Introduction

Robots have begun to play a large part in the factory automation process, substituting for humans in operations such as welding, painting, assembling, cutting, palletizing and general operations where the machine can perform the work cheaper, faster and more accurately. This article focuses on the system description and requirements from a motor control perspective.

Requirements

Whether the configuration of the robot architecture is linear or articulated, most applications demand high accuracy of the robotic arm motion. Therefore, the motor control strategy employs the position control loop where actual position is captured by the position sensor, usually the incremental or absolute encoder with very high resolution. The degrees of freedom (DOF) (number of moving joints) of the robotic system are equal to the number of motors used. Consequently, the higher the value of DOF, the higher the requirements on precision of movement of each motor, as position errors introduced by each motor are multiplied. It is not rare to find encoders with millions of pulses in these sorts of applications. The demands on the position control of the tool holder of the punching or drilling CNC machine are lower when compared to the welding or miller CNC machine, where the movement of the joints has to be precisely synchronized in order to maintain the required trajectory.

Concept

The high-level block diagram shown in figure 1 illustrates the components of a simple robotic system, in this case, a milling CNC machine. The top layer of the machine control architecture is the main CNC controller, which typically requires utilization of multiple MCU cores. The tasks and services it must perform include:

- Human-machine interface/display should enable entering, visualization and editing of the complete CNC program.
- System supervisor monitors and directs other MCUs, handles system exceptions and interrupt signals. It stores the CNC control program, tool calibration and tool offset parameters, as well as different user compensations and other settings.
- Axis control processor interprets the CNC program and calculates positional instruction by interpolating it to various coordinate systems and sending the information to particular motor controllers.

From the peripheral requirements point of view, the MCU should be capable of handling different kinds of industrial communication protocols and contain a large internal on-chip memory. On the other hand, there is no need for specific motor control peripheral modules.
The demands on the motor control layer are different from the upper layer. Applying a single MCU may not satisfy the application needs in each case. An additional monitoring safety MCU might be required. Beyond communication, the main MCU executes the motor control algorithm and handles the fault states of the particular drive. The motor control algorithm includes calculation of position, speed and current (torque) control loops. The optimal size of on-chip non-volatile memory is in the range of tens of kilobytes. The MCU must have dedicated motor control peripheral modules, including a timer for 6-channel PWM generation, a fast and accurate AD converter and an interface for processing the encoder signals.

Communication between the main CNC controller and the motor control MCUs is sometimes realized with optical bus in order to protect the position information in harsh, noisy environments.

The bottom layer represents power modules, each driving a single motor. These do not contain specific MCU logic, but can be equipped with an intelligent driver of IGBTs or power MOSFETs that can perform failsafe and diagnostic features. The information is passed over the fast communication interface to the motor control MCU. The power module measures the feedback signals (phase currents, voltage) used in the control algorithm.

Robotic systems often include additional components that have to be controlled by an MCU, such as an automatic tool changer and tool coolant control, or in the case of the CNC lathe machine, a spindle drive control.

Implementation of Freescale MCUs

Each layer of the control chain can be equipped by a Freescale MCU.

As mentioned, the top layer requires significant computation power to perform multiple tasks, though it does not require specific motor control peripherals. The Freescale portfolio of 32-bit solutions offers several options to meet this need:

- Vybrid controller solutions built on single- or dual-core ARM Cortex-A5/Cortex-M4 solutions
- Kinetis K70 MCU based on the ARM Cortex-M4 core

These MCUs contain safety features that are equipped with floating point unit, possess high-performance cores and are suitable for trajectory calculation.

Dedicated motor control MCUs are available in the following Freescale families:

- MC56F84xx, 32-bit/100 MHz DSC based on 56800EX Core
- Kinetis K40, K60 MCUs based on the ARM Cortex-M4 core

These solutions have dedicated motor control peripheral modules, including PWM generation with synchronized ADC. Floating point unit is not required, as the performance of the core is sufficient for execution of the vector control algorithm.

Freescale Enablement

Additional information, including reference designs, application notes and embedded software libraries for motor control is available at freescale.com/motorcontrol.
Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright license granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

For more information, visit freescale.com/motorcontrol

Freescale, the Freescale logo, and Kinetis are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. Vybrid is a trademark of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. ARM is the registered trademark of ARM Limited. ARM Cortex-A5, Cortex-M4 and Cortex-M0+ are trademarks of ARM Limited. © 2013 Freescale Semiconductor, Inc.

Document Number: SRVROBINDAPPART REV 1