305341 • EMBEDDED SYSTEM 1

41 Interrupt Mechanism

นายธีรภัทร ภู่ระย้า • 66362416 • กลุ่ม 1 • ลำดับ 4

Interrupt คือการถ่ายโอนการควบคุมตามเหตุการณ์ [1][2]

กลไกที่ทำให้ CPU เปลี่ยนจากโปรแกรมปัจจุบันไปยัง ISR เมื่อคำขอผ่านเงื่อนไขการตอบรับ จากนั้นคืนบริบทและดำเนินงานต่อ ณ ตำแหน่งเดิม

EVENT-DRIVEN	เหตุการณ์สร้างคำขอแบบอะซิงโครนัส
STATE-PRESERVING	เก็บสถานะที่จำเป็นก่อนเปลี่ยนเส้นทาง
VECTORED	Vector table ระบุตำแหน่งของ handler
PRIORITY-AWARE	Enable, mask และ priority กำกับการตอบรับ

การตอบสนองแบ่งเป็น 3 ระยะต่อเนื่อง [1][2][3][4]

PHASE 01

REQUEST

1 Event / flag สร้างคำขอ

และเข้าสถานะ Pending

PHASE 02

ENTRY

2–4 Qualify → Save → Vector

ก่อนเริ่ม handler

PHASE 03

SERVICE / RETURN

5–6 ISR / acknowledge → restore

แล้ว resume โปรแกรมเดิม

ความถูกต้องของ Interrupt ขึ้นกับ State และ Latency [2][3][4]

01 • QUALIFY

ENABLE / MASK / PRIORITY

คำขอต้องผ่านเงื่อนไขการตอบรับก่อนเปลี่ยน control flow

02 • PRESERVE

CONTEXT INTEGRITY

ขอบเขตการ Save / Restore กำหนดโดยสถาปัตยกรรมและ ABI

03 • COMPLETE

BOUNDED ISR

Acknowledge แหล่งคำขอ และทำเฉพาะงานจำเป็นเพื่อลด latency และ jitter

เอกสารอ้างอิง

[1] Mazidi et al. (2011).

M. A. Mazidi, S. Naimi และ S. Naimi. The AVR Microcontroller and Embedded Systems. หน้า 375–376, Prentice Hall.

[2] Microchip Technology Inc. (2018).

ATmega328/P AVR Microcontroller Datasheet Complete. หน้า 33–34, Microchip Technology.

[3] Yiu, J. (2010).

The Definitive Guide to the ARM Cortex-M3. 2nd ed. หน้า 172–175, Newnes.

[4] Arm Ltd. (2010).

Cortex-M3 Devices Generic User Guide (ออนไลน์). สืบค้นจาก https://developer.arm.com/documentation/dui0552/a [10 สิงหาคม 2569]