



Projet de semestre

FPGA Bruteforce Attack

h e p i a

Haute école du paysage, d'ingénierie
et d'architecture de Genève

Kandiah Abivarman

17.02.2024

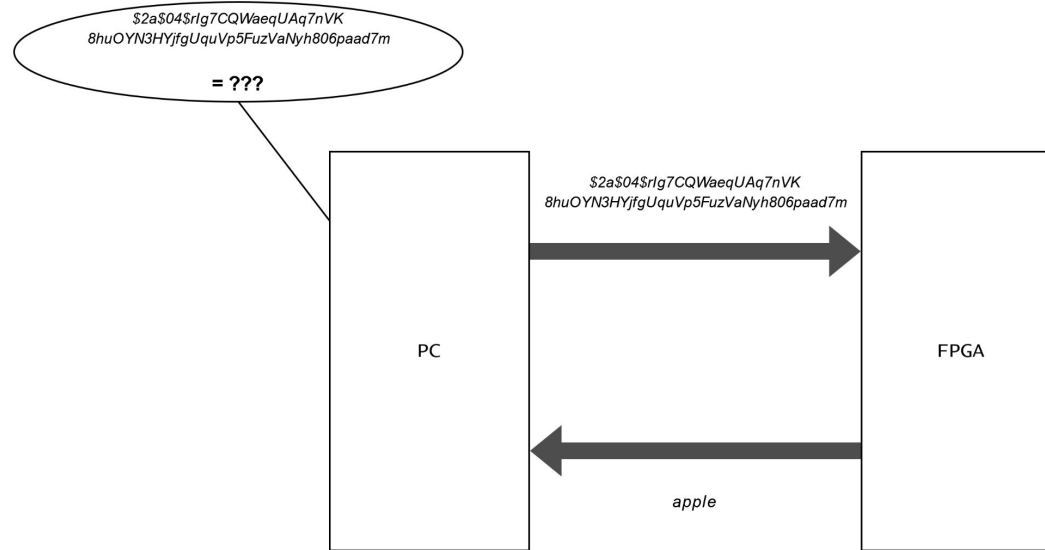
Table des matières :



- Objectif
- Bcrypt
- Implémentation existante
- Fonctionnement & Test
- Interface PC - FPGA
- Conclusion

Objectif

Objectif - Schéma



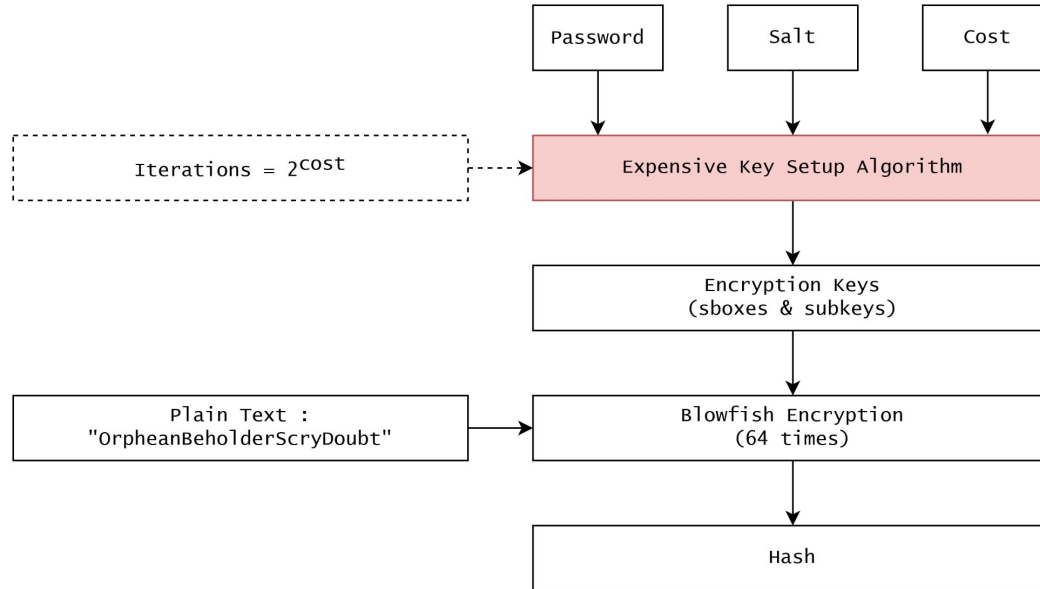
Objectif - FPGA vs CPU vs GPU



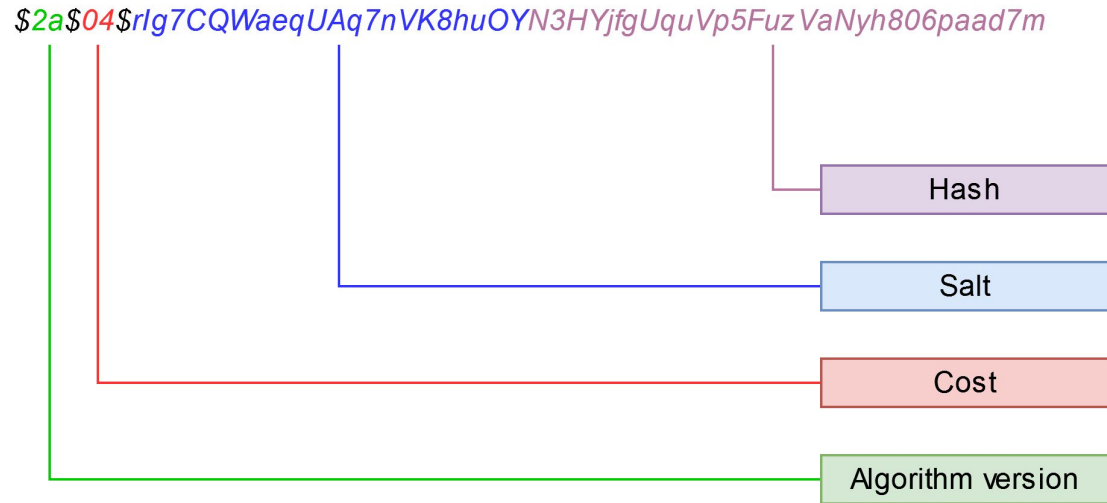
- Cout
- Consommation
- Hashrate

Bcrypt - Algorithme de hash

Bcrypt

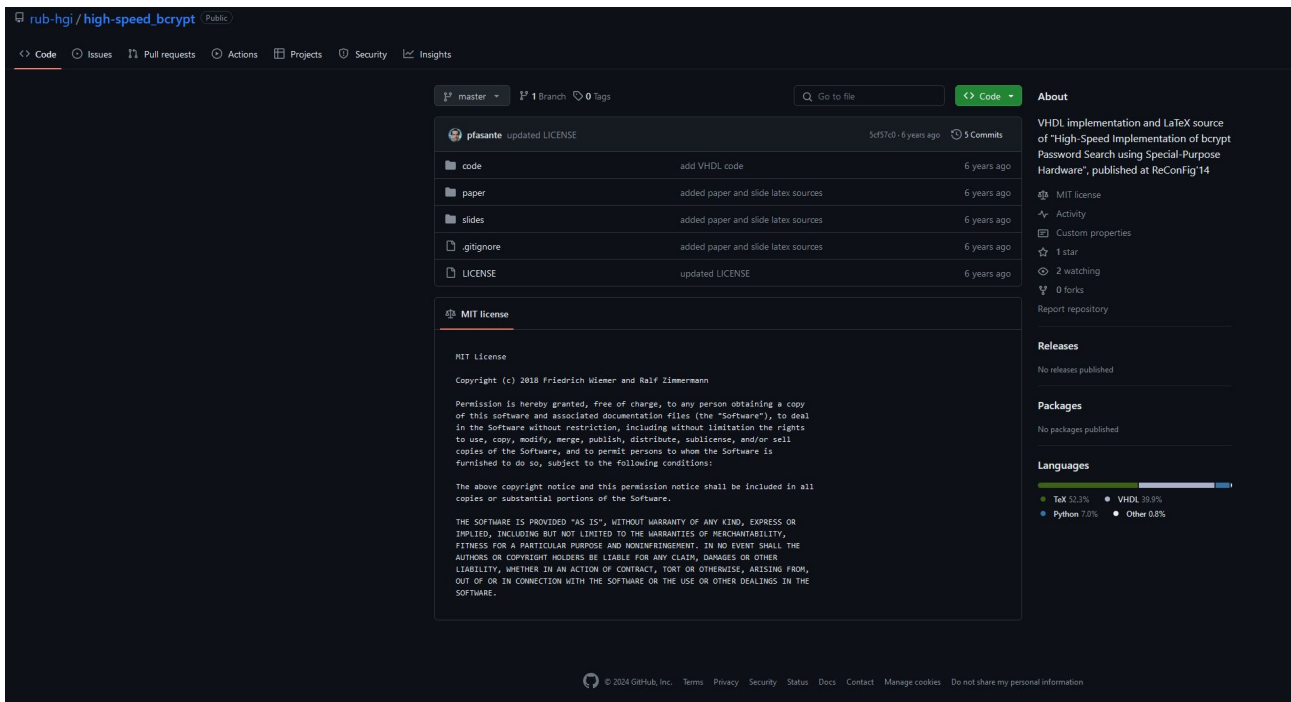


Bcrypt - Format du hash



Implémentation existante

Implémentation existante



rub-hgi / high-speed_bcrypt Public

<> Code Issues Pull requests Actions Projects Security Insights

master 1 Branch 0 Tags

Go to file

Code

pfasante updated LICENSE 3d37d · 6 years ago 5 Commits

code	add VHDL code	6 years ago
paper	added paper and slide latex sources	6 years ago
slides	added paper and slide latex sources	6 years ago
.gitignore	added paper and slide latex sources	6 years ago
LICENSE	updated LICENSE	6 years ago

MIT license

MIT License

Copyright (c) 2018 Friedrich Mliener and Ralf Zimmermann

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

About

VHDL Implementation and LaTeX source of "High-Speed Implementation of bcrypt Password Search using Special-Purpose Hardware", published at ReConFig'14

MIT license

Activity

Custom properties

1 star

2 watching

0 forks

Report repository

Releases

No releases published

Packages

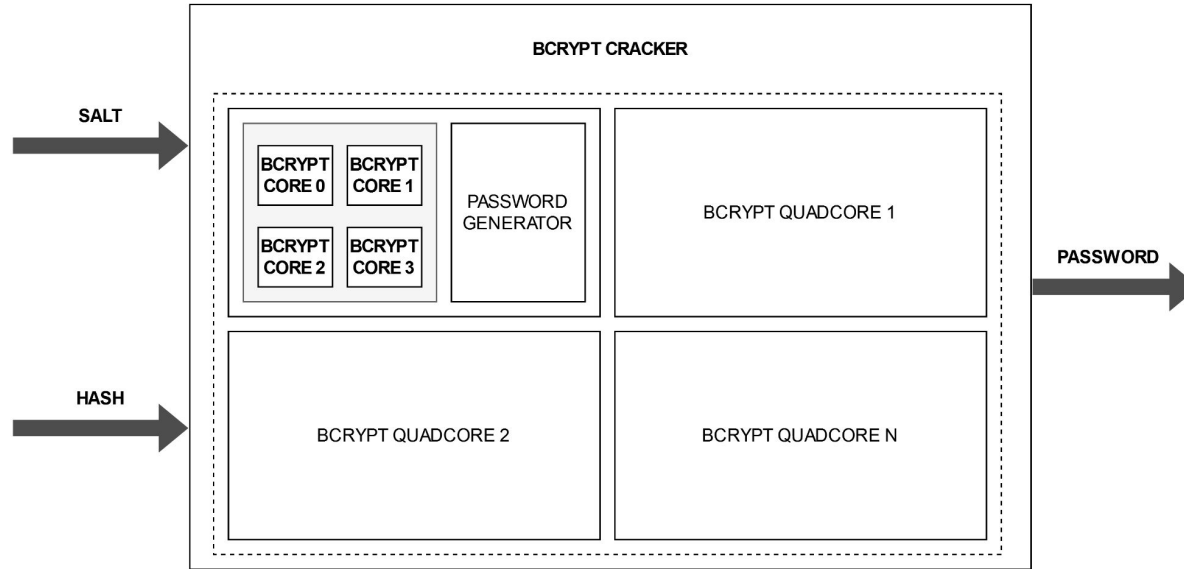
No packages published

Languages

52.3% VHDL 39.9%
7.0% Python 0.8%

© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

Implémentation existante - Schéma



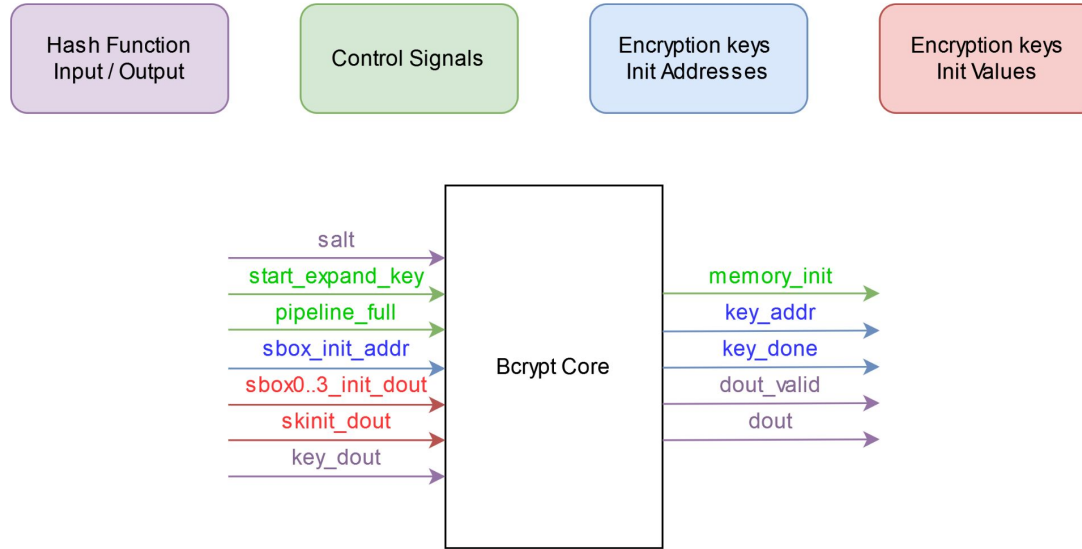
Implémentation existante - Problèmes



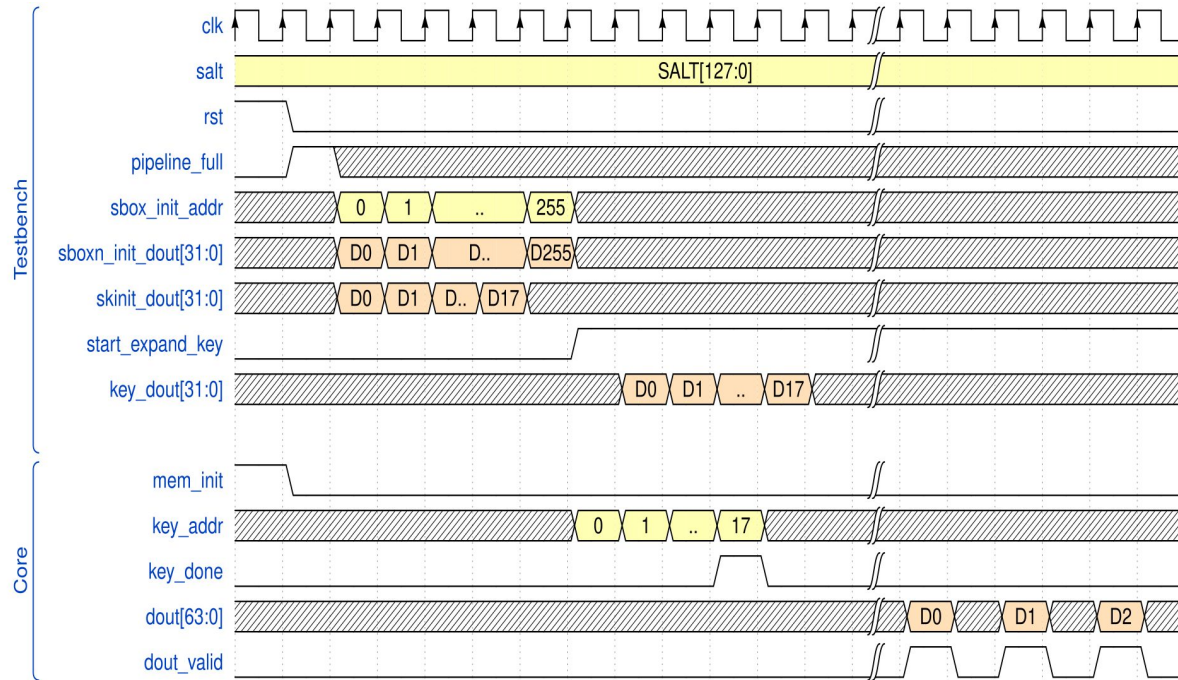
- Documentations
- Versions - Incohérences
- Testbenches incomplets
- Petites erreurs

Fonctionnement & Test

Bcrypt Core Interface



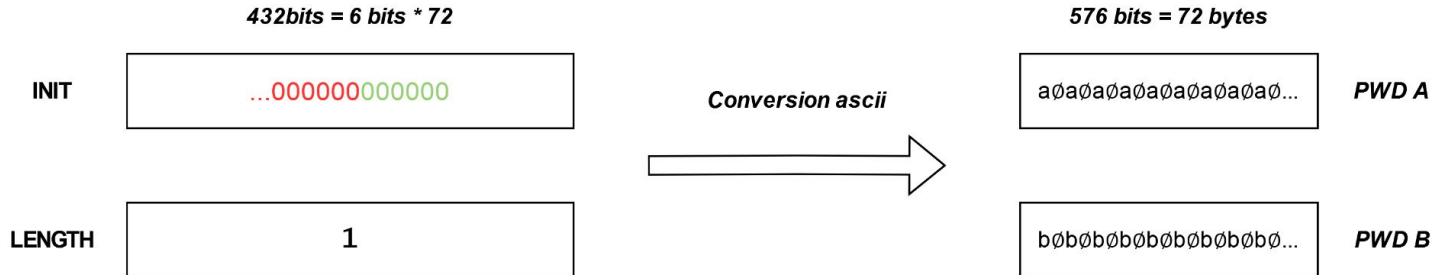
Bcrypt Core Timing



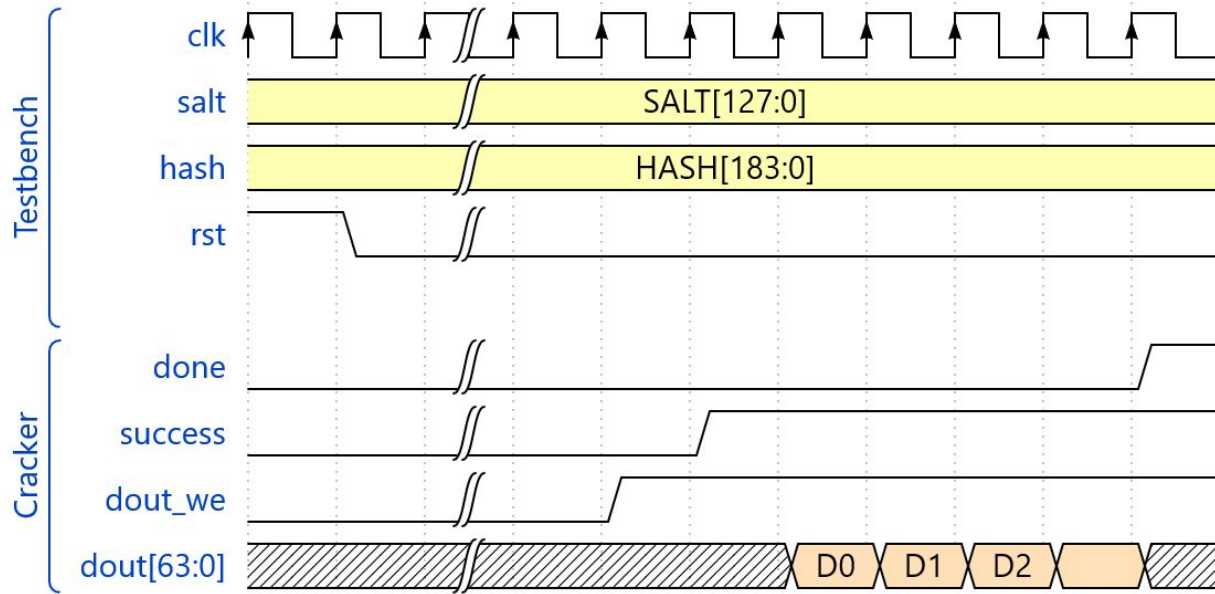
Password Generator



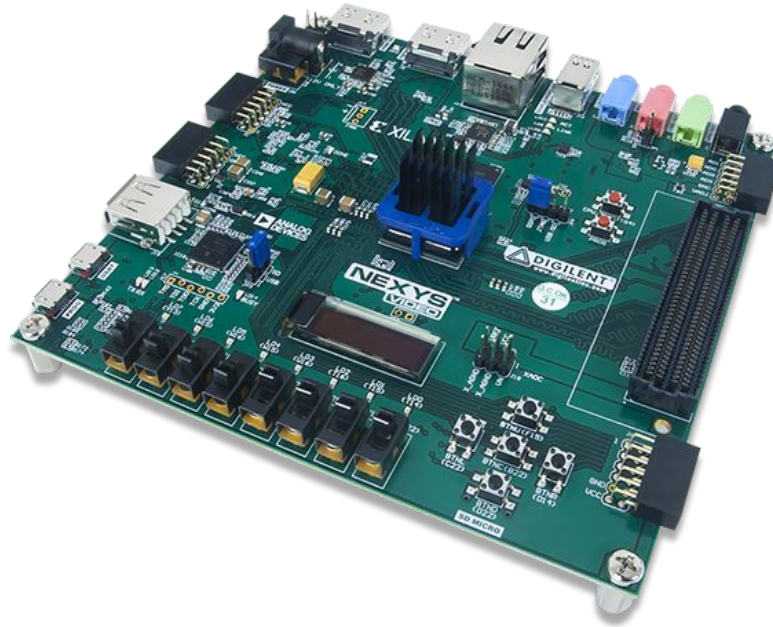
Conversion Table				
0x00	0x01	0x02	0x1b	0x35
NULL ∅	'a'	'b'	'A'	'0'



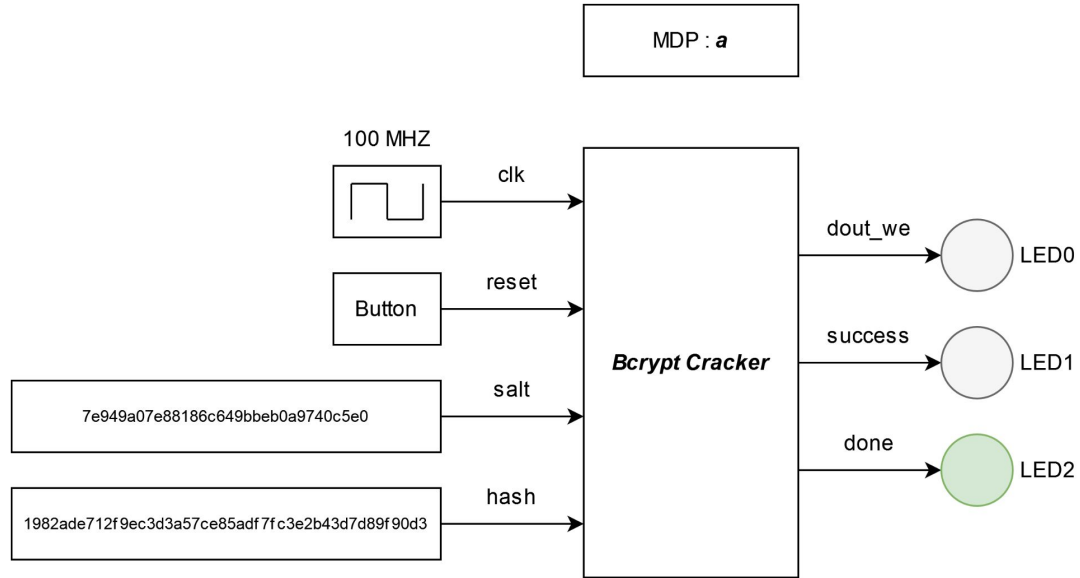
Bcrypt Cracker Timing



Bcrypt Cracker Test Board - Nexys Video



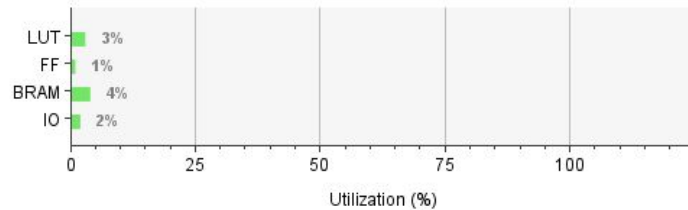
Bcrypt Cracker Test - Schéma



Bcrypt Cracker - Bilan



Resource	Utilization	Available	Utilization %
LUT	3640	134600	2.70
FF	2878	269200	1.07
BRAM	13	365	3.56
IO	6	285	2.11



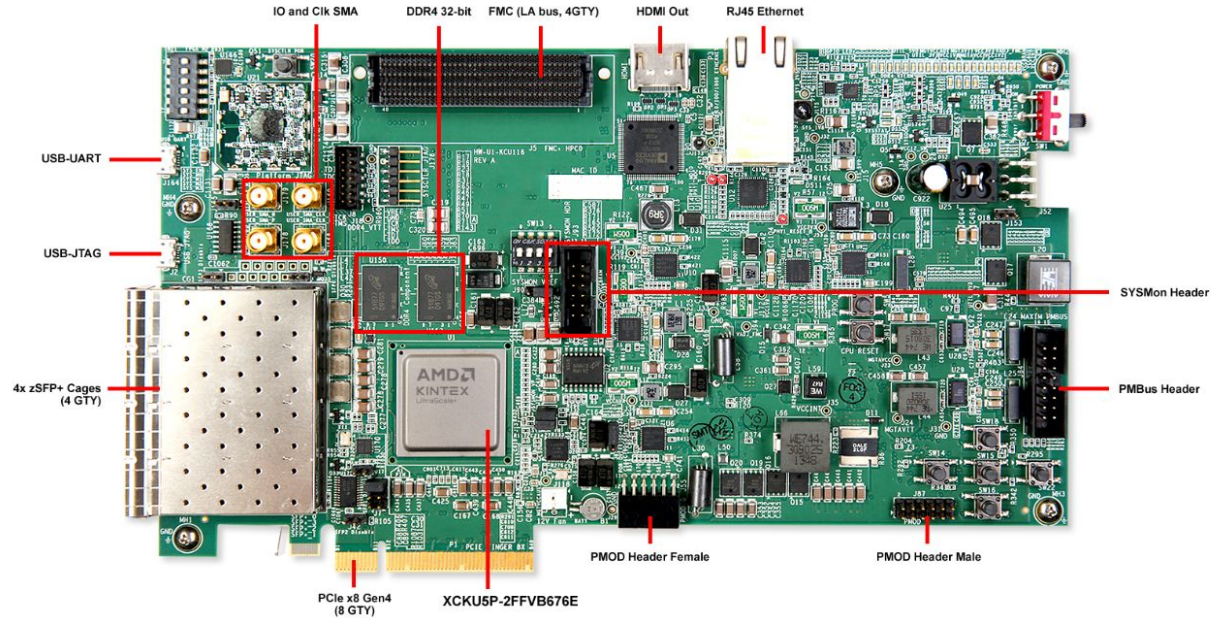
Cost = 4

Quadcores = 1

Hashrate = 1205.57 [Hash/s]

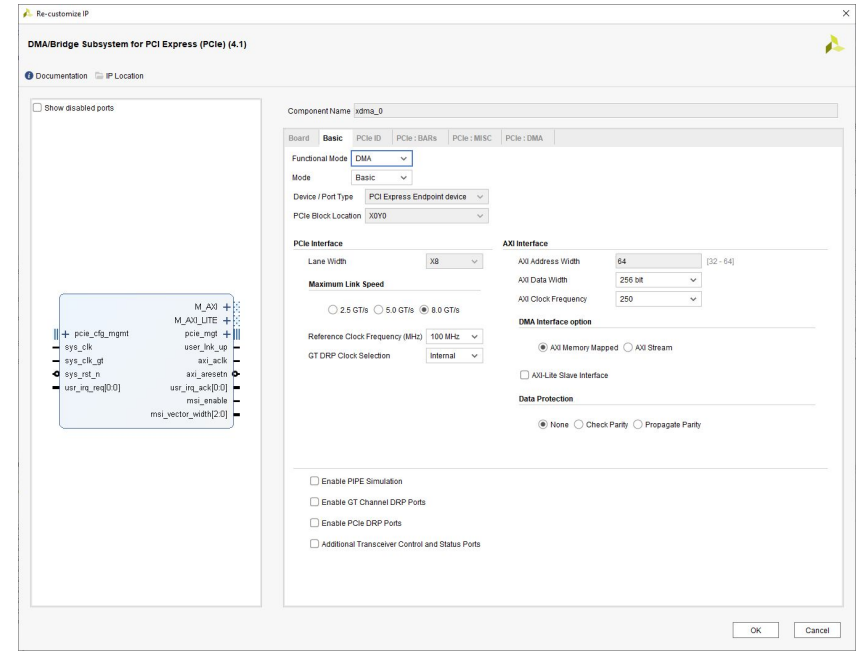
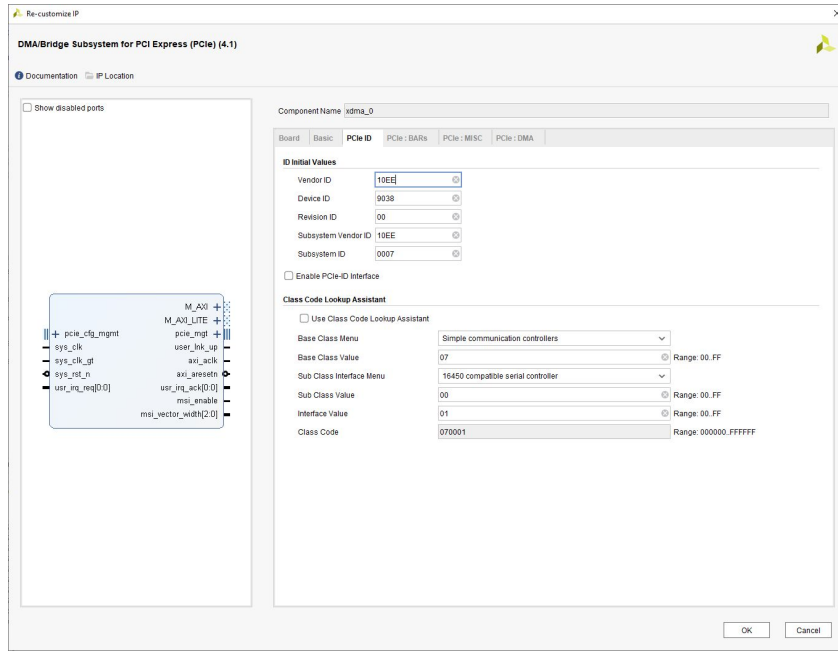
Interface PC - FPGA

Interface PCIe - Kintex Ultrascale +

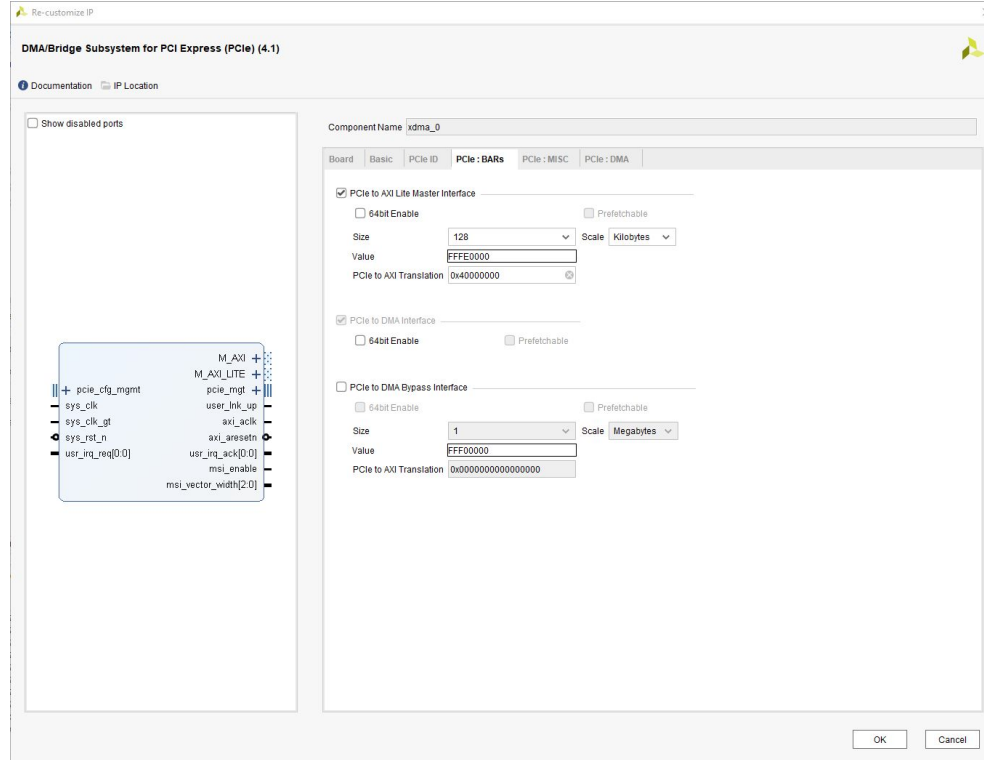




Interface PCIe - Config xdma



Interface PCIe - Config xdma



Interface PCIe - lspci

```
sudo lspci -vv -d 10ee:9038
01:00.0 Serial controller: Xilinx Corporation Device 9038 (prog-if 01 [16450])
    Subsystem: Xilinx Corporation Device 0007
    Control: I/O- Mem+ BusMaster- SpecCycle- MemWINV- VGASnoop- ParErr- Stepping- SERR+ FastB2B-
DisINTx-
    Status: Cap+ 66MHz- UDF- FastB2B- ParErr- DEVSEL=fast >TAbort- <TAbort- >MAbort- >SERR- <PERR-
INTx-
    Interrupt: pin A routed to IRQ 16
    Region 0: Memory at ef000000 (32-bit, non-prefetchable) [size=128K]
    Region 1: Memory at ef100000 (32-bit, non-prefetchable) [size=64K]
    Capabilities: [40] Power Management version 3
        Flags: PMEClk- DSI- D1- D2- AuxCurrent=0mA PME(D0-,D1-,D2-,D3hot-,D3cold-)
        Status: D0 NoSoftRst+ PME-Enable- DSel=0 DScale=0 PME-
    Capabilities: [48] MSI: Enable- Count=1/1 Maskable- 64bit+
        Address: 0000000000000000 Data: 0000
    Capabilities: [70] Express (v2) Endpoint, MSI 00
        DevCap: MaxPayload 1024 bytes, PhantFunc 0, Latency L0s <64ns, L1 <1us
        ExtTag+ AttnBtn- AttnInd- PwrInd- RBE+ FLReset- SlotPowerLimit 75.000W
    DevCtl: CorrErr+ NonFatalErr+ FatalErr+ UnsupReq+
        RlxdOrd+ ExtTag+ PhantFunc- AuxPwr- NoSnoop+
        MaxPayload 256 bytes, MaxReadReq 512 bytes
    DevSta: CorrErr+ NonFatalErr- FatalErr- UnsupReq+ AuxPwr- TransPend-
    LnkCap: Port #0, Speed 8GT/s, Width x8, ASPM not supported
        ClockPM- Surprise- LLActRep- BwNot- ASPMOptComp+
    LnkCtl: ASPM Disabled; RCB 64 bytes, Disabled- CommClk+
        ExtSynch- ClockPM- AutWidDis- BWInt- AutBWInt-
    LnkSta: Speed 8GT/s (ok), Width x8 (ok)
        TrErr- Train- SlotClk+ DLActive- BWMgmt- ABWMgmt-
    DevCap2: Completion Timeout: Range BC, TimeoutDis+ NROPrPrP- LTR-
        10BitTagComp- 10BitTagReq- OBFF Not Supported, ExtFmt- EETLPPrefix-
        EmergencyPowerReduction Not Supported, EmergencyPowerReductionInit-
        FRS- TPHComp- ExtTPHComp-
        AtomicOpsCap: 32bit- 64bit- 128bitCAS-
```

...

Interface PCIe - sysfs

file	function
class	PCI class (ascii, ro)
config	PCI config space (binary, rw)
device	PCI device (ascii, ro)
enable	Whether the device is enabled (ascii, rw)
irq	IRQ number (ascii, ro)
local_cpus	nearby CPU mask (cpumask, ro)
remove	remove device from kernel's list (ascii, wo)
resource	PCI resource host addresses (ascii, ro)
<u>resource0..N</u>	<u>PCI resource N, if present (binary, mmap, rw(!))</u>
re-source0_wc..N_wc	PCI WC map resource N, if prefetchable (binary, mmap)
revision	PCI revision (ascii, ro)
rom	PCI ROM resource, if present (binary, ro)
subsystem_device	PCI subsystem device (ascii, ro)
subsystem_vendor	PCI subsystem vendor (ascii, ro)
vendor	PCI vendor (ascii, ro)

```
int main()
{
    uint32_t* bar0;
    int fd;

    fd = open("/sys/bus/pci/devices/0000:01:00.0/resource0", O_RDWR | O_SYNC);

    if (fd < 0)
    {
        perror("test");
        fprintf(stderr, "Failed to open bar0 file\n");
        return -1;
    }

    bar0 = mmap(NULL, 131072, PROT_READ | PROT_WRITE, MAP_SHARED, fd, 0);
    close(fd);

    if (bar0 == MAP_FAILED)
    {
        fprintf(stderr, "Failed map bar0\n");
        return -1;
    }

    printf("Etat interrupteurs : 0x%x\n", bar0[0]);

    munmap(bar0, 131072);
    return 0;
}
```

Conclusion :



- Faire fonctionner sur la carte Nexys Video
- Tester le PCIe avec un driver linux
- Réfléchir à des améliorations au système
- Faire le rapport