Projet de semestre

Bruteforce Password Attack on FPGAs

Kandiah Abivarman

17.04.2024

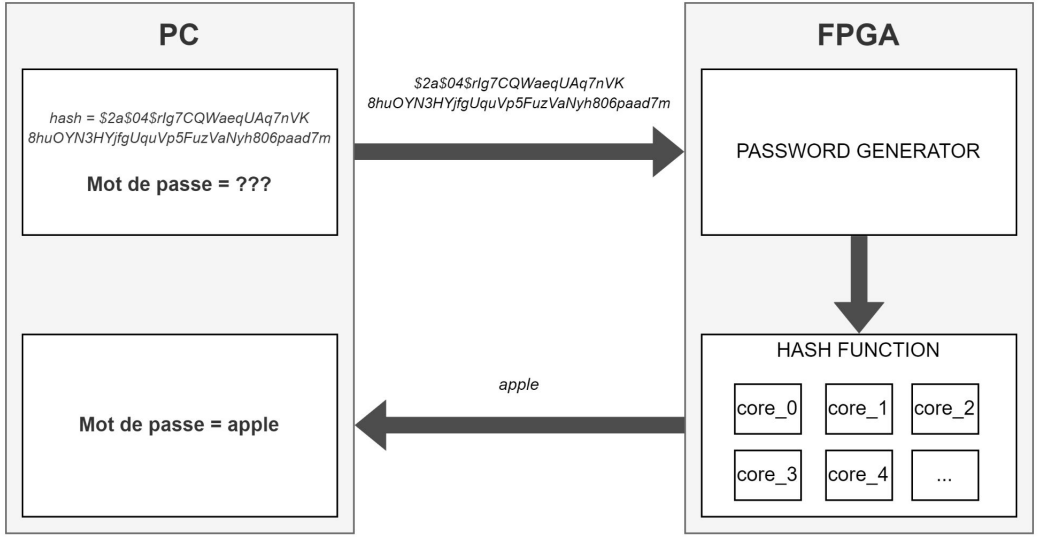
h e p i a
Haute école du paysage, d'ingénierie
et d'architecture de Genève

Table des matières :

Objectif

Objectif - Elca Security

Objectif - Schéma

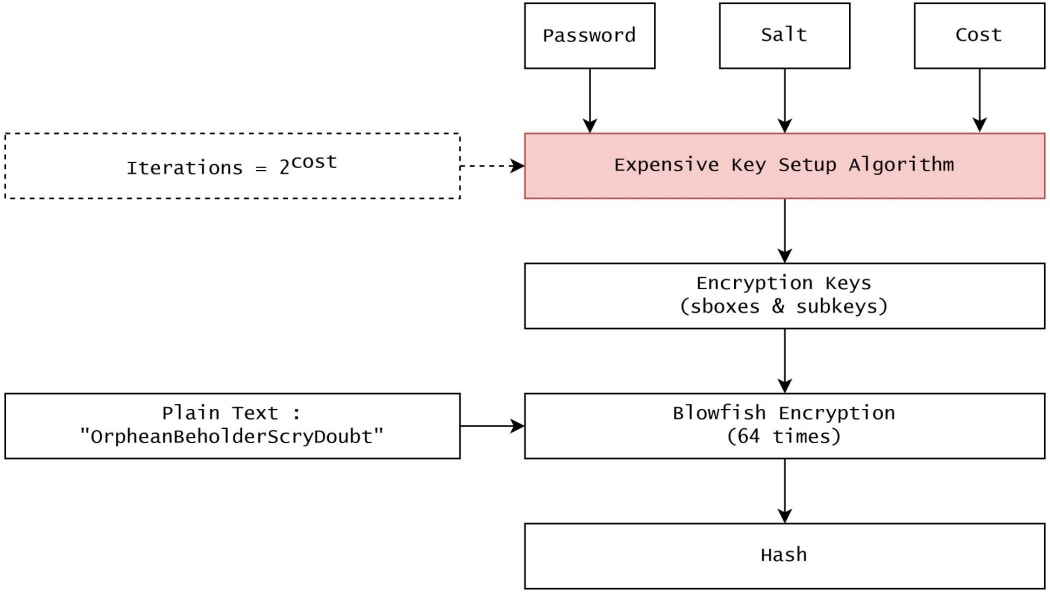


Objectif - FPGA vs CPU vs GPU

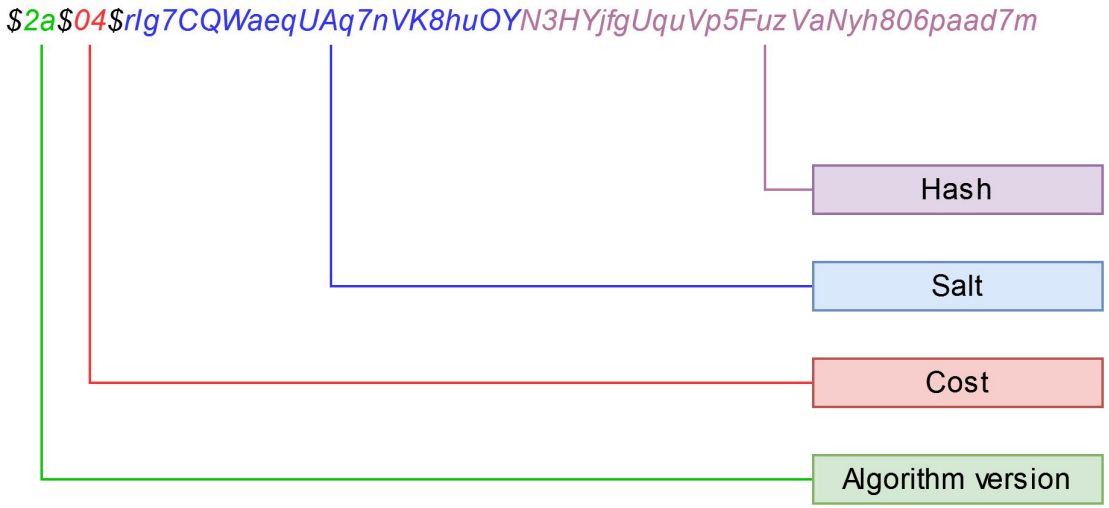
- Consommation
- Hashrate
- Coût

Bcrypt - Algorithme de hash

Bcrypt


Password
Salt
Cost
Iterations = 2^cost
Expensive Key Setup Algorithm
Encryption Keys
(sboxes & subkeys)
Plain Text :
"OrpheanBeholderScryDoubt"
Blowfish Encryption
(64 times)
Hash

Bcrypt - Format du hash



Implémentation existante

Implémentation existante

rub-hgi/**high-speed_bcrypt**

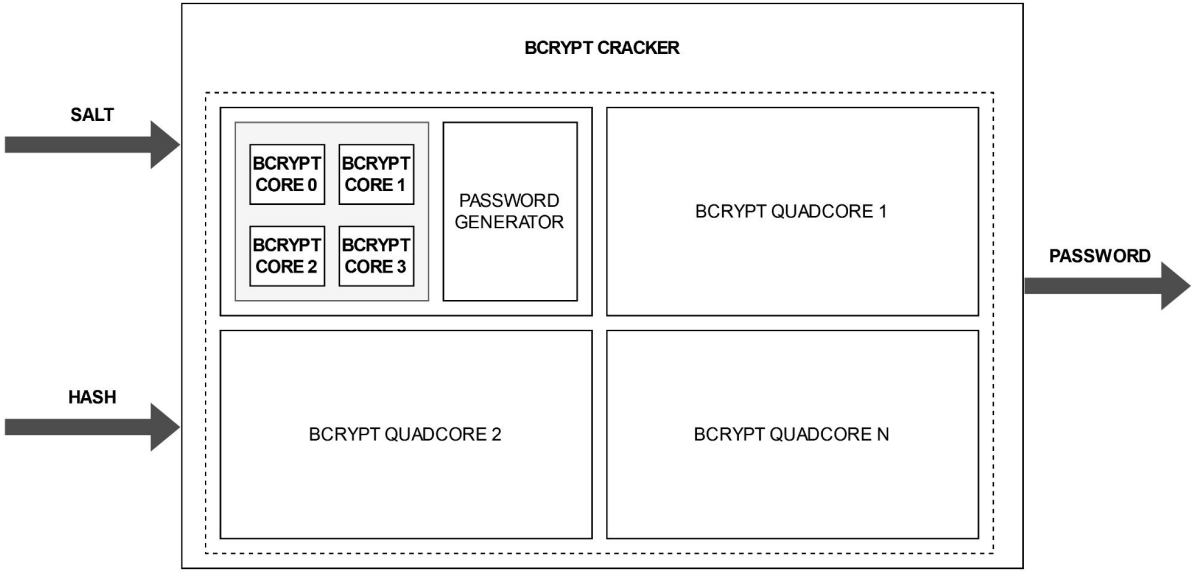
Horst Görtz Institut für IT-Sicherheit

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

1 Contributor | 0 Issues | 1 Star | 0 Forks

https://github.com/rub-hgi/high-speed_bcrypt

Implémentation existante - Schéma

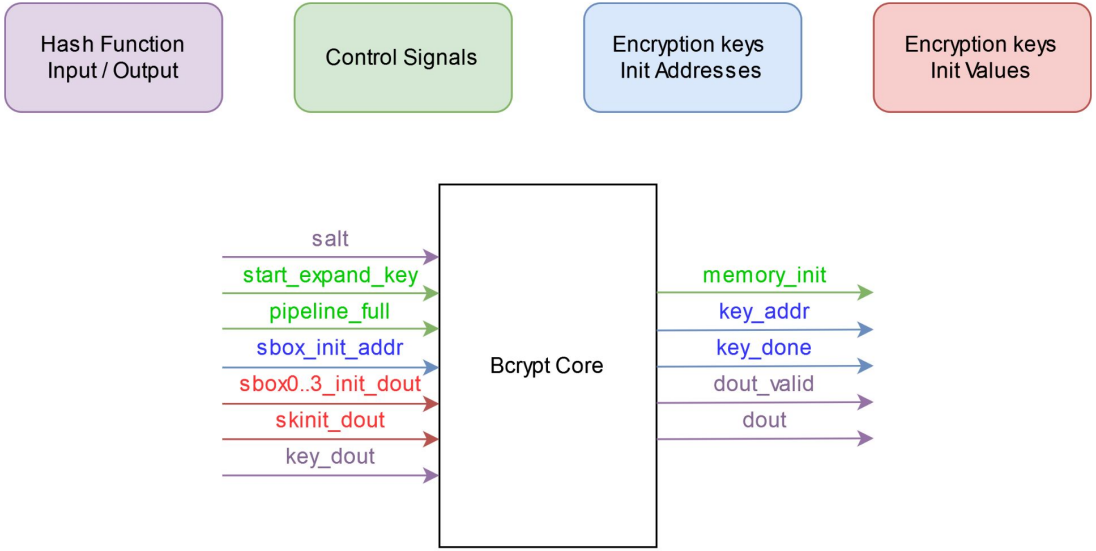


Implémentation existante - Problèmes

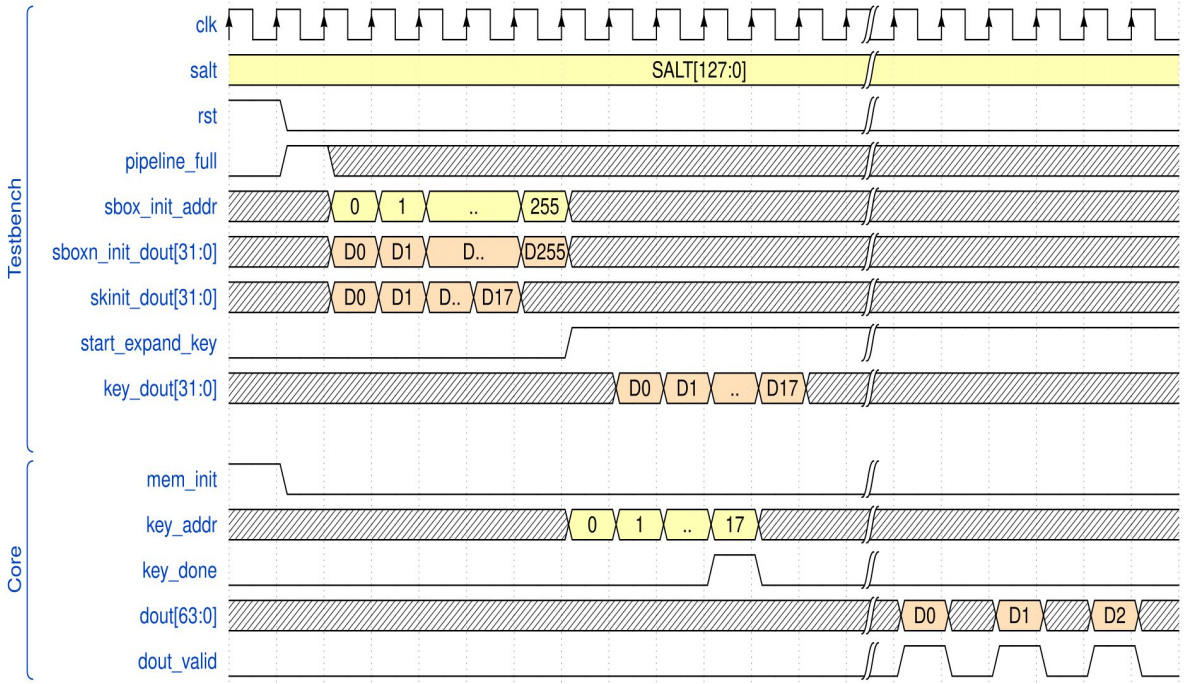
- Documentations
- Versions - Incohérences
- Testbenches incomplets
- Petites erreurs

Fonctionnement & Test

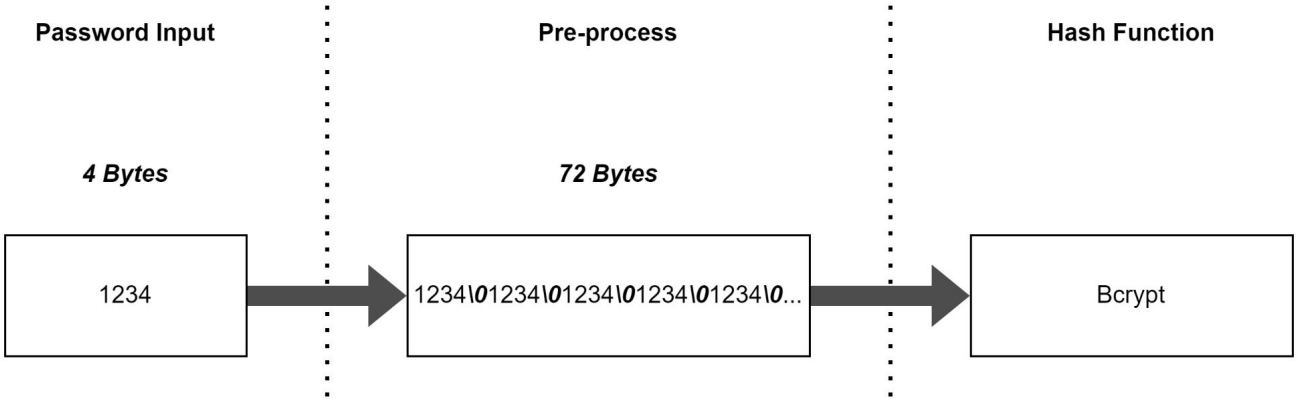
Bcrypt Core Interface



Bcrypt Core Timing

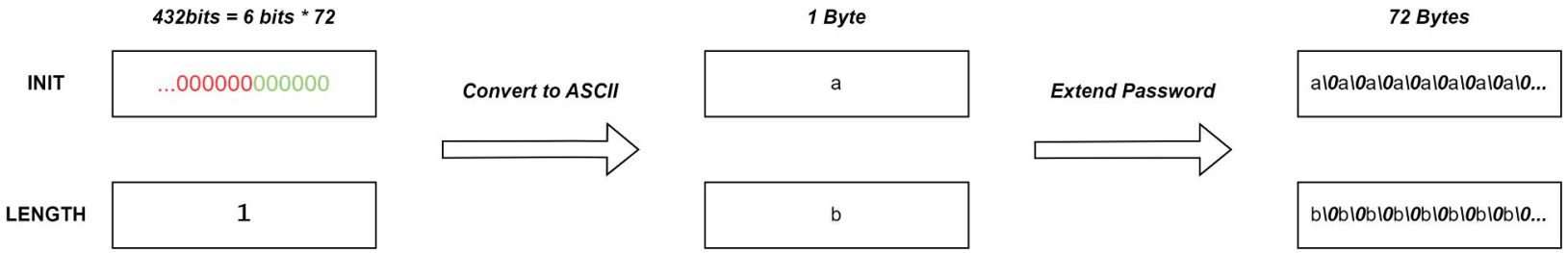


Bcrypt - Password Hashing

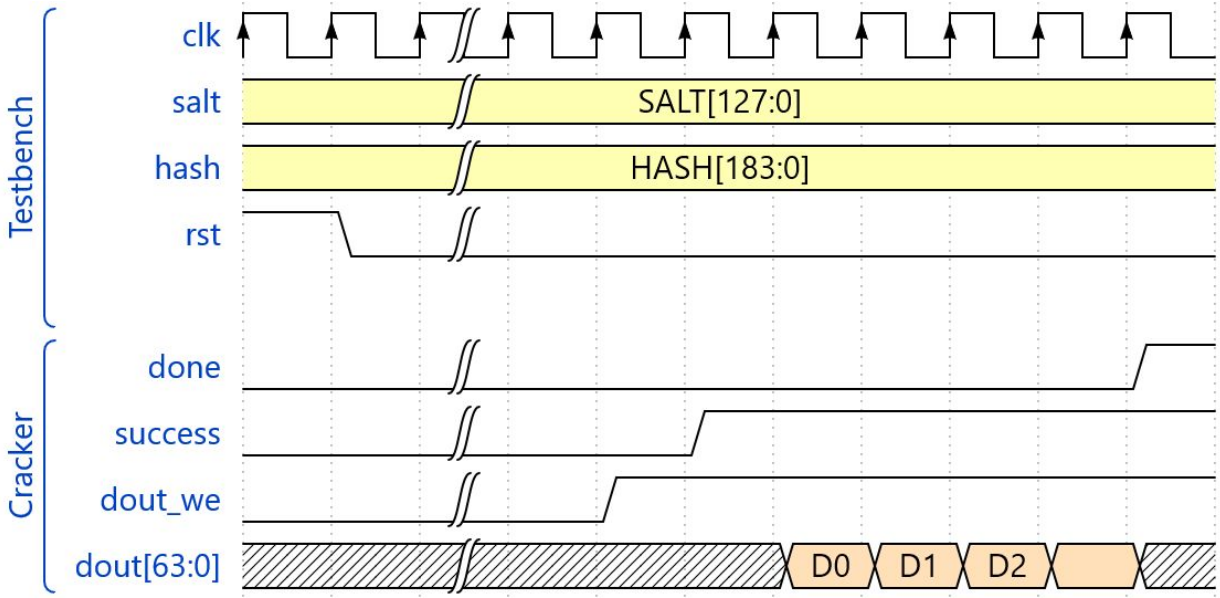


Password Generator

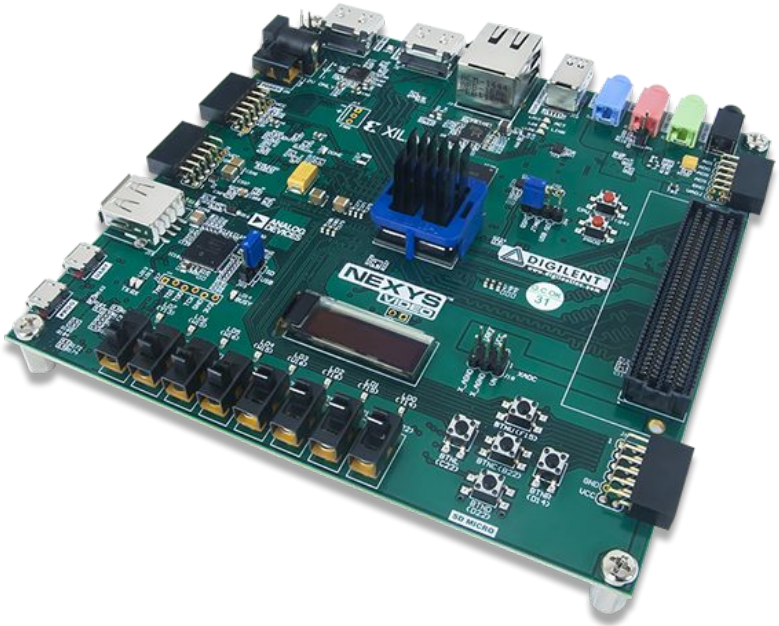
Conversion Table				
0x00	0x01	0x02	0x1b	0x35
NULL '\0'	'a'	'b'	'A'	'0'



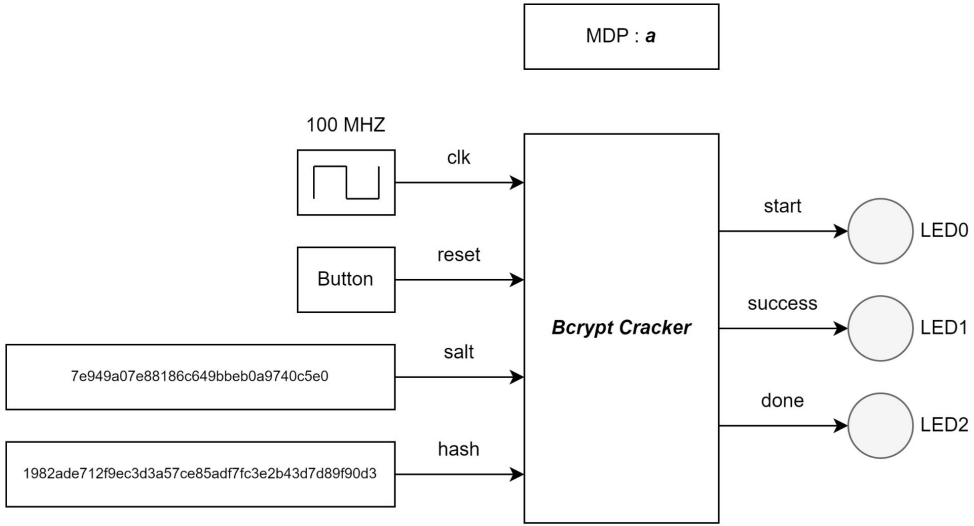
Bcrypt Cracker Timing



Bcrypt Cracker Test Board - Nexys Video

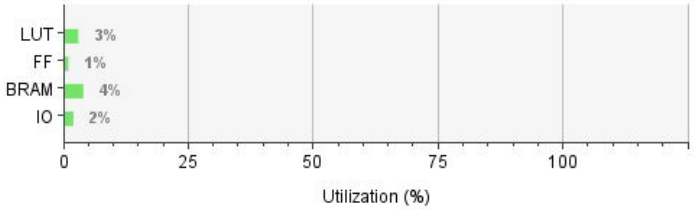


Bcrypt Cracker Test - Schéma



Bcrypt Cracker - Bilan Ressources

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



1 Quadcore => ***3.56 %***

25 Quadcore => ***89 %***

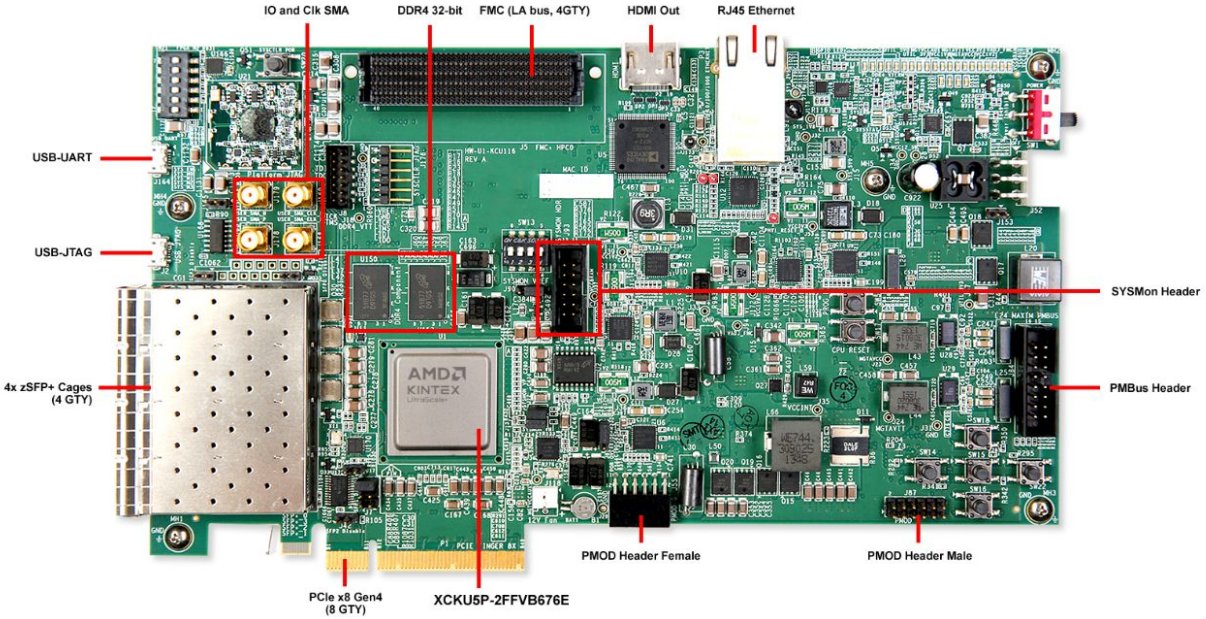
Bcrypt Cracker - Bilan Performances

Cost 5	GTX Titan X	16’625 Hash/s
	Nexys Video (*100 MHz, 25 Quadcores*)	15’400 Hash/s

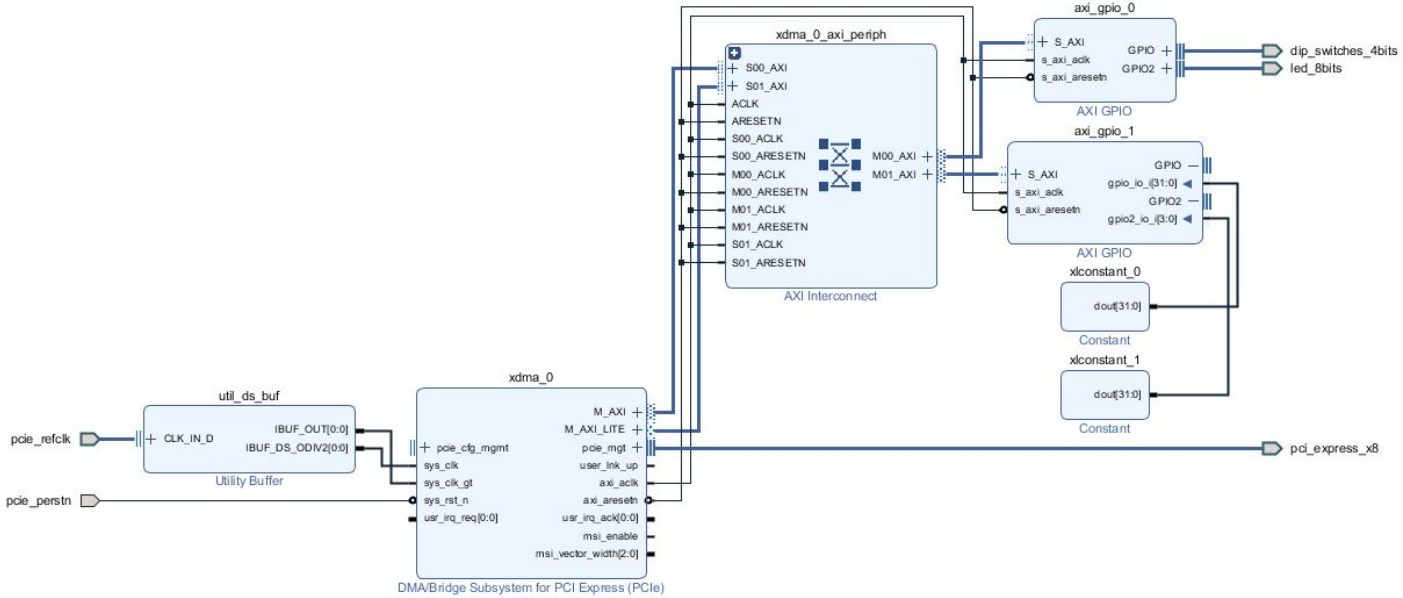
https://gist.github.com/epixoip/63c2ad11baf7bbd57544

Interface PC - FPGA

Interface PCIe - Kintex Ultrascale +



Interface PCIe - Block Design



Interface PCIe - Config xdma

- Vendor ID : 0x10EE, Device ID : 0x9038
- Maximum Link Speed : 8 GT/s
- Lane Width : x8
- Region Size : 128 kB

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)
resource0..N	PCI resource N, if present (binary, mmap, rw[1])
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)

```c
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 :

- Optimisation de l’implémentation
- Interface PCIe avec driver linux
- Mesures et comparaison avec GPU