



Så kan en systemkrets konfigureras för Realtidlinux

Låt inte cachemissar och virriga prediktorer rubba determinismen

Linux och realtid – låter som ett omaka par. Men Microchip har en resolut lösning som utnyttjar att minnessystem och Risc-V-kärnorna är konfigurera i FPGA-systemkretsen PolarFire. Vi kommer att berätta mer om lösningen senare, men först några grundläggande ord om "realtid" och vad det egentligen innebär.

Definitionen av ett realtidssystem, i sin enklaste form, är ett inbyggt system som periodvis exekverar deterministiskt. Determinism är ett grundläggande krav på realtidssystem eftersom de typiskt styr maskiner. Ingen vill ha en numeriskt styrd bormaskin som tar sig från punkt A till punkt B på 10 millisekunder på tisdag men behöver 20 millisekunder på onsdag. Likaledes bör en pilots styrsystem kontrollera flygplanets vingyta på exakt samma sätt, varje gång och under alla förhållanden.

Figur 1 visar ett deterministiskt system där det sker periodiska avbrott och avbrottsrutinen hanterar tidskritisk kod. Exekveringstiden för den koden är och måste vara deterministisk, annars får man ett system som uppför sig som figur 2, som störs av oregelbundna uppdateringar av maskinvaran.

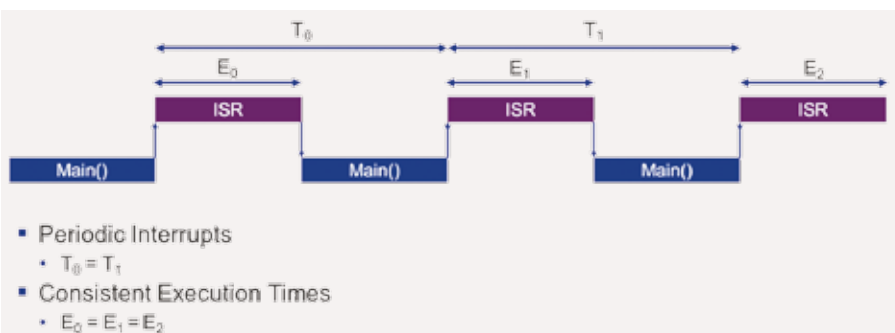
INBYGGNADSSYSTEM BEHÖVER OFTA den funktionsrikedom som finns i Linux och tillhörande mellanvara. Men Linux kräver en minneshanteringsenhet (Memory Management Unit, MMU) för virtualisering av fysiskt minne. Processorer med inbyggd MMU har även cache, åtminstone L1-cache och i många fall även L2-cache. Tyvärr är cache och determinism ömsesidigt uteslutande, vilket demonstreras i figur 3 – cachemissar i L1- och L2-cache introducerar tidsjitter. Orsaken är att processorns pipeline måste pausa medan cache-linorna fylls på. Ett större cache skulle kunna minska frekvensen av cachemissar, men aldrig ta bort dem helt och hållet.

I processorer som kan köra Linux är prediktorn (branch predictor) ytterligare en källa till exekveringsjitter. En prediktor gissar vilken väg exekveringen kommer att ta ge-

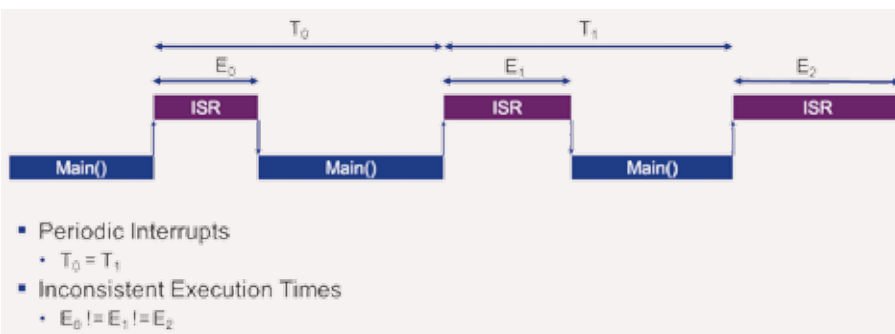


Av Tim Morin, Microchip

Tim Morin är chef för produktmarknadsföring på affärsenheten för FPGA:er. Han har drygt 30 års erfarenhet av teknik och marknadsföring inom halvledar- och försvarsindustrin. På Texas Instruments jobbade han med bildbehandlingssystem och på Atmel var han bland annat produktplanerare för SoC FPGA-familjerna SmartFusion- och SmartFusion2. Tim Morin har en examen i elektroteknik från Purdue University.



Figur 1. Exempel på deterministisk exekvering.



Figur 2. Avbrottsrutin (Interrupt Service Routine, ISR) med exekveringstider som varierar.

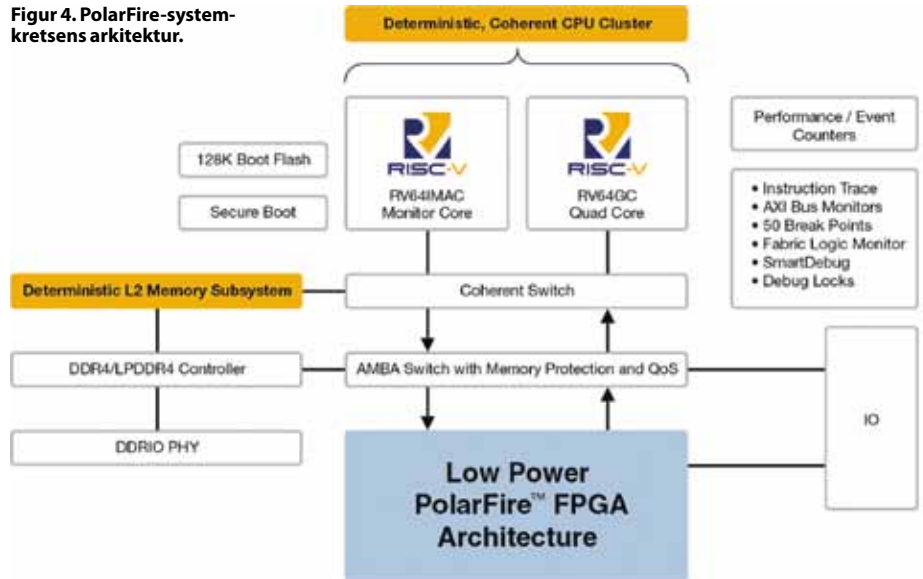


Figur 3. Missar i L1- och L2-cache påverkar determinismen.



”Vissa processorer kan köra Linux men inte exekvera kod deterministiskt och andra kan exekvera kod deterministiskt men inte köra Linux.”

Figur 4. PolarFire-systemkretsens arkitektur.



►► nom koden, för att kunna fylla rörledningen i förväg och på så vis öka genomströmningen. Oavsett implementering gissar prediktorn fel ibland. Vid sådana misstag töms rörledningen vilket rubbar determinismen.

NÄR EN AVBROTTSROUTIN startar baseras prediktorns exekveringshistorik på programhoppet i huvudrutinen, inte i avbrottsrutinen. Det leder till tömningar av rörledningen, vilket gör att exekveringstiden varierar från en avbrottsrutin till en annan. Genom att använda en processor som tillåter programmet att stänga av prediktorn kan du ta kontrollen över denna indeterminism, om än till priset av att du offerar en del prestanda.

Vissa processorer kan köra Linux men inte exekvera kod deterministiskt och andra kan exekvera kod deterministiskt men inte köra

Linux. Visst vore det trevligt med en arkitektur i verktygslådan som kunde bådadera? Och det finns faktiskt, i Microchips Risc-V-baserade FPGA-arkitektur för PolarFire.

Figur 4 visar fyra stycken Risc-V-cpu:er (av arkitekturen RV64GC) som kan köra Linux eftersom de har MMU, och en femte (RV64IMAC) som saknar MMU och därmed inte kan göra det. I övrigt är skillnaden mellan RV64IMAC och RV64GC att RV64GC arbetar med flyttal i dubbel precision.

För att öka arkitekturs determinism kan kärnornas prediktorer stängas av, enligt tidigare resonemang. Det kan antingen göras direkt vid uppstart eller under det att avbrottsrutiner körs.

För att ytterligare trimma determinismen saknar alla fem kärnorna out-of-order-rörledningar. Detta stoppar som bonus cyber-

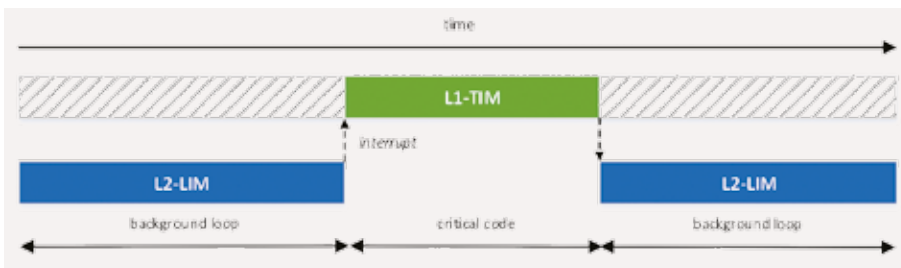
attacker av typerna Spectre och Meltdown.

Hittills har vi endast diskuterat determinism i cpu-kärnor. Men kod exekveras från minnet, så låt oss ta en titt även på minnesystemet i PolarFire. För det första hålls minnet koherent (samstämmigt) vilket betyder att alla kärnor ser samma innehåll i sina lokala kopior av minnesceller. Minnesceller som bara existerar i en enda kopia är uppenbarligen redan koherenta, medan minnesceller som finns i flera kopior i olika delar av minneshierarkin hanteras i en så kallad koherens-manager.

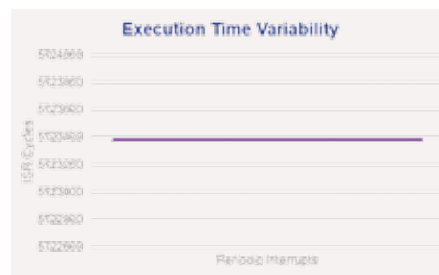
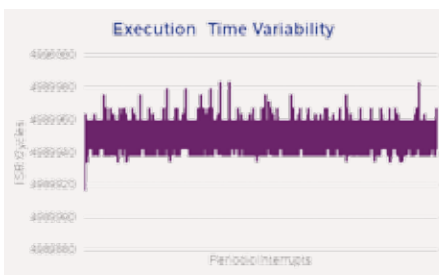
MINNESHIERARKIN I POLARFIRE har tre cachnivåer: L1, L2 och L3 där det sistnämnda utnyttjar en härdad 36-bitars styrkrets för LPDDR3/LPDDR4 och DDR3/DDR4. De fyra extrabitarna är felrättande koder för SECDED (single error correct, double error detect).

RV64GC-kärnorna, som används för att köra tillämpningsprogram, har ett 32 kbyte åttavägars mängdassociativt I\$TIM (instruktionscache av typen TIM, Tightly Integrated Memory) och ett dito D\$TIM för data. De kan båda konfigureras ner till ett evägs-cache. RV64IMAC används som övervakningskärna (Monitor) och har 16 kbyte tvåvägars instruktionscache och 8 kbyte datacache. D\$TIM är en scratchpad, ett kladdpapper för data som kod kan exekveras från. L1-TIM är deterministiskt med kort åtkomsttid och stöder SECDED.

L2-MINNESYSTEMET består av 2 MB SECDED-minne som kan konfigureras på tre sätt: som ett 16-vägars mängdassocierat cache, ett LIM (Loosely Integrated Memory) eller ett scratchpad. LIM-minne kan fästas vid en processor och storleksbestämmas med cachevägar. LIM-minnen kan med andra ord kon-



Figur 5. Deterministisk exekvering med LIM och TIM.



Figur 6. Prediktorns misstag rubbar determinismen. Figur 7. Deterministiskt beteende.

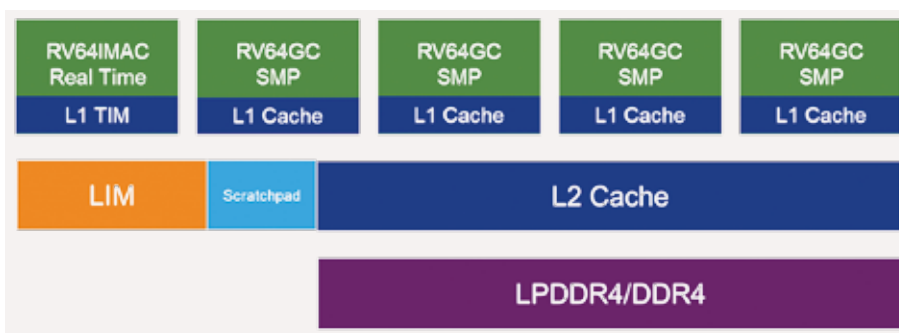
strueras i 128 kbyte-block (vägar) och tilldelas exklusivt tillträde till en processor.

Konfigurerat som LIM får L2 deterministisk åtkomst till sin kärna och blir automatiskt koherent eftersom inga kopior av innehållet delas med L1 eller L3. LIM är idealiskt för deterministisk exekvering av kod i både huvudtillämpning och avbrottsrutiner. Figur 5 visar ett deterministiskt system där L2 är konfigurerat som ett LIM och L1 som ett TIM.

EFTERSOM PREDIKTORNS förutsägelser om programhopp ibland slår fel, kan det tyvärr fortfarande uppstå variation i exekveringstiden för avbrottsrutiner, även när L2 konfigurerats som LIM. Figur 6 visar exekveringen av en tillämpning när L1 är konfigurerat som ett TIM och L2 konfigurerat som ett LIM. Den horisontella axeln visar avbrott och den vertikala visar cykeltiden för avbrottsrutiner. Som synes varierar exekveringen för avbrottsrutiner över tiden.

I figur 7 stänger vi av prediktorerna – och erhåller som belöning vår önskade determinism.

Precis som LIM, kan scratchpad-minnet konfigureras i block om 128 Kbyte och tilldelas till cpu-kärnor. En scratchpad passar utmärkt som en delad minnesresurs mellan en processor som exekverar kod från LIM



Figur 8. En konfiguration av PolarFire SoC Microprocessor Subsystem. Meddelanden överförs via delning av det automatiskt koherenta minnet.

och processorer som exekverar kod – vanligen Linux– från delsystemen L1/L2 och L3. Om RV64IMAC-programmet skriver data i sin scratchpad och en kopia av den minnesplatsen samtidigt existerar någon annanstans i L1/L2/L3 ser koherens-managern till att koherensen garanteras. På så vis kan ett realtidsprogram på ett koherent sätt dela data med en användarprogram i Linux.

Figur 8 visar ett av flera sätt att konfigurera det som kallas PolarFire SoC Microprocessor Subsystem. RV64IMAC-kärnan hanterar realtidsfunktioner och RV64GC-kärnorna kör Linux. Om en realtidsfunktion kräver flyttalsprestanda kan du använda RV64GC till

detta eftersom dess prediktorer kan stängas av och minnessystemet L1 kan konfigureras som ett TIM.

Sammanfattning

Determinism är fundamentalt för realtids-system. Det finns många processorer på marknaden som kan köra Linux men inte kan exekvera kod deterministiskt och det finns andra som kan exekvera kod deterministiskt men inte kan köra Linux. PolarFire SoC har ett unikt, flexibelt minnes-delsystem som gör att hårda realtidsapplikationer och Linux-applikationer kan samexistera på ett flexibelt, koherent sätt. ■