

5. Event and Trigger

Young W. Lim

2016-03-18 Fri

1 Events and Triggers

“Software Engineering for Embedded Systems...”, R Oshana and M Kraeling, 2013

I, the copyright holder of this work, hereby publish it under the following licenses: GNU head Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled GNU Free Documentation License.

CC BY SA This file is licensed under the Creative Commons Attribution ShareAlike 3.0 Unported License. In short: you are free to share and make derivative works of the file under the conditions that you appropriately attribute it, and that you distribute it only under a license compatible with this one.

- real world - communicate, react
- real time - at the right time
- synchronize with event
- use trigger to work something at a given time

ISR (Interrupt Service Routine)

heavily loaded ISR can increase the interrupt latency time unless nested interrupts are allowed

- Non-nested Interrupt: during performing ISR, another interrupt must wait
 - Nested Interrupt : priority based waiting
- 1 light weight ISR
 - 2 only the work that cannot be deferred

- ① handle quickly what cannot be deferred
 - ISR sets an event flag (light weight)
- ② deferring what can wait
 - event handler process asynchronously (heavy work)

Mutual Exclusive Access to the flags

- 1 Disable/Enable Interrupts
 - simple, but does not preserve interrupt state
- 2 EnterCritical / ExitCritical
 - register saving macros, preserve the state of the interrupt
- 3 SemaphoreTake / SemaphoreGive
 - OS, might not be unavailable

- a time-triggered fashion
- a hardware functionality
- Tick timer
 - a time base : tick counter
 - compare current tick counter value with a predefined value