

# HYGON

海光信息技术股份有限公司

**Haiguang Information Technology Co., Ltd.**

---

## **Hygon Register Access Tool (HRAT) User Guide**

The information contained herein is for informational purposes only, and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and Chengdu Haiguang Integrated Circuit Design Co., Ltd.(Hygon) is under no obligation to update or otherwise correct this information. Hygon makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of non-infringement, merchantability or fitness for particular purposes, with respect to the operation or use of Hygon hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of Hygon's products are as set forth in assigned agreement between the parties or in Hygon's Standard Terms and Conditions of Sale.

### **Trademarks**

Hygon, the Hygon logo, and combinations thereof are trademarks of Chengdu Haiguang Integrated Circuit Design Co., Ltd. Other product names used in this publication are for identification purposes only and may be trademarks of the irrespective owners.

Reverse engineering or disassembly is prohibited.

# 目录

目录 .....	iii
版本记录 .....	iv
概述 .....	1
使用说明 .....	2
功能描述 .....	2
命令描述 .....	3
参考用例 .....	4
IO Access .....	4
Memory 操作 .....	5
PCI Access .....	6
SMBus Access .....	7
I2C Access .....	8
MSR Access .....	9
CPUID Access .....	10
SMBIOS Access .....	11
SMN Access .....	11
FWVERSION Access .....	12
GPIO Access .....	11

## 版本记录

Date	Revision	Description
Sep.2020	1.01	Initial preliminary release
Oct.2020	1.02	Fix I2C write not exist device cannot return error issue
Apr.2021	1.03	Add GPIO setting function Improve FW version function
Jun.2021	1.04	Fix SMBUS access error on Hygon CPU

## 概述

---

HRAT 提供了 UEFI 和 Linux 两种版本，可以在 UEFI Shell 或者 Linux 环境下访问 CPU 寄存器

```
HRAT(Hygon Register Access Tool) -- Rev 1.01 BUILD 2020-09-15
***** Copyright(C) All Rights Reserved. *****
Help list(Hygon):
CMD:
  /IO:   Io Access;           /MEM:   Memory Access;
  /PCI:  Pci Access;          /SMBUS: Smbus Access;
  /MSR:  MSR Access;          /SMBIOS: SMBios Access;
=====
OPT:/R1 [Addr, ...]:          Read Bit;
  /R8 [Addr, ...]:            Read Byte;
  /R16 [Addr, ...]:           Read Word;
  /R32 [Addr, ...]:           Read Dword;
  /W1 [Addr, ...] [Bit-Data]:  Write Bit;
  /W8 [Addr, ...] [Byte-Data]: Write Byte;
  /W16 [Addr, ...] [Word-Data]: Write Word;
  /W32 [Addr, ...] [Dword-Data]: Write Dword;
More details please refer UserGuide.
```

# 使用说明

## 功能描述

功能	命令	支持操作
IO 访问	-io	读/写
内存访问	-mem	读/写
PCI 访问	-pci	读/写
SMBUS 访问	-smbus	读/写
I2C 访问	-i2c	读/写
MSR 访问	-msr	读/写
CPUID 访问	-cpuid	读
SMBIOS 访问	-smbios	读
SMN 访问	-smn	读/写
FW Version	-fwversion	读
GPIO 访问	-GPIO	读/写

## 命令描述

功能	命令	说明
读数据	-r1	读 1 bit
	-r8	读 8 bit
	-r16	读 16 bit
	-r32	读 32 bit
	-r64	读 64 bit
写数据	-w1	写 1 bit
	-w8	写 8 bit
	-w16	写 16 bit
	-w32	写 32 bit
	-w64	写 64 bit
dump	-dump	Dump 多个 byte 数据
读文件	-rf	读取内容到文件或者读取文件内容，在某些特定访问中会被使用
写文件	-wf	暂未支持

搜索	-s	搜索
读取 GPIO	-rgpio	读取每个 Die 的 GPIO 所在状态
	-rmux	读取每个 GPIO 支持的 IOMUX Function
设置 GPIO	-wmux	设置 GPIO 的 IOMUX Function
	-wgpio	设置 GPIO 的状态

## 参考用例

## IO Access

HRAT 提供 IO 访问命令，可以以 byte/word/dword 位宽对 IO 进行读写访问，或者一次性 dump 多字节数据。

命令	说明
HRAT -io -w8 <i>IoAddr Value</i>	-io: IO 访问 -w8: write-8bits-width (-w16/-w32 同理) IoAddr: IO 地址 Value: 向 IO 地址中写入的数据
HRAT -io -r8 <i>IoAddr</i>	-io: IO 访问 -r8: read-8bits-width (-r16/-r32 同理) IoAddr: IO 地址
HRAT -io -index -dump <i>IoIndex IoData</i>	-io 表示访问功能为 IO 操作;



	-index 表示按照间接方式访问 IO;  -dump 表示从 <i>IoIndex/IoData</i> 组合 port 中读取多字节, 默认为 dump 256 byte。
--	---

## Memory 操作

HRAT 提供 Memory 地址访问命令, 可以以 byte/word/dword 位宽对 Memory 地址进行读写访问, 或者一次性 dump 多字节数据。

命令	说明
HRAT -mem -w8 <i>Addr Value</i>	-mem: memory 访问  -w8: write-8bits-width (-w16/-w32 同理)  Addr: memory address  Value: Value to write
HRAT -mem -r8 <i>Addr</i>	-mem: memory 访问  -r8: read-8bits-width (-r16/-r32 同理)  Addr: Address to read
HRAT -mem -dump <i>Addr Size</i>	-mem: memory 访问  -dump: 从 <i>Addr</i> 地址开始读取 <i>Size</i> 字节; <i>Size</i> 项可以缺失, 默认为 256 byte  Addr: 起始地址  Size: 一次性读取内存大小, 以字节为单位

## PCI Access

HRAT 提供 PCI 地址访问命令，可以以 byte/word/dword 位宽对 Memory 地址进行读写访问，或者一次性 dump 多字节数据。

命令	说明
HRAT -pci -w8 bus device function offset Value	-pci: PCI 地址空间访问  -w8: write-8bits-width (-w16/-w32 同理) dword 等写操作  bus: PCI bus number  device: PCI device number  function: PCI function number  offset: PCI offset  Value: 写入数据
HRAT -pci -r8 bus device function offset	-pci: PCI 地址空间访问  -r8: read-8bits-width (-r16/-r32 同理)  bus: PCI bus number  device: PCI device number  function: PCI function number  offset: PCI offset
HRAT -pci -dump bus device function	-pci: PCI 地址空间访问  -dump: 从 Bus: Device.Function 地址中读取 256 byte  bus: PCI bus number

	device: PCI device number
	function: PCI function number

## SMBus Access

HRAT 提供 SMBus 访问命令，可以以字节为单位进行读写访问，或者一次性读取多个字节，也可以搜索有效的 SMBus 地址

命令	说明
HRAT -smbus -w8 Addr Offset Value	-smbus: SMBUS 访问 -w8: write-8bits-width Addr: Slave Device Address Offset: Byte offset of slave device Value: byte to write
HRAT -smbus -r8 Addr Offset	-smbus: SMBUS 访问 -r8: read-8bits-width (-r16/-r32 同理) Address: Slave Device Address Offset: Byte offset of slave device
HRAT -smbus -dump Addr	-smbus: SMBUS 访问 -dump: 表示从 Addr 地址中读取 256 或者 512 字节 Addr: Slave Device Address
HRAT -smbus -s StartAddr EndAddr	-smbus: SMBUS 访问 -s: 搜索

	<p>StartAddr: Smbus slave device 起始地址</p> <p>EndAddr: Smbus slave device 结束地址</p> <p>如搜索成功，打印有效设备的地址</p>
--	--

## I2C Access

HRAT 提供 I2C 访问命令，可以以字节为单位进行读写访问，或者一次性读取多个字节，也可以搜索有效的 I2C 地址

命令	说明
HRAT -i2c -w8 I2cNo Addr Offset Value	<p>-i2c: I2C 访问</p> <p>-w8: write-8bits-width (-w16 表示写 word)</p> <p>I2cNo: I2C Bus number</p> <p>Addr: I2C slave device address</p> <p>Offset: Register offset of I2C slave device</p> <p>Value: Value to write 同理还支持-w16 写一个 word。</p>
HRAT -i2c -r8 I2cNo Addr Offset	<p>-i2c: I2C 访问</p> <p>-r8: read-8bits-width (-r16 表示读 word)</p> <p>I2cNo: I2C Bus number</p> <p>Addr: I2C slave device address</p> <p>Offset: Register offset of I2C slave device</p>
HRAT -i2c -s I2cNo StartAddr EndAddr	-i2c: I2C 访问

	-s: 搜索有效设备  I2cNo: I2C Bus number  StartAddr: I2C slave device 起始地址  EndAddr: I2C slave device 结束地址  如搜索成功, 打印有效设备的地址
--	---

## MSR Access

MSR 操作提供了读命令, 可以读指定 CPU core 的 MSR 或者读写所有 CPU core 的 MSR 寄存器。

命令	说明
HRAT -msr -w32 CpuNo Msr Edx Eax	-msr: MSR 访问  -w32: write-32bits-width  CpuNo: CPU core 编号  Msr: MSR 寄存器  Edx: 写 MSR 值高 32 bit  Eax: 写 MSR 值低 32 bit
HRAT -msr -r32 CpuNo Msr	-msr: MSR 访问  -r32: read-32bits-width  CpuNo: CPU core 编号  Msr: MSR 寄存器
HRAT -msr -w32 all Msr Edx Eax	-msr: MSR 访问  -w32: write-32bits-width

	all: 所有的 CPU core Msr: MSR 寄存器 Edx: 写 MSR 值高 32 bit Eax: 写 MSR 值低 32 bit
HRAT -msr -r32 all Msr	-msr: MSR 访问 -r32: read-32bits-width all: 所有的 CPU core Msr: MSR 寄存器

## CPUID Access

CPUID 操作提供了读命令，可以读指定 CPU core 的 CPUID 或者读写所有 CPU core 的 CPUID 值。

命令	说明
HRAT -cpuid -r32 CpuNo CpuidReg	-cpuid: CPUID 访问 -r32: read-32bits-width CpuNo: CPU core 编号 CpuidReg: CPUID 寄存器
HRAT -msr -r32 all CpuidReg	-cpuid: CPUID 访问 -r32: read-32bits-width all: 所有 CPU core CpuidReg: CPUID 寄存器

## SMBIOS Access

SMBIOS 操作提供了读命令，可以读取所有 Smbios type 信息或者指定 type 的信息。

命令	说明
HRAT -smbios	-smbios: SMBIOS 访问  无其他参数，表示依次暂停打印所有 SMBIOS table 的信息（回车下一个 type），如需中间不停，-smbios 后加参数 <i>log</i>
HRAT -smbios -type <i>TypeNo</i>	-smbios: SMBIOS 访问  -type: 以指定 type 方式访问 SMBIOS table 中的某个 type  TypeNo: SMBIOS type
HRAT -smbios -s <i>String</i>	-smbios: SMBIOS 访问  -s: 搜索  String: SMBIOS 中的字符串  如搜索成功，打印 String 字符串所在位置

## SMN Access

SMN 操作提供了读写命令，可以读写所有 DIE 上的允许读写的 SMN 寄存器。

命令	说明
HRAT -smn -r32 <i>DieNo</i>	-smn: SMN 访问

<i>SmnAddr</i>	-r32: read-32bits-width  DieNo: 访问的 DIE 编号  SmnAddr: 读取目标 SMN 的地址
HRAT -smn -w32 <i>DieNo</i> <i>SmnAddr Value</i>	-smn: SMN 访问  -w32: write-32bits-width  DieNo: 访问的 DIE 编号  SmnAddr: 读取目标 SMN 的地址  Value: 写入的值
HRAT -smn -dump <i>DieNo</i> <i>SmnAddr NoOfDword</i>	-smn: SMN 访问  -dump: 一次性读取多个 SMN 寄存器  DieNo: 访问的 DIE 编号  SmnAddr: 读取目标 SMN 的起始地址  NoOfDword: 一次性读取的大小, 以 dword 为单位

## FWVERSION Access

FWVERSION 操作提供了读取当前 SBIOS 里的 FW 版本或者指定目标 BIOS 文件里的 FW 版本。FW 指 MP0 和 MP1 等 FW。

命令	说明
HRAT -fwversion	-fwversion: 显示 FW 版本  无参数表示读取当前运行平台中的 SBIOS 里的 FW 版本



HRAT -fwversion -rf <i>FileName</i>	-fwversion: 显示 FW 版本 -rf: 读取指定文件 FileName: 目标 BIOS 的文件名
-------------------------------------	---

## GPIO Access

GPIO 操作提供了读写命令，可以读写所有 DIE 上的 GPIO 状态。

命令	说明
HRAT -gpio -dump	-gpio: GPIO 访问 -dump: 读取所有 GPIO 状态
HRAT -gpio -rgpio <i>GpioNum</i>	-gpio: GPIO 访问 -rgpio: 读取所有 Die 上的指定的 GpioNum 状态
HRAT -gpio -rmux <i>GpioNum</i>	-gpio: GPIO 访问 -rmux: 读取指定的 GpioNum 所支持的 IOMUX function
HRAT -gpio -wmux <i>DieNo</i> <i>GpioNum</i> <i>IomuxFuntion</i>	-gpio: GPIO 访问 -wmux: 设置 GPIO IOMUX Function DieNo: 访问的 DIE 编号

	GpioNum: 访问的 GPIO 编号 Value: 写入的 GPIO IOMUX Funtion 的值
HRAT -gpio -wgpio DieNo GpioNum GpioDirection	-gpio: GPIO 访问 -wgpio:设置 GPIO 状态 DieNo: 访问的 DIE 编号 GpioNum: 访问的 GPIO 编号 GpioDirection: 写入的 GPIO 为输入/输出。输出为 1, 输入为 0.
HRAT -gpio -wgpio DieNo GpioNum GpioDirection GpioLevel	-gpio: GPIO 访问 -wgpio:设置 GPIO 状态 DieNo: 访问的 DIE 编号 GpioNum: 访问的 GPIO 编号 GpioDirection: 写入的 GPIO 为输入/输出。输出为 1, 输入为 0. GpioLevel: 写入的 GPIO 输出的高/低电平。高电平是 1, 低电平是 0.