Stepper Motor Controller Using Altera MAX Series

You can use the MAX® II, MAX V, and MAX 10 devices to implement the motor control logic. Their low power, easy power-on feature, and unique built-in internal oscillator make them ideal programmable logic devices to implement motor controlling systems.

The advantage of precision control, open-loop control of the motor, self contained braking, and the absence of brushes, makes the stepper motor a convenient choice for a variety of specialized applications. Printers and plotters, disk drives, robots, computer numerical control machines, and other precision machines are common examples where the stepper motor is used extensively.

The operation of a stepper motor can be explained by considering a series of electromagnets arranged in a circle to encapsulate a rotor made up of a magnetic material. When these solenoids, or electromagnets, are energized in sequence, the magneto motive force developed in them interacts with the rotor and causes it to re-align to the magnetic field, thereby causing it to rotate in a clockwise or counterclockwise direction. You can control the angular displacement of the motor by simply switching these electromagnets on or off in a fixed pattern for the desired motion of the motor.

Related Information

- **Design Example for MAX II**
  Provides the design files for this application note (AN 488).
- **Design Example for MAX 10**
  Provides the MAX 10 design files for this application note (AN 488).
- **AN422: Power Management in Portable Systems Using MAX II CPLDs**
- **AN428: MAX II CPLD Design Guidelines**

Stepper Motor Controllers Using Supported Altera Devices

The motor controller design is implemented in the device to govern the performance and operation of a unipolar permanent magnet stepper motor. The design uses a few switches and buttons on the demo board to serve as the user interface.

This motor controller design offers the following advantages:

- Two types of control for starting and stopping the motor and selecting forward or reverse rotation: manual control (through the user interface) or automatic control (through a microcontroller).
- Two modes of operation: continuous mode and step mode.
- Eliminates the need for an external clock signal as the devices have a unique internal oscillator which is incorporated in this design.

ISO 9001:2008 Registered

ALTERA®

The following figure shows the organization of the stepper motor controller. The driver circuitry is supplied by a DC source because of the requirements of the motor's windings. In accordance with the controller logic, the ports connected to the device drive the inputs of the motor driver, thereby driving the stepper motor. The controller incorporates logic for half stepping the motor.

Figure 1: Implementing a Stepper Motor Controller Using a Supported Altera Device
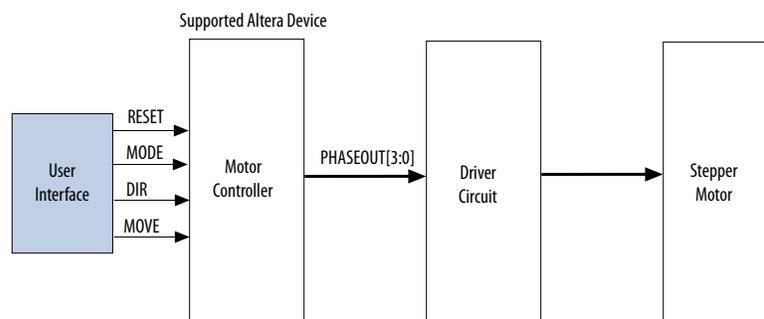
Table 1: Input and Output Signals of Motor Controller

Signal	Direction	Description
RESET	Input	Resets the position of the motor to the initial reference position.
MODE	Input	Switches between the continuous mode (normal motor) or the step mode of operation.
DIR	Input	Enables the rotor to rotate counterclockwise.
MOVE	Input	Uses in step mode for the rotor to move in a particular direction by half step (45°).
PHASEOUT[3:0]	Output	4-bit output of the device that goes to the MOSFET driver circuit.

The rotation steps are controlled by the excitation on each of the coils that follows. A 4-pole stepper motor has four electromagnets uniformly placed over its circumference. The motor controller must provide the following sequence of inputs to these electromagnets through its output ports (PHASEOUT) to rotate the rotor by four times the step size of the motor being used:

1000, 1100, 0100, 0110, 0010, 0011, 0001, 1001, 1000...

The preceding half-step sequence of operation causes the rotor of a 200-tooth, 4-pole motor to rotate by 0.9° at each step.

This design offers continuous mode and step mode for motor control.

- Continuos mode (set MODE input low)—the motor rotates normally without any discontinuity in rotor displacement.
- Step mode (set MODE input high)—the motor operates every time you press the MOVE button. The MOVE input is normally pulled high by a pull-up resistor. Every time a negative edge is detected on the MOVE input, the motor advances by half a step.

The RESET input returns the motor to its reference position. The rotor works until it reaches the position where PHASEOUT is 1000, regardless of its initial displacement.

Implementation

The detailed description of the implementation is based on the MAX II devices. This application can also be implemented in MAX V and MAX 10 devices.

Design Example: Targeting a MDN-B2 Demo Board (MAX II Devices)

You can implement this design with an EPM240 or any other MAX II device.

The stepper motor works by supplying inputs through a user interface, which form the controller inputs to the MAX II device. You can input mode, direction, and step control through user buttons and switches that are assigned to various general purpose I/Os (GPIOs) of the MAX II device.

The following details the implementation of this design on the MDN-B2 demo board using this design source code. For illustrative purposes, this design is made using LEDs. Each LED represents a winding of the motor. The winding energizing sequence can thus be "seen" on the demo board. However, note that the LEDs glow on a logic 0 and are turned off when the port detects a logic 1. You can replace these LEDs by a motor driver circuit to run a motor. All the lines are parallel and independent of each other. Each line excites one separate coil winding of the motor and causes the rotor to rotate in a specific direction and manner as set by the user control.

Table 2: EPM240 Pin Assignments for MDN-B2 Demo Board

Unused pins are assigned as input tri-stated in the Device and Pin option settings in the Quartus® II software prior to compilation.

Signal	Pin
dir	pin 38
mode	pin 37
move	pin 82
rst	pin 77
phaseout[0]	pin 71
phaseout[1]	pin 72
phaseout[2]	pin 73
phaseout[3]	pin 74

To demonstrate this design on the MDN-B2 demo board, follow these steps:

1. Select the EPM240GT100C3 device, compile the source code, and assign the pins (as shown in the pin assignments table).
2. Switch on power to the board using slide switch SW1. Download the design to the MAX II device using the JTAG header JP5 on the MDN-B2 demo board and a conventional programming cable (ByteBlaster™ II, USB-Blaster™).
3. Press switch SW4 on the demo board to begin the download process. Remove the JTAG connector and switch off power after programming the MAX II device.
4. Switch on power to the demo board by sliding slide switch SW1 to ON.

This version of stepper motor control displays the phase excitation on the LEDs on the demo board. The four `phaseout` outputs are connected to red LEDs (D5, D6, D8, and D10) on the demo board.

5. Select the mode and direction using the following switches of the DIP switch SW3 (Control Switch):

- Switch #1—assigned to set the mode to either continuous mode or step mode.
- Switch #2—assigned to set the direction.

6. Use push button SW9 for the "move" function.
   Note the LEDs progress in the excitation sequence that was set in step 5 as you push this button.

- When the mode is set to step mode and SW9 is pressed, the LEDs progress one step for each push.
- When the mode is set to continuous mode and SW9 is pressed, the LEDs appear to be moving one step after another without any user intervention.

The direction switch changes the direction of this sequence. Push button SW6 on the board is used to reset the demo board.

Acknowledgments

Document Revision History

Date	Version	Changes
September 2014	2014.09.22	Added MAX V and MAX 10 devices.
December 2007	1.0	Initial release.