AVR494: AC Induction Motor Control Using the constant V/f Principle and a Natural PWM Algorithm

AVR Microcontrollers

Application Note

1. Features

- Cost-effective and flexible 3-phase induction motor drive
- Interrupt driven
- Low memory and computing requirements

2. Introduction

Electrical power has been used for a long time to produce mechanical motion (either rotation or translation), thanks to electromechanical actuators. It is estimated that 50% of the electrical power produced in the United States is consumed by electrical motors. More than 50 motors can typically be found in a house, and nearly as many in a car.

To preserve the environment and to reduce green-house effect gas emissions, governments around the world are introducing regulations requiring white goods manufacturers and industrial factories to produce more energy efficient appliances. Most often, this goal can be reached by an efficient drive and control of the motor speed. This is the reason why appliance designers and semiconductor suppliers are now interested by the design of low-cost and energy-efficient variable speed drives.

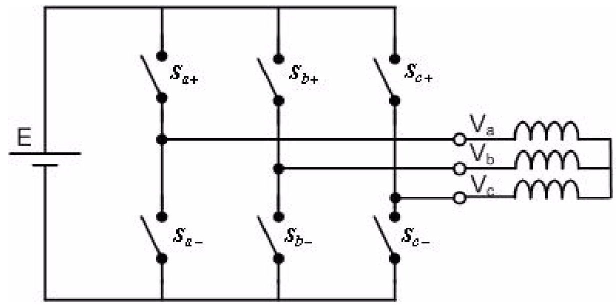
Because of their high robustness, reliability, low cost and high efficiency ($\approx$80%), AC induction motors are used in many industrial applications such as

- appliances (washers, blowers, refrigerators, fans, vacuum cleaners, compressors …);
- HVAC (heating, ventilation and air conditioning);
- industrial drives (motion control, centrifugal pumps, robotics, …);
- automotive control (electric vehicles)

However, induction motors can only run at their rated speed when they are connected to the main power supply. This is the reason why variable frequency drives are needed to vary the rotor speed of an induction motor. The most popular algorithm for the control of a three-phase induction motor is the V/f control approach using a natural pulse-width modulation (PWM) technique to drive a voltage-source inverter (VSI), as shown on Figure 2-1. The aim of this application note is to show how these techniques can be easily implemented on a AT90PWM3, an AVR RISC based microcontroller dedicated to power control applications.

7545A–AVR–12/05

Figure 2-1. Typical Structure of an Inverter-fed Induction Motor



3. AT90PWM3 Key Features

The control algorithms have been implemented on the AT90PWM3, a low-cost low-power single-chip microcontroller, achieving up to 16MIPS and suitable for the control of dc-dc buck-boost converters, permanent magnet synchronous machines, three-phase induction motors and brushless DC motors. This device integrates:

- a microcontroller with an 8-bit AVR advanced RISC architecture (similar to the ATmega 88),

- 8 kBytes of In-System-Programmable flash memory allowing up to 4096 instructions for the boot program and the application program,

- 512 bytes of static ram to store variables and lookup tables used in the application program,

- 512 bytes of EEPROM to store configuration data and look-up tables,

- one 8-bit timer and one 16-bit timer,

- a programmable watchdog timer with an internal oscillator,

- an 11-channel 10-bit ADC and a 10-bit DAC.

The main features that make this device suited to motor control applications are the three "power-stage controllers" (called PSC). These peripherals are 12-bit up/down counters with two comparators, whose output can drive the power transistors of an inverter leg. This allows to generate any three-phase waveform by pulse width modulation, with an easy management of the inverter dead times.

4. Theory of Operation

4.1 The Asynchronous Motor

In opposition to the brush or brushless DC motors, the asynchronous AC motor has no permanent magnets. The rotor is made of a squirrel cage where the rotating electric field induces a magnetic flux. Thanks to the speed difference between electric field in the stator and the magnetic flux in the rotor, the motor can deliver torque and turn.

4.2 The Constant V/f Principle

The constant Volts per Hertz principle is today the most common control principle used in adjustable-speed drives of induction machines [1,2]. Hence, many real-life motor control applications do not need a high dynamic performance, as long as the speed can be efficiently varied in the full range. This allows to use a sinusoidal steady state model of the induction motor, in which the magnitude of the stator flux is proportional to the ratio between the magnitude and the frequency of the stator voltage. If this ratio is kept constant, the stator flux will remain constant, and so the motor torque will only depend on the slip frequency.

More precisely, starting from the usual model of an induction motor expressed in a fixed reference frame

$$\frac{d\phi_s}{dt} + R_s I_s = V_s; \frac{d\phi_s}{dt} - j\,\omega_m \phi_r + R_r I_r = 0\,;$$
$$\phi_s = L_s I_s + L_m I_r;\ \phi_r = L_r I_r + L_m I_s;$$
$$C_{em} = \frac{3p}{2} L_m \operatorname{Im}(I_s I_r^*);\ \Omega_m = \frac{\omega_m}{p}$$

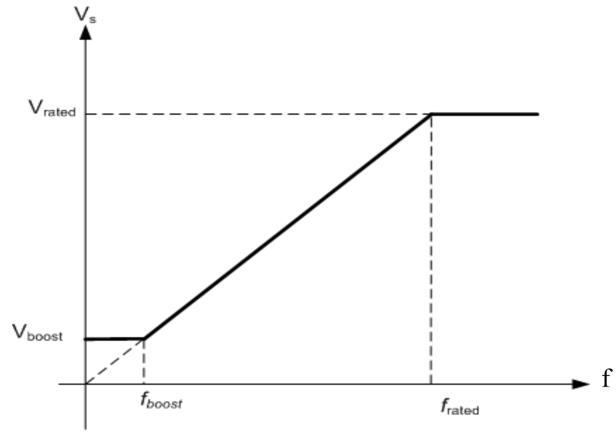
where V_s, ϕ_s, ϕ_r, I_s, I_r are respectively the stator voltage, stator and rotor magnetic fluxes, stator and rotor currents, and R_s, R_r, L_s, L_r, L_m and ω_m are respectively the global stator resistance, rotor resistance, stator inductance, rotor inductance, global leakage inductance and mechanical pulsation. If the motor is fed with a sinusoidal 3-phase voltage with a pulsation ω_s, $V_s = V_{sm}\, e^{j\omega_s t}$, the steady-state currents in the rotor and the stator will also be sinusoids with pulsation ω_s : $I_s = I_{sm}\, e^{j(\omega_s t + \varphi_s)}$ and $I_r = I_{rm}\, e^{j(\omega_s t + \varphi_r)}$. The previous equations lead to

$I_s = \frac{R_r + j\,L_r\,\omega_{slp}}{\Delta} V_s$, $I_r = -\frac{j\,L_m\,\omega_{slp}}{\Delta} V_s$ and $\phi_r = \frac{L_m R_r}{\Delta} V_s$, with $\omega_{slp} = \omega_s - \omega_m$

and $\Delta = (R_s + j\,L_s\,\omega_s)(R_r + j\,L_r\,\omega_{slp}) + L_m^2\,\omega_{slp}\,\omega_s$. Hence, the rotor magnitude ϕ_{rm} can be kept constant if the ratio $\frac{V_{sm}}{|\Delta|}$ is kept constant. At high speed, $\Delta \approx j\,R_r\,L_s \omega_s$,and the rotor flux magnitude is kept constant if the ratio $\frac{V_{sm}}{\omega_s}$ is kept constant: $\phi_{rm} \approx \frac{L_m R_r}{R_r\,L_s\,\omega_s} V_{sm} = \frac{L_m}{L_s}\frac{V_{sm}}{\omega_s}$.

The motor torque is then proportional to the slip frequency: $C_{em} = \frac{3p}{2}\frac{\phi_{rm}^2}{R_r}\omega_{slp}$. These expressions show that a desired motor torque C_{em} and a desired motor speed ω_m can be obtained if $\omega_s = \omega_m + \frac{2\,C_{em}\,R_r}{3\,p\,\phi_{rm}^2}$. At low speed, $\Delta \approx R_s\,R_r$, and $\phi_r \approx \frac{L_m}{R_s} V_s$. When the stator frequency fails under a given frequency threshold (called the boost frequency), the voltage magnitude must be kept at a given level (called the boost voltage) to keep the rotor flux magnitude constant. At the opposite, when the frequency becomes higher than the rated value, the voltage magnitude is also kept to the rated value, to take the saturation of the inverter into account. The rotor flux is no more constant and the torque decreases.

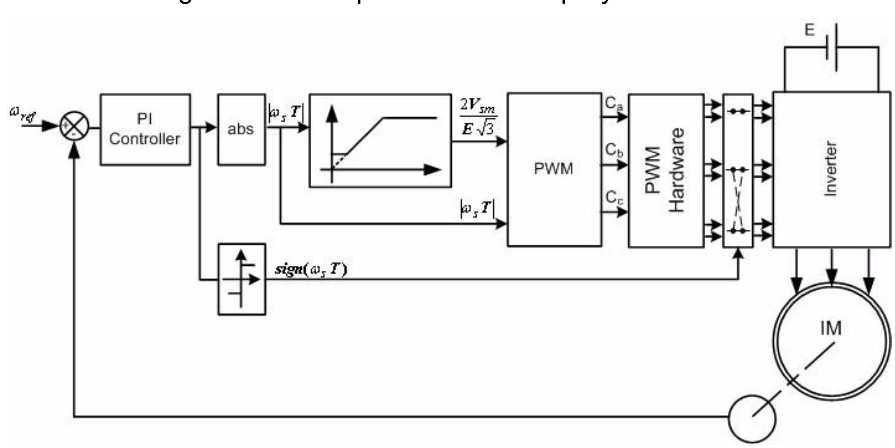
Figure 4-1. Stator Voltage Magnitude Versus the Stator Frequency Deduced from the V/f Principle



Roughly speaking, the scalar V/f control principle consists in feeding the motor windings with a 3-phase sinusoidal voltage whose amplitude is proportional to the frequency, except below the boost frequency and over the rated frequency, as shown on Figure 4-1. In practice, the slope that defines the relation between the voltage magnitude and the voltage frequency is deduced from the rated terminal supply voltage and the rated supply frequency written on the motor name plate, and the boost frequency is chosen equal to a percentage (say 5%) of the rated frequency.

This principle can be used to build a speed control loop (Figure 4-2.) in which the difference between the desired speed and the measured speed feeds a PI controller that determines the stator voltage frequency. To decrease the complexity of the controller, the input of the V/f law and of the space vector PWM algorithm is the absolute value of the stator voltage frequency. If the output of the PI controller is a negative number, two of the switching variables driving the power transistors of the inverter are interchanged. It should be noticed that the control principle described here can only be used in applications where the speed is kept constant whatever the load torque. Applications where the load torque must be kept constant whatever the motor speed require stator current measurements and more sophisticated control principles.

Figure 4-2. Block-diagram of a V/f Speed Control Loop System



4.3 The Natural PWM Principle

So as to feed the stator windings with a 3-phase sinusoidal voltage through an inverter, a first solution is to use a sine table to generate three sine waves with 120 degrees phase shift to each other. For this, the stator pulsation ω_s is used to feed three discrete-time integrators which compute the instantaneous phase of each stator voltage,

$$\theta_1[k] = \theta_1[k-1] + \omega_s[k]T_s$$
$$\theta_2[k] = \theta_2[k-1] + \omega_s[k]T_s$$
$$\theta_3[k] = \theta_3[k-1] + \omega_s[k]T_s$$

with $\theta_1[0] = 0$, $\theta_2[0] = -\frac{2\pi}{3}$, $\theta_3[0] = -\frac{4\pi}{3}$, T_s being the sampling period of the control algorithm. When one of these angles becomes higher than 2π, 2π is subtracted to it to keep it between 0 and 2π. A sine table is the used to compute the three voltages that should be applied to the stator,

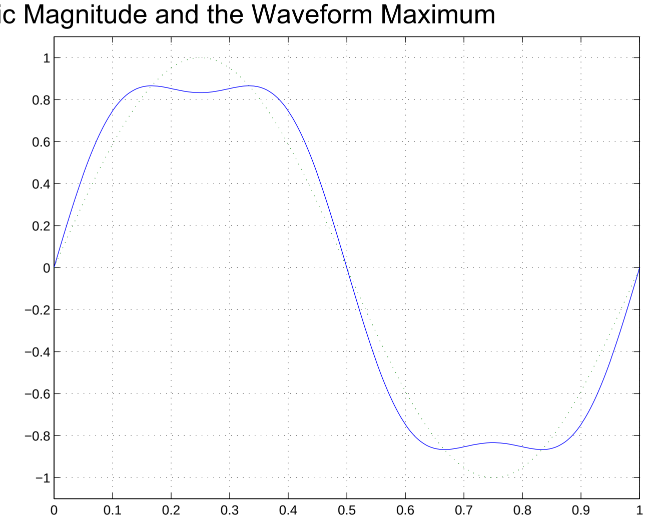
$$V_a[k] = V_{sm}(\omega_s[k])\ \mathrm{sita}(\theta_1[k])$$
$$V_b[k] = V_{sm}(\omega_s[k])\ \mathrm{sita}(\theta_2[k])$$
$$V_c[k] = V_{sm}(\omega_s[k])\ \mathrm{sita}(\theta_3[k])$$

where $V_{sm}(\omega_s)$ is the stator voltage magnitude deduced from the constant Volts per Hertz principle and $\mathrm{sita}(\theta) = \sin(\theta)$.

A slight improvement can be obtained by adding to the pure sine wave of the sine table a third harmonic, $\mathrm{sita}(\theta) = \sin(\theta) + \frac{1}{6}\sin(3\theta)$, since it has no effect on the motor behavior and it allows to generate a signal whose first harmonic has an amplitude which is 15.47% higher ($\frac{2}{\sqrt{3}}$) than the signal maximum (see Figure 4-3).

With this improvement, we can generate more AC voltage with the same DC bus voltage, so we can increase the speed of the motor with keeping constant the V/F ratio.

Figure 4-3. Use of a Non-sinusoidal Waveform to Increase the Ratio Between the First Harmonic Magnitude and the Waveform Maximum

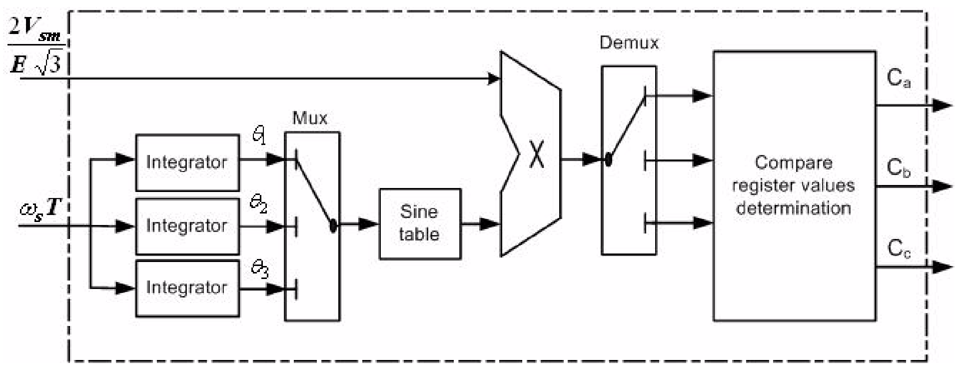


These values are compared to the output of an up/down counter (used as a triangle generator). When the up/down counter output oversteps one of these values, the corresponding output of the comparator toggles. As a result, the duty cycle of each PWM channel is proportional to the corresponding stator voltage value. Since this up/down counter with three comparators would be very heavy to implement by software, such a device must be included in a microcontroller so as to suit AC motor control applications. This is of course the case of the AT90PWM3, which provides three power stage controllers (PSCs). Taking the first phase as an example, the duty cycle stored in the compare register of the corresponding PSCs will be proportional to

$\frac{T_s}{2}(1+\alpha\frac{V_a[k]}{V_{s\max}})$, with $\alpha = 1-\frac{2\delta}{T_s}$, $V_{s\max}$ and δ are respectively the highest value of the stator

voltage magnitude and the dead time of the inverter switches. The resulting data-flow diagram is shown on Figure 4-4.

Figure 4-4. Natural PWM Data Flow Diagram



4.4 How Many Bytes are Needed to Store a Sine Table

As presented in the previous section, the natural PWM algorithm requires a sine table to compute $\sin(\theta)$ for all values of θ between 0 and 2π. Thanks to the properties of the trigonometric functions, several solutions are possible to reduce the length of this look-up table. The most efficient uses a look up table of the values of the sinus function for θ between 0 and $\frac{\pi}{3}$ only, since

$\sin(\theta) = \sin(\theta - \frac{\pi}{3}) + \sin(\frac{2\pi}{3} - \theta)$ for θ between $\frac{\pi}{3}$ and $\frac{2\pi}{3}$

$\sin(\theta) = \sin(\pi - \theta)$ for θ between $\frac{2\pi}{3}$ and π

$\sin(\theta) = -\sin(\theta - \pi)$ for θ between π and $\frac{4\pi}{3}$

$\sin(\theta) = -\sin(\theta - \frac{4\pi}{3}) + \sin(\frac{5\pi}{3} - \theta)$ for θ between $\frac{4\pi}{3}$ and $\frac{5\pi}{3}$

$\sin(\theta) = -\sin(2\pi - \theta)$ for θ between $\frac{5\pi}{3}$ and 2π

However, this solution does not easily allow to add a third harmonic to the sinus function, as explained in the previous section.This is the reason why we advise to use a look-up table

$\mathrm{sita}(\theta)$ with the values of either $\sin(\theta)$ or $\sin(\theta) + \frac{1}{6}\sin(3\theta)$ for θ between 0 and $\frac{\pi}{2}$, and to use the following relationships to compute $\mathrm{sita}(\theta)$ between $\frac{\pi}{2}$ and 2π:

$sita(\theta) = sita(\pi - \theta)$ for θ between $\frac{\pi}{2}$ and π

$sita(\theta) = -sita(\theta - \pi)$ for θ between π and $\frac{3\pi}{2}$

$sita(\theta) = -sita(2\pi - \theta)$ for θ between $\frac{3\pi}{2}$ and 2π

The latter solution allows to easily interchange between the two possible look-up tables.

4.5 The PI Regulator Principle

A PI controller is an algorithm that can be implemented without resorting to any heavy control theory. The aim of such an algorithm is to determine the plant input (in our case the stator voltage frequency) that will make the measured output (in our case the speed of the rotor) reach the reference (the speed the user wishes to have). PI stands for Proportional and Integral, two terms which describe two distinct elements of the controller:

- a proportional term, which is equal to the product of the error signal (the measured plant output subtracted to the reference) by a constant called the proportional gain. The proportional term mainly determines the short-term behavior of the controller since it determines how the controller strongly reacts to reference changes;

- an integral term, which adds long-term precision to the controller. This term is the product of the sum of all the previous error signal values by a constant called the integral gain. This sum keeps all the previous error signal values in memory, and evolves as long as the error is not zero. It allows the controller to cancel the difference between the measured output and the reference, but it usually makes the closed loop system slower and decreases its stability, however.

These two terms are sometimes added to a third one, proportional to the derivative of the error signal. The resulting regulator is then called a PID (Proportional, Integrator and Derivative). To control the speed of and induction motor by the V/f principle, this third term is not useful. It increases the speed of the closed loop, but it also derivates noises and it decreases the stability of the closed loop. So, the D term is tricky to adjust.

4.6 Sensors for Motor Control

Speed sensors play a critical role in a control loop. Several solutions are possible to obtain the speed and direction of the rotor.

The most precise, but also the most expensive, is to use absolute or incremental encoders. These optical sensors may be as expensive as the induction motor itself.

Another solution that we used in our experiments is to use the output of a tachometer generator that is connected to the rotor shaft. An analog-to-digital converter is needed to interface the output of this sensor with the microcontroller.

A third solution is to use Hall effect sensors. These cheap non-contact sensors are now proposed in small IC packages including the sensor and a sensor signal conditioning circuit. They provide an output that can be directly connected to the input-output port of a microcontroller.

5. Hardware Description (ATAVRMC200)

This application is available on the ATAVRMC200 evaluation board. This board provides a way to start and experiment asynchronous motor control.

ATAVRMC200 main features:

- AT90PWM3 microcontroller
- 110-230VAC motor drive
- Intelligent Power Module (230V / 370W board sized)
- ISP & Emulator interface
- RS232 interface
- Isolated I/O for sensors
- 0-10V input for command or sensor

6. Software Description

All algorithms have been written in the C language using IAR's embedded workbench and AVR Studio as development tools. The CPU is clocked at 8 MHz using the internal calibrated RC oscillator. In this application, 3 components of the microcontrollers play an important role:

The 8-bit timer0 is used to generate an interruption every 1 ms, which is the sampling period of the ADC and of the PIO controller. This timer is used in CTC mode (clear timer on compare) and clocked at 32 kHz. The 16-bit timer1 is free for other tasks.

The PSCs are clocked by the PLL at 64 MHz and are used as three synchronous counters, the third one (PSC2) being the “master” and PSC0 and PSC1 being the “slaves”. In this configuration, the modifications of the values of the compare registers of PSC0 and PSC1 are taken into account when the PSC2 compare registers are modified. This allows the three PSCs to evolve simultaneously. They are configured in centered mode with a switching frequency of 12 kHz (the value 2666 is stored in the RB registers, so that the PWM frequency approximately equals 64 MHz/(2* 2666)=12 kHz).

The analog-to-digital converter is also configured to generate an interruption when the conversion is finished. This allows to have a constant delay between two samples of the measured speed. The voltage reference of the converter is chosen as Vcc.

The digital-to-analog converter can also be used during the tests to watch how internal variables evolve. For the natural PWM algorithm, a table of the rounded values of $127\sin(\frac{2\pi k}{480})$ or $127(\sin(\frac{2\pi k}{480})+\frac{1}{6}\sin(\frac{6\pi k}{480}))$ for k between 0 and 120 is used. The length of this table (121 bytes) is a good trade-off between the size of the available internal memory and the quantification of the rotor shaft speed. For a bidirectional speed control, the values stored in two of the comparators are interchanged when the output of the PI regulator is a negative number.

Figure 6-1 and Figure 6-2 show the speed responses and the stator voltages obtained with the microcontroller for speed reference steps between +700 and -700 RPM. These results were obtained on a 750 W induction machine (with a load less than 370W). These figures show that the desired speed is reached after a 1 second transient, and that when the stator frequency ω_s obtained at the output of the PI regulator nears zero, the stator voltage magnitude is equal to the boost voltage. These figures also confirm that the same speed and torque can be obtained with a lower stator voltage peak-to-peak amplitude thanks to the use of a third harmonic component, at the price of a less regular transient.

Figure 6-1. Experimental Results Obtained with Purely Sinusoidal Lookup Table

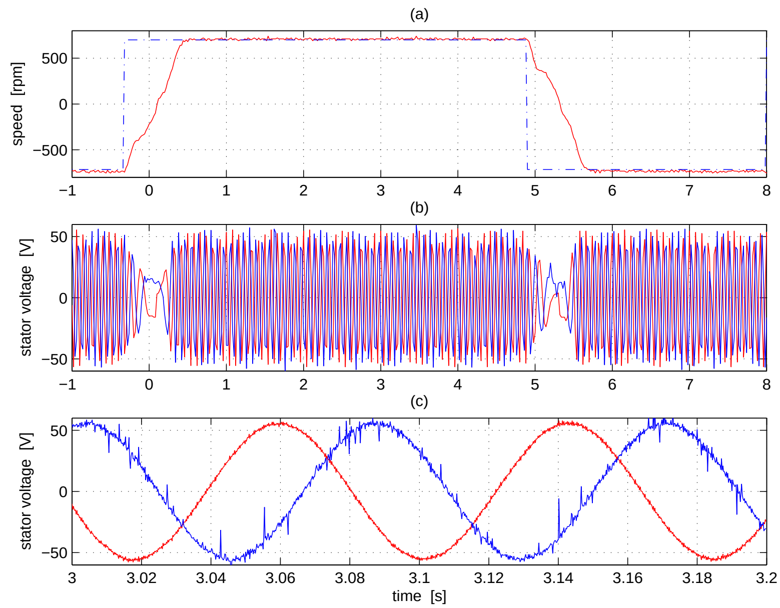
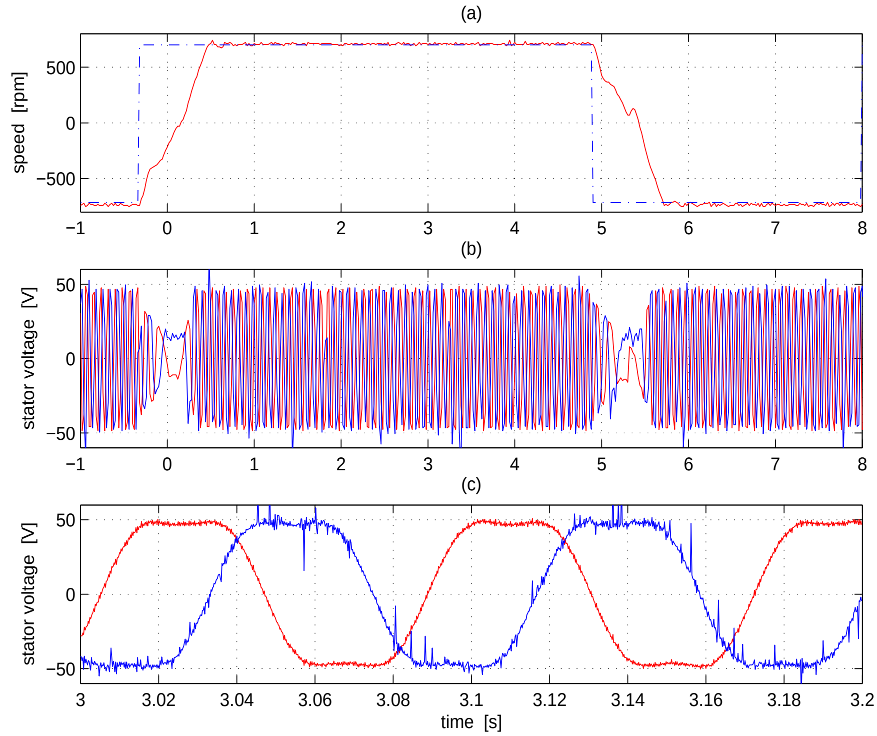


Figure 6-2. Experimental Results Obtained with a Lookup Table Including a Third Harmonic Component



7. Resources

Code Size: 1 947 bytes

Ram Size: 246 bytes (including sine table)

CPU Load: 30% (without PI regulation) / 55% (with PI regulation)

8. References

1. W. Leonhard, “Control of electrical drives”, 2nd Ed, Springer, 1996.

2. F.A. Toliyat, S.G. Campbell, “DSP-based electromechanical motion control”, CRC Press, 2004.

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

Printed on recycled paper.

7545A–AVR–12/05