

STANDBY 指令

這個指令可以使 cpu 進入待機模式，能節省電力，在 AndeStart_ISA_UM025_V1.4.pdf 裡面有說明。

節錄如下：

STANDBY (Wait For External Event)

Format:

| | | | | | | | | | | | | |
|----|----------------|-------|------------|-----|---------|------------------|---|---|---|---|---|---|
| 31 | 30 | 25 | 24 | 20 | 19 | 10 | 9 | 7 | 6 | 5 | 4 | 0 |
| 0 | MISC 110010 | 00000 | 0000000000 | 000 | SubType | STANDBY 00000 | | | | | | |

* N12 implementation for this instruction, please refer to chapter 9.2

Syntax: STANDBY SubType (= no_wake_grant, wake_grant)

Purpose: It is used for a core to enter a standby state while waiting for external events to happen.

Description: This instruction puts the core and its associating structures into an implementation-dependent low power standby mode where the instruction execution stops and most of the pipeline clocks can be disabled. The core has to enter the standby mode after all external memory and I/O accesses have been completed.

In general, the core leaves the standby mode when an external event happens that needs the core's attention. However, to facilitate the need for an external power manager to control the clock frequency and voltage, the wakeup action may need the external power manager's consent. Thus two flavors of STANDBY instruction are defined to distinguish the different usages. The SubType field definitions are listed as follows.

Table 59 STANDBY instruction SubType definitions

| SubType | Mnemonic | Wakeup Condition |
|---------|---------------|---|
| 0 | no_wake_grant | The STANDBY instruction immediately monitors and accepts a wakeup event (e.g external interrupt) to leave the standby mode without waiting for a wakeup_consent notification from an external agent. |
| 1 | wake_grant | The STANDBY instruction waits for a wakeup_consent notification from an external agent (e.g. power management unit) before monitoring and accepting a wakeup_event (e.g. external interrupt) to leave the standby mode. |
| 2 | wait_done | The STANDBY instruction waits for a wakeup_consent |

| SubType | Mnemonic | Wakeup Condition |
|---------|----------|---|
| | | notification from an external agent (e.g. power management unit). When the wakeup_consent notification arrives, the core leaves the standby mode immediately. |

The wakeup external events include interrupt (regardless of masking condition), debug request, wakeup signal, reset etc. And the instruction execution restarts either from the instruction following the STANDBY instruction or from the enabled interrupt handler which cause the core to leave the standby mode. When entering an interrupt handler, the IPC system register will have the address of the instruction following the STANDBY instruction.

An implementation can export the standby state to an external agent such as a power/energy controller to further regulate the clock or the voltage of the processor core for maximum energy savings. However, if any such clock or voltage regulation causes any core/memory state loss, software is responsible to preserve the needed states before the core enters standby mode. And if such state loss has happened, the only sensible way to bring the core into action is through a reset event.

沒有全部貼上來，看文件比較清楚。