Using the PIC16F84 Microcontroller in Intelligent Stepper Motor Control

As.ing. Papazian Petru, As.ing. Băbăiţă Mircea, Conf.dr.ing. Popovici Adrian*,
* Applied Electronics Departament,
„Politehnica" University of Timişoara, Electronics and Telecommunications Faculty,
Timişoara, România, mircea.babaita@etc.upt.ro

Abstract – For applications requiring critical rotor positioning, usually stepper motors are the best choice. Stepper motors operate differently from other motors; rather than voltage being applied and the rotor spinning smoothly, stepper motors turn on a series of electrical pulses to the motor's windings. Each pulse rotates the rotor by an exact degree and a series of pulses must be generated to perform a complete rotation. An intelligent controlling device was conceived and built based on a PIC16F84 microcontroller and UFDC-1 interface.

Keywords: stepper motor, driver, microcontroller

I. PIC16F84 MICROCONTROLLER

Suitable for various digital applications, the PIC16F84 RISC microcontroller has 13 I/O port lines, 1K Flash program memory, EEPROM data memory and an 8 bit timer/counter. The pin diagram of PIC16F84 is shown in figure 1.

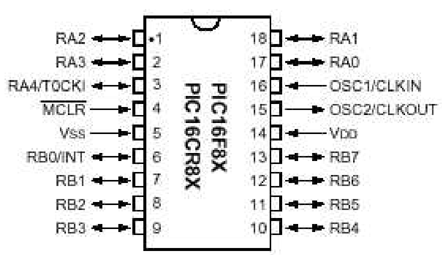

Fig. 1 PIC16F84 pin diagram

At a clock rate of 4 MHz, which is obtained by connecting a crystal oscillator between OSC1 and OSC2 pins, one instruction cycle is accomplished in 1 µs – a satisfactory speed for our application. The built in power on reset circuitry provides a safe start-up, therefore the Master Clear pin (/MCLR) is tied to VDD. Port A (RA0 – RA4) provides 5 bi-directional I/O lines and Port B (RB0 – RB7) 8 bi-directional I/O lines, independently configurable. Each line can source up to 20 mA current.

The PIC16FXX family has special features to reduce external components, thus reducing cost, enhancing system reliability and reducing power consumption.
The SLEEP (power-down) mode offers power saving. The user can wake the chip from sleep through several external and internal interrupts and resets. A highly reliable Watchdog Timer with its own on-chip RC oscillator provides protection against software lock-up.

The devices with Flash program memory allow the same device package to be used for prototyping and production. In-circuit reprogrammability allows the code to be updated without the device being removed from the end application. This is useful in the development of many applications where the device may not be easily accessible, but the prototypes may require code updates. This is also useful for remote applications where the code may need to be updated [1].

In-circuit reprogrammability can be achieved using I/O pins RB6 (serial clock) and RB7 (serial data). Programming voltage (Vpp) is applied through /MCLR pin and it's value is 13.8 V. To develop application software and to program the PIC16F84 microcontroller, an original user interface was designed, including a source code editor, a compiler (Parallax Compiler) and an unique featured program mode interface which can send to the device program code, data memory code and configuration information.

A. I/O lines initialization

At start-up, each port must be initialized i.e. control bytes must be written in registers TRIS A and TRIS B which gives the I/O direction of each port line. A '1' written in TRIS x will designate the corresponding Rx line as input, while a '0' will make it an output. For example, the following sequence will make lines RA3 – RA0 inputs, RA4 – output, all RB lines output.

```
clrf    PORTA
clrf    PORTB
bsf     STATUS, RP0
```

```
movlw    0x0F
movwf    TRISA
movlw    0x00
movwf    TRISB
```

To send a byte to PORTB lines (8 bit) a mov sequence must be issued, like follows:

```
movlw    0x0FF
movwf    PORTB
```

By this, all PORTB lines are set to '1', i.e. '11111111'. PORTA being only 5 bit wide, bits 7-5 are ignored, regardless of the value written in TRISA or PORTA registers. To read a port value (a full byte) we can use the sequence:

```
mov    PORTA, 0
```

which writes the 5 bit word of PORTA into the working register W. For testing a single bit of a port line, a BTFSS or BTFSC instruction must be used. BTFSS tests the corresponding bit and if it's value is '1' skips the following instruction. If BTFSC is used, then the next instruction is skipped if the bit is '0'. This technique is useful to make conditional branches in scanning external switch states.

II. ULN 2003 POWER DRIVER

In order to control a stepper motor, the coils of each phase must be driven by a certain pulse signal which is generated by the PIC16F84 microcontroller. Since I/O lines do not source a sufficient current, an integrated power driver is used. We choose a simple, low-cost device, ULN 2003, including seven darlington arrays with TTL or CMOS compatible inputs. Each channel is rated at 500mA and can withstand peak currents of 600mA. Suppression diodes are included for inductive load driving and the inputs are pinned opposite the outputs to simplify board layout [2].
Figure 2 shows the pin connection of ULN 2003.

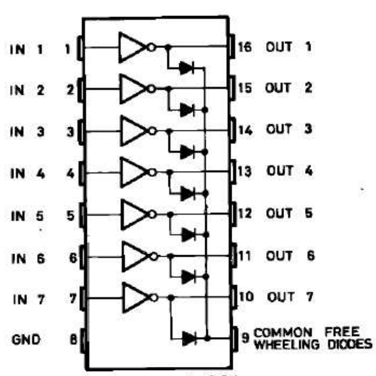

Fig. 2 ULN 2003 pin diagram

Each driver has a input, a common pin and an output as shown in figure 3.

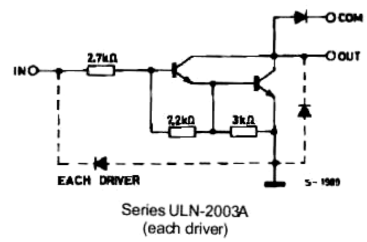

Fig.3 Internal driver schematic.

The device is powered by the external voltage supply which also drives the stepper motor's coils.

III. THE UNIVERSAL FREQUENCY-TO-DIGITAL CONVERTER (UFDC-1)

The Universal Frequency-to-Digital Converter (UFDC-1) is a universal 2-channel high precision multifunctional converter based on novel frequency-time conversion methods. It is perfectly suited to any applications where frequency-time parameters but also other sensors output signals have to be measured and transmitted to a central processing unit as a feedback in various automation systems [3].

Figure 4 shows the pin configuration of PDIP package for the UFDC-1 and Table 1 pin functionality.

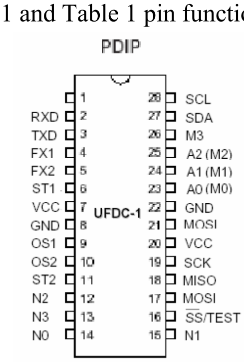

Fig.4 UFDC-1 pin configuration.

Name	Pin Descriptions
V_{DD}	Supply voltage
GND	Ground
FX1, ST1	Input channel 1
FX2, ST2	Input channel 2
OS1, OS2	Oscillator
M0...M3	Mode selection (see table 2)
N0...N3	Accuracy selection (see table 3)
A0...A2	Addresses (for I^2C bus communication)
TEST	Calibration signal
SCK	SPI Bus Master clock Input
MISO	SPI Bus Master Input/Slave Output
MOSI	SPI Bus Master Output/Slave Input
SS	SPI Bus Master Slave select
SCL	Two-wire (I^2C compatible) Serial Bus Clock Line
SDA	Two-wire (I^2C compatible) Serial Bus Data Input/Output Line
RXD	UART Input Pin
TXD	UART Output Pin

Table 1. Pin description

This device offers high performance with flexibility and requires minimum possible number of external components It can be easily included into digital environment controlled by an external microcontroller with software configured function selection.

IV. STEPPER MOTOR

The application hereby considers an unipolar stepper motor, which has logically two windings per phase, one for each direction of current. Since in this arrangement a magnetic pole can be reversed without switching the direction of current, the commutation circuit can be made very simple for each winding. A typical winding configuration is illustrated in figure 5.

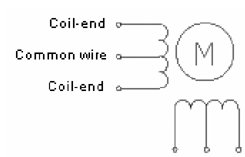

Fig. 5 Unipolar stepper motor windings

Stepper motor performance is strongly dependent on the drive circuit. Therefore we approach the latest chopper drive circuit method [4] building a digital intelligent controlling device which implements linear constant current control in combination with a pulse width modulation scheme.

V. CONTROL METHOD AND EXPERIMETAL DEVICE

We use a common unipolar stepper motor named KP4M4-001 which has four phases and 100 steps, each step being 3.6º. It has 5 terminals as shown in figure 6.

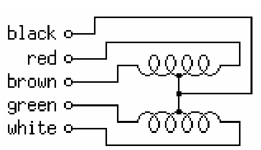

Fig. 6 KP4M4-001 motor terminals

The black terminal is common and will be connected to a 12V power supply.

Linear constant current control in combination with a pulse-width modulation scheme (chopper circuit) reduces the effect of motor winding's time constant (L/R) upon current increase during each input pulse and also cuts power dissipation too.

A four-phase unipolar stepper motor (as shown in figure 6) could use a Darlington like the ULN 2003 as a driver and a current sensing circuitry built with operational amplifiers which output in the end a frequency signal proportional with the measured winding current. This signal is fed into the UFDC-1 interface which communicates with the PIC16F84 microcontroller via the I2C protocol.

The classical topology for winding current sensing in a chopper circuit is illustrated in figure 7.

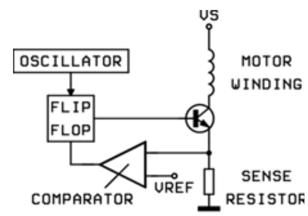

Fig. 7 Classic driver circuit with current sensing.

Our method will replace the whole sensing circuitry with a current-frequency converter connected to the UFDC-1 and controlled by the PIC16F84 microcontroller.

A. Current-frequency converter

Using operational amplifiers is an efficient and common method for such circuits. The sensing resistor will provide an input voltage for the first stage which is an integrating circuit. The resulting signal is fed to a comparator, resulting the output frequency signal. Since the UFDC-1 accepts only TTL input levels a signal conditioning circuit is required. It may be a simple transistor interface. Figure 8 shows the current-frequency converter.

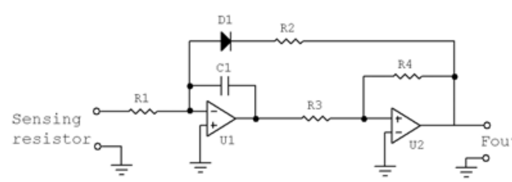

Fig. 8 Current to frequency converter.

First stage output voltage will be:

$$v_1 = -\frac{U_{in}}{R_1 C} t \qquad (1)$$

and generated output signal:

$$v_2 = \frac{U_{in}}{R_1 C} \frac{R_4}{R_3} t \qquad (2)$$

where Uin is the voltage on the sensing resistor, fed to the input of first operational amplifier (U1).
Regarding (1) and (2) and considering V11 and V12 the positive and negative saturation voltages of the comparator (U2) we have the expression of the frequency signal:

$$f = \frac{U_{in}}{R_1 C} \frac{R_4}{R_3 (V_{11} - V_{12})} \quad (3)$$

Thus we obtain a linear dependence of the frequency f and input signal's voltage Uin also proportional with the winding's current.

Since the stepper motor has four terminals for the windings, we can use an analog CMOS multiplexer to fed the sensed voltages into the converter shown in figure 8. The TTL interface will provide the rectangular pulses for the UFDC-1's input.

The block diagram of the experimental device is shown in figure 9.

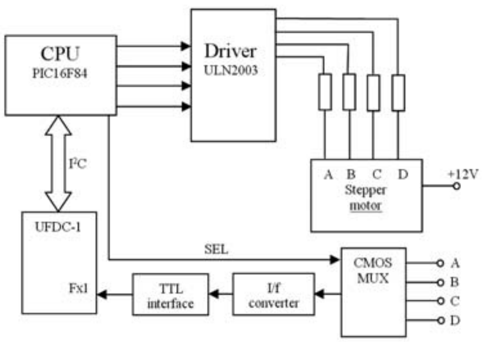

Fig. 9 Bloc diagram of control unit

Control signals for the four phase inputs are provided by the PIC16F84 microcontroller, based upon a specific algorithm. The sequence can be stored into the PIC's internal flash memory or can be read from an external I2C memory device connected on the same bus as the UFDC-1. The chopping algorithm is also provided by the microcontroller's software based upon readings from the current sensing circuit and translated by the UFDC-1. The CMOS multiplexer provides sequentially to the I/f converter the signals from the four motor windings terminals. The CPU (Central Processing Unit) triggers the multiplexer accordingly to assure the right value to be read.

VI. CONCLUSIONS

An intelligent controlling device was conceived and built based on a PIC16F84 microcontroller and UFDC-1 interface. Considering classic linear constant current control it was designed a digital sensing system for the winding's current, using current to frequency conversion and a powerful interface like UFDC-1. The Central Processing Unit is capable of rendering and processing all the tasks needed to perform a control using pulse width modulation scheme (chopping method), thus providing a simple and flexible device for stepper motor control applications.

REFERENCES

[1] Microchip, PIC16F84 Data sheet.
[2] *** , ULN2003 Data sheet.
[3] IFSA, UFDC-1 Specification and Application Note.
[4] ST, Stepper-motor performance constant-current chopper drive.
[5] Papazian P., Băbăiță M., Popovici A., Avram A., Lie I., "Signal Generating Techniques using PIC16F84 Microcontrollers", ICMCS-05, Proceedings of the 4'th International Conference on Microelectronics and Computer Science 15-17 September 2005, Technical University of Moldova & Academy of Sciences of Moldova, Chișinău, Vol.II, ISBN 9975-66-040-1, pg.401-404
[6] Papazian P., Popescu V., Băbăiță M., Filip A., "Using the PIC16F84 Microcontroller in Signal Generating Techniques", EMES 2006, Proceedings of the 9'th International Conference on Engeneering of Modern Electric Systems, 26-28 May, 2006, Analele Universității din Oradea, Fascicola Electrotehnică, Secțiunea Electronică, ISSN 1454-9239, pg.25-28.