

Laveffekt ASIP design for komplekse applikasjoner

Blant tilgjengelige prosessorkjerner vil man sjeldent finne en prosessor som er effektoptimal for en kompleks applikasjon. Dersom applikasjonen allerede er optimalisert i software (SW) kan man redusere effektforbruket i prosessoren ytterligere ved å legge til applikasjonsspesifikke hardware (HW) enheter. Dette kan gjøres ved å tilpasse prosessoren til applikasjonen.

Av Yahya H. Yassin, vinner av Mikroelektronikkprisen 2009

Høy ytelse og lavt effektforbruk har alltid vært en av de mest diskuterte temaene innen innvedde systemer (embedded systems). Det har blitt forsket mye i dette feltet for å finne optimaliserings-teknikker som resulterer i en optimal kombinasjon av ytelse og effekt. HW/SW Co-Design er en metodelære som bidrar med teknikker for å optimalisere SW og HW slik at de fungerer optimalt sammen.

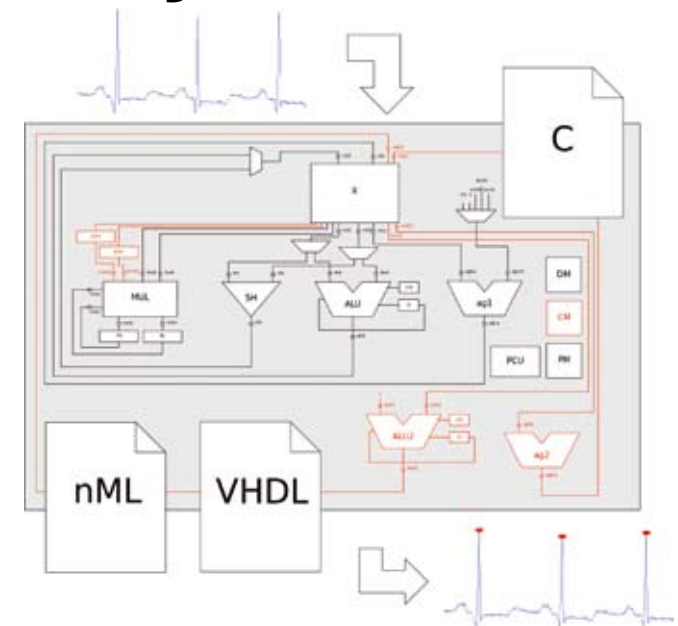
Denne artikkelen vil først beskrive bruk av HW/SW Co-Design for å få ned effektforbruket, og fordelene med design av applikasjonsspesifikke prosessorer (Application Specific Instruction-set Processor, ASIP). Deretter vil arbeidet i min masteroppgave bli presentert, der

HW/SW Co-Design ble tatt i bruk for å lage en ASIP som er optimalisert for å kjøre en avansert hjerterytmedetektor mest mulig effektivt med tanke på effektforbruk.

HW/SW Co-Design

For noen applikasjoner er det ønskelig å redusere kjøretiden. Dersom en applikasjon med mye dataprosessering er optimalisert i SW, og en raskere applikasjon er ønskelig, kan man benytte seg av optimaliseringsteknikker innen HW/SW Co-Design for å løse problemet.

Effektforbruket domineres i hovedsak av dynamisk effekt (den effekten en applikasjon krever når den kjører), og statisk effekt (når applikasjonen ikke kjører). Den statiske effekten domineres av lekkasjestrøm (konstant tap av effekt som skyl-



des lekkasje i transistorer).

De fleste digitale systemer i dag bruker komponenter med et instruksjonssettarkitektur (ISA), der det er mulig å ta i bruk HW/SW Co-Design. Blant de tilgjengelige plattformene finner man blant annet applikasjonsspesifikke integrerte kretser (ASIC), digitale signalprosessorer (DSP) og ASIP.

Fordelene med ASIP design

En ASIP er en mikroprosessor

som er spesialisert for en eller flere applikasjoner. Meningen med ASIP er å fylle gapet mellom ASIC og DSP. En ASIC har høy ytelse, men mangler fleksibiliteten for endringer i applikasjonen, mens en DSP tillater gjenbruk av SW (men med lavere ytelse og høyere effektforbruk sammenlignet med ASIC). En ASIP klarer å nå bedre ytelse og effektforbruk ved å utnytte applikasjonens karakteristikk. Prosesseringskraften forbedres

Scanditest
... viser veien



Scanditest Norge

Brüel & Kjær - nytt agentur

Vi kan nå også tilby kvalitetsinstrumenter og tilbehør fra Brüel & Kjær.

Brüel & Kjær leverer utstyr innen akustikk og datainnsamling for f.eks. måling av trafikkstøy, vibrasjon i bilmotorer eller måling av akustikk i bygninger.



Scanditest Norge AS, Postboks 244, Langgt 21, 3080 Holmestrand
Tlf: 33 05 34 50 - e-post: roar@scanditest.no - www.scanditest.no

ved å introdusere skreddersydd HW moduler, og arealet forbedres ved å fjerne moduler som man ikke trenger.

Generelt sett er ikke design av applikasjonsspesifikke kretser en enkel oppgave, og det krever at en setter seg godt inn i arkitekturen som skal implementeres. I tillegg kreves det komplekse verktøy for å evaluere og analysere designet underveis i designfasen. Det finnes forskjellige måter å designe en ASIP på, men den mest vanlige måten er beskrevet i det følgende.

Først må applikasjonens designbegrensninger analyseres. For å kunne designe en ASIP, må en kjenne godt til applikasjonen og hvordan den fungerer. Videre må en identifisere mulige arkitekturer som oppfyller applikasjonensbegrensningene. I tillegg kan ytelsen estimeres for de foreslåtte arkitekturer, slik at en kan velge ut den arkitekturen som oppfyller ytelses- og effektkravene på best mulig måte. I det neste steget lager man instruksjonssettet for applikasjonen og arkitekturen. En skreddersydd kompilator eller en "re-targetable" kode generator blir så brukt for å syntetisere koden for applikasjonen. Helt til slutt blir plattformen syntetisert ved å bruke et HW beskrivende språk som VHDL eller Verilog.

Den komplekse algoritmen

Under masteroppgaven var applikasjonen en avansert hjerterytmedetektor, modellert i Matlab, som brukte en kontinuerlig wavelet transform modulus maxima (CWTMM) definert som følger:

```

480 2336 // sv r3,r6
>= 476 4100 // ld r0,de(r3++)
>= 473 4017 // ld r7,de(r3++)
>= 472 2807 // make r0,r7
472 2300 // sv r0,r1
>= 474 4028 // add r3,r3,r0
>= 473 1608 // ss r0,r0
>= 476 8300 // addc r0,r0,r0
477 8390 // ld r0,de(r3++)
    
```

```

for(n = n_min; n <= n_max; n++){
//temp = temp + _a->array[n-1] * _b->array[n-1];
temp = temp + _a->array[n-1] * _b->array[n-1];
}
// Divide result with the root of a
z =
= FIXED POINT output;
    
```

Figur 1. Reduksjon av assemblyinstruksjoner (markert i gult).

CWT:

$$T(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \cdot W\left(\frac{b-t}{a}\right) dt$$

Modulus maxima:

$$\frac{d|T(a,b)|^2}{db} = 0$$

Algoritmen er avhengig av å bufre opp 3 sekunder av hjerterytmesamplere (med 0,5 sekunders overlapp mellom etterfølgende samplere), før de bufrede samplingsdataene blir sendt til prosessoren for deteksjon av pulser. Sanntidskravet for dette systemet er å fullføre prosesseringen av samplingsdataene før neste sett av data er klar (2 sekunder i dette tilfellet). Et tilleggskrav for masteroppgaven var å optimalisere effektforbruket. Velkjente databaser med opptak av hjerterytmesamplere fra Massachusetts Institute of Technology (MIT) ble blant annet brukt for å teste applikasjonen.

SW utviklingen

Før en applikasjon kan implementeres på en prosessor, må programmet være skrevet i et språk som designverktøyet kan implementere. I dette tilfellet måtte applikasjonen konverteres fra Matlab-modellen til C kode. Siden Matlab-modellen var optimalisert med bruk av innebygde

Matlabfunksjoner, måtte hele koden skrives om bit for bit i Matlab med kode som lett kunne konverteres til C kode (dvs. kun inneholdt if-setninger, for- og while-løkker). Alternativet var å skrive hele koden på nytt i C, men fordelene med å lage en Matlab-prototype var at man kunne konvertere koden i flere steg og verifisere en bit av gangen med den samme testdatabasen i Matlab. Etter at prototypekoden var ferdig, ble den raskt konvertert til C kode.

Den fungerende C-koden ble deretter optimalisert ved bruk av kjente SW optimaliseringsteknikker som fixed-point konvertering, reduksjon av løkker, og eliminering av komplekse operasjoner (som divisjon og røtter) ved omskriving av koden. I tillegg ble minneforbruket analysert og minnekravet ble redusert ved å utnytte det faktum at et menneske ikke kan ha en puls høyere enn 300 slag i minuttet. Koden ble deretter analysert med profileringsverktøy for å identifisere flaskehalsene i SW (hvilke funksjonskall som ble brukt mest) og minnelekkasjer. Resultatet av optimaliseringen er en implementerbar C kode som hadde mindre minneforbruk, og sammenlignet

med originalmodellen var avvirket i hjerterytmedeteksjon mindre enn 0,5 % (som er mer enn godt nok for denne applikasjonen).

HW utviklingen

En enkel prosessor, modellert i et prosessorbeskrivende språk kalt nML, ble brukt som utgangspunkt. Den optimaliserte C koden ble mappet på arkitekturen ved hjelp av utviklingsverktøyet IP designer fra Target Compiler Technologies.

Ved å benytte seg av profileringsverktøyet til IP designer, ble den kritiske stien i koden identifisert, inkludert hvilke assemblyoperasjoner som ble kjørt, og kjøretiden i form av antall klokkesykler. C koden brukte omlag 27,7 millioner klokkesykler på å prosessere et sett med data på den uoptimaliserte prosessoren.

Kjente HW optimaliseringsteknikker, som parallellisering av instruksjonssettet, ble brukt for å kombinere assemblyinstruksjoner. For å få til dette ble det designet skreddersydd HW enheter. Dette ble gjort i flere iterasjoner helt til alle kritiske stier var optimalisert. I tillegg ble dataminnene delt i to separate moduler for å kunne lese to forskjellige minneområder samti-

Møt morgendagens utfordringer! Innbydelse til seminar

Den Norske Utleggerforening (DNU)
inviterer til seminar i Sandvika/Oslo den 18. og 19. mars 2010

På første dag blir det parallelle sesjoner på de forskjellige verktøyplattformene i regi av DAK-leverandørene.

Dagen avsluttes med en sosial sammenkomst på kvelden.

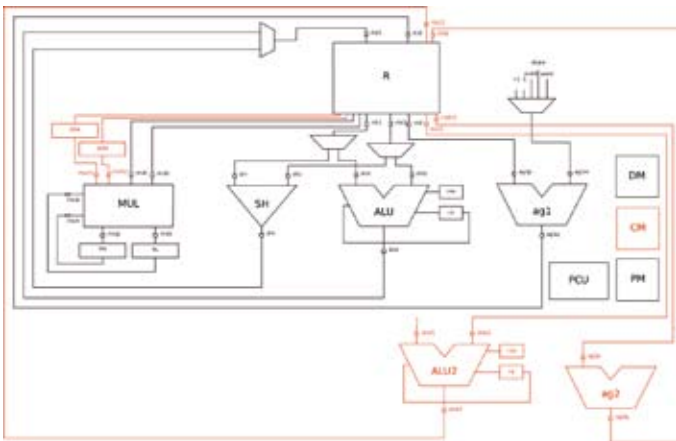
Neste dag innledes med årsmøte i DNU. Deretter blir det felles faglig seminar.

Nærmere opplysninger og påmelding finnes på www.dnu.no



DNU
Den Norske Utleggerforening

Den Norske Utleggerforening er en tverrfaglig interesseforening tilknyttet norsk elektronikkindustri. Formålet er å bidra til et høyt faglig nivå både hos våre medlemsbedrifter og i bransjen i sin alminnelighet. Det gjør vi blant annet ved å avholde seminarer hvor vi tar opp aktuelle problemstillinger, og gjennom informasjon om bruk og erfaring med ny teknologi, nyutviklet programvare og nytt teknisk utstyr. Vi legger også stor vekt på sosialt og kollegialt samvær som et virkemiddel for å bygge gode profesjonelle relasjoner både mellom individer og bedrifter.



Figur 2. Oversikt over prosessorens moduler med skreddersydde HW moduler i rødt.

dig. Figur 1 viser hvordan flere assemblyoperasjoner (markert i gult) ble erstattet med en enkelt operasjon som resultat av en skreddersydd HW modul. Figur 2 viser en oversikt over modulene i den optimaliserte prosessoren (med de skreddersydde HW enhetene markert i rødt).

Ved hjelp av IP Designer verktøyet ble det generert VHDL filer av den optimaliserte prosessoren. Utviklingsverktøyet fra Cadence og Synopsis ble brukt til syntese og utlegg. Ved å benytte laveffekt 90nm standardcellebibliotek fra TSMC, kunne man gjennom nettlitesimuleringen få ut informasjon om effektforbruket i såkalte value change dump (VCD) filer. Ved hjelp av PrimeTime fra Synopsis ble effekt tallene ekstrahert fra VCD filene.

Resultater og videre arbeid

Den optimaliserte prosessoren trengte ikke mer enn 5,3 millioner klokkesyklus for å kjøre C koden. Dette tilsvarer en reduksjon av kjøretiden på 81 %. Samtidig brukte prosessoren 55 % mindre samlet effekt enn den uoptimaliserte versjonen. Av den samlede effekten ble det dynamiske effektforbruket redusert med 78 %, mens lekkasjestrømmen ble redusert med 13 %. Andelen av tiden prosessoren brukte på å prosessere data med hensyn på sanntidskravet på 2 sekunder, ble redusert fra 14 % til 2.8 %.

For videre arbeid ble det estimert at man kan redusere effektforbruket ytterligere med 35 % ved å benytte seg av en teknikk kalt Dynamic Voltage and Frequency Scaling (DVFS). Dette baserer seg på å redusere den

dynamiske effekten ved å skru ned klokkefrekvensen. Alternativt kan man fokusere på å redusere lekkasjestrømmen ved slå av prosessoren etter at den er ferdig med å prosessere data og aktivere den like før neste sett av samplingsdata er klare. Det statiske effektforbruket vil da bli eliminert i denne perioden. Dette kan man få til ved bruk av såkalt power gating istedenfor DVFS. Estimater indikerer at dette kan redusere det nåværende effektforbruket ytterligere med 62 %.

Oppsummering

Denne artikkelen beskriver kort bakgrunnen for mitt arbeid og designet som ble utført i min masteroppgave. Ved å analysere en kompleks applikasjon grundig, kan man ved hjelp av kjente SW og HW optimaliseringsteknikker designe en prosessor som har betydelig høyere ytelse og lavere effektforbruk enn en standard prosessor.

Resultatet av mitt arbeid er blitt videreutviklet av IMEC, og de har tatt hensyn til mine råd for videre optimalisering av applikasjonen for å redusere minneforbruket og effektforbruket ytterligere.

For mer informasjon om SW og HW optimaliseringsteknikker anbefaler jeg interesserte lesere å se på Benini og De Michelli "System-level power optimization: techniques and tools", ACM Transactions on Design Automation of Electronic Systems, 5(2):115-192, April 2000. For spørsmål angående masteroppgaven min kan jeg kontaktes på følgende e-post: yhyassin@gmail.com.



S.E.E.

SCANDINAVIAN
ELECTRONICS
EVENT

Elektronikkbransjens nye, store møtested!

S.E.E. 2010 nærmer seg, og vi kan slå fast at interessen og engasjementet fra alle kanter er imponerende. Nærmere 200 ledende elektronikkleverandører fra hele verden står allerede i startgropen for årets store bransjemanifestasjon. Og mange flere er på vei inn.

Nå begynner også besøksaktivitetene å ta form – her er noen eksempler:

Semiconductor Meeting Point

13 av verdens største halvlederprodusenter deltar. Et unikt møtested for halvlederspesialister og -konstruktører. Utstilling og seminarer.

Learn Shops, Testing & Måling

Den nyeste teknologien og trendene innen testing og måling. Seminarer.

Embedded Awards and Seminars

Fremgangsrike produkter, nominert til Embedded Awards i årenes løp, blir presentert. Seminarprogram om Embedded Technology.

Live Production

En levende produksjonslinje hvor hele kjeden er med, fra konstruksjon til ferdig kretskort, med fokus på pålitelighet.

Wireless Inspiration

Nye muligheter med trådløse applikasjoner. Presentasjoner og workshops.

Showcase Sweden

En manifestasjon av svensk elektronikkindustri.

Les mer om alle aktivitetene på nettstedet vårt, www.see-event.se



13.-15. april 2010
Stockholmsmässan

For fri adgang, registrer deg på:
www.see-event.se

Arrangører: Branschorganisationen Svensk Elektronik og Stockholmsmässan.