

Jump-Start Your Industrial Drives and Servo-Control Development

Single-chip microcontrollers geared to servo and other industrial/EV control applications help engineers cost-effectively design multiple-axis systems.

As manufacturing industry continues to replace manual processes with automated versions, servo motors are replacing pneumatic and hydraulic systems (Fig. 1). It's resulted in a rapid increase in, and greater demand for, precision motion-control designs.

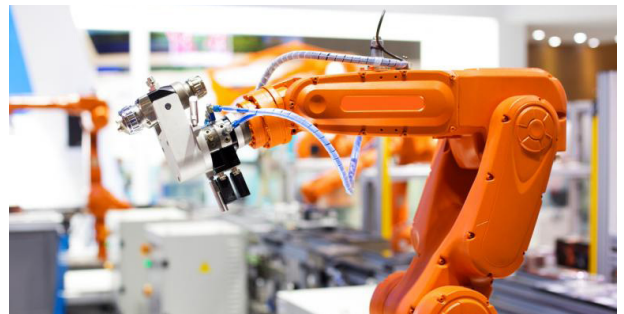
Motor-control and servo-drive advances enabled by servo-drive and servo-motor hardware enhancements, such as the transition of inverter switches away from insulated gate bipolar transistors (IGBTs) toward higher-speed gallium-nitride (GaN) and higher-power silicon-carbide (SiC) technology, remain essential for competing in the global economy.

In this fast-moving market, servo-drive manufacturers are always looking for improvements in performance, connectivity, ease of use, and cost. Some applications will always require the highest performance and highest precision. However, given the progress already made in digital drive control performance, the greater challenge today is reducing the cost of standard motion performance applications, rather than the maximum attainable control bandwidth for speed and position loops.

Most industrial automation machines have multiple motor axes, so the cost per axis is a suitable measure. Since some drive electronics functions—and their costs—can be shared across multiple axes, it makes sense to consider servo drives that can support two or more axes instead of the traditional one-drive/one-motor approach.

COTS Motor-Drive Solutions Lower Costs, Simplify Design

When designers began designing servo-motor controllers based on digital technology, they found that the available off-the-shelf microcontrollers (MCUs) couldn't support some of the specialized requirements, such as intricate protection



1. Precision servo drives are ubiquitous on the factory floor. (Source: "Delivering the next generation of low-cost, dual-axis servo drives," Electronic Design).

schemes for the power drivers, complex encoder interfaces, or current loops that were too fast for a software implementation. Consequently, designers added external logic devices to handle these functions.

Options include a field-programmable gate array (FPGA) or a complex programmable logic device (CPLD). However, these devices, while certainly fast, are expensive and notoriously hard to program because they require an intimate knowledge of the underlying hardware structure and a dedicated set of development tools. Programming the FPGA or making updates adds considerable cost and time to the development cycle. For more information, consult this [detailed discussion](#) of SoC-vs.-FPGA design tradeoffs.

As digital control became the preferred approach, MCU manufacturers began introducing system-on-chip (SoC) products that included the standard motor-control building functions, flexible peripherals that allowed for modifications, plus the ability to make changes using the standard

TMS320F28004x		Temperatures	125C	Q100
Sensing		Processing		
ADC1: 12-bit, 3.45 MSPS, 8ch	Configurable Logic Block 4 Tiles Position Manager: Flexible Absolute Encoder Interfaces System Modules 3x 32-bit CPU Timers NMI Watchdog Timer 192 Interrupt PIE	C28x™ DSP core 100 MHz	Actuation	
ADC2: 12-bit, 3.45 MSPS, 8ch		FPU	8x ePWM Modules 16x Outputs (16x High-Res)	
ADC3: 12-bit, 3.45 MSPS, 8ch		TMU	Fault Trip Zones	
7x Windowed Comparators Subsystem w/Integrated 12-bit DAC		VCU-I	2x 12-bit DAC	
7x PGAs		Connectivity		
4x Sigma Delta Channels (2x Filters per channel)		CLA core 100 MHz	2x UART, 1x LIN/UART	
Temperature Sensor		Floating-Point Math	2x I2C (1x true PMBus)	
2x eQEP		6ch DMA	2x SPI	
7x eCAP (2x HRCAP)		Memory		
		Up to 256 KB Flash (dual-bank) +ECC	Power & Clocking	
		Up to 100 KB SRAM +parity	2x 10 MHz 0-pin OSC	
		2x 128-bit Security Zones	1.2V VREG	
		Boot ROM	POR/BOR Protection	
		InstaSPIN™ Motor ROM	Debug	
		cJTAG/Real-time JTAG		
		Embedded Real-time Analysis And Diagnostic unit (ERAD)		

2. The Piccolo F28004x contains many specialized blocks for motor control. (Source: TI Microcontrollers: C2000 Piccolo 32-bit microcontrollers)

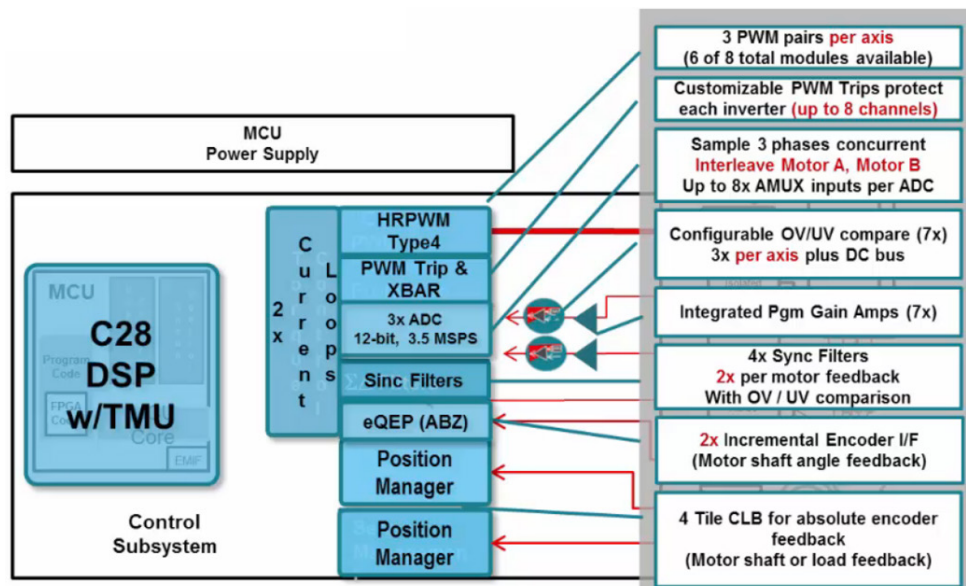
development environment and programming languages such as C.

Switching to a single dedicated device reduces bill-of-materials (BOM) costs, cuts development time, and provides a single point of contact to help with tricky design issues.

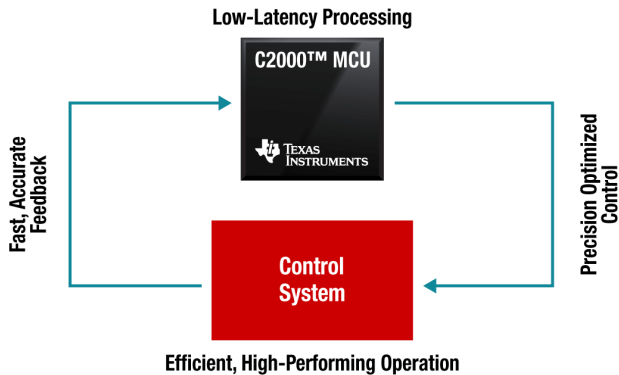
C2000: An MCU Family Designed for Real-Time Control

Texas Instruments' C2000 is a family of high-performance 32-bit MCUs purpose-built for real-time control in demanding applications such as electric vehicles, industrial drives, and digitally controlled power conversion.

The family has two product lines. The [Piccolo](#) line contains six devices and offers cost-effective performance with high levels of integration for a broad set of closed-loop control



3. The F28004x contains several specialized blocks for servo-drive applications. (Source: TI Training Video: "Dual Axis Servo Control on Piccolo F280049C")



4. A real-time closed-loop control system must minimize the time taken to move data from one stage to the next. (Source: TI E2E Blog: “Cycle scavenging on C2000 MCUs, part 1: introduction”)

applications. The [Delfino](#) line of four devices offers premium performance and expanded feature sets for the most advanced closed-loop control applications.

Piccolo F20004x Two-Axis Servo Controller

Although designers are always pushing the envelope in search of ever-increasing levels of performance, a more germane challenge for many applications is how to implement multi-axis servo motor control in the most cost-effective manner.

The [F28004x](#) series of MCUs (*Fig. 2*) are the newest members of the Piccolo portfolio; these devices incorporate many of the features of the high-end Delfino F2837x floating-point performance MCUs plus added features for ac drives and servo applications. The result is a dual-axis servo drive controller on a single chip that reduces BOM cost while delivering a high level of motion-control performance.

The F28004x is based on TI’s 32-bit C28x CPU running at 100 MHz, plus specialized accelerators for several algorithms that are common in real-time motor control.

For instance, the trigonometric math unit (TMU) enables fast execution of algorithms with trigonometric operations such as transforms and torque loop calculations. The Viterbi/Complex Math Unit (VCU-I) speeds complex math operations commonly found in encoded applications. And the programmable control-law accelerator (CLA), an independent 32-bit floating-point math accelerator that executes in parallel with the main C28x CPU, allows significant offloading of common tasks.

Figure 3 shows some of the specialized Piccolo F28004x peripheral blocks dedicated to real-time servo control. These include:

- Eight high-resolution pulse-width modulators (HRPWMs); type 4 with immediate PWM update mode
- Highly configurable PWM trip logic for maximum

protection options

- Three high-performance 12-bit ADCs operating at 3.5 Msamples/s with configurable start-of-conversion triggering for concurrent operation
 - A configurable logic block (CLB) that can work with other peripheral blocks to implement complex functions without requiring external logic. The CLB can perform many functions, such as inverter protection circuits; PWM generation; pulse train output (PTO) generation; and absolute encoder protocol master logic.
 - Two enhanced quadrature encoder peripheral (eQEP) blocks with error detection logic
 - Four integrated sigma-delta filters
 - Seven programmable-gain amplifiers (PGAs)
- See the [C2000 Real-Time Control Peripherals Reference Guide](#) for more information on these peripherals.

Cycle Scavenging Minimizes Delay Between Functions

C2000 microcontrollers include several architectural improvements to provide enhanced motor-control performance. At a high level, a real-time control system can be viewed as three blocks (*Fig. 4*):

- The **sensing stage** acquires data that characterizes the operation of the power system—drive voltage and current, motor position, etc.—with measurements from active or passive sensors.
- The **processing stage** calculates required changes to the control output based on the desired control input and measurements from the sensor stage.
- The **actuation stage** takes the output from the processing stage and calculates the drive signal for the power electronics.

One of the big challenges faced by engineers when designing real-time control systems is minimizing the time delay between these three stages to meet the timing demands of the system. The C2000 architecture incorporates the concept of “cycle scavenging.” This minimizes the sample-to-output latency to help meet real-time control-loop performance requirements.

The central processing unit (CPU), peripherals, and device architecture minimize the CPU cycles needed to transition from the sample acquisition phase, to processing the data in the CPU, to updating the pulse-width modulation (PWM) calculations. Some examples of cycle-scavenging techniques include:

Sample acquisition stage: Zero-wait-state analog-to-digital converter (ADC) transfers allow the results to transfer to the CPU in a single cycle as soon as they’re available. There are dedicated data taps for each master, so multiple masters can read ADC outputs simultaneously without having to waiting for arbitration by a data bridge.

CPU processing stage: C2000 CPUs integrate hardware accelerators for fast execution of complex math operations. The TMU, for example, reduces the CPU cycles required to

calculate sine, cosine, arctangent, and 1/X functions. The TMU also speeds common real-time control algorithms such as Park and inverse Park transforms, fast Fourier transforms (FFTs), and others. The cycle savings can be dramatic: An unassisted MCU can easily take more than 100 cycles to execute a Park transform; with a TMU, the same transform can occur in just 13 cycles.

PWM updating: C2000 microcontrollers integrate a hardware block that performs valley-switching without requiring computationally intensive software control.

Find out more details on C2000 cycle-scavenging techniques in this multi-part [blog series](#).

DesignDRIVE—the Other Piece of the Puzzle

A high-performance servo drive is more than a collection of integrated circuits. Real-world requirements include building in the ability to handle different motor topologies, accommodating a range of encoder types, meeting the new functional safety regulations, and more.

Texas Instruments offers DesignDRIVE, an integrated set of hardware and software resources for the C2000 family of MCUs that help engineers get started quickly with their industrial drive applications.

DesignDRIVE offers support for a wide variety of motor types, encoder standards, sensing technologies, and communications networks, as well as easy expansion to develop with real-time Ethernet communications and functional safety topologies. The result is a more comprehensive, integrated system solution.

There's also the associated [DesignDRIVE Development Kit](#), a reference design for servo and ac inverter drives.

Practical Example: Dual-Axis Servo

Let's see how these features help in a dual-axis servo design that uses the F28004x and DesignDRIVE modules. *Figure 5* shows the key elements.

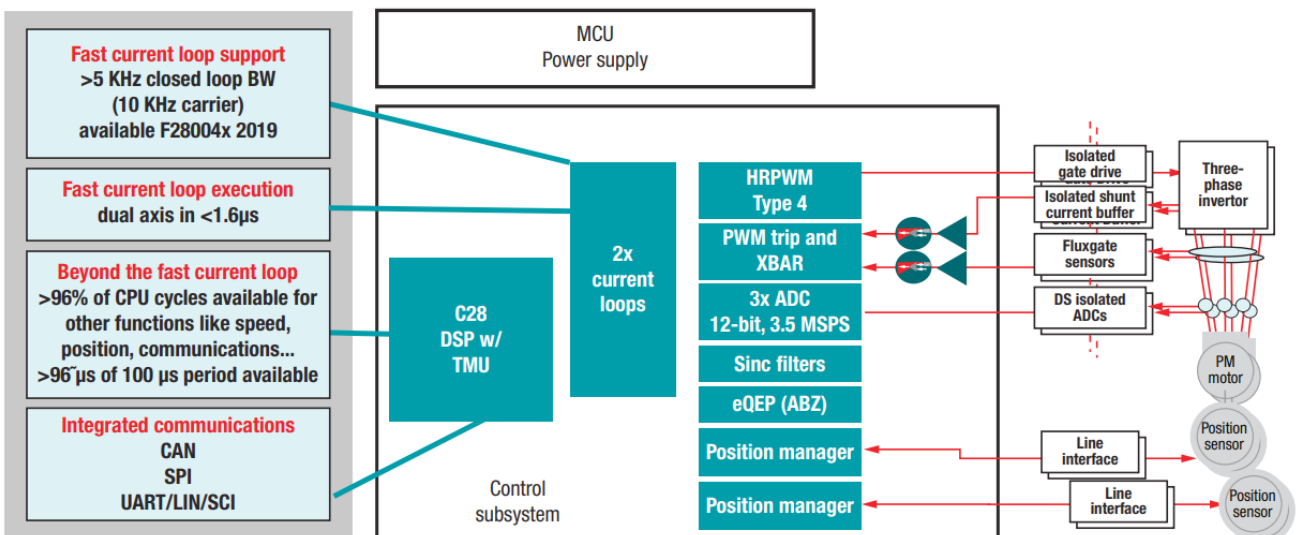
H-bridge control block: A three-phase inverter in a permanent-magnet synchronous motor or an ac induction motor power block requires two switches in a half-bridge configuration on each phase.

Each ePWM module on the F28004x contains a pair of complementary PWM-generating timers and can control both the high- and low-side of an H-bridge, with an integrated dead-band to avoid shoot-through.

Only six of the eight PWM modules are required for the inverters in the dual-channel servo design. The F28004x still has two PWM pairs available for additional functions, such as controlling a power-factor-correction (PFC) circuit to supply high-voltage dc power for the inverters; generating precision clocks for the encoder interfaces; or implementing real-time braking circuits.

Control loop: The F28004x features TI's fast-current-loop (FCL) architecture that's optimized to take advantage of the C2000 architecture. FCL first appeared on Delfino. Running on the F28004x, the time taken for the field-oriented-control (FOC) processing will double, but the 12-bit ADC conversion time doesn't change. As a result, on a 100-MHz Piccolo, the FCL can complete in less than 1.6 μ s for a single axis. This is nearly a full microsecond faster than traditional current loops on the fastest microcontrollers.

By interleaving the ADC sampling between each axis, the sampling and FOC processing for each axis can run concurrently, allowing for a closed-loop bandwidth of 5 kHz. The FCL consumes less than 4% of a 100- μ s loop, leaving the



5. Here, the F28004x is implemented in a dual-axis servo control application. (Source: TI "Piccolo C2000 MCUs enable the next generation of low-cost, dual-axis servo drives" PDF)

rest available for other tasks: the speed and position loops, communication, etc.

Current and voltage sampling: The F28004x has three ADCs with between four and nine multiplexed analog inputs per ADC, depending on the pinout of the device selected. There are up to 21 total multiplexed input channels. This configuration provides flexibility to support the interleaved three-phase sampling of at least four different sets of current and/or voltage sensing, on even the smallest F28004x device. Additional analog inputs are available to monitor dc bus voltage, temperature, or other signals.

Quadrature encoder interface: With two eQEPs, the F28004x can track two ABZ encoders, which are standard equipment on many servo motors. ABZ encoders track incremental position changes. They have two output signals, A and B, that provide quadrature-encoded signals, plus a Z output that gives an index position once per revolution. ABZ encoders have low latency and can track high speeds, but they require the controller to track the pulses to determine the position of the motor shaft, using the Z output as a reference. At power-on, the controller doesn't know the motor's position until it sees the first Z pulse.

Absolute encoder interface: The industry is trending toward absolute encoders to avoid the incremental encoder indexing problem. Absolute encoders are especially helpful for measuring the load angle; a dual-axis drive will need to support two interfaces to absolute encoders. The DesignDRIVE Position Manager solutions for absolute encoders combine the CLB and the serial port to create solutions for EnDat 2.2, BISS-C, the Tamagawa T format, and other absolute encoder standards. The CLB on the F28004x can handle two channels of any of these protocols.

In a dual-axis servo drive, the F28004x can process the motor incremental encoder and the load angle absolute encoder inputs concurrently. The feedback is direct to the MCU with no need for external logic.

Resources for Servo-Drive Developers

This article has discussed dual-axis servo control with the Piccolo F28004x MCU, but C2000 devices cover a wide range of servo and other real-time control applications. In addition, Texas Instruments has many resources available for developers of industrial servo drives. The Motor Drives [home page](#) contains links to applications including CNC and robotics servo drives; AC inverters and VF drives; and brushed and brushless DC drives, among others.