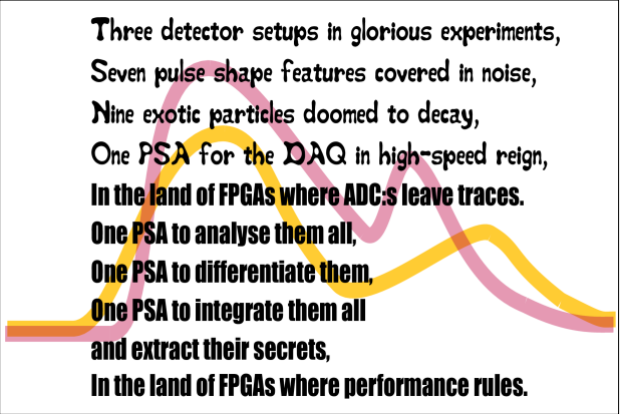


Pulsformsanalys i FPGA

Bakgrund

Moderna datainsamlingssystem går mer och mer emot att digitalisera hela pulsformen av varje detektorsignal. Den klassiska metoden är att bearbeta signalerna med analog elektronik och därefter digitalisera några väl valda storheter, oftast amplitud och tid vid en specifik del av pulsformen.

Förutom att en analys av hela pulsformen kan ge mer information finns ytterligare två skäl till utvecklingen: det är först nu som snabba analog-till-digitalomvandlare (ADC:er) med tillräckligt hög upplösning för att fånga de korta pulser som används i t.ex. kärn- och partikelfysikexperiment blivit tillgängliga. Den stora datamängden som ges av pulsformer (som har oundvikligt brus) blir också lätt ohanterlig om den inte snabbt bearbetas.



**Three detector setups in glorious experiments,
Seven pulse shape features covered in noise,
Nine exotic particles doomed to decay,
One PSA for the DAQ in high-speed reign,
In the land of FPGAs where ADC:s leave traces.
One PSA to analyse them all,
One PSA to differentiate them,
One PSA to integrate them all
and extract their secrets,
In the land of FPGAs where performance rules.**

Problembeskrivning

För att dra nytta av fördelarna med att digitalisera hela pulsformer, men samtidigt reducera datamängden och höja dess informationsdensitet, används så kallade FPGA:er (Field-Programmable Gate Array) direkt i datainsamlingen efter ADC:n. En FPGA kan utföra massivt parallella beräkningar och är väl lämpad för denna uppgift. Kommersiella moduler för pulsformsinsamling kommer med egna rutiner för pulsformsanalys, men är oftast hemliga och erbjuder begränsad funktionalitet. Det är därför på tiden att utveckla en öppen och generell kod för pulsformsanalys, som med hög prestanda och effektivitet kan konfigureras och användas oberoende av märke och modell av FPGA.

Arbetsätt

Bli bekant med FPGA-programmering i VHDL (Very high speed integrated circuit Hardware Description Language). Förstå hur pulsformsanalys går till och hur den kan brytas ned i små delkomponenter. Utveckla dessa tillsammans med ett motsvarande simuleringsverktyg i C för att testa och verifiera funktionalitet. Både VHDL- och C-komponenterna skall konfigureras med användarvänliga textfiler, lämpligen via lex/yacc.

Gruppstorlek

3-4 studenter. Projektet kan inte dubbleras.

Målgrupp

F, GU eller TM. Interesse för avancerad programmering. Projektet har koppling till kursen FUF050/FYP204 i subatomär fysik under LP4.

Litteraturtips

Douglas L. Perry, *VHDL : Programming By Example*, The McGraw-Hill Companies, Inc., 2002.

Handledare

Hans T. Törnqvist, tornqvih@chalmers.se, F8101, 031-772 3249, Subatomär fysik
Håkan T. Johansson, f96hajo@chalmers.se, F8109, 031-772 3253, Subatomär fysik