

Inbyggd grafik utmanar



Det ska vara stort, snabbt och billigt

För mig personligen är bildminnet en av de svåraste detaljerna vid konstruktion av grafik för inbyggda system. Det ska vara stort, snabbt och billigt.

Tyvärr måste man ofta göra kompromisser när man adderar minne till inbyggd grafik. I bästa fall blir kompromisserna enbart dyra irritationsmoment som naggar på lönsamheten. I värsta fall leder de till att konstruktionen måste läggas ut eller att någon ny talang måste anställas för att ta hand om det hela.

Den här artikeln diskuterar de överväganden som man kan komma att behöva göra vid integration av stora avancerade minnen i grafiktillämpningar som körs på styrkretsar (MCU). Likaså diskuteras här hur man minimerar, och till och med tar bort, negativa effekter som dessa minnen kan orsaka.

Det finns många fördelar med att använda en MCU istället för en MPU (mikroprocessor) för inbyggd grafik. En MPU är definitivt nödvändig när det grafiska gränssnittet (GUI) är avancerat över en viss nivå, men många tillämpningar klarar sig bra utan den extra kostnad och komplexitet som en MPU medför.

Den största fördelen med standardiserade MCU:er är kanske den höga integrationen – de erbjuder mängder av storlekar på både SRAM och flashminne, specifik kärn- och klockhastighet, kommunikationsgränssnitt, IO-portar samt analoga periferenheter. På marknaden finns dessutom ett i det närmaste oändligt utbud av enheter som passar de flesta behov.

När en inbyggd tillämpning kräver ett GUI kan det tyvärr bli svårt att få till en enkel, liten och billig lösning med en MCU. I det läget är det många konstruktörer som frågar sig om en styrkrets kan tillfredsställa deras behov eller om de borde ta steget till en dyrare och mer komplicerad mikroprocessor.

Det första övervägandet: "Hur kommer grafiken att drivas?"

I allmänhet kräver inbyggd grafik tre funktioner: rendering, drivning och lagring.



Av Kurt Parker, Microchip

Kurt Parker ansvarar för produktmarknadsföring på Microchips affärsenhet MCU32. Det betyder att han ansvarar för att definiera, hantera och marknadsföra företagets styrkretsfamiljer PIC och SAM samt relaterade utvecklingsverktyg. Tidigare har han arbetat på bland annat Intel, Netronome och Wyle Electronics. Han har en kandidatexamen i elektroteknik och en magister i systemteknik.



Figur 1. Till vänster syns en demo av nästan fotorealistisk bilddistorsion som kräver ett icke-flyktigt minne på runt 12 Mbyte. Till höger, en kaffebryggare, som inte har lika krävande grafik men ändå nyttjar cirka 3 Mbyte.

Rendering handlar om hur bilden skapas och manipuleras. Enklare konstruktioner kan använda styrenhetens CPU för detta. Avancerade specialanpassade MCU:er har en dedikerad grafik kärna, en GPU, som tar hand om renderingsfunktioner, såsom att rita linjer, fylla rektanglar, förflytta bildobjekt och applicera bitoperatorer (så kallade blitter).

Drivning syftar på hur det genererade bildinnehållet flyttas till skärmen. Det kan ske med en DMA-enhet (Direct Memory Access), över en extern parallellport hos styrkretsen eller med en skräddarsydd grafikcontroller. Den sistnämnda lägger till funktioner som överlagring och rotation, vilket skapar möjlighet för ett bättre slutresultat.

Lagring syftar på platsen där informationen om vad som ska visas är sparad. Det är denna del som resten av denna artikel är inriktad på.

Det andra övervägandet: "Var ska du lagra din GUI-design?"

Idag är det tillgängliga integrerade SRAM:et hos de flesta avancerade MCU:er omkring 512 Kbyte.

Det kan vara tillräckligt för att köra enkla statiska GUI:er som enbart kräver en bildbuffert, eller GUI:er som bara använder åtta bitar per färgpixel och har en liten skärm.

Trenden är dock att slutanvändarna vill ha lika bra grafik i sitt inbyggda system som de har i favoritappen på sin smartmobil.

Likaså vill företag att det grafiska gränssnittet ska representera deras varumärke korrekt, och skapa varumärkesidentitet

och lojalitet. Att driva ett sådant GUI kan kräva flera bildbuffertar, flera överlappande lager och större färgdjup. Det senare är särskilt viktigt om grafiken ska vara fotorealistisk eller exakt matcha en viss varumärkesfärg (se figur 1)

Det tredje övervägandet: "Bör du lagra ditt GUI i ett externt minne?"

Först vill jag påminna om att ett integrerat SRAM i en typisk high-end-MCU toppar på cirka 512 Kbyte. De två exemplen i figur 1 kräver betydligt mer minne än vad de flesta MCU:er på marknaden erbjuder. Lösningen är ett externt minne med hög densitet, hög prestanda och hög tillgänglighet.

Asynkrona SRAM är då ett alternativ. De ger en minnesboost på 8 Mbyte och är relativt lätt att konstruera in, då de har icke multiplexade adresser och anslutningar som passar bra med många styrkretsar. Nackdelen är densiteten (8 Mbyte är mycket men inte tillräckligt mycket för många grafikintensiva tillämpningar), kostnad (ofta dyrare än själva MCU:n) samt den extra yta minnet tar upp på kortet.

Många styrkretsar har gränssnitt för SDRAM som kan användas för att lagra bilder. Mest prisvärda är SDRAM mellan 8 Mbyte och 16 Mbyte. De är relativt lätta att få tag på och betydligt mer kostnadseffektiva än externa SRAM.

Som redan nämnts är 8 Mbyte en undre gräns för vad GUI-tillämpningar behöver, och det är ibland ändå för lite. Likaså måste man ta hänsyn till hur kortet utformas vid konstruktion med SDRAM – med bussar

minnet

som når 120 MHz gäller speciella konstruktionskrav. För vissa tillämpningar rekommenderas att kretskort med SDRAM ska ha sex lager. Det betyder att ett sådant minne kan kräva upp till fyra extra kretskortslager, vilket adderar många kronor till systemets inköpslista (BOM).

SDRAM kan också ha problem att nå tillräcklig prestanda. Den maximala teoretiska dataakten är 200 Mbyte/s på en typisk 16-bitars buss i 100 MHz.

En 800 x 480 WVGA-skärm som uppdateras i 60 Hz och har ett färgdjup på 16 bitar per pixel kräver att data strömmas med 46 Mbyte/s. Tar man dessutom hänsyn till bildbehandlingen som CPU eller GPU sköter, och adderar stöd för överlagrad bild – vilket har blivit norm – så hamnar konstruktionen nära eller överstiger vad ett SDRAM-baserat system klarar. Och med tanke på tidigare nämnda förväntningar hos användarna så kan kraven inte annat än att bli ännu tuffare framöver.

Vill man garantera slutanvändarna samma upplevelse som i smartmobilen krävs nyare minnesteknik som ger högre prestanda och mer lagringsutrymme.

Det fjärde övervägandet: "Finns det internminne för GUI-tillämpningar?"

Klart högre densitet – upp till 128 Mbyte – erbjuder DDR2 SDRAM vars gränssnitt dessutom klockas minst dubbelt så högt som SDRAM.

DDR är kort för Double Data Rate. Det betyder att data överförs till eller från minnet två gånger för varje klockcykel. Det ger ett minne som är minst fyra gånger snabbare än den SDRAM som finns på marknaden idag.

Ett av de stora problemen med att använ-

da DDR2-minnen är att kunna dra nytta av denna prestanda.

I och med att bussgränssnittet ligger på minst 200MHz och dataöverföringen sker varje halvcykel så ställs helt andra krav än för SDRAM för att samtidigt kunna säkerställa korrekt signalintegritet och isolera övriga kortet.

Vid konstruktion med DDR måste man också vara uppmärksam på andra detaljer såsom tajta referensförhållanden, olika spänningar och lämplig avkoppling. Liksom layoutnära aspekter som ledningsbredd, avstånd mellan ledningar samt ledningsdragning.

Samtidigt kan man konstatera att när inbyggda grafiktillämpningar blir större, djupare och mer komplexa så växer behovet av DDR2-minnesstorlek och snabb dataöverföring.

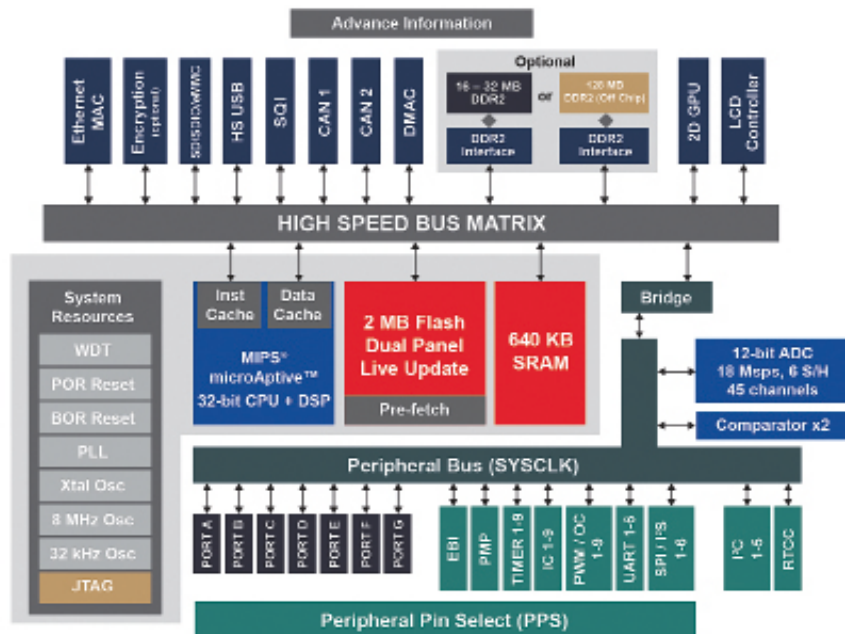
Konstruktörer som för bara några år sedan använde MCU:er på 150 MHz behöver

nu dubbelt så snabba styrkretsar med minnesgränssnitt som matchar den interna klockhastigheten. De flesta kretstillverkare erbjuder också olika former av konstruktionshjälp.

Men visst vore det trevligt med en smidigare lösning som ger hög prestanda utan att addera de ovan nämnda utmaningar som följer med en DDR2-konstruktion? Liksom att slippa den högre kostnaden i form av extra kretskortslager?

Microchips PIC32MZ DA är en av de få MCU:er på marknaden med gränssnitt till DDR2-minne. Styrkretsfamiljen integrerar 32 Mbyte DDR2 DRAM på chipet med hjälp av staplad minnesteknik. Således krävs inget externt minne (se figur 2). Chipet inkluderar även en grafikcontroller i tre lager och en högpresterande GPU.

Resultatet är en styrkretsfamilj för grafik med en överträffad integrationsgrad och prestanda. ■



Figur 2. Blockdiagram över styrkretsfamiljen PIC32MZ DA.