芯来科技
NUCLEI

Nuclei™ N100 系列处理器内核简明数据手册

版权声明

联系我们

若您有任何疑问，请通过电子邮件 support@nucleisys.com 联系芯来科技。

修订历史

版本号	修订日期	修订的章节	修订的内容
1.0	2019/9/11	N/A	1. 初始版本
2.0	2020/1/11	N/A	1. 修订了若干笔误，重塑了文档章节结构

目录

版权声明 **0**

联系我们 **0**

修订历史 **1**

表格清单 **4**

图片清单 **5**

1. N100 系列内核概述 **6**

1.1. N100 系列内核特性列表 6
1.2. N100 系列内核指令集与架构 7
1.3. N100 系列内核层次结构图 7

2. N100 系列内核功能简介 **9**

2.1. N100 系列内核时钟域介绍 9
2.2. N100 系列内核电源域介绍 10
2.3. N100 系列内核接口简介 10
2.4. N100 系列内核地址空间分配 10
2.5. N100 系列内核的特权模式 11
2.6. N100 系列内核的私有设备 11
2.7. N100 系列内核的调试机制 11
2.8. N100 系列内核的中断和异常机制 12
2.9. N100 系列内核的 CSR 寄存器 12
2.10. N100 系列内核的性能计数器 12
2.11. N100 系列内核的计时器单元 13
2.11.1. 调试模式时的计时器行为 *13*
2.11.2. 正常模式时的计时器行为 *13*
2.12. N100 系列内核的低功耗机制 14
2.12.1. 进入休眠状态的时钟控制 *14*
2.12.2. 退出休眠状态的时钟控制 *14*

3. N100 系列内核接口介绍 **16**

3.1. 时钟和复位接口 16
3.2. 四线 JTAG 与两线 JTAG 调试接口 16
3.3. 外部中断接口 19
3.4. 总线接口 20
3.5. 其他功能接口 22

4. N100 系列内核配置选项介绍 24

表格清单

表 2-1 N100 内核地址空间分配 10
表 3-1 时钟和复位接口 16
表 3-2 调试接口 17
表 3-3 外部中断接口 19
表 3-4 MEM AHB-LITE 接口信号表 20
表 3-5 MEM ICB 接口信号表 21
表 3-6 其他功能接口信号表 22
表 4-1 N100 系列内核配置选项 24

图片清单

图 1-1 N100 系列内核顶层示意图 8
图 2-1 N100 内核时钟域示意图 9
图 3-1 实例 JTAG 连接方式 18
图 3-2 可控上下拉的 IO PAD 19
图 3-3 MTIME_TOGGLE_A 信号生成示意图 23

1. N100 系列内核概述

Nuclei N100 系列处理器内核（简称 N100 系列内核）是由芯来科技开发的一款全国产自主可控的商用 RISC-V 处理器内核系列，主要面向极低功耗与极小面积的场景而设计，非常适合替代传统的 8 位内核或 16 位内核升级需求、或者 ARM Cortex-M0/M0+内核，应用于数模混合、IoT 或其他超低功耗场景。

1.1. N100 系列内核特性列表

N100 系列内核的特性列表如下：

- CPU 内核（CPU Core）
  - 2 级变长流水线架构，采用一流的处理器架构设计，实现业界最小的面积与最低的成本。
  - 仅支持机器模式（Machine Mode Only）。
  - 指令和数据地址宽度为 20 位（总共可以寻址到 1MB 的内存空间）以节省面积。
- 支持指令集架构（ISA，Instruction Set Architecture）
  - N100 系列处理器核支持 32 位的 RISC-V 指令集架构，支持 RV32EC 指令子集的组合。不支持硬件的乘除法指令。
- 总线接口
  - 支持可配置的系统总线接口（支持标准的 AHB-Lite 或 ICB 接口协议），用于访问外部指令和数据。
- 调试功能
  - 支持标准的四线 JTAG 和两线 JTAG 接口。
  - 支持 RISC-V 调试标准。
  - 可配置 2 个硬件断点（Hardware Breakpoints）。
  - 支持可配置的调试器连接超时功能。
  - 支持成熟的交互式调试工具。
- 低功耗管理

- 支持 WFI（Wait For Interrupt）与 WFE（Wait For Event）进入休眠模式。
- 支持两级休眠模式：浅度休眠与深度休眠。

- 内核私有的计时器单元（Core Priviate Timer，简称 TIMER）
  - 可配置 32 比特宽的实时计时器，产生计时器中断。
- 内核私有的性能计数器单元（Core Priviate Performance Monitor）
  - 可配置 64 比特宽的周期计数器（mcycle）和指令计数器（minstret）。
- 内核私有的中断控制器（Core Priviate Interrupt Controller，简称 IRQC）
  - 支持上述的软件中断、计时器中断和存储器访问错误中断（内核自己产生）和以及中断（可配置数目）。
  - 不同的中断来源拥有不同的优先级。
  - 支持快速向量中断处理机制。
- 软件开发工具：
  - N100 系列处理器核支持 RISC-V 标准的编译工具链，以及图形化集成开发环境（Integrated Development Environment，IDE）。

1.2. N100 系列内核指令集与架构

N100 系列支持的指令集和架构详情请参见《Nuclei_N100 系列指令架构手册》。

1.3. N100 系列内核层次结构图

N100 系列内核的顶层如图 1-1 所示。Nuclei N100 系列处理器内核的组织结构主要包含如下要点：

- Core Wrapper 为整个处理器内核的顶层。除了 Core 以外，在 Core Wrapper 下面还包含了如下主要组件：
  - DEBUG：处理 JTAG 接口和相关的调试功能。

- Reset Sync：对异步复位信号进行同步。

- uCore 位于 Core 层次结构之下，为处理器内核的主体部分。
- 除了 uCore 之外，在 Core 层次结构之下还包含了如下主要组件：
  - IRQC：中断控制单元。
  - TIMER：计时器单元。
  - BIU：对外部 PPI 接口和 MEM 接口的控制。
  - Misc Ctrl：其他控制模块。

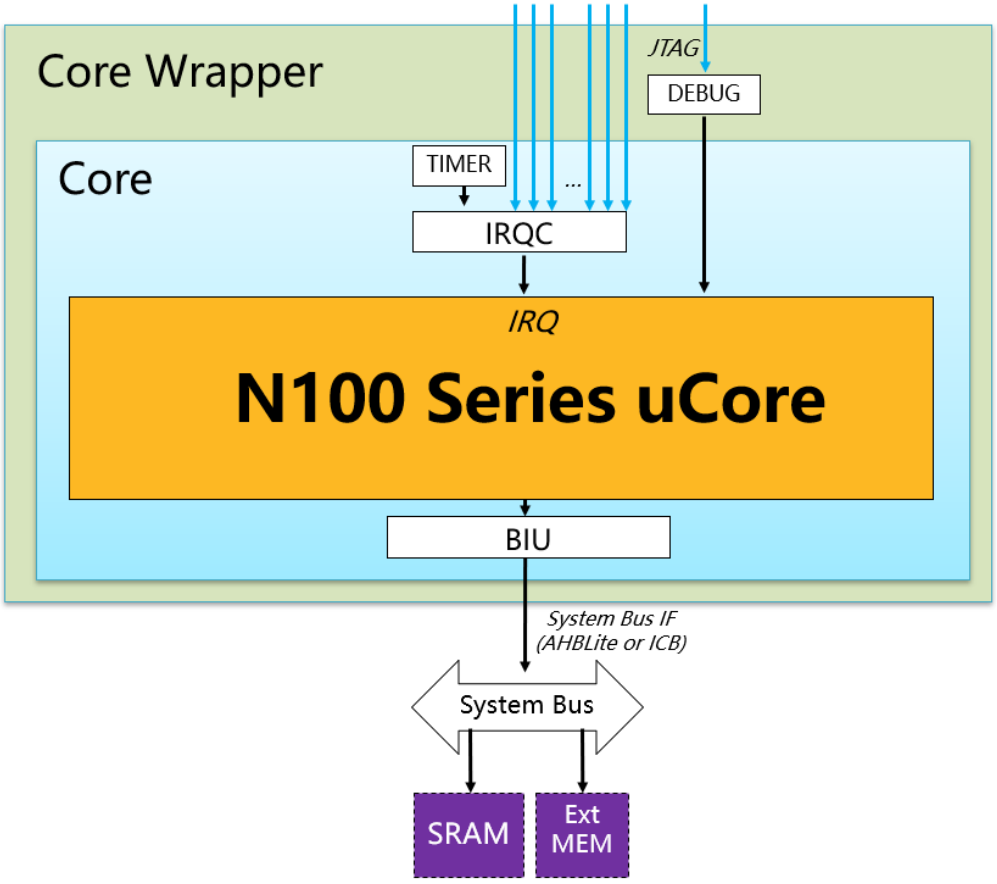

图 1-1 N100 系列内核顶层示意图

2. N100 系列内核功能简介

2.1. N100 系列内核时钟域介绍

N100 系列内核的时钟域划分如图 2-1 中所示，整个处理器内核分为两个彼此异步的时钟域：

- 工作时钟域，由输入的时钟 core_clk 和 core_clk_aon 驱动处理器内核的绝大部分功能逻辑。注意：
  - core_clk 和 core_clk_aon 为来自于同一个时钟源的同频同相时钟。
  - core_clk 为主工作时钟，驱动处理器内核内部的主要工作逻辑，并且可以在系统层面上被全局门控。
  - core_clk_aon 为常开时钟，驱动内核中的常开（Always-On）逻辑，主要包括 IRQC、TIMER 以及 DEBUG。有关 IRQC 和 TIMER 的详情请参见《Nuclei_N100 系列指令架构手册》。
- JTAG TCK 时钟域，由输入的信号 jtag_TCK 驱动处理器内核的 JTAG 调试相关逻辑。
- JTAG TMS 时钟域，由输入信号 jtag_TMS 信号驱动处理器内核的两线 JTAG 调试相关逻辑。注意：由于 N100 系列处理器内核需要用到 jtag_TMS 作为时钟，来驱动两线/四线切换的逻辑，因此，即使用户不用两线模式，也需要将 jtag_TMS 设为异步时钟。

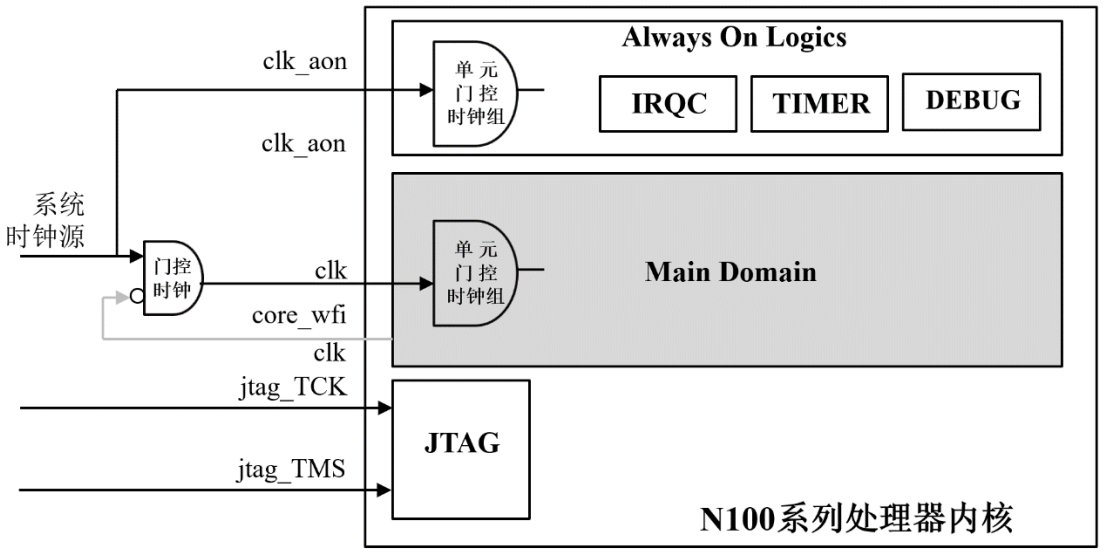

图 2-1 N100 内核时钟域示意图

上述三个时钟域之间完全异步（实际上 jtag_TMS 和 jtag_TCK 是相关的时钟，但为了时序约

束简单，可以将两者当成异步时钟来约束），在处理器内核的内部实现中已经进行了异步跨时钟域的处理，用户需要在 SoC 层面将三者约束成异步时钟。

2.2. N100 系列内核电源域介绍

N100 系列内核的内部并没有划分电源域，SoC 系统集成者可以根据 N100 系列内核的层次结构自行划分电源域和进行跨电源域处理。

2.3. N100 系列内核接口简介

Nuclei N100 系列处理器内核包含如下几类接口：

- 时钟和复位接口
- 调试接口
- 外部中断接口
- 总线接口
- 其他功能接口

请参见第 3.4 节了解接口的详细信息。

2.4. N100 系列内核地址空间分配

N100 系列内核的地址空间分配如表 2-1 中所示。

表 2-1 N100 内核地址空间分配

模块或接口	基地址	偏移地址区间	描述
DEBUG	可配置	0x000~ 0xFFF	■ DEBUG 单元的地址空间 ■ 注意： 1. DEBUG 主要用于调试器使用，普通软件程序不应该使用此区间。 2. DEBUG 单元位于 Core 内部，属于私有于 Core 的外设。
系统总线接口	N/A	N/A	■ 除了上述 DEBUG 单元的地址之外的地址区间，均会通过系统总线接口对外进行访

			问。
注：有关可配置参数的详细信息请参见本文档第 4 章。			

2.5. N100 系列内核的特权模式

N100 系列内核只支持一个特权模式（Privilege Modes），即仅支持机器模式（Machine Mode Only）。有关 Privilege Modes 的详情请参见《Nuclei_N100 系列指令架构手册》。

2.6. N100 系列内核的私有设备

如图 1-1 中所示，N100 系列内核的 Core 层次结构之下，除了 uCore 之外，还包含了如下私有设备：

- DEBUG：处理 JTAG 接口和相关的调试功能，使用存储器地址寻址方式进行访问，有关其具体的地址区间分配请参见第 2.4 节。
- IRQC：内核中断控制单元，使用 CSR 指令寻址方式进行访问。
- TIMER：内核私有计时器单元，使用 CSR 指令寻址方式进行访问。

2.7. N100 系列内核的调试机制

N100 系列内核支持 JTAG 调试接口，以及成熟的交互式调试工具（譬如：GDB、IAR，J-Link，Trace32 等）。有关调试工具详情请参见《Nuclei_N100 系列 IDE 使用说明》和《Nuclei_N100 系列 SDK 使用说明》。注意：

- N100 系列内核可以配置是否支持硬件断点（Hardware Breakpoint）。硬件断点主要用于向只读区间（譬如 Flash）设置断点。

N100 系列内核定义了一根输入信号，i_dbg_stop 可以通过其输入信号的值来进行控制：

- 如果 i_dbg_stop 信号的值为 1，则处理器内核的调试功能被关闭。
- 如果 i_dbg_stop 信号的值为 0，则处理器内核的调试功能正常工作。

2.8. N100 系列内核的中断和异常机制

有关 N100 系列的中断和异常机制详细介绍，请参见《Nuclei_N100 系列指令架构手册》。

2.9. N100 系列内核的 CSR 寄存器

RISC-V 的架构中定义了一些控制和状态寄存器（Control and Status Register，CSR），用于配置或记录一些运行的状态。CSR 寄存器是处理器核内部的寄存器，使用其专有的 12 位地址编码空间。详情请参见《Nuclei_N100 系列指令架构手册》了解其详情。

2.10. N100 系列内核的性能计数器

N100 系列内核定义了如下两种性能计数器：

- 时钟计数器（Cycle Counter）：
  - 一个 64 位宽的时钟周期计数器，用于反映处理器执行了多少个时钟周期。只要处理器处于执行状态时，此计数器便会不断自增计数。
  - CSR 寄存器 mcycle 反映了该计数器低 32 位的值，CSR 寄存器 mcycleh 寄存器反映了该计数器高 32 位的值。有关 mcycle 和 mcycleh 的详情请参见《Nuclei_N100 系列指令架构手册》。
- 指令完成计数器（Instruction Retirement Counter）：
  - 一个 64 位宽的指令完成计数器，用于反映处理器成功执行了多少条指令。只要处理器每成功执行完成一条指令，此计数器便会自增计数。
  - CSR 寄存器 minstret 反映了该计数器低 32 位的值，CSR 寄存器 minstreth 反映了该计数器高 32 位的值。有关 minstret 和 minstreth 的详情请参见《Nuclei_N100 系列指令架构手册》。

时钟计数器（Cycle Counter）和指令完成计数器（Instruction Retirement Counter）通常用

于测量性能。

默认情况下，计数器在内核复位后的值为 0，然后一直不断的自增计数。由于考虑到计数器计数会消耗某些动态功耗，因此在 N100 系列处理器内核的实现中，在自定义的 CSR 寄存器 mcountinhibit 中额外增加了若干位控制域，软件可以配置相应的控制域分别将不同的计数器关停，从而在不需要使用它们之时停止计数以达到省电的作用。

有关 CSR 寄存器 mcountinhibit 的详情请参见《Nuclei_N100 系列指令架构手册》。

2.11. N100 系列内核的计时器单元

N100 系列内核定义了一个 32 位的计时器（Timer Counter），该计时器的值实时反映在寄存器 mtime 中。N100 系列内核还定义了一个 32 位的寄存器 mtimecmp，该寄存器作为计时器的比较值，假设计时器的值 mtime 大于或者等于 mtimecmp 的值，则产生计时器中断。

在 N100 系列处理器内核的实现中，mtime/mtimecmp 均由 TIMER 单元实现，有关 N100 系列内核的 TIMER 单元详情请参见《Nuclei_N100 系列指令架构手册》。

2.11.1. 调试模式时的计时器行为

当 N100 系列内核在处于调试模式时会偶尔执行一些调试器（Debugger）设定的代码（DEBUG 单元中，对于用户透明不可见）以支持调试器的功能。如果在执行这些调试器设定的代码时计时器仍然计数，则无法真实反映被调试的程序的真实行为。因此 N100 系列内核在执行调试器设定的代码时，计时器会自动停止计数。

2.11.2. 正常模式时的计时器行为

默认情况下，计时器在内核复位后的值为 0，然后一直不断的自增计时。由于考虑到计时器计数会消耗某些动态功耗，因此在 N100 系列处理器内核的实现中，在自定义的寄存器 mstop 中设置了一个位控制域，软件可以配置该控制域将计时器关停，从而在不需要使用它们之时停止计数以达到省电的作用。有关寄存器 mstop 的详情请参见《Nuclei_N100 系列指令架构手册》。

2.12. N100 系列内核的低功耗机制

N100 内核的低功耗机制体现在如下几个方面：

- N100 系列内核内部的各个主要单元的时钟在空闲时都会自动地会被门控关闭以节省静态功耗。
- N100 系列内核能够通过常见的 WFI（Wait for Interrupt）和 WFE（Wait for Event）机制支持休眠（Sleep）模式以实现较低的动态和静态功耗，有关 “Wait for Interrupt”和“Wait for Event”的详情请参见《Nuclei_N100 系列指令架构手册》。

2.12.1. 进入休眠状态的时钟控制

N100 系列内核可以通过执行 WFI 指令进入休眠状态，有关“如何进入休眠状态”的具体详情请参见《Nuclei_N100 系列指令架构手册》。

N100 系列内核的输出信号 core_sleep_value 可以用于指示不同的休眠模式（0 或者 1），通常可以用休眠模式 0 作为浅度休眠，休眠模式 1 作为深度休眠。注意：当进入深度休眠模式之后，处理器内核将不能够再被调试接口进行调试。

处理器内核进入休眠状态时的时钟控制（参考方案）要点如下：

- 如图 2-1 中，当成功的执行了 WFI 后，N100 系列内核的输出信号 core_wfi_mode 会拉高，指示此处理器核处于执行 WFI 指令之后的休眠状态；SoC 系统层面可以使用 core_wfi_mode 控制外部的总门控时钟将处理器内核的主工作时钟 core_clk 关闭。
- 如果 N100 系列内核进入的是深度休眠模式（core_sleep_value 为 1），SoC 系统可以根据其实际情况决定是否将处内核的常开时钟 core_clk_aon 也关闭。

2.12.2. 退出休眠状态的时钟控制

处理器内核可以被中断（Interrupt）、事件（Event）唤醒，有关“如何退出休眠状态”的具体详情请参见《Nuclei_N100 系列指令架构手册》。

处理器内核退出休眠状态时的时钟控制要点如下：

- 如果是等待中断（Interrupt）的唤醒，由于 N100 系列内核的中断需要经过 IRQC 单元的处理和分发，中断只有通过了使能和优先级阈值等条件的判断之后，才能够唤醒内核。除此之外，还需特别注意处理器内核的常开时钟（core_clk_aon）是否关闭：
  - 如第 2.1 节中所述，由于 TIMER 受 core_clk_aon 驱动，因此：
    - 假设 SoC 系统层面已经将处理器内核的常开时钟（core_clk_aon）关闭，则 TIMER 单元由于无时钟，因此其无法产生计时器中断和软件中断。
  - 如第 2.1 节中所述，由于 IRQC 受 core_clk_aon 驱动，因此：
    - 假设 SoC 系统层面已经将处理器内核的常开时钟（core_clk_aon）关闭，外部中断信号线拉高之后必须一直保持，直到 SoC 系统层面将处理器内核的的常开时钟（core_clk_aon）再次打开。否则处理器内核的 IRQC 单元由于没有时钟，无法采样到外部中断信号，从而导致处理器内核无法被唤醒。
- 如果是等待事件（Event）的唤醒，则内核一旦（通过 core_clk_aon 时钟）采样到输入信号 rx_evt（Event 信号，高电平有效），便从休眠状态中被唤醒。除此之外，还需特别注意处理器内核的常开时钟（core_clk_aon）是否关闭：
  - 假设 SoC 系统层面已经将处理器内核的常开时钟（core_clk_aon）关闭，输入信号 rx_evt 拉高之后必须一直保持，直到 SoC 系统层面将处理器内核的的常开时钟（core_clk_aon）再次打开。否则处理器内核的 Event 采样逻辑由于没有时钟，无法采样到 Event，从而无法被唤醒。
- 处理器被唤醒后则会马上将输出信号 core_wfi_mode 拉低。假设 SoC 系统层面使用了 core_wfi_mode 控制内核的 core_clk 门控时钟，则随着 core_wfi_mode 信号的拉低，处理器内核的工作时钟 core_clk 将会重新被打开。

3. N100 系列内核接口介绍

3.1. 时钟和复位接口

N100 系列内核的时钟和复位接口信号如表 3-1 中所示。

表 3-1 时钟和复位接口

信号名	方向	位宽	描述
core_clk_aon	Input	1	■ 此常开时钟用于驱动 N100 系列处理器内核内部的常开逻辑（Always-On Logics）。请参见第 2.1 节了解详情。
core_clk	Input	1	■ 此工作时钟用于驱动 N100 系列处理器内核内部的主要工作逻辑。请参见第 2.1 节了解详情。
por_reset_n	Input	1	■ 上电复位信号。该信号低电平有效，此复位信号将复位整个 N100 系列处理器内核，包括 JTAG 调试部分。 ■ 注意：此信号在内核内部会进行异步时钟同步处理，即将其处理成为“异步置位同步释放”的复位信号。
core_reset_n	Input	1	■ 系统复位信号。该信号低电平有效，此复位信号将复位除了 JTAG 调试部分之外的 N100 系列处理器内核主要功能部分。 ■ 注意：此信号在内核内部会进行异步时钟同步处理，即将其处理成为“异步置位同步释放”的复位信号。
reset_bypass	Input	1	■ 如上所述，异步复位信号需要在 N100 系列处理器内核内部进行“异步置位同步释放”的处理，需要使用到“几级寄存器同步（Synchronizer）”电路。 ■ 如果输入信号 reset_bypass 为高，则将此 Synchronizer 旁路（Bypass）掉，以便于测试目的（Design For Test）。 ■ 注意：如果输入信号 reset_bypass 为高，core_reset_n 复位信号会被旁路，仅有 por_reset_n 复位信号有效。
clkgate_bypass	Input	1	■ 如上所述，N100 系列处理器内核内部会使用到门控时钟。 ■ 如果输入信号 clkgate_bypass 为高，则将时钟门控（Bypass）掉，以便于测试（Design For Test）。

3.2. 四线 JTAG 与两线 JTAG 调试接口

N100 系列内核支持标准四线 JTAG 与两线 JTAG 接口（两种模式同时支持），兼容 IEEE 1149.7 T4 Wide 标准。支持标准两线调试模式中的 Oscan0 和 Oscan1 扫描格式和标准 4 线 JTAG 调试模式。

N100 系列内核的两线调试接口信号如表 3-2 中所示。

表 3-2 调试接口

信号名	方向	位宽	描述	备注
jtag_TCK	Input	1	标准 JTAG/两线 JTAG TCK 信号。	需约束为异步时钟
jtag_TMS_in	Input	1	标准 JTAG/两线 JTAG TMS 输入信号。	支持两线 JTAG 模式，需约束为异步时钟
jtag_TMS_out	Output	1	两线 JTAG TMS 输出信号。注意，如果选择不支持两线 JTAG, jtag_TMS_out 信号不需要连到 TMS 引脚。	
jtag_BK_TMS	Output	1	两线 JTAG TMS 总线保持信号，当该信号拉高时，会使能 TMS 引脚的总线保持功能。 注意：如果选择支持两线 JTAG 接口，TMS 的引脚需要有总线保持功能，如果选择不支持两线 JTAG，jtag_BK_TMS 信号不需要连到 TMS 引脚。	该信号详细功能和实现方式将在下文描述
jtag_DRV_TMS	Output	1	两线 JTAG TMS 的输出使能信号，当 TMS 进行有效的输出之时，此使能信号为高电平，否则为低电平。 注意，如果选择不支持两线 JTAG，jtag_DRV_TMS 信号不需要连到 TMS 引脚。	
jtag_TDI	Input	1	标准 JTAG TDI 信号。注意，如果选择支持两线 JTAG 接口，在 jtag_BK_TMS 信号拉高时，TDI 引脚可以被复用成其它功能。	如果选择不支持标准四线 JTAG 调试接口时，该信号可以不引出到 SoC 顶层。
jtag_TDO	Output	1	标准 JTAG TDO 信号。注意，如果选择支持两线 JTAG 接口，在 jtag_BK_TMS 信号拉高时，TDO 引脚可以被复用成其它功能。	同上
jtag_DRV_TDO	Output	1	JTAG TDO 的输出使能信号，当 TDO 进行有效的输出之时，此使能信号为高电平，否则为低电平。	同上
jtag_dwen	Output	1	N100 系列内核调试模块中的两线调试功能指示信号，高电平有效。当调试模块进入到两线调试握手阶段或数据传输阶段时，该信号有效。	若 SoC 系统中含有其他 JTAG TAP 串联在整个 JTAG 链上，两线调试机制可能导致其他 TAP 误动作。因此，可使用该信号关闭其他 TAP，防止误动作。

两线 JTAG 调试接口实例连接如下图所示。

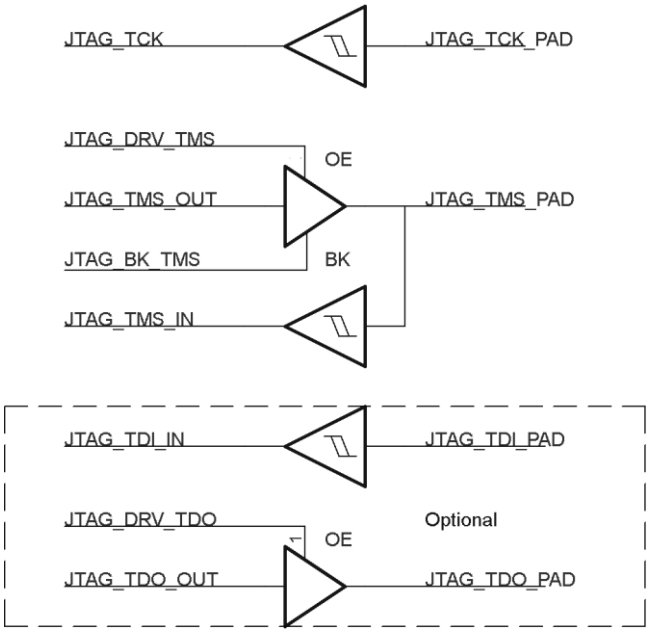

图 3-1 实例 JTAG 连接方式

其中，标注为 Optional 的部分可以不实现。若不实现，则系统仅支持两线调试模式（即兼容 IEEE 1149.7 T4 Narrow 标准），而不支持两线/四线调试混合调试模式。

在 IEEE 1149.7 规范中规定，支持两线调试功能的 TAP 要求芯片的 JTAG_TMS_PAD 有总线保持（Bus Keep）功能用于满足要求的电气特性，否则会导致通讯失败。芯来科技定制的调试器（HBird Debugger Kit）在内部已经内置总线保持器，可以支持不规范设备（芯片的 JTAG_TMS_PAD 没有 Bus Keep 功能）的两线调试。但是第三方调试器一般不会内置总线保持器。因此，如果用户希望芯片产品完整兼容 IEEE 1149.7 规范的两线 JTAG 调试功能，则建议实现含有 Bus Keep 的 JTAG_TMS_PAD。

若用户难以直接获得带有 Bus Keep 功能的 IO PAD，可通过“可控上下拉 IO PAD”实现总线保持器功能。

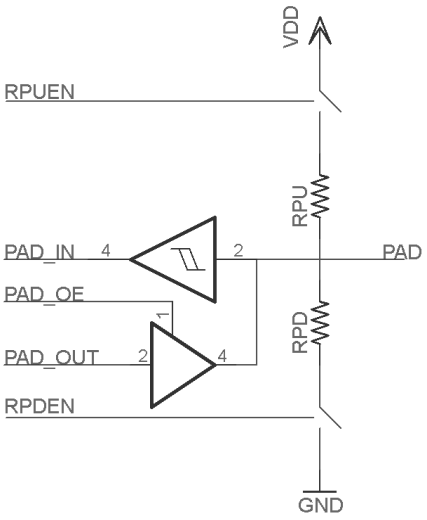

图 3-2 可控上下拉的 IO PAD

上图描述了典型的可控上下拉 IO PAD 的内部原理图，用户可以使用如下方式基于该 PAD 实现总线保持功能，以下列出实现总线保持功能的实例代码。

```
//默认上拉，当总线保持模式时，弱输出 pad_in 值，保持电平

   assign rpuen = bk_en?  pad_in : 1'b1;
   assign rpden = bk_en? ~pad_in : 1'b0;

   //注意：上述代码的 bk_en 信号可以直接来自于 jtag_BK_TMS，或者结合芯片上的其他软件可配置的使能位，譬如 bk_en = jtag_BK_TMS & ctrl_bk_enable。ctrl_bk_enable 可以是来自芯片上的某个软件可配的使能寄存器，则软件可以通过编程该寄存器强制关闭该 Bus Keep 功能。
```

3.3. 外部中断接口

N100 系列内核的外部中断接口信号如表 3-3 中所示。

表 3-3 外部中断接口

信号名	方向	位宽	描述

clic_irq	Input	可配置	■ 来自外部系统的外部中断信号，每个比特对应一个外部中断信号。 ■ 注意： 1. clic_irq 信号在处理器内核内部并没有进行异步时钟同步处理，因此，如果外部中断源与处理器内核处于异步时钟域，则系统集成者需要在外部对其进行异步时钟同步处理。 2. 有关中断机制的详细介绍请参见《Nuclei_N100 系列指令架构手册》。

3.4. 总线接口

N100 系列内核只有一个系统总线接口（System Memory Interface，简称 MEM）。

MEM 接口用于 N100 系列处理器内核访问外部的系统总线，MEM 接口可以配置为 AHB-Lite 接口或者 ICB 接口（芯来科技自定义的总线接口）。

- 当配置为 AHB-Lite 时，MEM 接口信号如表 3-4 中所示。
- 当配置为 ICB 时，MEM 接口信号如表 3-5 中所示。

表 3-4 MEM AHB-Lite 接口信号表

信号名	方向	位宽	描述
htrans	Output	2	■ AHB-Lite 协议的 HTRANS 信号。 ■ 注：在 MEM 接口，可以发出 IDLE 和 NONSEQUENTIAL 两种类型的 Transaction。
hwrite	Output	1	■ AHB-Lite 协议的 HWRITE 信号
haddr	Output	20	■ AHB-Lite 协议的 HADDR 信号
hsize	Output	3	■ AHB-Lite 协议的 HSIZE 信号。 ■ 注：在 MEM 接口，可以发出 8、16 或者 32 比特的 Transaction。
hburst	Output	3	■ AHB-Lite 协议的 HBURST 信号。 ■ 注：在 N100 系列内核的 MEM 接口，该信号的值固定是 b000。
hprot	Output	4	■ AHB-Lite 协议的 HPROT 信号。 ■ 注：在 N100 系列内核的 MEM 接口： ➢ HPROT[3]的值固定是 0（表示 Non-Cacheable）。 ➢ HPROT[2]的值固定是 0（表示 Non-Bufferable）。

			➢ HPROT[1]的值固定是 1（表示这是 Machine Mode 下的访问）。 ➢ HPROT[0]的值可以是 1（表示这是数据访问）或者是 0（表示这是取指令访问）。
hmastlock	output	1	■ AHB-Lite 协议的 HLOCK 信号。 ■ 注：在 N100 系列内核的 MEM 接口，该信号的值固定是 0。
master	Output	2	■ 该信号不是 AHB-Lite 的标准信号。 ■ 注：在 MEM 接口，此信号的值可以是 b01（表示这是 Debug-Mode 下的访问），或者是 b00（表示这是普通的数据访问），或者是 b10（表示这是普通的取指令访问）。
hwdata	Output	32	■ AHB-Lite 协议的 HWDATA 信号。
hrdata	Input	32	■ AHB-Lite 协议的 HRDATA 信号。
hresp	Input	2	■ AHB-Lite 协议的 HRESP 信号。 ■ 注：在 MEM 接口，仅支持 OKAY 和 ERROR 类型的反馈。
hready	Input	1	■ AHB-Lite 协议的 HREADY 信号。

表 3-5 MEM ICB 接口信号表

信号名	方向	位宽	描述
mem_icb_cmd_valid	Output	1	■ 如果该信号为 1，则表示是一个有效的访问。
mem_icb_cmd_ready	Input	1	■ 如果该信号为 1，则表示当前的访问被接受。
mem_icb_cmd_addr	Output	20	■ 该访问的地址信号。
mem_icb_cmd_read	Output	1	■ 如果该信号为 1，则表示该访问是读操作，否则表示写操作。
mem_icb_cmd_dmode	Output	1	■ 如果该信号为 1，则表示该访问是 Debug-Mode 下的操作。
mem_icb_cmd_wdata	Output	32	■ 该访问的写数据。
mem_icb_cmd_wmask	Output	4	■ 该访问的写数据字节使能。
mem_icb_cmd_size	Output	2	■ 该访问的数据大小： ➢ 0：字节访问； ➢ 1：半字访问； ➢ 2：字访问。
mem_icb_cmd_hprot	Output	4	■ 该访问的附加信息： ➢ mem_icb_cmd_hprot[3:2]：保留位；

			➢ mem_icb_cmd_hpro [1]的值固定是 1（表示这是 Machine Mode 下的访问）。 ➢ mem_icb_cmd_hpro [0]的值可以是 1（表示这是数据访问）或者是 0（表示这是取指令访问）。
mem_icb_rsp_valid	Input	1	■ 该访问的反馈有效信号，为 1 表示当前是有效的反馈。
mem_icb_rsp_ready	Output	1	■ 该访问的反馈接受信号。
mem_icb_rsp_err	Input	1	■ 该访问的反馈错误标志。
mem_icb_rsp_rdata	Input	32	■ 该读访问的反馈数据。

3.5. 其他功能接口

表 3-6 其他功能接口信号表

信号名	方向	位宽	描述
tx_evt	Output	1	■ N100 系列处理器内核可以通过此输出信号 txevt 产生一个单周期脉冲信号，作为对外发送的 Event 信号。请参见《Nuclei_N100 系列指令架构手册》了解 CSR 寄存器 txevt 的详细行为。
rx_evt	Input	1	■ 输入信号作为 Wait For Event 的唤醒信号，请参见第 2.12.2 节了解此输入信号的详情。请参见《Nuclei_N100 系列指令架构手册》了解 Wait For Event 机制的详情。
mtime_toggle_a	Input	1	■ 来自于 SoC 系统层面的脉冲信号，用于驱动 Core 内部 TIMER 单元的计时器。该信号通常用于 System Tick 功能。 ■ 注意： ● 该信号被当做异步输入信号。 ● 在 Core 内部会对该信号进行异步信号同步处理（使用几级寄存器进行同步）。 ● 在进行同步处理之后，会根据 Core 的主时钟对此信号进行上升沿和下降沿的检测，任何一个边沿检测到就会触发 TIMER 的计时器自增加一。 ● 建议此信号使用慢速时钟驱动的寄存器作为输入信号，处理器内核内部进行上下边沿检测后产生的自增频率即等于慢速时钟的频率，如图 3-3 中所示。慢速时钟的频率越低，则内部计时器的自增频率越低，可以降低动态功耗。
dbg_toggle_a	Input	1	■ 来自于 SoC 系统层面的 Debug 超时计数器的脉冲信号，用于防止调试器意外掉线后，Reset Halt 置位导致系统在复位后挂起的问题。 ■ 注意： ● 该信号只有在配置了调试器连接超时功能才存在。 ● 该信号被当做异步输入信号。在 Core 内部会对该信号进

			行异步信号同步处理（使用几级寄存器进行同步）。 ■ 为了达到 170-320ms 的超时时间，建议使用 25kHz~50kHz 的频率驱动该输入信号。
hart_id	Input	1	■ 该 Core 的 HART ID 指示信号，在 SoC 集成时，可以将此信号赋予某个常数值或者信号值。该 ID 号会体现在 Core 内部的 CSR 寄存器 mhartid 中。在单核情形下，可以将此信号直接接 0。
reset_vector	Input	32	■ 该输入信号用于指定处理器被 reset 后的 PC 初始值。在 SoC 层面可以通过控制此信号达到控制处理器核上电 PC 初始值的效果。
hart_halted	Output	1	■ 该输出信号如果为高电平，则指示此 Core 处于调试模式状态。
i_dbg_stop	input	1	■ 该信号如果位高电平，可以将 Core 的调试功能关闭，从而让外界的 JTAG Debugger 没法对 Core 进行调试。
ndmreset	output	1	■ 该信号是 JTAG 调试器发出的对整个 System 进行 reset 的请求。系统集成者可以用此信号 reset 整个 SoC，并且通过连接到 Core 的 core_reset_n（注意：不能够连接到 por_reset_n）达到复位 Core 的工作域的效果。
core_wfi_mode	Output	1	■ 该输出信号如果为高电平，则指示此 Core 处于执行 WFI 指令之后的 Sleep 状态。
core_sleep_value	Output	1	■ 该输出信号指示 Core 的 Sleep 模式。请参见《Nuclei_N100 系列指令架构手册》了解“进入休眠状态”的详情。

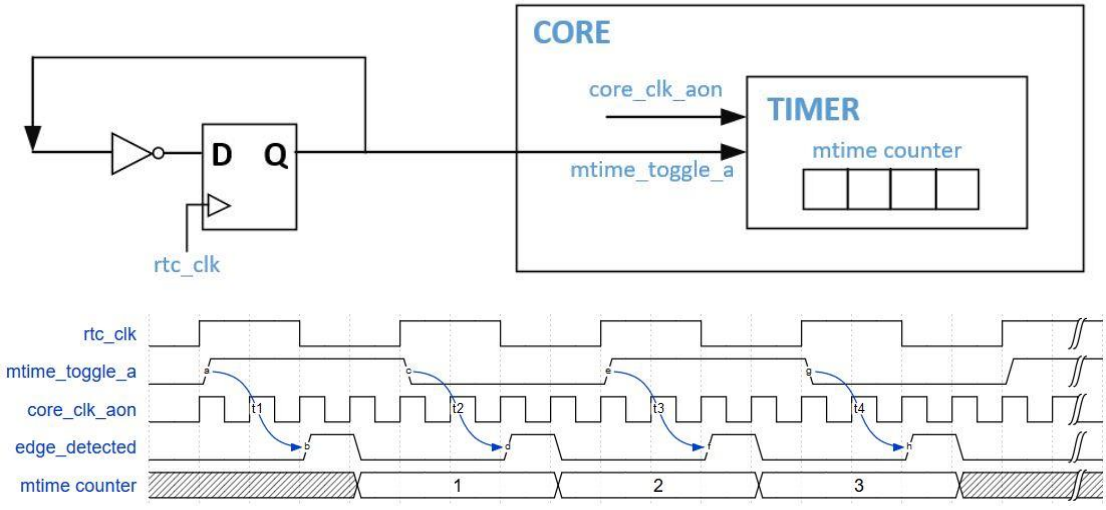

图 3-3 mtime_toggle_a 信号生成示意图

4. N100 系列内核配置选项介绍

N100 系列的每款处理器均具有一定的可配置性，具体配置选项如表 4-1 所示。如何对 N100 系列内核进行配置生成相应的 RTL 源代码，请参见《N100 系列内核快速集成手册》。

表 4-1 N100 系列内核配置选项

类别	功能	宏	描述
调试模块	是否有调试模块	N100_CFG_HAS_DEBUG	■ 如果配置了此选项，则使用调试模块。 ■ 注意：调试模块会增加大约 5K Gate 的逻辑开销。
	Debug 模块的基地址	N100_CFG_DEBUG_BASE_ADDR	■ 配置 Debug 模块的地址区间基地址。注意：Debug 模块占据 4K 的地址空间。
	是否有 Hardware Trigger	N100_CFG_DEBUG_TRIGM	■ 配置是否支持 Hardware Trigger（如果支持，则固定为 2 组） ■ 注意：每一组 Trigger 需要消耗约 128bits 寄存器的开销。
	是否支持调试器连接超时功能	N100_CFG_DEBUG_TIMEOUT	■ 如果添加了此宏，则能够在调试器意外拔除后，自动退出调试状态，使处理器内核进入正常运行模式。
	调试连接超时计数器位宽	N100_CFG_DEBUG_COUNTLEN	■ 超时时间计算公式为为 $2^{\wedge}$N100_CFG_DEBUG_COUNTLEN/(2*$f_{dbg\_toggle\_a}$)。 ■ 注意：用户需要合理配置此宏，使得超时时间控制在 170-320ms 的范围内。

TIMER 相关	是否配置私有 TIMER 模块	N100_CFG_HAS_TMR_PRIVATE	■ 如果配置了此选项，则使用私有 TIMER 模块。 ■ 注意：TIMER 模块会增加大约 1K Gate 的逻辑开销。
IRQC 相关	中断数量	N100_CFG_IRQ_NUM	■ 配置外部中断个数（1~29）
MEM 接口相关	MEM 接口类型	N100_CFG_MEM_TYPE_AHBL	■ 定义 MEM 接口的类型，配置此宏，则支持 AHB-Lite 总线接口，否则支持 ICB 总线接口。
性能计数器相关	是否配置有性能计数器	N100_CFG_HAS_PMONITOR	■ 如果配置了此选项，则具有性能计数器模块。 ■ 注意：性能计数器模块会增加大约 2K Gate 的逻辑开销。
MTVT 相关	中断向量表基地址	N100_CFG_MTVT_BASE_ADDR	■ 设置中断向量表的基地址
MTVEC 相关	异常入口地址	N100_CFG_MTVEC_BASE_ADDR	■ 设置异常入口地址