TI Rediscovers CISC With ’C2700 DSP

New DSP Adopts Decades-Old Techniques to Appeal to Microprocessor Users

by Jim Turley

In developing its newest digital-signal-processing architecture, DSP behemoth Texas Instruments has turned over a new leaf—backwards. The new ’C2700 DSP family borrows from CISC microcontrollers to create a new kind of hybrid DSP/MCU that makers of small embedded systems will find interesting and useful. TI is taking particular aim at disk-drive manufacturers, which frequently use separate microcontrollers and DSP chips in very high volume products.

The ’C2700 family’s biggest contribution to TI’s already well-stocked pool of DSP architectures is support for a relatively orthogonal register set, instructions that modify memory directly, and C language programming. Each of these features is calculated to appeal to users and programmers of conventional 16-bit microprocessors and microcontrollers. TI hopes to herd this large community into the DSP fold.

TI is not planning any standard parts in the ’C2700 family. Instead, the core will be used in the company’s custom DSP (cDSP) line of ASICs. The first of these chips is expected to sample in 3Q98.

Internal Architecture Looks Like Most DSPs

At its heart, TI’s ’C2700 architecture is like most other 16-bit fixed-point DSPs. It bears many similarities to TI’s existing ’C2xx family of DSP chips, with which the ’C2700 shares some passing software compatibility. As Figure 1 shows, the ’C2700 core maintains separate program and data buses, as well as separate read and write ports (Harvard architecture). These are traditional features for a DSP.

The core is architecturally limited to two 22-bit (8M) address spaces. Separate address spaces for code and data are provided. The first 2K of code space is duplicated in the data space so programs may access embedded constants (or implement self-modifying code).

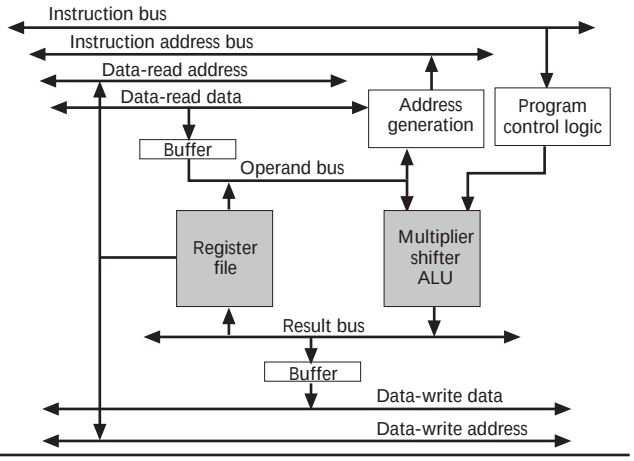

Figure 1. Like most DSPs, TI’s ’C2700 core separates the program, data, and address buses.

Registers, Instructions Appeal to CPU Users

TI tried hard to hide the ’C2700’s DSP origins, dressing it up as a 16-bit microprocessor with a slightly peculiar register and instruction set. As Figure 2 illustrates, the programmers’ model for the ’C2700 looks similar to that of many 16-bit microprocessors and microcontrollers. The register mnemonics may be unusual, but their function is familiar. In particular, the upper and lower halves of the accumulator, AH and AL, will jog many fond memories.

In true DSP fashion, most instructions operate on specific registers, particularly the accumulator. The ’C2700 does not, however, maintain explicit X and Y data pointers as most DSPs do. In this regard, it leans more toward a CPU-style programming model, with operands directly addressable through registers or in memory.

Rather than follow the load/store model of other DSPs and all RISC processors, the ’C2700 can add, subtract, and otherwise modify operands in memory in one indivisible operation. This CISC-like attitude toward memory is the least DSP-like aspect of the architecture.

All the instructions one might reasonably expect can be used with memory-based operands. Arithmetic and logical operations like ADD, SUB, AND, OR, and XOR all work with

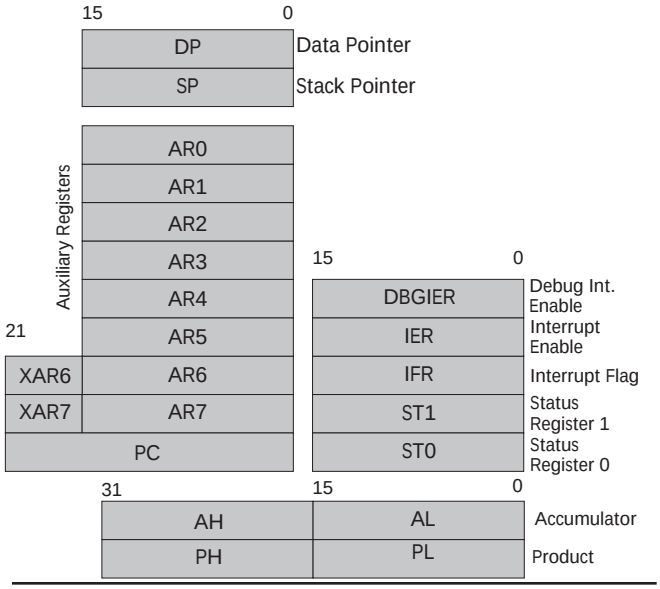

Figure 2. The ‘C2700 register set includes an accumulator and other special registers, which can be addressed in a regular fashion.

Mnemonic	Description	▲	■	●
Arithmetic				
ABS	Absolute value	▲		
ADD	Add	▲		●
ADDB	Add immediate	▲		
ADDCU	Add acc w/carry	▲		●
ADDL	Add accumulator long	▲		●
ADDU	Add acc, unsigned	▲		●
ADRK	Add to auxiliary register			
SBBU	Subtract imm w/borrow	▲		●
SBRK	Subtract auxiliary register			
SUB	Subtract	▲	■	●
SUBB	Subtract with borrow	▲		
SUBCU	Subtract conditionally	▲		●
SUBL	Subtract long	▲		
SUBU	Subtract unsigned	▲		●
MPY	Multiply	▲	■	●
MPYA	Multiply accumulate		■	●
MPYB	MPYA, immediate	▲	■	
MPYS	Multiply subtract		■	●
MPYU	Multiply, unsigned	▲	■	●
MPYXU	Multiply signed/unsigned	▲	■	●
MAC	Multiply accumulate		■	●
NEG	Negate accumulator	▲		
INC	Increment			●
DEC	Decrement			●
SXTB	Sign-extend accumulator	▲		
SAT	Saturate accumulator	▲		
NORM	Normalize accumulator	▲		
▲	Works with accumulator			
■	Works with P register			
●	Works with memory			

Mnemonic	Description	▲	■	●
Logical				
AND	Logical AND			●
ANDB	Logical AND, immediate	▲		
OR	Logical OR	▲		●
ORB	Logical OR immediate	▲		
XOR	Logical exclusive-OR	▲		●
XORB	Logical XOR immediate	▲		
NOT	Invert value	▲		
ASR	Shift right, arithmetic	▲		
LSL	Shift left logical	▲		
LSR	Shift right logical	▲		
ROL	Rotate accumulator left	▲		
ROR	Rotate accumulator right	▲		
SFR	Shift accumulator right	▲		
SPM	Set shift mode			
CLRC	Clear status bit			
CMP	Compare	▲		
CMPB	Compare immediate	▲		
CMPL	Compare long	▲		
SETC	Set status bit			
TBIT	Test bit			●
TEST	Test accumulator with zero	▲		
System				
PUSH	Push onto stack			●
POP	Pop from stack		■	●
PREAD	Read program space			●
PWRITE	Write program space			●
ASP	Align stack pointer			
NASP	Unalign stack pointer			
IDLE	Wait for interrupt			
NOP	No operation			

Mnemonic	Description	▲	■	●
Transfer				
MOV	Move	▲	■	●
MOVA	Move and accumulate			●
MOVB	Move immediate	▲		●
MOVH	Move upper byte	▲		●
MOVL	Move long	▲		●
MOVP	Load, move to acc			●
MOVS	Load, subtract acc			●
MOVU	Move w/ zero-extend	▲		●
MOVW	Load data pointer			
Miscellaneous				
EALLOW	Enable emulation			
EDIS	Disable emulation			
ESTOPn	Emulator breakpoint			
ABORTI	Abort interrupt			
TRAPn	Software trap [0...15]			
ITRAPn	Instruction trap [0...1]			
IACK	Interrupt acknowledge			
INTR	Software interrupt			
IRET	Interrupt return			
Flow Control				
B	Branch, conditional			
SB/LB	Branch short/long			
BANZ	Branch if aux not zero			
CALL	Call subroutine			
LOOPZ	Loop if zero			●
LOOPNZ	Loop if not zero			●
FFC	Fast function call			
RET	Return from subroutine			
RETE	Return, enable interrupts			
RPT	Repeat next instruction			

Table 1. TI's 'C2700 DSP architecture, unlike most DSPs, can modify operands directly in memory like a CISC microprocessor. Most instructions, however, work like a conventional DSP with reliance on a few accumulator registers.

either register or memory contents; the chip can also perform multiply-accumulation (using MAC, MPYA, MPYB, MPYS, and MPYU) using operands in memory, although the product register (P) still accumulates the results.

Interlock Avoids Inconsistent Data

Each of these instructions executes in a single cycle whether the operands are in memory or in registers, because the read, write, and ALU operations occupy separate stages of the pipeline. Slow memory will obviously delay the operand-fetch stage; it will also stall the entire 'C2700 pipeline while it waits for data. TI doesn't expect slow memory will be a problem in real systems, because the company's product plans call for all 'C2700 devices to include copious on-chip memory.

The only other stall potential comes from consecutive accesses to the same memory address. If a pipelined read access follows one cycle behind a read-modify-write access to the same address, the operand will be read twice before it's written, updating stale data. RISC chips are theoretically prone to the same failure but include pipeline interlocks to prevent such an occurrence. The TI 'C2700 does too, catching such conflicts in hardware.

Although the memory-reference instructions could potentially take more time, they require less code space. For example, a simple arithmetic or logical operation on a value in memory can be done in one 16-bit instruction, versus three instructions (load, modify, and store) in a conventional load/store DSP design. The total execution time may be the same, but the code density is greatly increased. For disk-drive makers, in particular, better code density is important, as it reduces the cost of associated memory.

Single Instructions Taken From CPU Realm

As Table 1 shows, the 'C2700 instruction set is rich with arithmetic operations for both register- and memory-based operands. Loop constructs are supported with the LOOPZ/LOOPNZ pair and RPT (repeat next instruction). Unusual is the TBIT (test bit) instruction, which examines single bits located in memory, a very microcontroller-like operation.

The FFC (fast function call) instruction provides an interesting alternative to standard subroutine calls. Rather than pushing the return address onto the stack, FFC simply copies it into auxiliary register XAR7 and jumps to the specified address. An LB (long branch) instruction returns to the

Price & Availability

Texas Instruments expects the first custom part in the 'C2700 family to sample in 3Q98. Clock speeds will be around 100 MHz, and prices are expected to be below $10 in 10,000-piece quantities.

For more information contact TI (Dallas, Texas) at 800.477.8924 ext. 4094 or visit *www.ti.com/sc/c2700.*

calling routine. In addition to being faster, FFC eliminates two doubleword references to memory.

Software Tools Part of the Appeal

To strengthen the 'C2700's appeal to programmers who are used to microprocessors, TI expended a lot of effort in creating familiar software-development tools. As it did with its 'C6xx VLIW architecture (see MPR 2/17/97, p. 14), TI is offering a C compiler for the DSP. The company expects most 'C2700 development will be done using C rather than assembly language.

The 'C2700 is touted as being software-compatible with TI's existing line of low-end DSP chips in the 'C2xx family. Software compatibility, such as it is, is only at the mnemonic level. Assembly-language source code for a 'C2xx can be reused, but the 'C2700's software tools will assemble the mnemonics into different opcodes. There is no binary compatibility between the 'C2xx and the 'C2700 families, although the latter does implement a superset of the instructions of the former.

Again bowing to the needs of its audience, TI aliased many DSP mnemonics with forms more familiar to CPU programmers. For example, the LACL (load memory to accumulator) instruction can also be written as MOV AL, mem.

Chips Expected in 3Q98

TI has working silicon of the 'C2700 core, built in a 0.35-micron process, running at 75 MHz. General sampling of the first 'C2700 part, which has not been announced, is scheduled to begin in 3Q98. Production-level parts will all be built using TI's 0.25-micron process, with clock speeds expected to be in the 100-MHz range. By late 1999, TI plans to move some 'C2700 production to its 0.18-micron line. These are all very aggressive plans for a brand-new product line and strong evidence that TI expects the 'C2700 family to become one of its major volume drivers in the coming years.

Disk-drive makers, which are the 'C2700's primary audience, will appreciate the lower power consumption the new process will give them. The higher clock speeds will also enable more exotic, enhanced PRML algorithms for a new generation of disk drives, potentially with microactuators on the heads. TI has also hinted that upcoming 'C2700 devices will manage an IEEE-1394 (FireWire) interface, expected become more popular on disk drives within a few years.

A Microcontroller for DSP Fans

TI's bold claim of "rendering microcontrollers obsolete" is perhaps premature, but the 'C2700 is an interesting new entry in the hybrid CPU/DSP race nonetheless. The company has correctly identified the demand for such a product and has responded as one might expect, given the company's background. There's no question that the 'C2700 is a DSP, but TI has done more than just alter the mnemonics and wave its hands to make it into a microcontroller. The chip's approach to memory references and register addressing make it a distinctly different beast to program.

TI is not, however, the first company to take this route. Siemens also rediscovered some retro-CISC techniques in creating TriCore (see MPR 11/17/97, p. 13). TriCore looks far more like a CPU than a DSP, though. It includes some bit-insert and -extract functions, but only on register contents. In fact, TriCore sticks with the load/store model even as it tries to be a microcontroller. TriCore also handles Q-format (pseudo-FP) and SIMD (packed) data formats, something the 'C2700 wasn't designed for. Neither architecture is very good at random bit manipulation, something microcontrollers are often called upon to perform. In the end, TriCore is better equipped for image manipulation, simple audio processing, and security/encryption work than the 'C2700, which excels at traditional DSP tasks, including signal extraction and motor control.

Neither company has begun general sampling of parts yet, but both should appear at nearly the same time, later this year. Siemens had made plain its plan to include embedded DRAM in its parts—TriCore was designed around its availability—but TI has been a bit more coy. New 'C2700 parts with SRAM, ROM, and flash memory seem certain. Embedded DRAM would be a logical addition as well.

Siemens's new Carmel DSP architecture, which is also intended for ASIC designs, is more DSP-like than TriCore and a bolder move into TI's turf. Just as TI discovers the advantages of microcontrollers, Siemens is encroaching into DSP territory. Unlike TI, Siemens has already licensed its DSP to outside companies.

There now are about half a dozen different CPU/DSP product lines to choose from. Hitachi, ARM, Hyperstone, Siemens, and Motorola have all weighed in with designs that lean predominantly one way or the other. Some maintain two instruction sets (or a single instruction set with extensions), while others have just one, like the 'C2700. TI's entry stirs the pot even more, especially given the company's reputation in signal processing.

Now that TI has taken its first step into the microprocessor/microcontroller arena, a new band of customers will follow, crossing the line under TI's tutelage. While that's good news for the market as a whole, because it leads to tighter integration between CPUs and DSPs, competing vendors may find these customers tough to win over. For potential customers with a DSP background, competitors will find that TI has the upper hand. M