# Cost-Effective Variable-Speed Motor Control

## *Washing machines are a target market*

*The turbulent and steadily increasing price of energy, combined with increasing government levies on resource usage, is sharpening homeowners' focus on appliance energy efficiency.*

*By Aengus Murray, International Rectifier*

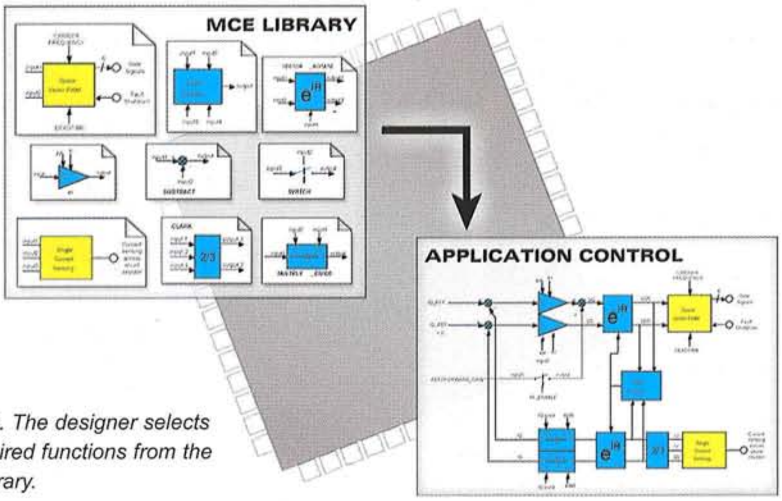


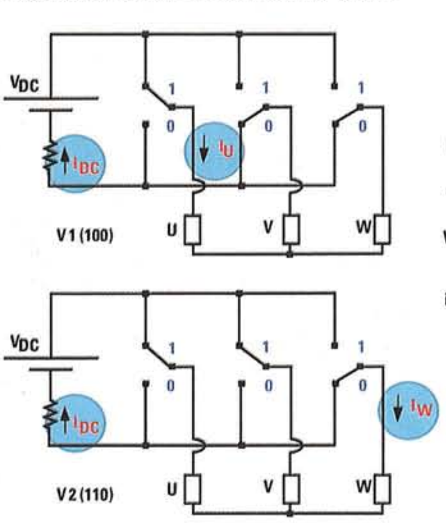
*Figure 2. The designer selects the required functions from the MCE library.*

Migrating appliances such as washing machines from fixed-speed motor operation to more flexible variable-speed operation can save energy while enabling design techniques such as advanced wash programs that also reduce water usage.

Washing machine designers have identified the permanent magnet synchronous motor (PMSM) as the most cost-effective and easily controlled choice for variable-speed motor operation. Speed and torque control are much easier to establish than with traditional induction motors, and the PMSM typically delivers greater efficiency in terms of torque per amp. This enables a smaller motor, thereby reducing per-unit costs and simplifying electrical and mechanical design.

A barrier to the arrival of the next-generation, ultra-efficient washer, however, is the need to implement additional sensing components such as Hall Effect sensors, to generate rotor positional information. Alternatively, the software challenges associated with coding a sensorless motor control algorithm introduce large risks to the project and require specialist DSP or RISC programming skills.

Designers can overcome this barrier with a sensorless motor control algorithm in hardware tailored to meet specific system requirements. This approach allows designers to implement variable-speed motor drives quickly and cost effectively. Among other important benefits, executing the algorithm at a higher speed in dedicated hardware results in better speed and torque control compared to a software-based approach. This provides greater flexibility for washer designers to create efficient programs that use less electrical energy.



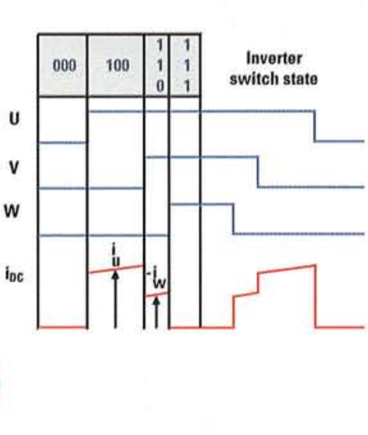


*Figure 1. Motor phase currents are calculated by sensing in the inverter DC link.*

**Sensorless Control Algorithm in Hardware**

In sensorless motor control, the algorithm determines rotor position typically by estimating the motor winding currents, because directly sensing actual current in the windings would require expensive signal isolation circuits. A number of estimation techniques may be used, but the technique International Rectifier has adopted for the Motion Control Engine (MCE) algorithm deployed in its iMOTION platform for washers, derives accurate values from actual measurements of current in the inverter DC link used to drive the PMSM, as shown in figure 1.

This produces an accurate representation of the motor winding because, in each PWM switching cycle, there are two inverter-switching states when the current flowing in the DC bus matches the current in a motor winding. Hence, by sampling the DC link current twice in the PWM cycle, it is possible to measure two of the motor phase currents. A calculation for the third phase current is generated from the two measured currents since the three currents must sum to zero. From this directly measured link current, therefore, IR's MCE is able to extract accurate estimates for rotor angle and speed to drive speed and torque control algorithms.

The MCE actually provides a library of motor control functional blocks as soft IP (figure 2) that the designer selects using a graphical configuration tool.

When the desired motor control functions have been selected, a dedicated compiler translates the control design into the MCE sequencer instructions that connect the hardware macro blocks in the correct sequence to implement the control system. This eliminates software coding from the development process, simplifying the implementation of sensorless speed control. This not only saves time and reduces dependence on specialised motor control design skills, but also reduces design errors.

A sample algorithm, implemented using functional blocks from the IR MCE, is shown in figure 3. The MCE uses a two-phase equivalent model of the PMSM circuit, to simplify rotor angle estimation. Hence, in the algorithm shown, the three rotor current values reconstructed from the DC link measurements are converted into equivalent two-phase values using the Clarke transformation function. Rotor flux functions are then derived using the two-phase circuit equations, from the current values produced by the Clarke transformation and the voltage values produced by the forward vector rotation function (ej?). The rotor phase PLL then ensures accurate estimated values for rotor angle and speed.

The Field Oriented Control (FOC) blocks shown in figure 3 decouple the transformed AC motor winding currents into two DC components representing torque (IQ) and flux (ID). This simplifies the controller design by allowing current loop tuning to become independent of speed. The speed loop then becomes an outer loop, used to calculate the torque reference command for the IQ loop based on the speed error. The RAMP function shown maintains motor acceleration within specified limits, and the LIMIT function is implemented to prevent the motor current exceeding the manufacturer's recommended maximum.

In the algorithm shown, an additional function implements reluctance torque control for an interior permanent magnet motor (IPM). Reluctance torque is an extra component of torque enabled by the unique construction of an IPM, and is not present in surface permanent magnet (SPM) motors. IPM control allows the designer to maximise the IPM's torque output by adjusting the phase advance. Since the MCE is configurable to meet individual motor control requirements, the designer has the flexibility to exclude the IPM control block if not required.

## Optimised Digital Control ICs

By implementing the MCE as configurable hardware IP, International Rectifier opened the door to an additional level of integration in which one piece of silicon delivers the MCE and other motor control functions to reduce component count and development time even further. Among the initial second-generation motion control ICs to feature the MCE on-chip, the IRMCF341 for washer motor control applications, features an 8051-based microcontroller co-integrated on the same silicon (figure 4) The MCU is capable of running the washer application software including sequencing, user interface, host communication, and upper layer control tasks. The MCE and 8051-based MCU have shared access to a dual-port RAM, allowing set points such as target speed, to be made available to the MCU and the motion control engine. This allows set points such as motor speed to be adjusted under control of the application.

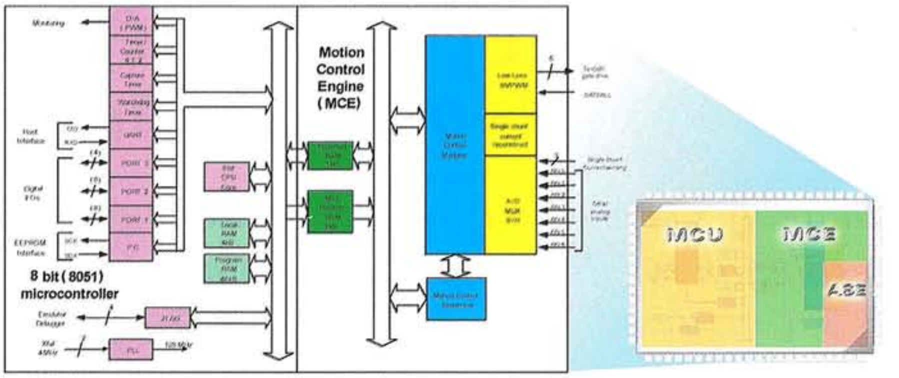


*Figure 4. MCE and co-integrated 8051 microcontroller, with A/D also on-chip.*

An analogue signal engine (ASE) also is integrated on-chip, as shown in figure 4. The ASE implements special hardware tailored to the single-shunt inverter architecture to enable a direct connection of the shunt resistor to the IC without requiring discrete analogue and digital circuitry.

A 12-bit ADC assists designers needing to adapt the gain of the amplifiers to meet specific system parameters, requiring only external resistor additions. Also, a number of spare analogue input channels are available for application-specific requirements. For instance, connecting the gate driver temperature-sensing circuit to the ASE and graphically inserting the appropriate blocks inside the MCE provides a route to implementing temperature protection.

Another advantage is that, when fine-tuning the digital control algorithm, the designer can rewrite control parameters and system variables quickly via the 8051 MCU communication ports, using the IR MCEDesigner PC-based configuration tool. This allows rapid fine-tuning of the motor control algorithm, to further streamline development of variable speed washer motor controls.

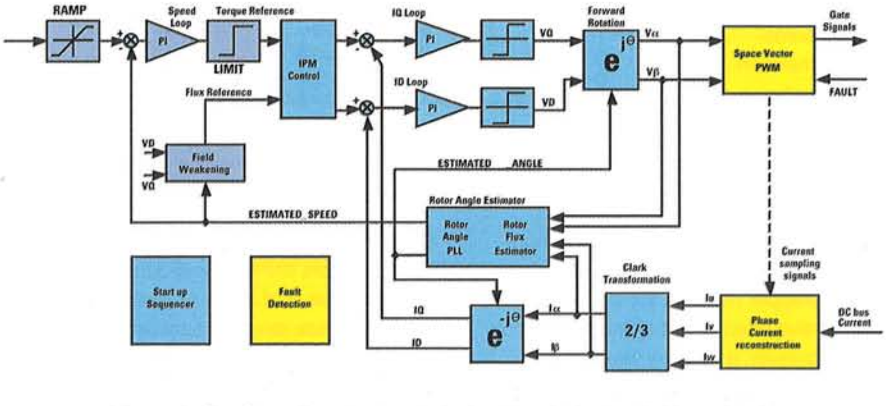


*Figure 3. Configured motion control algorithm including field weakening.*