Brian Burton

ECE 5780

2/9/2022

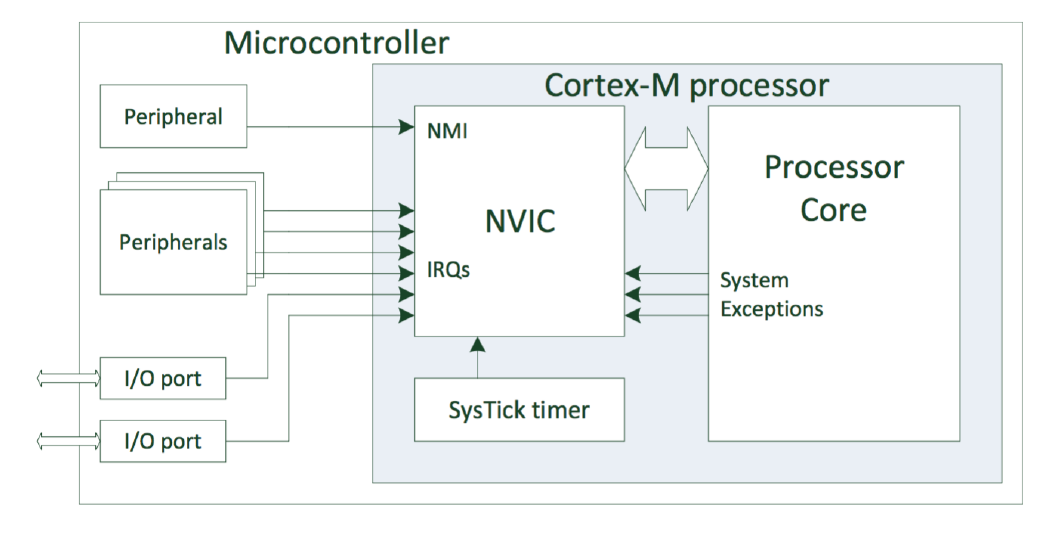
**Prelab 02 (Intro/GPIO)**

1. What is the purpose of the NVIC peripheral?

The primary responsibilities of the NVIC are enabling and disabling interrupts, indicating requests

waiting for servicing, canceling pending interrupt requests, and establishing how multiple interrupts

interact through configurable priorities.



1. What is the difference between interrupt tail-chaining and nesting?

The difference between interrupt tail-chaining and nesting is the ability/inability to interject other interrupt handlers. Tail-chaining can sometimes prevent lower interrupts from executing while nesting does not.

Tail-Chaining:

Some embedded processors have only a built-in hardware ordering between interrupts. In these systems, if multiple interrupt trigger concurrently or during a handler, they execute one after each other in succession according to the hardware priority. In this mode known as tail-chaining, interrupt handlers do not interrupt each other. Tail-chaining may use a simple save and restore mechanism for transitioning from the main thread, but it has the disadvantage of allowing a rapidly-triggering or long-running interrupt high on the hardware priority to “starve”, or prevent lower interrupts from executing.

The NVIC will tail-chain interrupts configured to the same software priority within the IPR registers. If multiple interrupts with the same software priority trigger simultaneously, the built-in hardware ordering determines the next handler to launch.

Nesting:

Unlike systems having only hardware interrupt priorities, the NVIC allows important interrupts to

interrupt lower priority handlers: this process, called nesting, requires a more complex context-switch mechanism but otherwise works identically to how interrupts pause execution of the main application thread.

Allowing nested interrupts introduces some complications: some interrupt tasks require uninterrupted processing without losing or corrupting data (e.g., interrupts which move data between communication peripherals). Many of these have limited buffer space and will overwrite data if the interrupt execution delays or pauses for too long.

1. In what file are the CMSIS libraries that control the NVIC?

The core\_cm0.h file are the CMSIS libraries that control the NVIC.

These CMSIS functions are located after the peripheral structure and register

definitions in the core\_cm0.h file.

1. What is the purpose of the EXTI peripheral?

The Extended Interrupts and Events Controller (EXTI) peripheral allows non-peripheral sources to

trigger interrupts. While its typical use is to generate interrupts from the GPIO pins of the device,

it may also monitor various internal signals such as the brownout protection circuitry (low-voltage

shutdown).

1. What is the purpose of the SYSCFG pin multiplexers?

Hello

1. What file has the defined names for interrupt numbers?

Hello

1. What file has the Vector table implementation?

Hello