

Embedded Systems Essentials with Arm: Getting Started

Module 3

KV3 (3): GPIO Design and Microcontrollers

Because just a few physical pins must serve many functions in most processors, physical pins are assigned dynamically to different functions, such as the timer, clock, or GPIOs.

This means that GPIOs need to be assigned separately for each physical pin.

Each physical pin is connected to a multiplexer that allows each pin to perform several functions, and optimizes functionality for small packages.

The multiplexer is managed by a pin control register, or PCR.

A GPIO pin can represent a single bit which can take the logical value of 0 or 1. To save memory, GPIOs are grouped in ports according to the register size of the controller. Each port is connected to the microcontroller via a peripheral bridge. If the controller arranges the registers into groups of 32 bits, the GPIOs are combined into ports of 32 pins.

In order to save energy the clock is disabled by default in certain areas, such as the GPIO resource block. Therefore, the clock needs to be enabled for each resource individually.

All GPIOs are controlled with the GPIO controller, which is connected via an internal bus interface. Because the GPIO may be used bi-directionally or half duplex, it is necessary to set its mode to input or output with the direction register.

For data management, we need three types of registers. First, GPIO control registers set the general GPIO configuration such as direction management. Second, data control registers manage and manipulate the data sent to the GPIO. And third, data registers contain sent or received data.

In some cases the GPIO pin is configured for output, the data output register can be written with a new value in which the set output register is masked with a 1.

If the GPIO pin is configured for input, it should be pulled up or pulled down with a pull-up or pull-down resistor.