzt, um weitere Versionen der Hardware bauen zu lassen. Wir raten jedem Maker massiv von der eigenen Herstellung derartiger anspruchsvoller Boards ab und empfehlen die Nutzung von Firmen, die die Herstellung von Prototypen Boards anbieten auf basis der von ns zur Verfügung gestellten Baupläne und dokumentation

### **3.3 Arbeitspaket 3 - Software**

*Kurzbeschreibung der Haupttätigkeiten*

Die Software Lösung besteht aus dem Synthesewerkzeug Yosys und dem Place&Route Werkzeug nextpnr. Diese beiden Softwarewerkzeuge wurden ursprünglich von unserem Team entwickelt und im Rahmen dieses Projektes erweitert, um den Hochgeschwindigkeits Chip ECP5 der Firma Lattice zu unterstützen.

Zu diesem Zweck musste Software geschrieben werden zum automatischen reverse engineering des Chips, da der Chiphersteller keine ausreichende Dokumentation zur Verfügung stellt.

*Besondere Erfolge/ Probleme*

Während des Projektes hat sich die Firma Lattice als Hersteller des ECP5 Chips dazu entschlossen, unser Projekt offiziell zu unterstützen und auf seiner eigenen Webpage zu bewerben.

<https://www.latticesemi.com/en/Solutions/Solutions/SolutionsDetails01/CommunitySourced>

*Gab es große Abweichungen zum Plan? Warum?*

Zusätzlich zu dem Team von Onsite Broadcast arbeiteten auch Teammitglieder der Symbiotic GmbH im großem Umfang an der Entwicklung der Software mit. Alleine mit den von Netidee zur Verfügung gestellten Ressourcen wäre das Projekt nicht machbar gewesen. Symbiotic GmbH hatte zum Glück die zusätzlichen Ressourcen zur Verfügung gestellt.

Der Abgang von David Shah war für das Projekt zu früh. Softwarearbeiten mussten von Clifford Wolf übernommen werden. Es kam dadurch zu Verzögerungen und zusätzlichem personellen Aufwand

#### *Erkenntnisse zur Vorgangsweise*

Es gibt nur eine ganz kleine Anzahl von Menschen weltweit, die die Kompetenz besitzen, dieses reverse engineering von FPGAs durchzuführen. Derzeit beschäftigen sich 3 andere Teams mit dem Reverse engineering von anderen FPGAs. Diese Teams machen einen viel langsameren Fortschritt. Wir hatten viel Glück, derart hochqualifizierte und motivierte Teammitglieder zur Verfügung zu haben.

### **3.4 Arbeitspaket 4 - Vermarktung**

#### *Kurzbeschreibung der Haupttätigkeiten*

Das Projekt wurde vor allem über die gut vernetzte open hardware Twitter community beworben. Unser Twitter account @ico\_TC konnte 6000 Follower versammeln. Weiters wurde das Projekt auf 3 Konferenzen vorgestellt.

Besonders schöne Projektbeispiele wurden über die Website netidee.at als Blogbeiträge veröffentlicht.

#### *Besondere Erfolge/ Probleme*

Das Land Argentinien verwendet unsere Software in allen Universitäten des Landes für die Ausbildung von Elektronik Ingenieuren. Unsere Software wird in acht der weltweit top-10 Technischen Universitäten eingesetzt.

#### *Gab es große Abweichungen zum Plan? Warum?*

Die Kosten für die Reisen und Aufenthalte für die Konferenzen wurde direkt von Symbiotic GmbH übernommen. Deshalb sind im projekt diese geplanten Kosten nicht aufgelistet.

#### *Erkenntnisse zur Vorgangsweise*

Die Vermarktung über Twitter hat sich als erfolgreich herausgestellt. Die Betreuung des Twitter account sehr zeitintensiv, aber trotzdem der kostengünstigste Vermarktungsweg.

Mit einer Vermarktung über LinkedIn sind vor allem Personen in Indien erreichbar, jedoch kaum Maker.

## 4 Liste Projektergebnisse

Kurzbeschreibung der erreichten Projektergebnisse jeweils mit Open Source Lizenz und Webadresse (netidee Vorgaben beachten!)

1	<p><i>Open Hardware ECP5 FPGA board Trellis Board</i>  <i>Es wurde eine Platine mit einem leistungsfähigen FPGA Baustein entwickelt und produziert.</i></p>	<p>Lizenz  ISC License</p>	<p>netidee/          &lt;projekt&gt;[link]  <a href="https://github.com/daveshah1/TrellisBoard">https://github.com/daveshah1/TrellisBoard</a></p>
2	<p><i>Open Source Software Programmierwerkzeuge für ECP5 FPGA</i>  <i>Es wurde Open Source Software entwickelt, die die Konfigurationsdatei (=bitstream) für den leistungsfähigen FPGA erstellen kann. Dazu gehören die Erweiterung des Synthese Werkzeug Yosys als auch die Erweiterung des place&amp;route Werkzeuges nextpnr. Beide Werkzeuge wurden maßgeblich von den Teammitgliedern mit finanzieller Unterstützung der Symbiotic GmbH für andere FPGAs entwickelt.</i></p> <p><i>Dazu notwendig wurde die Dokumentation des FPGA Chips durch reverse engineering erstellt.</i></p>	<p>Lizenz  ISC License</p>	<p><a href="https://github.com/YosysHQ/nextpnr">https://github.com/YosysHQ/nextpnr</a>   <a href="https://prjtrellis.readthedocs.io/en/latest/">https://prjtrellis.readthedocs.io/en/latest/</a></p>

## 5 Verwertung der Projektergebnisse in der Praxis

Angaben zur Verwertung der Projektergebnisse in der Praxis

Es gibt bereits eine Vielzahl von Projekten, die unsere Software nutzen.

Ein zentrales Element der Sicherheit der verschlüsselten Kommunikation im Internet ist der Einsatz eines "root of Trust" Element.

<https://github.com/kazkojima/device-key> in Japan hat ein solches entwickelt mit Hilfe unserer Software.

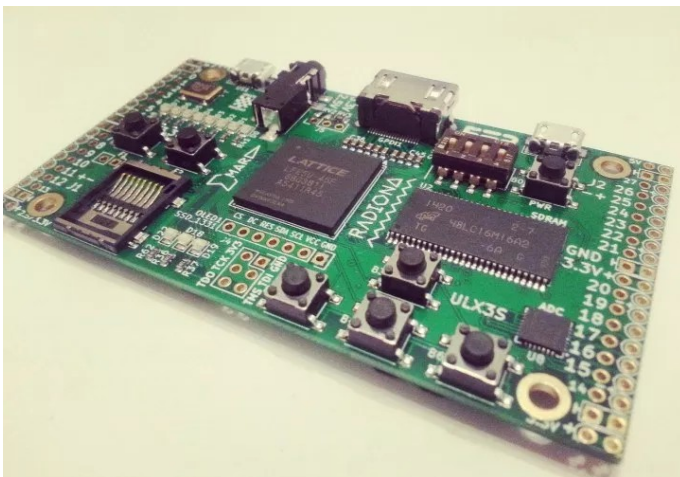
Bis jetzt nutzen bereits 12 neue Hardwareprojekte unsere FPGA Software. Im Antrag haben wir unter Punkt 10.6 angegeben, dass es ein ausgezeichneter Erfolg wäre, wenn 6 neue Hardwareprojekte unsere Software nutzen.

Die folgend aufgelisteten Projekte wurden ausschließlich wegen der Verfügbarkeit der im Rahmen von CroG2 entwickelten open source FPGA Werkzeuge entworfen und produziert.

### **Project ULX3S**

<https://www.crowdsupply.com/radiona/ulx3s>

Dieses FPGA board wurde von einem Team in Kroatien entwickelt und dient vor allem der akademischen Ausbildung von Elektronik Ingenieuren. Es wurden im Rahmen einer Crowdfunding Kampagne bereits über 1000 Boards weltweit verkauft. Diese FPGA board nutzt ausschließlich die von uns entwickelte FPGA design Software.

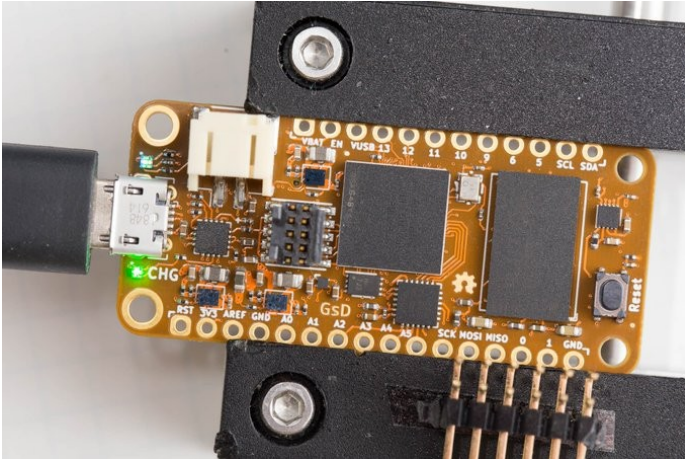


### **Project Orange Crab**

FPGA board welches ausschließlich unsere Software nutzt.

<https://groupgets.com/manufacturers/good-stuff-department/products/orangecrab>





## Project Butterstick

FPGA board welches ausschließlich unsere Software nutzt.

<https://www.hackster.io/gatoninja236/the-butterstick-development-board-hackster-spotlight-6f10d4>



## Hackaday Badge

[https://github.com/Spritetm/hadbadge2019\\_pcb](https://github.com/Spritetm/hadbadge2019_pcb)

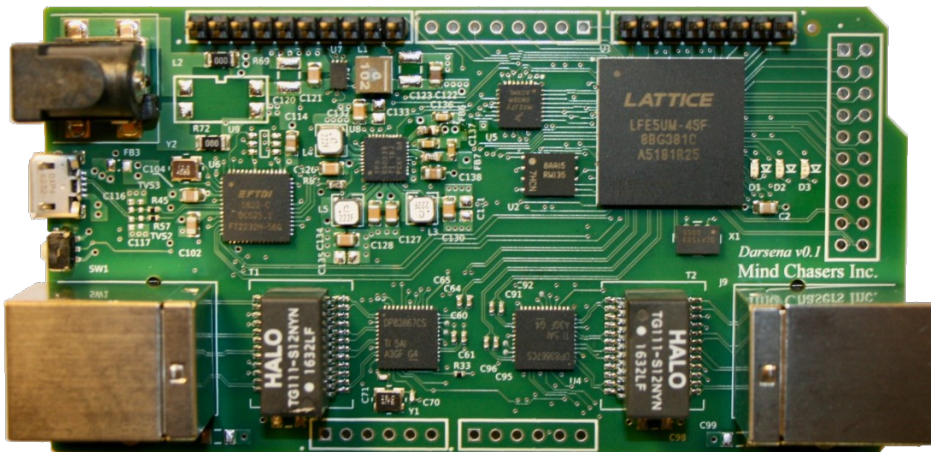
<https://hackaday.com/2019/11/16/behind-the-scenes-of-the-2019-superconference-badge/>



## Project Darsena

Das Hardware Projekt nutzt unsere Software Werkzeuge zum Entwurf eines Ethernetnetwork System zur Überwachung der Sicherheit des Internet Verkehr:

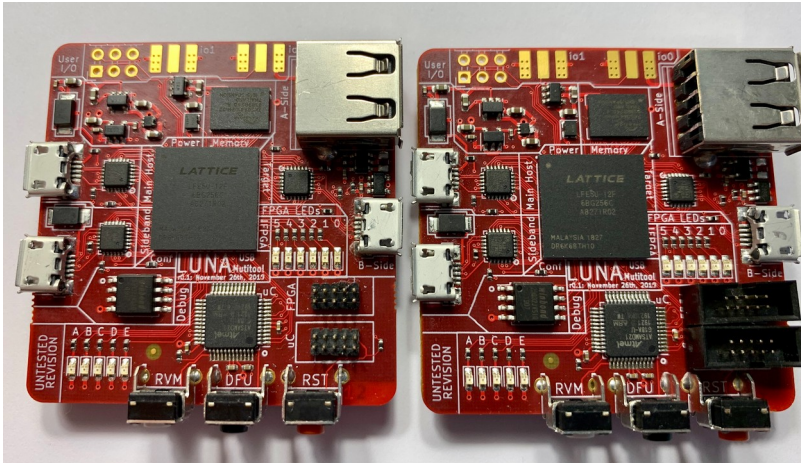
<https://mindchasers.com/dev/darsena-hw-spec>



## Project Luna

Unsere FPGA Werkzeuge waren ausschlaggebend für die Entwicklung des Luna Boards und der Luna Software, die für Sicherheitsforschung zum USB 2.0 Interface Standard eingesetzt wird.

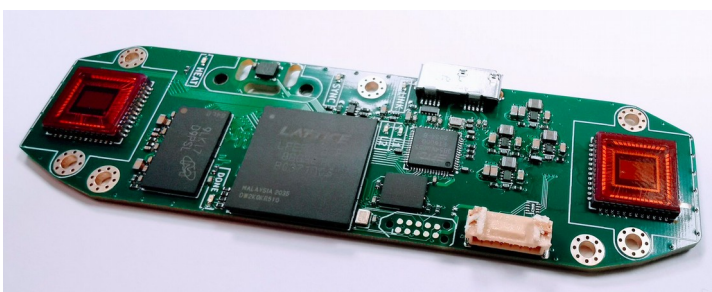
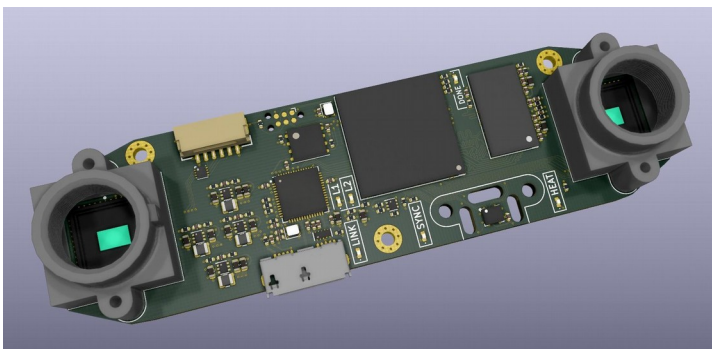
<https://luna.readthedocs.io/en/latest/features.html>



### Project stereo camera

A stereo camera board with 2x of the classical MT9V034 cameras, DDR3 memory, FT600 USB3.0 FIFO and a heated IMU. Using the open source FPGA tools of CroG2.

[https://github.com/korken89/ovio\\_core](https://github.com/korken89/ovio_core)

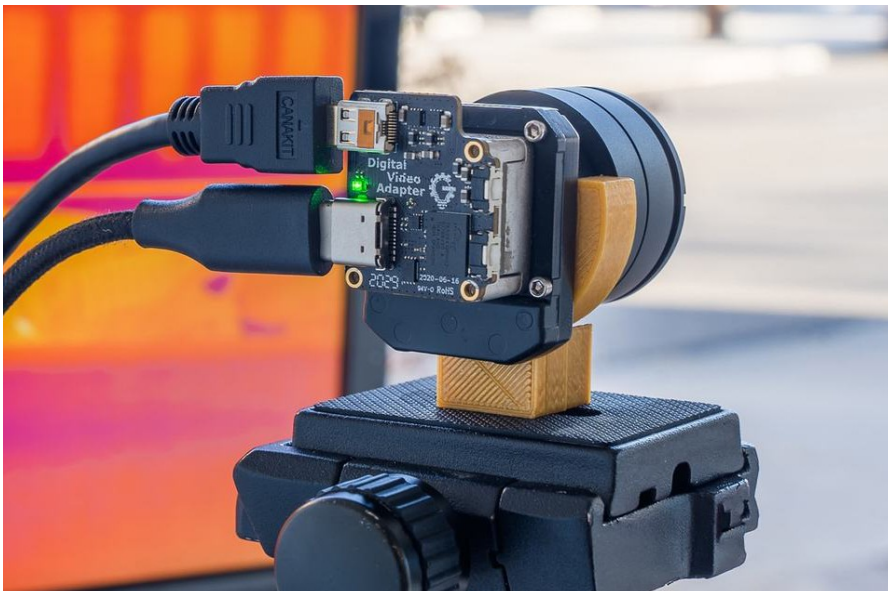
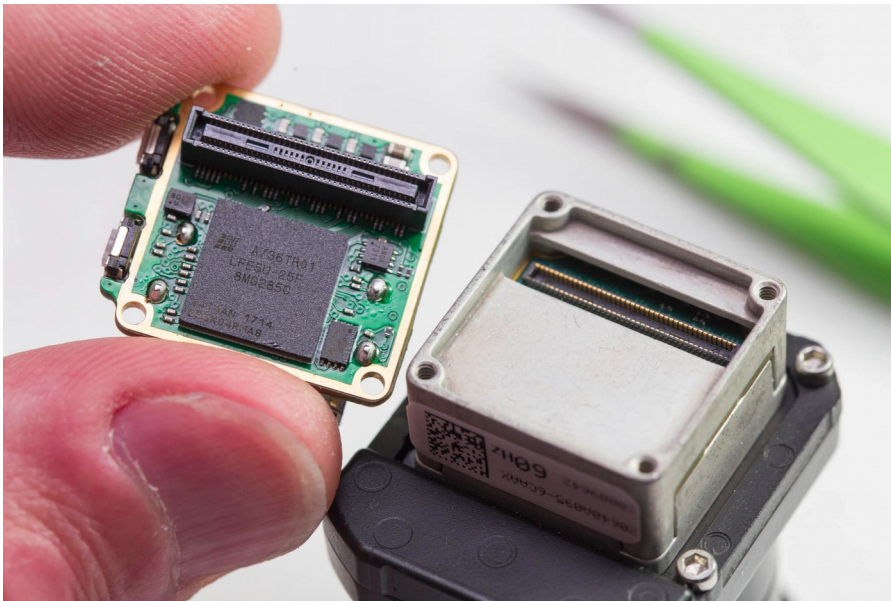


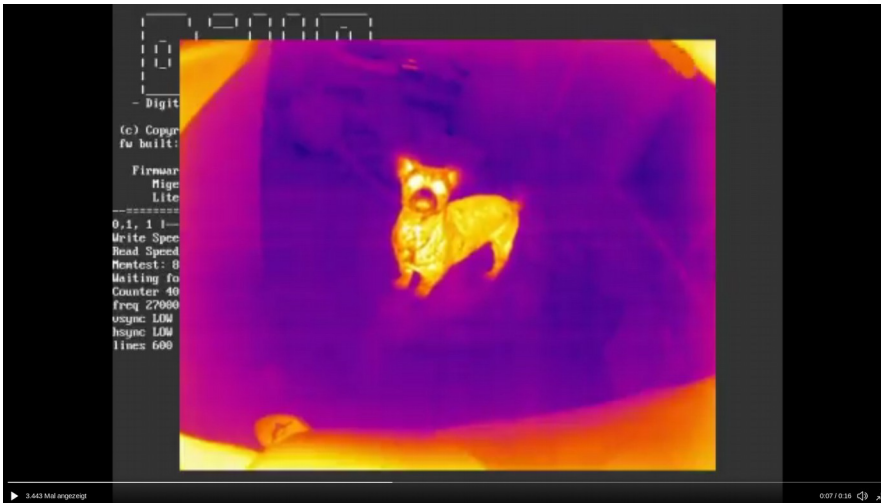
[https://github.com/korken89/ovio\\_core](https://github.com/korken89/ovio_core)

### Project FIR Camera

<https://store.groupgets.com/collections/frontpage/products/flir-boson-digital-video-adapter>

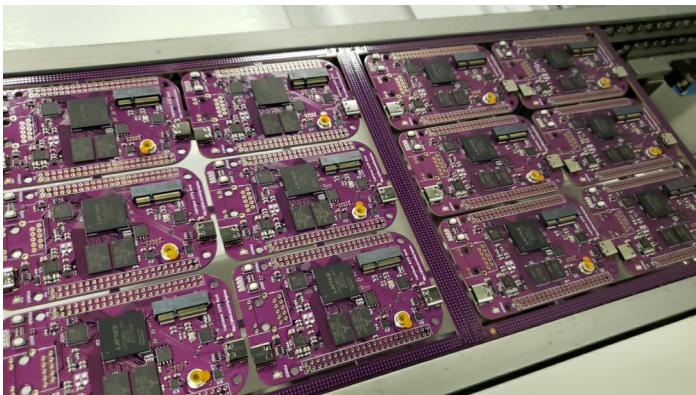
A camera board was developed to interface to a high quality Infrared sensor and output the video signal over HDMI. The FPGA on the board is used to do image processing like colour adjustment or scaling.





### project Logicbone

Dieses Board ist eine Aufsteck Platine für einen Beaglebone Linux Rechner und ermöglicht die Ein- und Ausgabe über beliebig erstellbare Schnittstellen, die auf dem FPGA implementiert werden.



<https://github.com/oskirby/logicbone>

### Project NUT2NT

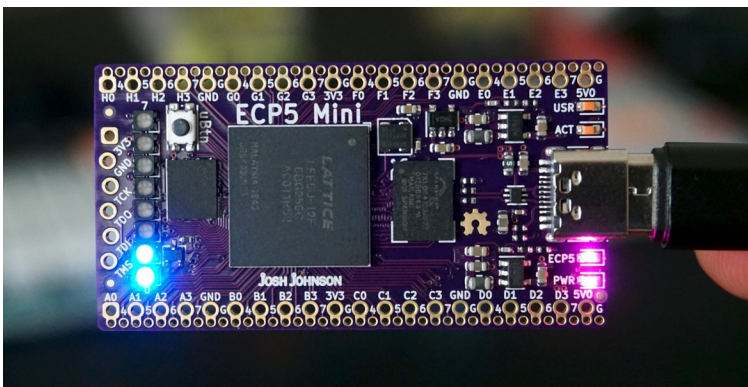
Diese ECP5 FPGA basierte Hardware ist vor allem für die Entwicklung von GPS Lösungen entworfen worden. Sie nutzt unsere Software für das Programmieren des FPGA.

<https://www.crowdsupply.com/amungo-navigation/nut2nt-plus>



### Project ECP5 mini

Das ECP5 mini board ist eine Hardware, die die Entwicklung von Digitalschaltungen ermöglicht. Das Board kann entweder auf einem Steckbrett verwendet werden, oder auf eine eigene einfache Platine gelötet werden.

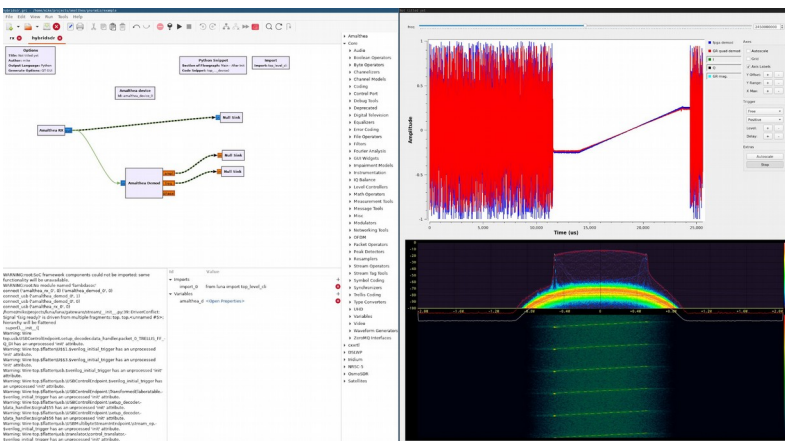
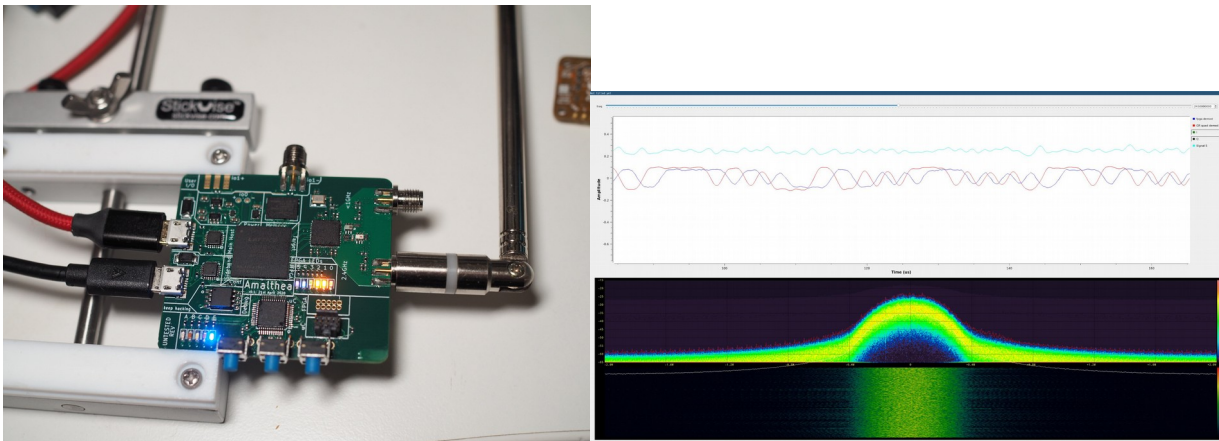


<https://github.com/joshajohnson/ecp5-mini>

### Project Amalthea

Ist ein FPGA board für das Empfangen, Senden, Bearbeiten, Erzeugen von beliebigen Radiosignalen im Bereich 400MHz bis 2,8 GHz. Es nutzt die von uns entwickelten open source FPGA Werkzeuge.

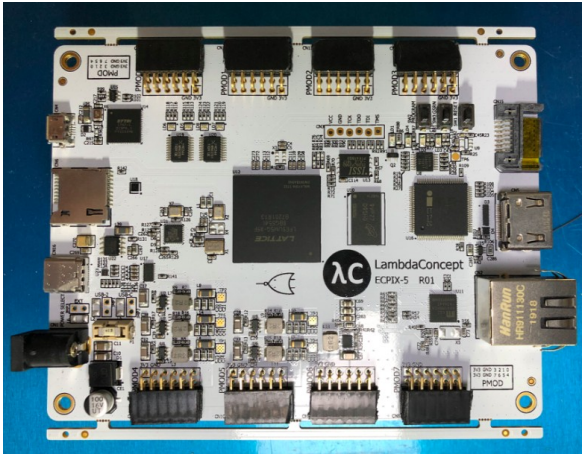
<https://github.com/greatscottgadgets/amalthea>



## Projekt ECPIX-5

<https://shop.lambdaconcept.com/home/46-1-ecpix-5.html>

ECPIX-5 is an FPGA development board designed around Lattice's ECP5 FPGA. It features an extensive set of peripherals and high speed connectors to ease your prototyping process. The FPGA on ECPIX-5 is fully supported by an open source toolchain compatible with major operating systems. No hardware-enforced licensing systems, no bloated IDE to download, and faster compile times!



## Project BenchBox

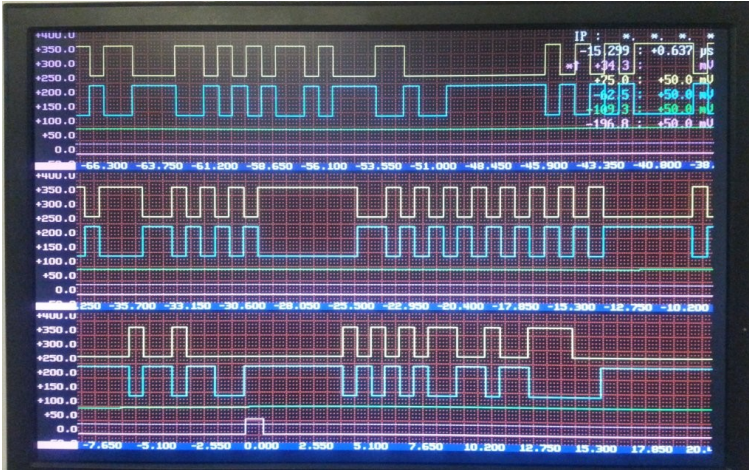
Project EEC Benchbox is a open source power supply project .

<https://www.crowdsupply.com/envox/eez-bb3>



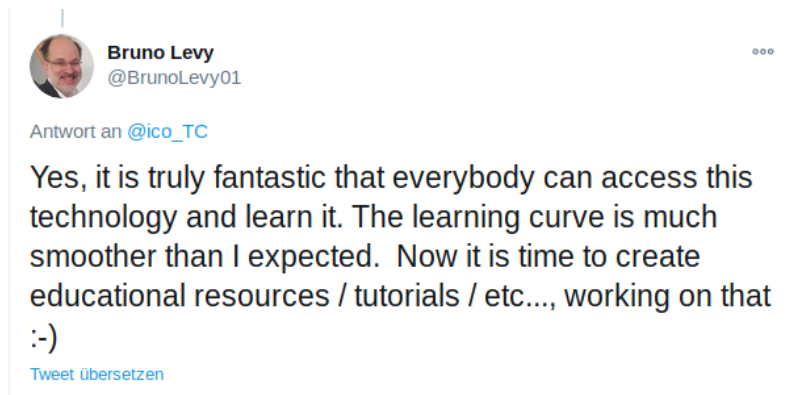
For maximum Speed and functionality, it uses an ECP5 FPGA and uses our open source tools to generate the configuration file. This way functionalities usually only available in very expensive Lab equipment is not available for Makers and students.





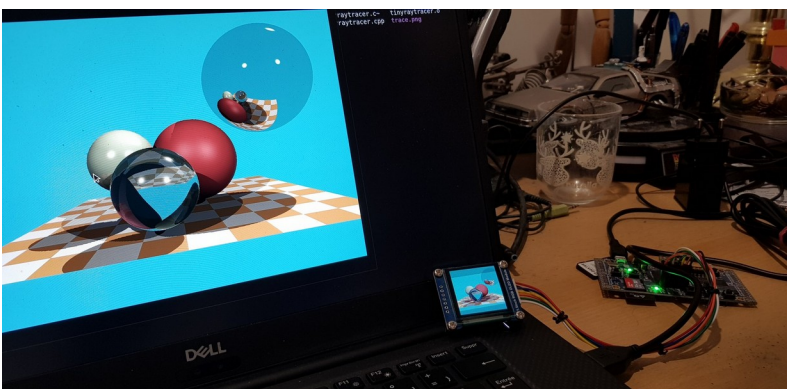
Logic analyzer function of the EEC Labbench.

## Teaching



Bruno is Professor and did design a complete CPU and a graphics card for generating HDMI Video output and also implemented an operating system and wrote a raytracing render software for his CPU and deployed it on an ULX3S FPGA using our open source FPGA tools. He will use our open source FPGA tools to teach digital design at the University of Nancy/France.

<https://github.com/BrunoLevy/learn-fpga>

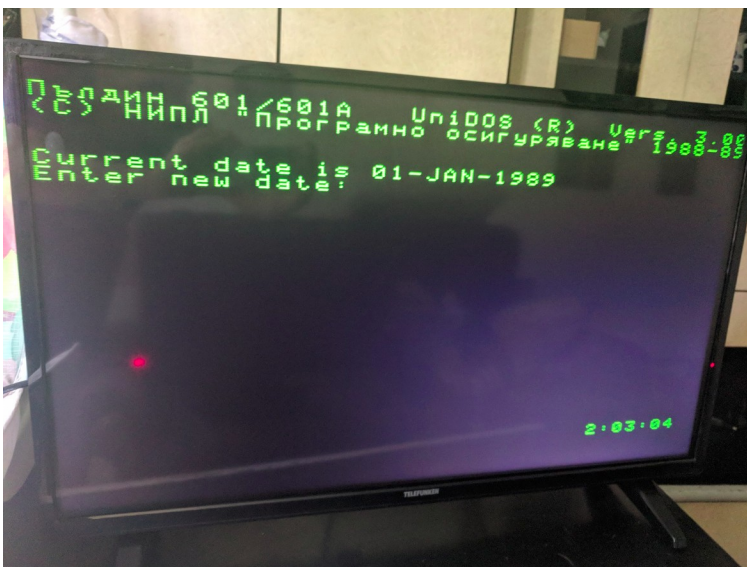


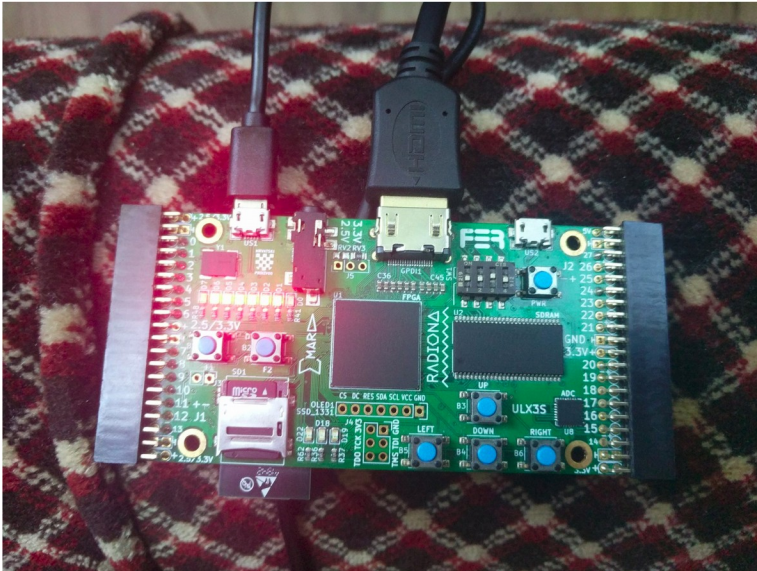


<https://twitter.com/robng15/status/1328618758206857217?s=20>

## Retro computing

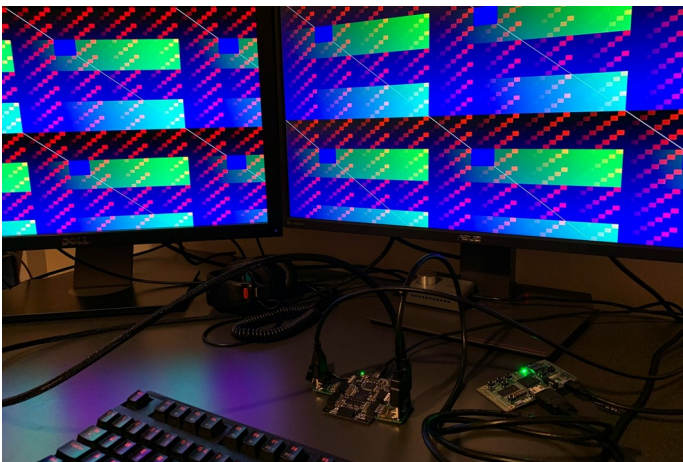
Old Bulgarian computer





## Video hacking

As described in our application, people will start hacking with Video signals. This is one example of <https://twitter.com/kbeckmann> doing so.



## Computer design

<https://twitter.com/lawriegriffiths> designed a complete CPU with its own memory controller, graphics card, storage interface, keyboard. This is transparency at max.



## Research

Eine neue Sprache für den Entwurf von Digitalschaltungen wurde entwickelt.

<https://github.com/sylefeb/Silice/>

[https://www.youtube.com/watch?v=\\_OhxEY72qxl](https://www.youtube.com/watch?v=_OhxEY72qxl)

Als Anwendungsbeispiel für die neue Designsprache wurde ein kompletter Chip entwickelt, auf dem man Doom spielen kann, jedoch keine CPU besitzt.



Das Ziel dieser Forschung ist es, die über 40 Jahre alten Chip Designsprachen Verilog und VHDL zu ersetzen.

Dieses Forschungsprojekt baut auf die im Rahmen von CroG2 entwickelten Designwerkzeugen auf.

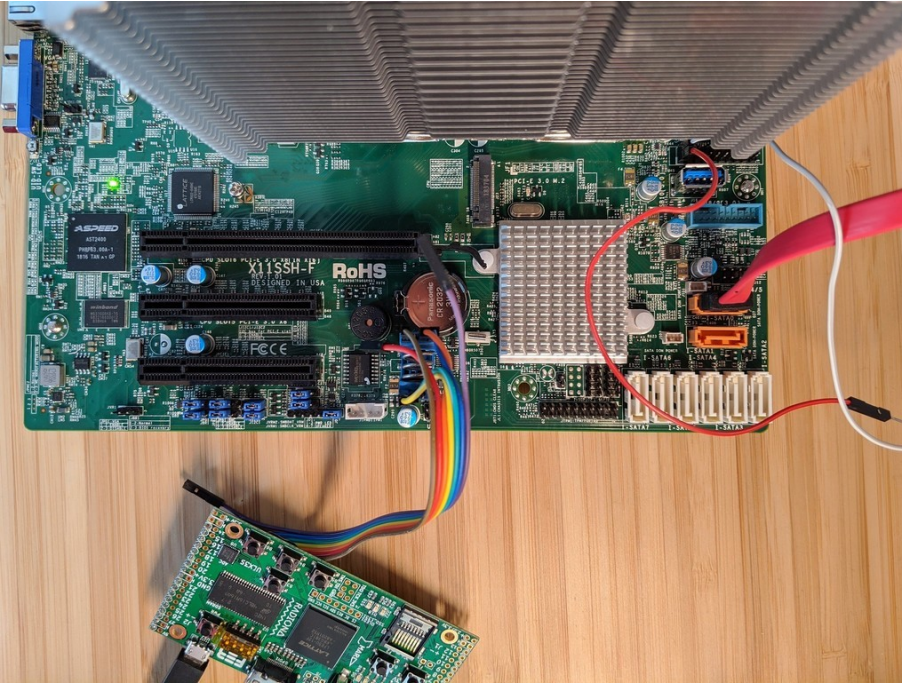
## Security

Dieses Projekt simuliert mit einem FPGA einen SPI Flash Speicher auf einem Motherboard und erlaubt so das Aushebeln jedes Datenschutzmechanismus in einem Computer.

The sleep attack can modify all of memory (including backdooring SMM) allowing an adversary with physical access, such as when clearing customs at an airport, to disable the lock screen process and access encrypted disks without requiring a password

<https://github.com/osresearch/spispy>

Diese Lösung wurde mit unseren open source FPGA Werkzeugen entwickelt und mit dem open Source FPGA board ULX3S implementiert.



## 6 Öffentlichkeitsarbeit/ Vernetzung

*Beschreibung der im Rahmen Ihres netidee-Projektes bereits erfolgten bzw. noch geplanten Öffentlichkeitsarbeit oder Vernetzung*

Die Arbeit dieses Projektes wurde vor allem über den Twitter account @ico\_TC mit mehreren tausend Tweets beworben. @ico\_TC hat über 6000 Followers weltweit. Weiters wurde das Projekt auf Konferenzen wie der RISC-V Konferenz in Zürich oder der Chipsalliance Group oder dem CCC in Leipzig vorgestellt .

Nachdem sich bereits eine umfangreiche und gesunde Community von über 1000 aktiven Entwicklern um die open source FPGA Werkzeuge gebildet hat, sind von uns keine weiteren Marketingmaßnahmen geplant.

## 7 Geplante Aktivitäten nach netidee-Projektende

*Sind weiterführende Aktivitäten nach dem netidee-Projektende geplant?*

Symbiotic GmbH plant die Software Werkzeuge weiter zu entwickeln im Rahmen seiner finanziellen Möglichkeit und unterstützt derzeit Forscher und Maker bei deren Anwendung.

## 8 Anregungen für Weiterentwicklungen durch Dritte

*Welche Nutzungs- und Weiterentwicklungsmöglichkeiten für Dritte ergeben sich durch Ihr netidee-Projekt bzw. empfehlen Sie?*

Unsere Werkzeuge sind bereits die Basis für die Entwicklung neuer Hardware Design Sprachen wie SpinalHDL, nMigen, Silice, Bluespec, und Chisel. Auch kommen sie weltweit in der Universitären Ausbildung von Elektronik Ingenieuren zum Einsatz. Auch wird die im Rahmen dieses Projektes entwickelte Software von Dritten bereits erweitert, um FPGA von anderen Herstellern zu unterstützen.

Ein derartiges Projekt wurde für FPGAs der Firma GoWin Semi gestartet.  
<https://github.com/YosysHQ/apicula>

Ein ähnliches Projekt wurde von einem anderen Team für die FPGAs der Firma Intel gestartet und es gibt erste Implementierungserfolge.  
<https://github.com/Ravenslofty/mistral>

Die im Rahmen von CroG und Crog2 entwickelten FPGA Werkzeuge wurden von Google übernommen und werden von Google im Rahmen des Symbiflow Projektes weiterentwickelt.

<https://symbiflow.github.io/>

Die Firma Quicklogic hat sich entschlossen, selber open source Werkzeuge zur Programmierung ihrer FPGAs zur Verfügung zu stellen. Dabie verwenden sie unser Synthese Werkzeug Yosys.

<https://www.quicklogic.com/software/qorc-mcu-efpga-fpga-open-source-tools/>