

# Snabb utveckling kräver kort, mjukvara & support

*Se till att få ut det mesta av öppen källkod för ditt inbyggda system*



**Av Chris Young, Silica**

Chris Young arbetar i en grupp som hjälper Silicas kunder med alla aspekter på programvaran under alla stadier av utvecklingsarbetet.

Innan han började på Silica hade han en likande roll på Xilinx med fokus på inbyggda processorer.

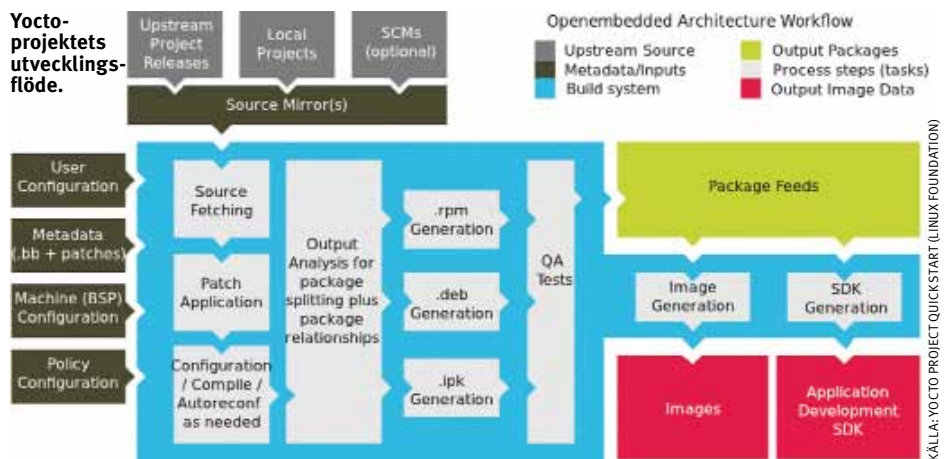


I takt med att inbyggda system blivit allt mer komplexa har nyckelfaktorn för att förutse utvecklingshastigheten hos ett projekt ändrats. Det har blivit betydligt viktigare att ha tillgång till prisvärda, fullt funktionella utvecklingskort som stöds av ett komplett ekosystem med mjukvaror och kvalitativ teknisk support. För tio år sedan var det möjligt för hårdvaruleverantörer att tillhandahålla enkla utvecklingspaket för styrkretsar och inbyggda processorer med endast begränsad tillgång till mjukvarustöd. Vanligtvis var storleken på en kunds applikation inte större än att de kunde utveckla all mjukvara i egen regi.

Idag gör kraven på kommunikation, användargränssnitt och fjärrhantering i moderna tillämpningar det extremt svårt för utvecklare av inbyggda system att implementera en produkt med endast sin egen mjukvaruexpertis. Mycket måste integreras från olika källor, men det skapar i sin tur merarbete eftersom utvecklarna måste lära sig hur mjukvarukomponenterna passar ihop och hur dessa element ska anpassas för att åstadkomma integrationen. Således ställs det krav på plattformar att tillhandahålla nödvändig infrastruktur i en användarvänlig form.

**ANVÄNDANDET AV ÖPPEN** källkod är ett svar på denna utmaning. Det erbjuder en rad möjligheter för utvecklare men kommer verkligen till sin rätt då det kombineras med en strategi som är utvecklad för att tätt koppla mjukvaran till ett brett utbud av hårdvarualternativ. Nyckeln till denna integration är utvecklingen av plattformar med öppen källkod runt operativsystemet Linux.

Linux erbjuder en hög grad av skalbarhet tillsammans med en snabbt växande mjukvaruinfrastruktur som produkttillver-



kare kan utnyttja för utveckling, intrimning och avlusning i syfte att snabbt få ut sina produkter på marknaden. Det finns verktyg baserade på öppen källkod tillgängliga som tillhandahåller stöd inte bara för kodunderhåll och versionshantering utan även för utveckling av grafikgränssnitt för det växande antalet inbyggda system som nu kräver avancerade användargränssnitt med kapacitiva pekskärmar.

Linuxmiljön är dock komplex. Det finns ett stort antal distributioner tillgängliga som inte är fullt kompatibla med varandra även om de erbjuder liknande funktioner. Det gör det svårare att flytta från en plattform till en annan i syfte att dra fördel av andra plattformars användbara funktioner i specifika produkter. Utvecklare av inbyggda system står också inför utmaningen att försöka konstruera om något som från början var avsett för stationära datorer och servrar till potentiellt mer begränsande exekveringsplattformar. Att trimma en distribution så att den exempelvis passar för ett displaylöst system, eller för ett minnesbegränsat system som måste rymmas i ett litet flashminne, kan bli en krävande uppgift.

Yocto-projektet startade 2010 som en lösning på de problem som orsakats av Linux tilltagande fragmentering inom området för inbyggda system. Istället för att tillhandahålla en enda komplett distribution som kan vara svår att trimma för det breda utbud av plattformskrav som är vanliga för inbyggda system så stöder Yocto anpassningen genom en modulär, skickad arkitektur som tagits fram för att minimera bristen på kompatibiliteter mellan olika konfigurationer.

**NYCKELN TILL YOCTOS** konfigurerbarhet finns i verktyget BitBake, ett projektbyggnadsverktyg som använder en uppsättning metadatafiler för att konfigurera inte bara Linuxkärnan utan även systemet och tillhörande applikationsmjukvara till en slutlig image-fil för systemet. BitBake hanterar alla byggsteg baserat på definitioner utvecklaren tillhandahåller, vilka definierar de olika konfigurationsalternativen och beroendeförhållandena. Genom noggrann styrning av metadata och genom att dra fördel av den skickade arkitekturen är det möjligt att flytta från en hårdvaruplattform till en annan med obetydligt mer arbete än ▶

en enkel omkompilering. Olika hårdvaruberoende förändringar, som byte från en nätverksadapter till en annan eller byte till en annan processor, tas om hand av kompilatorn och byggnadsverktyg genom att referera till de relevanta hårdvaruberoende metalagren.

Andra plattformar tvingar ofta utvecklarna att utföra storskaliga omskrivningar av koden till sina inbyggda produkter vid byte av processorer och hårdvaruplattformar. Med en Yoctobaserad infrastruktur kan övergången vara så enkel som att lägga till en kodrad som specificerar relevant BSP (Board Support Package).

Genom att använda applikationsramverk som Qt är det möjligt att utöka flyttbarheten ännu längre. Exempelvis kan en stor del av applikationsutvecklingen utföras på en stationär dator och med minimala kodändringar överförs direkt till ett inbyggt system som kör Qt. Det öppnar för att prototyper av användargränssnittet kan byggas i en stor utsträckning redan innan den slutliga hårdvaran är klar.

Det är dock långt ifrån enkelt att bygga en Yoctobaserad infrastruktur som stöder enkel flyttbarhet och integration. Valmöjligheterna i BitBake är omfattande och Yocto har en brant inlärningskurva för de som vill skapa plattformdefinitioner. Även om en välkonstruerad kombination av BSP:er och metadata kan minimera arbetet med att flytta mellan hårdvaruplattformar är det ofta svårt och tidskrävande att utveckla en arkitektur som håller vad den lovar, även för utvecklare som är väl insatta i de grundläggande alternativen. Yoctomiljön kräver dessutom att en Linuxvärd används för utveckling och korskompilering, vilken kanske inte finns tillgänglig för de som arbetar i en Windowsmiljö.

**ETT ALTERNATIV SOM** Silica erbjuder som en del av sin ArchiTech-strategi är att använda en virtuell maskin baserad på VirtualBox-teknik för att tillhandahålla en förkonfigurerad utvecklingsmiljö för Yocto på en standarddator och på så sätt kan ett stort antal utvecklare av inbyggda system ta del av fördelarna. Ekosystemet inkluderar en uppsättning utvecklingskort som tagits fram i samarbete med leverantörer av processorer, mikrokontrollers, FPGA:er och viktiga periferikretsar för exempelvis analoga in- och utsignaler samt trådlös kommunikation.

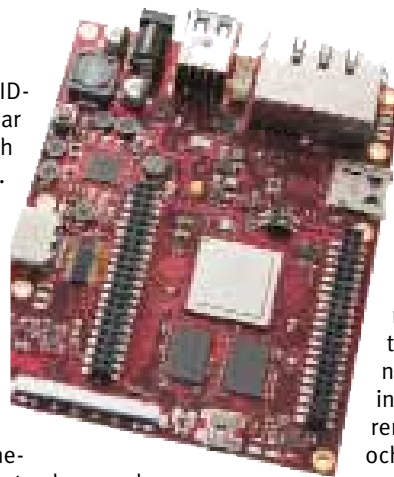
I motsats till de utvecklingskort för mikrokontrollers avsedda för generellt användande som vanligen erbjuds av halvledartillverkare erbjuder detta tillvägagångssätt speciella kort som är välanpassade till specifika applikationer och löser många av de problem runt konstruktion och integration som kunden möter när mjukvaru- och hårdvaruutveckling framskrider mot den färdiga produkten. Projektexempel sträcker

sig från utveckling för RFID- och NFC till tillämpningar för imbygnadsvideo och industriell anslutning. Övriga radiostandarder som stöds inkluderar wifi och IEEE802.15.4.

Yocto tillhandahåller en mycket effektiv mjukvaruinfrastruktur för sådana hårdvaruplattformar och klarar enkelt av plattformsförändringar genom de metalagren som används i Yoctos byggverktyg. Dessutom kan det färdiga systemets imagefil skräddarsys och trimmas med hjälp av Hobverktyget, ett grafiskt användargränssnitt som gör det enklare för användaren att välja vilka Linuxpaket som ska byggas in i målfilen.

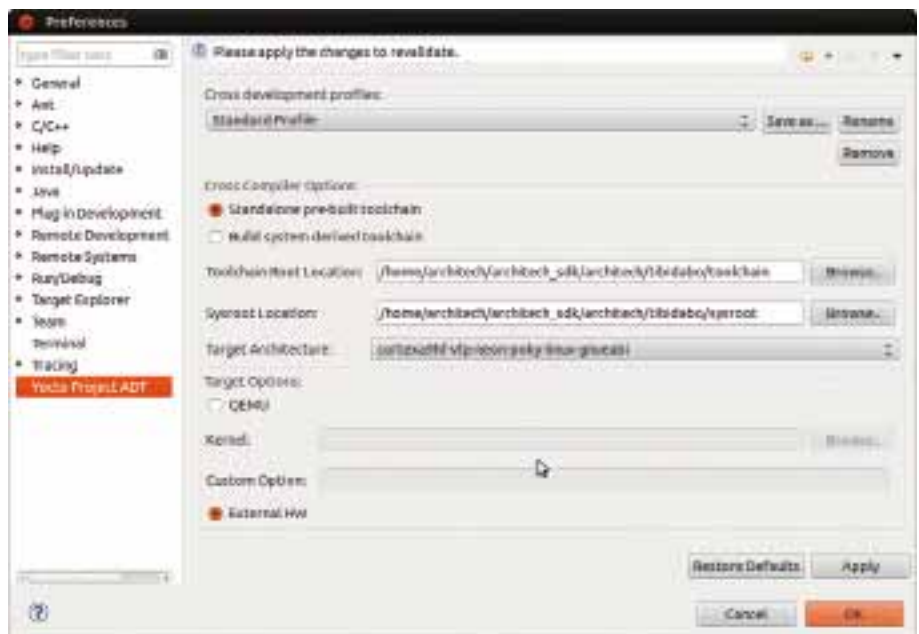
Yocto är dock inte en komplett utvecklingslösning för inbyggda system eftersom det saknar en integrerad utvecklingsmiljö (IDE, Integrated Design Environment). Här behövs en annan öppen källkodslösning, nämligen Eclipse. Eclipse är en fullt grafisk integrerad utvecklingsmiljö baserad på plugin-lösningar som gör det möjligt för utvecklare att bygga en skräddarsydd plattform för mjukvaruutveckling och test. Tillsammans förbättrar Eclipse och Yocto konstruktionseffektiviteten och låter ingenjörerna fokusera på en gemensam samling kärnverktyg istället för att behöva lära sig olika aspekter av Linuxdistributioner runt deras tillhörande verktyg allteftersom deras behov förändras.

**SOM EN OMFATTANDE** utvecklingsmiljö innefattande prisvärda hårdvaruplattformar, verktyg och mjukvaror som kan adressera många olika tillämpningar, kan Silica ge-



**Silicas ArchiTech Tibidokort.**

nom Architech erbjuda en innovativ och högeffektiv modell för att långsiktigt stöda sina kunder. Kombinationen av specialiserad hårdvara utvecklad i samarbete med ledande halvledartillverkare, en känd Linuxdistribution som stöds internt av Silicas ingenjörer, stark teknisk support och omfattande träning för produktutvecklande ingenjörer tar inbyggnadsutveckling till nästa nivå i termer av att snabbt komma igång samt säkerställa ett smidigt och snabbt projektgenomförande. ■



**Yoktoplugin för Eclipseintegrerad utvecklingsomgivning.**

