

# **Goldshell-HS1& HS1PLUS Serial communication protocol**

V0.5

# REVISION HISTROY

Version	Date	Description
V0.1	2020.6.12	Preliminary Version.
V0.2	2020.6.22	Add process description.
V0.3	2020.7.2	Added warning sign for fan power.
V0.4	2020.7.15	Add command to send restart and set LED.
V0.5	2020.8.27	<ol style="list-style-type: none"><li>1. Modify the receiving device information command and delete the HashRate Info field.</li><li>2. Shorten SN bytes to 18 bytes, HS1 and HS1PLUS ModelName are 16 and 20 bytes in length respectively. This change is for HS1 V0.0.4 and HS1PLUS machines.</li><li>3. The HS1 V0.0.4 software protocol version is changed to 0x20, the JOBID of HS1 V0.0.4 and HS1PLUS is set to 1-255, and the HS1 V0.0.3 is 1-15.</li></ol>

# Catalog

REVISION HISTROY .....	1
1 Overview.....	3
2 Goldshell-HS1 Protocol.....	3
2.1 Communication Protocol Introduction.....	3
2.1.1 Query device information[TX].....	3
2.1.2 Send Job data[TX].....	3
2.1.3 Set device parameters[TX].....	4
2.1.4 Send restart command[TX].....	4
2.1.5 Send LED commands[TX].....	4
2.1.6 Receive nonce[RX].....	5
2.1.7 Receiver status[RX].....	5
2.1.8 Receive device information[RX].....	6
2.1.9 Respond to the Job result command sent[RX].....	7
3 Introduction of software scheme.....	8
3.1 Finite State Machine.....	8
3.1.1 Goldshell-HS1 State transition.....	8
3.1.2 LED Indicator definition.....	8
4 Typical workflow.....	8
4.1 Specific workflow.....	8

# 1 Overview

This paper mainly introduces the communication protocol between Goldshell' small blockchain terminal equipment and PC and other upper computer terminals.

## 2 Goldshell-HS1 Protocol

### 2.1 Communication Protocol Introduction

0xA53C96	PKT...	0x69C35A
----------	--------	----------

A complete communication consists of the start verification section 0xA53C96, the transmission packet PKT, and the end section 0x69C35A. PKT includes many types, such as sending JOB packets, reading Nonce packets, etc. PKT consists of headers and data content.

#### 2.1.1 Query device information[TX]

Field name	describe	Start (byte)	length (byte)	value
Type	Packet type	0	1	0xA4
Version	Protocol version	1	1	0x10
Length	Packet length	2	4	

#### 2.1.2 Send Job data[TX]

Field name	describe	Start (byte)	length (byte)	value
Type	Packet type	0	1	0xA1
Version	Protocol version	1	1	0x10
Length	Packet length	2	4	
Target	JOB target value	6	8	0x0000ffffffffffff
StartNonce	Calculate start value	14	8	0x00000000
EndNonce	Calculate end value	22	8	0xffffffffffff
JobNum	Job number	30	1	1

<b>JobID</b>	This JOB ID number	31	1	1-255
<b>JobData</b>	This JOB data	32	128	

### 2.1.3 Set device parameters[TX]

Field name	describe	Start (byte)	length (byte)	value
<b>Type</b>	Packet type	0	1	0xA2
<b>Version</b>	Protocol version	1	1	0x10
<b>Length</b>	Packet length	2	4	
<b>Flag</b>	Set parameters or query device status	6	1	0xA2:Device parameters 0x52:Check status
<b>Voltage</b>	Set working voltage mV	7	2	
<b>Freq</b>	Set working frequency	9	2	
<b>Mode</b>	Extended Algorithm Mode	11	4	
<b>Temp</b>	Target operating temperature	15	1	65

### 2.1.4 Send restart command[TX]

Field name	describe	Start (byte)	length (byte)	value
<b>Type</b>	Packet type	0	1	0xAC
<b>Version</b>	Protocol version	1	1	0x10
<b>Length</b>	Packet length	2	4	3

### 2.1.5 Send LED commands[TX]

Field name	describe	Start (byte)	length (byte)	value
<b>Type</b>	Packet type	0	1	0xA6
<b>Version</b>	Protocol version	1	1	0x10
<b>Length</b>	Packet length	2	4	
<b>Set</b>	Whether to turn on the LED	6	1	0x0:LED off 0x1:LED on
<b>LED</b>	Set LED status	7	4	Bit[0:15]LED on time(ms) Bit[16:31]Led off time(ms)

## 2.1.6 Receive nonce[RX]

Field name	describe	Start (byte)	length (byte)	value
Type	Packet type	0	1	0x51
Version	Protocol version	1	1	0x10
Length	Packet length	2	4	
JobID	JobID corresponding to Nonce	6	1	1-15
ChipID	ChipID corresponding to Nonce	7	1	1-255
CoreID	CoreID corresponding to Nonce	8	1	
Nonce		9	8	0xffffffff
HashExist	Whether to return the hash value	17	1	1-255
Hash	Hash value	18	32	Hashvalue

## 2.1.7 Receiver status[RX]

Field name	describe	Start (byte)	length (byte)	value
Type	Packet type	0	1	0x52
Version	Protocol version	1	1	0x10
Length	Packet length	2	4	
Chips	Number of chips	6	1	
Cores	Number of chip cores	7	1	9
Goodcores	Number of good cores	8	1	9
scanbits	How many bits are in the scan range	9	1	32
scantime	Scan full space time (100ms)	10	2	
Voltage	Current working voltage(mv)	12	2	
Freq	Current working frequency(MHz)	14	2	
Mode	Current work expansion algorithm mode	16	4	

<b>Temp</b>	Equipment temperature ( °C )	20	1	
<b>Rebootcnt</b>	Chip restart times	21	1	
<b>Tempwarn</b>	Temperature warning sign	22	1	
<b>Fanwarn</b>	Fan warning sign	23	1	
<b>Powerwarn</b>	Power warning sign	24	1	
<b>Rpm</b>	Speed of the fan	25	2	

## 2.1.8 Receive device information[RX]

HS1 firmware version V0.0.3:

Field name	describe	Start (byte)	length (byte)	value
<b>Type</b>	Packet type	0	1	0x54
<b>Version</b>	Protocol version	1	1	0x10
<b>Length</b>	Packet length	2	4	
<b>ModelName Length</b>	ModelName effective length	6	1	<b>10</b>
<b>ModelName</b>	ModelName	7	16	Goldshell-HS1
<b>FWV Length</b>	Firmware version length	23	1	5
<b>FirmwareVersion</b>	Current firmware version	24	8	0.0.3
<b>SN</b>	Bar code	33	32	Byte[0]:Effective length Byte[1:31]:Bar code
<b>HashRate Info</b>	HashRate information	65	32	Byte[0]:Effective length Byte[1:31]:information
<b>Workdepth</b>	Work depth	97	1	

HS1 firmware version V0.0.4:

Field name	describe	Start (byte)	length (byte)	value
<b>Type</b>	Packet type	0	1	0x54
<b>Version</b>	Protocol version	1	1	0x20
<b>Length</b>	Packet length	2	4	
<b>ModelName Length</b>	ModelName effective length	6	1	<b>10</b>
<b>ModelName</b>	ModelName	7	16	Goldshell-HS1
<b>FWV Length</b>	Firmware version length	23	1	
<b>FirmwareVersion</b>	Current firmware version	24	8	0.0.4

<b>SN</b>	Bar code	33	18	Byte[0]:Effective length Byte[1:31]:Bar code
<b>Workdepth</b>	Work depth	51	1	

HS1PIUS:

Field name	describe	Start (byte)	length (byte)	value
<b>Type</b>	Packet type	0	1	0x54
<b>Version</b>	Protocol version	1	1	0x10
<b>Length</b>	Packet length	2	4	
<b>ModelName Length</b>	ModelName effective length	6	1	
<b>ModelName</b>	ModelName	7	20	Goldshell-HS1-Plus
<b>FWV Length</b>	Firmware version length	23	1	
<b>FirmwareVersion</b>	Current firmware version	24	8	0.0.1
<b>SN</b>	Bar code	33	18	Byte[0]:Effective length Byte[1:31]:Bar code
<b>Workdepth</b>	Work depth	54	1	

## 2.1.9 Respond to the Job result command sent[RX]

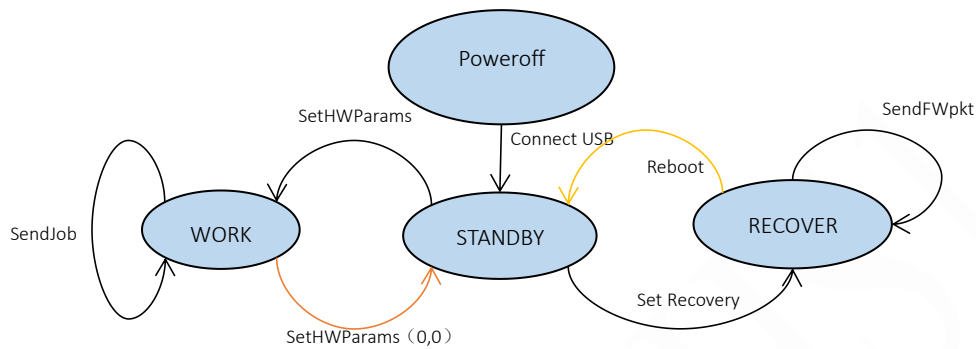
Field name	describe	Start (byte)	length (byte)	value
<b>Type</b>	Packet type	0	1	0x55
<b>Version</b>	Protocol version	1	1	0x10
<b>Length</b>	Packet length	2	4	



## 3 Introduction of software scheme

### 3.1 Finite State Machine

#### 3.1.1 Goldshell-HS1 State transition



#### 3.1.2 LED Indicator definition

status	Led status description	Remarks
STANDBY	Flash (on)	
WORK	Flash (on 300ms, off 100ms)	
RECOVERY	Flash (on 300ms, off 100ms)	
EXCEPTION	Flash (on 100ms, off 1s)	

## 4 Typical workflow

### 4.1 Specific workflow

1.Send quiry device information packet (A4), wait for receiving device information

packet (54), check whether the USB channel is connected

**send:**

A53C96A4100600000069C35A

**HS1 V0.0.3 version receives byte stream:**

A53C965410610000000D476F6C647368656C6C2D48533100000005302E302E33  
0000000E483130423032373245393045394400000000000000000000000000000000  
00  
0000000469C35A

**HS1 V0.0.4 version receives byte stream:**

A53C965420330000000D476F6C647368656C6C2D48533100000005302E302E340000000  
E48313042303237324539304539440000000469C35A

**HS1PLUS receive byte stream:**

A53C9654103700000012476F6C647368656C6C2D4853312D506C7573000005302E302E3  
20000000E48313142303335303339454144360000000869C35A

2. Send setting device parameter packet (A2), wait for receiving device status packet (52)

**send:**

A53C96A21010000000A2EE026400040000005069C35A

**receive:**

A53C9652101700000002280020FFFFEE0264000400000000000069C35A

When the Flag field of the device parameter packet is A2, it means the device parameter is set. At this time, it is necessary to set the device voltage and frequency to a non-zero value.

The device will run. When it is 52, it is to query the device parameters. At this time, the current device status package is returned. When the voltage and frequency are set to 0, the device will stop working and return to the standby state.

3. Send JOB data packet (A1), waiting to receive response packet (55)

**send:**

A53C96A1109C0000006F00  
FF010C00  
2B42056662B7FC0000000000000005B20195AFE280A27276D4517C7F8  
0F5A61843EAC78B1AF2C82962B1C413636E80378E4A3E33D2165220B  
27A03B7F0EB30BB515B160961657AF5B9F8AF4996C2566621ECD0F71  
C6A77CF80C2E6B6E816AAE44D07D91830CAA0769C35A

**receive:**

A53C9655100A000000000000000069C35A

The total amount of JOB packets sent is the number of workdepth in the first step of querying the device information, then send in sequence, After receiving the response packet, send the next JOB package until the end of sending. You cannot send multiple JOB packages at the same time before receiving the corresponding package. Set the upper 32 bits of startnonce as ntime value. The JOB scanning cycle is to obtain the scantime in the device parameter value. When the scantime arrives, the new work is updated to avoid repeated calculation. JOBID is 1-255(HS1 V0.0.3 is 1-15). Assuming that workdepth is 4, the JOBID of 4 consecutive JOBs are updated in turn with 4 different consecutive IDs in each group. For example, this JOBID is 1, 2, 3, 4, and the next one is 5, 6, 7, 8. When it comes to 255, the next ID starts from 1. The received nonce can judge which JOB is generated by the current JOB via the ID number if the workdepth is 8, et cetera.

#### 4. Polling to receive NONCE packets(51)

A53C965110120000000C0103B7F2920200000000069C35A