

# Smart avlusningsverktyg kortar utvecklingstiden för FPGA:er



**F**PGA-konstruktörer har många hinder att övervinna innan de kan lämna över sin design till produktionen. Normalt börjar de med att välja en lämplig krets. Sedan implementerar de konstruktionen i ett hårdvarubeskriande språk (HDL), gör ledningsdragningen (routingen) och avlusar slutligen hela FPGA:an innan den kan börja tillverkas.

Särskilt för dem som arbetar med industriella produkter och på inbyggnadsmarknaden finns det många olika kretsar som passar en specifik uppgift. Valet av FPGA-tillverkare avgörs ofta av tidigare erfarenheter av utvecklingsverktygen. Det finns dock en viktigare parameter att ta med i beslutsunderlaget. Jag syftar på möjligheterna avlusa både mjukvara och FPGA:an. Det finns många debugverktyg från leverantörer som Altera, Microsemi och Xilinx, men det finns numera smartare verktyg som bör vägas in i valet.

Alla stora FPGA-tillverkare erbjuder logikanalysatorer som debugverktyg. Det är en instansierad del av logiken som använder interna logikelement i FPGA:an och det interna blockminnet för att implementera samplingen av signalen. En konstruktör kan specificera vilka signaler som ska studeras och ställa in en trigger för att logikanalysatorn ska starta vid rätt tillfälle. När detta är gjort måste man göra om syntesen liksom ledningsdragningen så att funktionen kommer med i designen. När den är omkompilerad och laddad kan man börja studera signalerna i logikanalysatorn.

**DET ÄR VIKTIGT ATT NOTERA** att på grund av att dessa signaler måste samplas, så fångar de inte realtidsegenskaperna i data. Logikanalysatorn kan bara köras med en hastighet som tillåter den att sampla och lagra data i det interna minnet. En stor nackdel med det här avlusningsverktyget är att felet ofta försvinner eller maskeras när designen kompileras om. Sedan kan det dyka upp igen när man gör en ny ledningsdragning. En annan nackdel med logikanalysatorn är att samplingen sannolikt är långsammare än signalerna vilket innebär att den kan missa snabba förlopp.

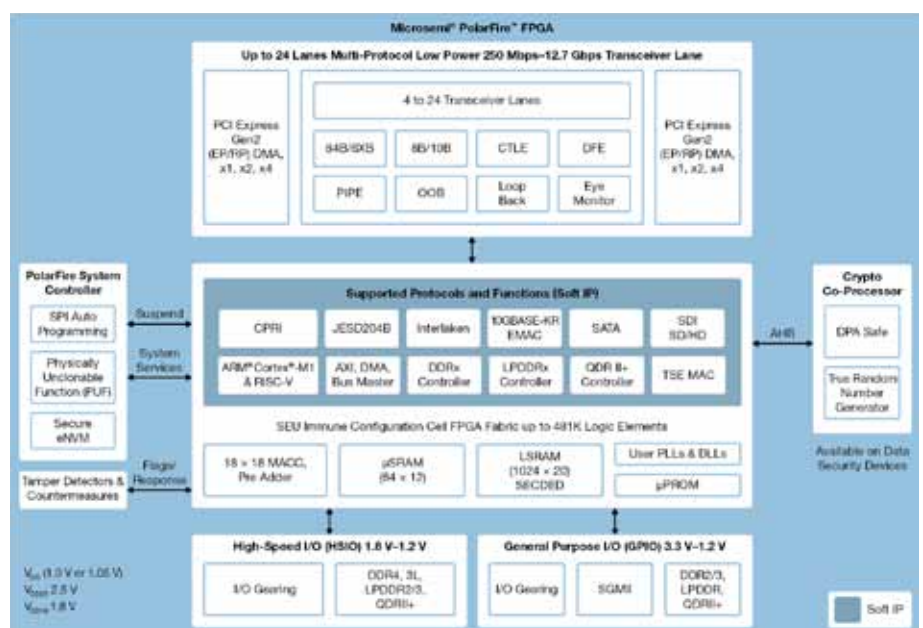


## Av Ted Marena, Microsemi

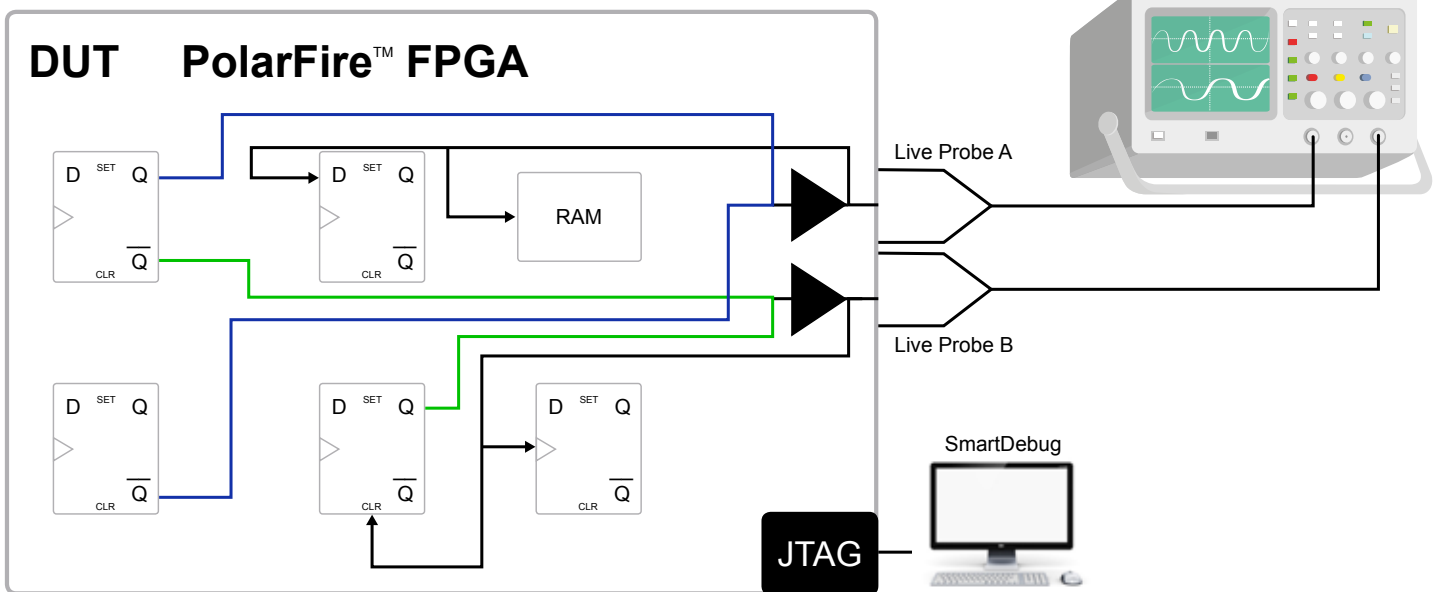
**Ted Marena** är ansvarig för marknadsföring av system-FPGA:er på Microsemi. Han har över 25 års erfarenhet av området och började sin karriär som konstruktör, bytte till FAE och därefter försäljning för att idag syssla med marknadsföring.

Trots dessa begränsningar så kan du se signalernas status – baserat på triggvillkoren – i logikanalysatorn vilket underlättar felsökningen av konstruktionen. Att använda en logikanalysator är en iterativ process – en konstruktör ser vad som händer, gör en uppdatering och kompilerar sedan om designen. Sedan studerar konstruktören de nya resultaten och upprepar processen gång på gång tills buggen är fixad. Tiden för varje iteration och varje bugg varierar men man kommer inte att hitta alla fel på grund av den långsamma samplingshastigheten hos logikanalysatorn.

Begränsningarna i logikanalysatorn har lett till att det utvecklats nya verktyg som snabbar upp valideringen av FPGA:er och kort. Synopsys har ett verktyg kallat Identify som integrerar en logikanalysator med företagets syntesverktyg Synplify vilket ger kortare cykeltider under avlusningen. Med en integrerad logikanalysator blir det enklare att titta in i konstruktionen och definiera triggvillkor. En konstruktör kan också ändra i designen och automatiskt få ändringarna mappade till RTL-kod. För att begränsa användningen av interna FPGA-resurser kan Identify sampla grupper av



**Polarfire är flashbaserad och finns i fyra modeller med 100 000 till 500 000 logikelement. Det är gott om minnesblock av olika typer liksom DSP-block, upp till 1024 stycken av de senare. Vidare finns upp till 24 transceivrar som klarar 12,7 Gbit/s liksom bland annat hårda block för säkerhetsfunktioner.**



Figur 1. Användarexempel med Live Probes.

signaler och multiplexera dem vilket underlättar starten av avlusningsprocessen när man fortfarande letar efter roten till problemet. Trots de fördelar som Identify ger hämmas alla dessa angreppssätt av kravet på omkompilering vilket påverkar den ursprungliga designen plus att det tar tid och sänker samplingshastigheten.

**DET SOM VERKLIGEN** skulle gagna utvecklarna är inte bara att ha en logikanalysator inuti FPGA:an utan också ett oscilloskop. Det skulle ge ett realtidsfönster som visade hur signalerna egentligen uppför sig. Det skulle vara idealiskt att prova noder i FPGA:an samtidigt som man lägger på olika signaler för att studera vad som händer. Dessutom skulle det vara användbart att kunna skriva och läsa till interna minnen och till testpunkterna i Serdes-transceivarna. Att kunna ha allt det här utan att det påverkar FPGA-designen skulle dramatiskt förenkla avlusningsprocessen.

Ett exempel på det här angreppssättet är SmartDebug, en verktygsuppsättning i Microsemis SoC-mjukvara Libero som används tillsammans med FPGA:erna Polarfire, Smartfusion2 och Igloo2. Verktögen gör det möjligt för konstruktören att avlusa FPGA-logiken, minnesblock och Serdesblock på samma sätt som om den hade ett oscilloskop. Med SmartDebug är det möjligt att använda de testpunkter som finns integrerade i FPGA-logiken vilket avsevärt snabbar upp och förenklar avlusningsprocessen. Det är också möjligt att välja olika testpunkter utan att kompilera om konstruktionen. Funktionen ger tillgång till alla logikelement och konstruktören kan kontrollera tillståndet på in- och utgångar i realtid.

De nyckelfunktioner som kan användas utan att göra om konstruktionen omfattar: Live Probes: Ger två dedicerade prober

som kan konfigureras för att studera valfri nod i FPGA-delen, exempelvis in- och utgångar av logikblocket. Probpunkterna väljs i mjukvaran varefter ett kommando skickas via kabeln till JTAG-anslutningarna och signalerna syns omedelbart på oscilloskopet. Dessa probpunkter är dynamiska och i realtid. Det behövs ingen omkompilering och laddning av FPGA:an och testpunkterna kan ändras efter behov och signalerna syns direkt.

**ACTIVE PROBES:** är en funktion som ger dynamisk och asynkron läsning eller skrivning till en vipa eller testpunkt. Konstruktören kan se register i FPGA-logiken eller minnesblocken. Den här funktionen gör det möjligt för användarna att snabbt observera effekten internt i logiken och experimentera med hur logiken reagerar när man skriver i en testpunkt. Det går att sätta ett valfritt antal signaler till förutbestämda värden och precis som med vanliga probar behövs ingen omkompilering och laddning av FPGA:an. Aktiva probar är speciellt användbart för att stoppa in fel i en konstruktion och sedan studera effekten.

FPGA-konstruktörer spenderar minst 30 procent av tiden med att avlusa sina system. Procentsatsen kan vara ännu högre beroende på storlek och typ av projekt. Till exempel kan det ta veckor att avlusa en medelstor FPGA. Processen är besvärlig eftersom den innehåller så många interaktioner med begränsad visibilitet och styrbarhet, många omkonstruktioner inklusive layout och tidsvillkor som sedan ska laddas i kretsen. Genom att dra nytta av avlusningsverktögen kan utvecklarna validera sina FPGA-konstruktioner mycket snabbare än när de använder den traditionella metoden med en intern logikanalysator. Till exempel kan en Polarfire med 300 000

uppslagstabeller inte bara använda logikanalysatorn utan också de ovan beskrivna oscilloskopfunktionerna. Konstruktören kan därmed se signalerna i realtid och styra tillstånden i olika delar av designen vilket ger snabbare avlusning.

**ETT ANTAL KUNDER** har rapporterat att de lyckats korta sin avlusningstid avsevärt med SmartDebug. Genom att ersätta interna logikanalysatorer med smart avlusning lyckades de identifiera problemen på några timmar, tidigare tog det veckor. En utvecklare fann att felet låg i en helt annan modul än den han studerade med logikanalysatorn. Konstruktionen förbättrades dessutom ytterligare genom att lägga på olika värden via de aktiva probarna och sedan studera att konstruktionen reagerade på korrekt sätt.

Nya avlusningsmöjligheter är en viktig förändring för FPGA-utvecklare. De senaste lösningarna kan korta avlusnings- och valideringstiden och ger helt nya möjligheter att studera och styra en FPGA-design. Konstruktörerna bör överväga avlusningsmöjligheterna innan de väljer sin nästa krets. De kan avsevärt korta utvecklingscyklerna och kostnaden samtidigt som de kortar tiden till marknaden. ■

