

# Motion-Control Chip Sets

## DSP and motor-control chips simplify DSP-based AC motor control hardware

by Aengus Murray

Variable-speed motor control systems have a wide range of applications, from high-end industrial robotics to ordinary home appliances, such as domestic washing machines. The control loops in these systems were first implemented using analog components. Typically, op amps were used in the feedback compensation circuits, and comparators were used to generate the control signals for the switching power converter. However, the advent of low cost microcomputers has increasingly led to the implementation of the control and user interface functions by digital means. The continuing reduction in the cost of processing power has made it possible to simplify the system hardware even further by implementing all the motor control functions on the same processor.

The Analog Devices Motion Control Group has been the leading supplier of resolver-to-digital conversion products to the ac servo motor market for the last ten years. Devices such as the AD2S80 and AD2S90 R/D converters are used to process the analog feedback signals for digital control of position and velocity in permanent-magnet ac servo drive systems. The latest offerings from the Motion Control Group, the ADMC200 and ADMC201 motion coprocessors, carry this concept farther. These multi-function devices combine, on a single chip, all the interface and signal-processing functions required to provide feedback-controlled current for ac motor control.

The ADMC201 provides the analog acquisition system to capture the motor current or voltage signals, the vector processing functions which condition these signals, and the pulse width modulator required to control the power converter. The ADMC201, combined with the ADSP-2105 digital signal processor, functions as a high performance control engine for ac motor drive systems. The following example describes the principal device features and their use in a motor-control application.

**AC servo motor control system:** A servo motor control system typically has two cascaded control loops, as shown in Figure 1. The *outer* motion loop controls the motor position and velocity based on feedback signals from a position or velocity sensor. The output of this loop is a demand for an increase or decrease in motor torque, which is fed to the *inner* current loop. The current loop generates signals for the power converter which supplies suitable motor currents to produce the desired output torque. The power flow from the dc supply rail to the motor is controlled by

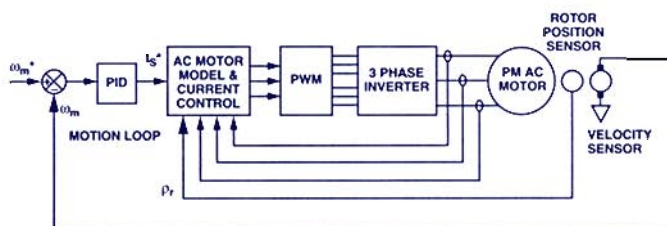


Figure 1. Servo-motor control system.

rapidly varying the on and off conduction periods of power semiconductor switches such as IGBTs or power MOSFETs. These control signals are typically fixed-frequency, variable duty-cycle waveforms that can be produced digitally using timing circuits.

In general, the motion loop design is independent of the motor type (ac or dc) but is solely a function of the mechanical properties of the system such as inertia, dynamic friction, etc. However, the current loop varies very much in sophistication depending on the motor type. In dc motors the torque is directly proportional to the dc current in the armature windings. But to control the torque in ac motors, the currents must be synchronized to the position of the rotating rotor field. One way to simplify the control of the motor torque is to transform the measured stator currents to a reference frame synchronized to the rotor field. This process (Figure 2) results in two equivalent dc motor current quantities: a torque-producing component,  $I_q$ , and a field-control component,  $I_d$ . The ac motor control system calculates the two quadrature voltages,  $V_d$  and  $V_q$ , required to force the  $I_q$  current to directly follow the torque demand and the  $I_d$  current to maintain a constant rotor field. An inverse transformation is then used to transform the "dc-motor"  $V_d$  and  $V_q$  voltages back to the stator reference frame to give the required winding voltages.

Figure 3 shows an all-digital implementation of a control scheme for a permanent magnet ac servo motor. A resolver-to-digital (R/D) converter derives digital angular position feedback information from the output signals of a shaft-mounted resolver. The outer position and velocity loop calculates the required motor torque current,  $I_q^*$ . Motor velocity is calculated from position measurements, using an estimation algorithm. The field-reduction component,  $I_d$ , is normally zero so as to maximize the motor torque output. However, the field weakening function can set a non zero  $I_d^*$  to effectively reduce the rotor field strength, thus increasing the motor speed range.

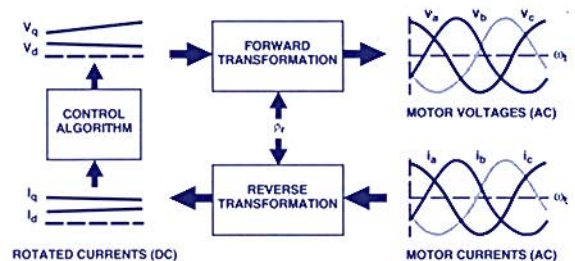


Figure 2. Vector-rotation function.

## IN THIS ISSUE

Volume 30, Number 2, 1996, 24 Pages

Editor's Notes (New Fellow), Authors .....	2
DSP-based chip set for AC motor control .....	3
Flexible low-cost wavelet video codec for image compression .....	7
Audio noise-reduction IC—25-dB of noise reduction without pre-encoding ..	10
A bibliography on EMC/EMI/ESD .....	11
Improved-fidelity single-chip digital stereo subsystem for wide range of sample rates ..	12
<b>New-Product Briefs:</b>	
A/D converters—Single supply, High speed, Low power .....	14
Two monolithic DACs and a Data-acquisition board .....	15
Amplifiers: Operational, Instrumentation, Isolation .....	16
Mixed Bag: DC-DC converters, Temperature sensors, Switch, Video encoders .....	17
Ask The Applications Engineer—21: Capacitance and capacitors .....	18
Worth Reading, More authors .....	22
Potpourri .....	23

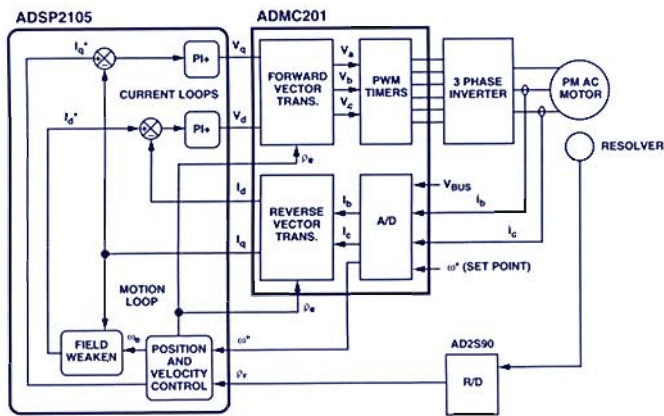


Figure 3. Control system for AC motor.

An A/D converter conditions the motor's stator current measurements, which are passed on as an input to the vector transformation. The reverse transformation takes two stator current signals—and the rotor electrical angle,  $\rho$ —and calculates the torque and field components,  $I_q$  and  $I_d$ . The third stator current signal is inferred, since all three stator currents sum to zero. There are two current loops, a torque loop and a field loop, with proportional and integral compensation (PI). The response of these loops can be improved by feeding forward the estimated winding back-cmf and winding-impedance drops (hence the '+' annotation). The  $V_d$  and  $V_q$  outputs of the calculation are then transformed in the vector-transformation block to digital equivalents of the three-phase stator voltages,  $V_a$ ,  $V_b$  and  $V_c$  for driving the motor.

The PWM timer block converts the digital inputs to pulse width modulated timing signals for the three phase inverter. The voltage applied to the motor windings is controlled by the conduction times of power transistor switches in each inverter leg. In the example below (Figure 4), when the upper inverter power transistor,  $Q_A$ , is turned on, winding "a" is connected to the +V bus power rail causing the winding current,  $i_a$ , to increase. When  $Q_A$  is turned off the winding current "free wheels" through lower inverter diode,  $D_{AP}$ , and connects the winding "a" to the -V bus power rail. The average stator winding voltage,  $V_A$ , is proportional to the conduction period,  $t_A$ , of power transistor,  $Q_A$ , and is given by:

$$V_a = \frac{(+V_{bus}) \times t_A + (-V_{bus}) \times (T - t_A)}{T} = V_{bus} \times \left( \frac{t_A - \frac{T}{2}}{\frac{T}{2}} \right) \quad \text{Eq. 1}$$

For negative stator currents, the winding current flows through lower transistor  $Q_{AP}$  and "free wheels" through upper diode  $D_A$ . In this case, the winding voltage is a function of the conduction

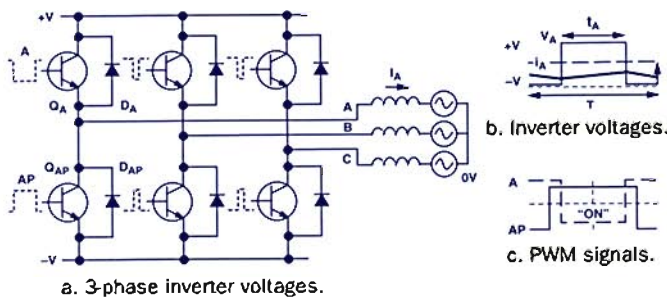


Figure 4. Three-phase inverter operation.

period of the diode  $D_A$ . In order to make the applied stator voltage independent of the stator current sense, the power transistor,  $Q_{AP}$ , is turned on when  $Q_A$  is turned off. However, to prevent the possibility of simultaneous conduction of these power transistors a brief "dead time" is inserted between the on signals for the upper and lower devices. The resultant active low PWM signals, shown in figure 4 (c), are complementary timing wave forms with a short "dead" time between the active portions.

The ac servo system described above can be built using three principal control components. The ADSP-2105 DSP implements the control loops, the ADMC201 interfaces to the three-phase inverter, and the AD2S90 interfaces to the resolver position sensor. The ADSP-2105 fixed point DSP has been optimized for high speed signal processing applications. It is very suitable for ac motor current control because of the short control loop cycle times—of the order of 50-100  $\mu$ s. The AD2S90 resolver-to-digital converter can be simply interfaced to the DSP using the serial port. [If the ADSP-2115 is used, it makes available an additional serial port.]. A companion oscillator IC, the AD2S99, is used for resolver excitation and provides loss-of-signal detection.

The ADMC201 provides the required interface functions between a DSP controller and the three phase inverter; it is suitable for use in controlling both permanent-magnet ac motors and ac induction motors. A detailed description of the ADMC201 and the interface to the ADSP-2105 follows.

**The ADMC200 Motion Coprocessor family:** The ADMC200 motion coprocessor has three principal functional blocks: a 4-channel, 11-bit, simultaneous-sampling A/D conversion system, a 12-bit zero-centered PWM timer block, and a vector-rotation block. The ADMC201 provides, in addition, three additional analog input channels and 6 bits of programmable digital I/O pins. The device has 25 internal memory-mapped registers for storing the peripheral input and output data. An embedded control sequencer decodes the chip-select line, Read- and Write lines, and 4 address lines—and maps these data registers directly into the DSP memory address space. This means that *all registers are directly accessible to the DSP at all times*. An on-chip Interrupt controller can interrupt the DSP at the end of an A/D conversion sequence or on completion of a vector transformation. The A/D converter's Start-of-Conversion line can be driven by the PWM timer block, to synchronize the control software and signal sampling to PWM frequency.

The ADMC200, designed on a CMOS process, combines low cost and low power consumption. The A/D converter, based on a CMOS-compatible switched-capacitor technique, is an 11-bit successive approximation device with a 4-channel simultaneous-sampling track-and-hold amplifier as its front end. This allows up to four motor current or voltage signals to be acquired without "skew" in less than 14.4  $\mu$ s. The ADMC201 has an internal 4:1 multiplexer, which provides an additional three asynchronous channels for slower signals, such as temperature or dc bus voltage. The converted values are in a two's-complement format to match the fixed-point DSP processor. The analog input range is 0 to 5 V, with 2.5 V equivalent to digital zero. The on-board reference has an absolute accuracy to within 5% (fully loaded). The overall accuracy of the analog to digital converter is 8 LSBs, while the channel to channel match is to within  $\pm 2$  LSB. A high Start-of-Conversion pulse acquires all four input channels and starts a conversion sequence for 2, 3 or 4 channels, depending on the

control-register settings. The end of conversion can be programmed to generate an interrupt pulse for the DSP, which can read the results registers in any order.

The 12 bit PWM block produces three pairs of constant-frequency variable-duty-cycle wave forms for the power converter switches, with a frequency ranging from 1.5 kHz to 25 kHz. The signals, described in Figure 5, are center based active-low signals so that the On (low) periods are symmetrical about the midpoint between timing pulses. This makes it easier to synchronize the current-sampling with the PWM wave forms. The wave forms are complemented, i.e., the power devices are switched in pairs: one device "on", the complementary device "off". To prevent the possibility of simultaneous conduction in the inverter power devices, complementary PWM wave forms are dead time adjusted (PWMDT). An active high PWMSYNC pulse, produced at the beginning of each PWM cycle, synchronizes the operation of the power inverter with the A/D converter.

The 12 bit hardware vector-rotation blocks can perform both forward- and reverse Park- and Clarke transformations between the stator (ac currents and voltages) and the rotor (dc machine equivalent) reference frames. The reverse transformation converts three-phase stator currents signals,  $I_a$ ,  $I_b$  and  $I_c$  to two orthogonal rotor referenced currents,  $I_d$  and  $I_q$ . The transformation consists of three stages (Table 1), where  $\rho$  is the angle of the rotor field.

**Table 1. Reverse Park transformation**

1. Third phase calculation:  $I_a = -I_b - I_c$

2. Transformation: 
$$\begin{pmatrix} I_x \\ I_y \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix} \begin{pmatrix} I_a \\ I_b \\ I_c \end{pmatrix}$$

3. Reverse Park Rotation: 
$$\begin{pmatrix} I_d \\ I_q \end{pmatrix} = \begin{bmatrix} \cos \rho & \sin \rho \\ -\sin \rho & \cos \rho \end{bmatrix} \begin{pmatrix} I_x \\ I_y \end{pmatrix}$$

The forward transformation converts two orthogonal rotor referenced voltages,  $V_d$  and  $V_q$ , to three-phase stator voltage signals,  $V_a$ ,  $V_b$  and  $V_c$ . The transformation consists of two stages (Table 2), where  $\rho$  is the angle of the rotor field.

**Table 2. Forward transformation**

1. Forward Park Rotation: 
$$\begin{pmatrix} V_x \\ V_y \end{pmatrix} = \begin{bmatrix} \cos \rho & -\sin \rho \\ \sin \rho & \cos \rho \end{bmatrix} \begin{pmatrix} V_d \\ V_q \end{pmatrix}$$

2. Inverse Clarke

Transformation: 
$$\begin{pmatrix} V_a \\ V_b \\ V_c \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ -1/2 & -\sqrt{3}/2 \\ -1/2 & \sqrt{3}/2 \end{bmatrix} \begin{pmatrix} V_x \\ V_y \end{pmatrix}$$

The ADMC201 digital I/O block has six digital lines, which can be configured as either inputs or outputs. They can also be configured as interrupt sources for system-protection functions. The I/O block is accessed via four memory-mapped registers.

**AC servo motor control software:** The software required to control an ac servo motor using the ADSP-2105 can take fewer than 500 lines of DSP code. Space limitations prevent a full description of the software here, but we will describe some of the core algorithms and code.

The current-control algorithm is synchronized to PWM frequency through an interrupt signal. The motor currents are sampled at the start of the PWM cycle by connecting the ADMC201's PWMSYNC pin (from the timer block) to the ADC's CONVST pin. The ADMC201's interrupt (IRQ) signal, connected to the ADSP-2105's IRQ2 pin, interrupts the DSP at the end of the A/D conversion cycle. The captured current signals represent the average winding current value since the sampling is at the midpoint of the current waveform. A set of current measurements are presented to the DSP after the start of each PWM cycle; and a new set of stator voltage values and PWM times are calculated before the next cycle.

The current loop signal flow diagram, in Figure 6, describes the flow of information between the ADSP-2105 DSP and the ADMC201 coprocessor. The algorithm starts when an interrupt from the ADMC201's A/D converter indicates that a new set of current samples are available. The DSP reads two phase-current values from the ADMC201's V and W registers, adjusts them for A/D and current sensor offsets, and writes them to the ADMC201 PHIP2 and PHIP3 vector transformation block, along with the rotor angle, RHO. The ADMC201 starts the reverse vector rotation, while the DSP may perform protection functions, such as overload detection or bus-voltage monitoring. The end of the transformation is signaled by an interrupt; then the DSP reads the ID and IQ registers and implements the current-loop control algorithm. The calculated VD and VQ values are written to the ADMC201's VD and VQ registers, along with the rotor angle RHOP. The ADMC201 starts the forward vector rotation, while the DSP may perform some further house keeping functions. The end of the transformation is signaled by another interrupt; the DSP reads the PHV1, PHV2 and PHV3 registers and scales these values according to the PWM period and bus voltage. The DSP then writes three new values to the PWM registers: PWMCHA, PWMCHB and PWMCHC, to close the current control loop.

The control algorithm consists of a number of control laws and some reads and writes to ADMC201 data registers. The ADI fixed point DSP is particularly well-suited to implementing control laws such as P-I (proportional + integral control) loops and state-space algorithms, for which there are many examples. The memory-mapped structure of the ADMC201 has the advantage that no

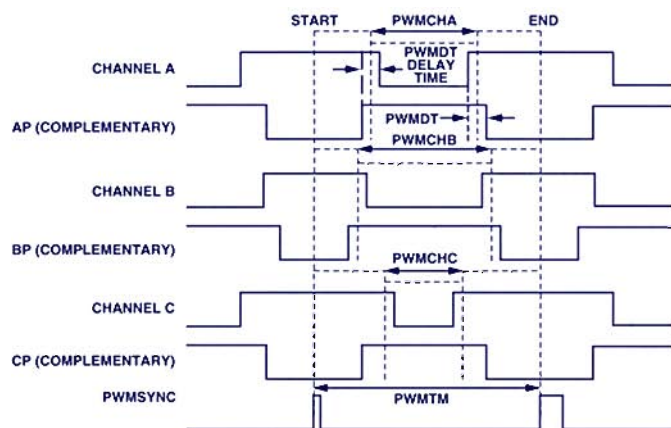


Figure 5. PWM waveform features in ADMC201.

special read or write sequence is required to access the data registers. The ADCM201 read and write registers are mapped to a block in the DSP external DM address space through the chip-select line on the device. A sample of the interrupt service routine code for an ADCM201 A/D converter interrupt is shown in the table below to illustrate this point. The first instruction is to read the ADCM201 system status register, using a data-memory read instruction. The AR register is then loaded with a constant which has the ADCM201 A/D interrupt bit set. If the A/D interrupt is not set then proceed to check for other sources of interrupt such as the PARK block interrupt. If bit is set then read the A/D registers and start the current loop algorithm.

```

IRQ2_ISR AY0 = DM(ADMC201_SYSSTAT); read ADCM201 stats register
:      AR = ADCM201_AD_INT;      ADCM201 A/D interrupt bit
      AR = AR AND AY0;          test: is this bit set?
      IF NE JUMP PARK_INTERRUPT; if not set then jump to next interrupt
      AX0 = DM(ADMC201_ADCV);   read the A/D channel V register
      DM(I_PH_V) = AX0;        save value in data memory
      AX0 = DM(ADMC201_ADCW);   read the A/D channel W register
      DM(I_PH_W) = AX0;        save value in data memory
      ....

```

**Conclusion:** We have selected an example of a control scheme for a permanent-magnet ac servo motor implemented with an ADSP-2105 (or ADSP-2115), ADCM201 and AD2S90 chip set.\* These hardware choices allow for a lot of flexibility in the system configuration. The ADSP-2115 can be upgraded to an ADSP-2101 or an ADSP-2181 if more processing "horsepower" is required. The lower-cost ADCM200 can be used if the additional

analog channels and digital I/O are not required. The ADSP-2105 and ADCM201 can also be used to control ac induction motors, with an encoder used in the place of the resolver.

An evaluation system is available,† using the ADSP-2101 EZLAB and an ADCM201 board. It comes with software, which illustrates the key features of the ADCM201 function blocks. This system can be used to build the control elements of a three-phase motor-control demonstration system. ▣

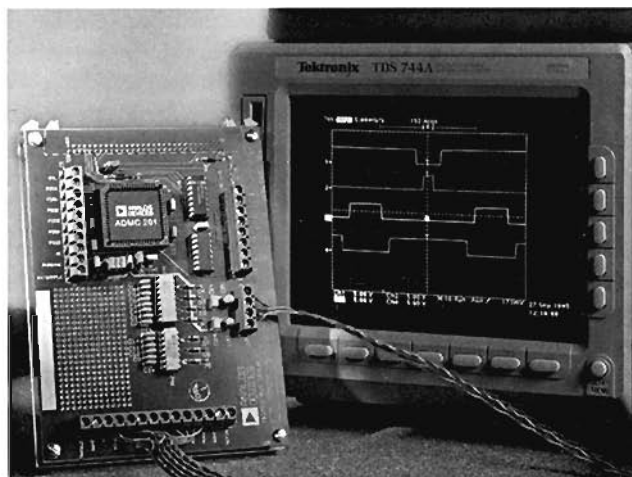


Figure 7. Motor-control evaluation system.

The ADCM200/201 core team included Art Lucia, Chris Coughlin, Jerry Hershkowitz, Tony Scavo, Bill Ahern, Lori Berenson, and Paul Kettle at ADI's Transportation and Industrial Products Division, in Wilmington, MA.

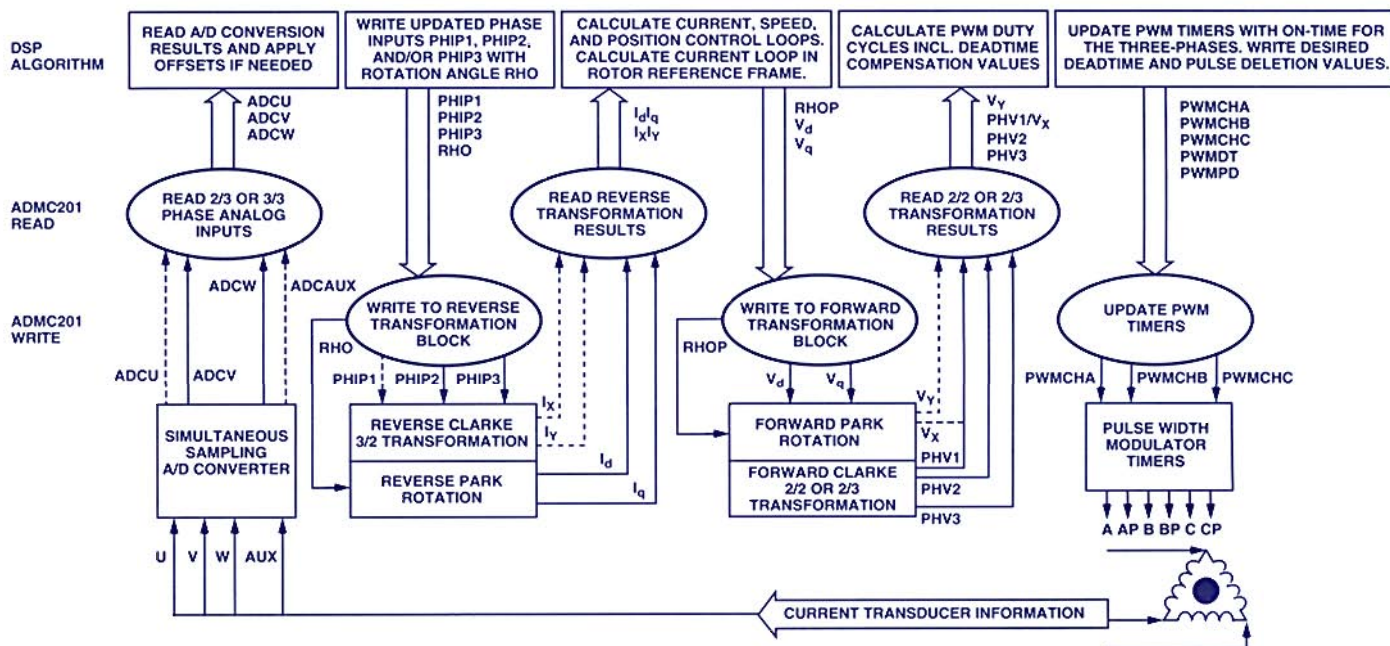


Figure 6. Control signal flow in AC motor control loop.

\*For data on these devices, Circle 1

†For information on the evaluation system, Circle 2