HC-05 - AT Command Set					
AT Command is case-sensitive, should end up with terminator “\r\n” (Carriage Return and New Line).					
Function	**AT Command**	**Response**	**Parameter**	**Observations**	**Examples**
Test	AT	OK	None	-	*AT* OK
Reset	AT+RESET	OK	None	-	*AT+RESET* OK
Get the software version	AT+VERSION?	+VERSION:Param1 OK	Param1: Version number	-	*AT+VERSION\r\n* +VERSION:2.0-20100601 OK
Restore default status	AT+ORGL	OK	None	The parameter of default status: 1. Device type: 0 2. Inquire code: 0x009e8b33 3. Module work mode: Slave Mode 4. Connection mode: Connect to the Bluetooth device specified 5. Serial parameter: Baud rate: 38400 bits/s; Stop bit: 1 bit; Parity bit: None. 6. Passkey: “1234” 7. Device name: “H-C-2010-06-01”	*AT+ORGL* OK
Get module bluetooth address	AT+ADDR?	+ADDR:Param1 OK	Param1: Bluetooth address	Bluetooth address will show as this way: NAP: UAP: LAP(Hexadecimal)	*Module Bluetooth address: 12:34:56:ab:cd:ef.* *AT+ADDR?\r\n* +ADDR:1234:56:abcdef OK
Set/Inquire device’s name	AT+NAME=Param1	OK	Param1: Bluetooth device name	Length up to 32 bytes; Supports special characters; AT+NAME="HC-05" Is the same as AT+NAME=HC-05 Default: “HC-05”	*Set the module device name to "HC-05":* *AT+NAME=HC-05\r\n* *OK* *AT+NAME?\r\n* +NAME:HC-05 OK
	AT+NAME?	*If Success:* +NAME:Param1 OK *If Failure*: FAIL			
Inquire remote bluetooth device’s name	AT+RNAME?Param1	*If Success:* +NAME:Param2 OK *If Failure:* FAIL	Param1: Remote Bluetooth device address Param2: Remote Bluetooth device name	Bluetooth address will show as this way: NAP:UAP:LAP (Hexadecimal)	*Bluetooth device address: 00:02:72:od:22:24; device name: Bluetooth.* *AT+RNAME?0002,72,od2224\r\n* +RNAME:Bluetooth OK
Set/Inquire module role	AT+ROLE=Param1	OK	Param1: module role: 0 -> Slave 1 -> Master 2 -> Slave-Loop	*Role introduction:* *Slave:* Passive connection; *Slave-Loop:* Passive connection, receive the remote Bluetooth master device data and send it back to the master device; *Master:* Inquire the near SPP Bluetooth slave device, build connection with it positively, and build up the transparent data transmission between master and slave device. Default: 0	*Set the module device role to Slave:* *AT+ROLE*=0 O*K* *AT+RO*LE? +ROLE=0 OK
		+ROLE:Param1 OK			
Set/Inquire device type	AT+CLASS=Param1	OK	Param1: Device type	Bluetooth device type is a 32-bit parameter indicates the device type and what type can be supported. For inquiring the custom Bluetooth device from around Bluetooth devices quickly and effectively, user can set the module to be non-standard Bluetooth device type, such as 0x1f1f (Hexadecimal). More information is provided at the appendix 1(device type introduction). Default: 0	-
	AT+ CLASS?	*If Success:* + CLASS:Param OK *If Failure:* FAIL			

Set/Inquire inquire access code	AT+IAC=Param1	*If Success:* OK *If Failure:* FAIL	Param1: Inquire access code	Access code is set to be GIAC type (General Inquire Access Code:0x9e8b33), and used for seeking (or being sought by) all the Bluetooth devices around. For inquiring (or being inquiring by) the custom Bluetooth device from around Bluetooth devices quickly and effectively, user can set the inquire access code to be the other type number (not GIAC nor LIAC), such as 9e8b3f. The more information is provided at the appendix 2(Inquire access code introduction). Default: 9e8b33	*AT+IAC=9e8b3f*\r\n OK *AT+IAC?*\r\n +IAC: 9e8b3f OK
	AT+ IAC?	+IAC: Param1 OK			
Set/Inquire inquire access mode	AT+INQM=Param1, Param2, Param3	*If Success:* OK *If Failure:* FAIL	Param1: Inquire access mode: 0 -> Standard 1 -> RSSI Param2: The maximum of Bluetooth devices response Param3: The maximum of limited inquiring time	The range of limited time: 1~48 (Corresponding time:1.28s~61.44s) Default: 1, 1, 48	*Set Inquire access mode: 1) has RSSI signal intensity indicator, 2) stop inquiring once more than 9 devices response, 3) limited time is 48*1.28=61.44s:* *AT+INQM=1,9,48*\r\n OK *AT+INQM*\r\n +INQM:1, 9, 48 OK
	AT+ INQM?	+INQM:Param1,Param2,Param3 OK			
Set/Inquire passkey	AT+PSWD=Param1	OK	Param1: Passkey	Default: “1234”	*Set the module device passkey to "1234":* *AT+PWD=1234*\r\n *(or AT+PSWD="1234"*\r\n*)* OK AT+PSWD?\r\n +PSWD:1234 OK
	AT+ PSWD?	+PSWD:Param1 OK			
Set/Inquire serial parameter	AT+UART=Param1, Param2, Param3	OK	Param1: Baud rate: 4800 -> 4800 bits/s 9600 -> 9600 bits/s 19200 -> 19200 bits/s 38400 -> 38400 bits/s 57600 -> 57600 bits/s 115200 -> 115200 bits/s 230400 -> 230400 bits/s 460800 -> 460800 bits/s 921600 -> 921600 bits/s 1382400 -> 1382400 bits/s Param2: Stop bit: 0 -> 1 bit 1 -> 2 bits Param3: Parity bit: 0 -> None 1 -> Odd parity 2 -> Even parity	Default: 9600, 0, 0	*Set baud rate to be 115200, stop bit to be 2 bits, parity bit to be even parity:* *AT+UART=115200,1,2,*\r\n OK *AT+UART?* +UART:115200,1,2 OK
	AT+UART?	+UART=Param1, Param2, Param3 OK			
Set/Inquire connection mode	AT+CMODE=Param1	OK	Param1: Connection mode: 0 -> Connect the module to the specified Bluetooth address. (Bluetooth address can be specified by the binding command) 1 -> Connect the module to any address (The specifying address has no effect for this mode.) 2 -> Slave-Loop	Default connection mode: 0	*Set the module device connection mode to Slave-Loop:* *AT+CMODE=2* OK *AT+CMODE?* +CMODE:2 OK
	AT+ CMODE?	+CMODE:Param1 OK			

Set/Inquire - bind Bluetooth address	AT+BIND=Param1	OK	Param1: Bluetooth address: needed to be bind	Bluetooth address will show as this way: NAP: UAP:LAP(Hexadecimal) This command is effective only when the module wants to connect to the specified Bluetooth address. Default: 00:00:00:00:00:0	*The module is at connection mode which connects to specified Bluetooth address, and the specified address is 12:34:56:ab:cd:ef.* *AT+BIND=1234, 56, abcdef\r\n* OK *AT+BIND?\r\n* +BIND:1234:56:abcdef OK
	AT+ BIND?	+BIND:Param1 OK			
Set/Inquire - drive indication of LED and connection status	AT+POLAR=Param1,Param2	OK	Param1: PI08 mode: 0 -> PI08 outputs low level and turn on LED 1 -> PI08 outputs high level and turn on Param2: PI09 mode: 0 -> PI09 output low level indicate successful connection 1 -> PI09 output high level indicate successful connection	Default: 1, 1	*PI08 outputs low level and turn on LED, PI09 outputs high level and indicates successful connection* *AT+POLAR=0, 1\r\n* OK *AT+POLAR?\r\n* +POLAR=0, 1 OK
	AT+ POLAR?	+POLAR=Param1, Param1 OK			
Set PIO single port output	AT+PIO=Param1,Param2	OK	Param1: PIO port number(Decimal): Param2: PIO port status: 0 -> low level 1 -> high level	HC-05 Bluetooth module provides the user with the ports (PI00~PI07 and PI010) which can extern another input and output ports.	*1. PI010 port outputs high level:* *AT+PI0=10, 1\r\n* OK *2. PI010 port outpust low level:* *AT+PI0=10, 0\r\n* OK
Set PIO multiple port output	AT+MPIO=Param1	OK	Param1: Mask combination of PIO ports number (Decimal)	(1) Mask of PIO port number = (1<<port number) (2) Mask combination of PIO ports number= (PIO port number mask 1|PIO port numbermask 2|.....) Example : PI02 port number mask=(1<<2) =0x004 PI010 port number mask =(1<<10)=0x400 Mask combination of PI02 and PI010 port number=(0x004|0x400)=0x404	*1. PI010 and PI02 ports output high level:* *AT+MPI0=404\r\n* OK *2. PI04 port output high level:* *AT+PI0=004\r\n* OK *3. PI010 port output high level:* *AT+PI0=400\r\n* OK *4. All ports output low level:* *AT+MPI0=0\r\n* OK
Inquire PIO port input	AT+MPIO?	+MPIO: <Param> OK	Param1: PIO port value (16bits) Param[0]=PI00 Param[1]=PI01 Param[2]=PI02 ... Param[10]=PI010 Param[11]=PI011	-	-

Set/ Inquire page scan and inquire scan parameter	AT+IPSCAN=Param1, Param2, Param3, Param4	OK	Param1:time interval of inquiring Param2: duration in inquiring Param3: time interval of paging Param4: duration in paging The above parameters are decimal.	Default:1024,512,1024,512	*AT+IPSCAN=1234,500,1200,250*\r\n OK *AT+IPSCAN?* +IPSCAN:1234,500,1200,250 OK
	AT+IPSCAN?	+IPSCAN:Param1, Param2, Param3, Param4			
Set/ Inquire—SNIFF energy parameter	AT+SNIFF=Param1, Param2, Param3, Param4	OK	Param1: maximum time Param2: minimum time Param3: test time Param4: limited time The above parameters are decimal.	Default : 0,0,0,0	-
	AT+SNIFF?	+SNIFF: Param1, Param2, Param3, Param4			
Set/ Inquire safe and encryption mode	AT+SENM=Param1, Param2	*If Success:* OK *If Failure:* FAIL	Param1: the value of safe mode: 0 -> sec_mode0+off 1 -> sec_mode1+non_secure 2 -> sec_mode2_service 3 -> sec_mode3_link 4 -> sec_mode_unknown Param2: the value of encryption mode: 0 -> hci_enc_mode_off 1 -> hci_enc_mode_pt_to_pt 2 -> hci_enc_mode_pt_to_pt_and_bcast	Default: 0,0	-
	AT+ SENM?	+SENM:Param1,Param2 OK			
Delete authenticated device in the Bluetooth pair list	AT+RMSAD=Param1	OK	Param1: Bluetooth device address	-	*Delete the device (12:34:56:ab:cd:ef) in the blue pair list:* *AT+RMSAD=1234,56,abcdef*\r\n *If Success*: OK *If Failure*: FAIL
Delete all authenticated devices in the pair list	AT+RMAAD	OK	None	-	*Move all devices away from the pair list:* *AT+RMAAD*\r\n OK
Seek the authenticated device in the Bluetooth pair list	AT+FSAD=Param1	*If Success:* OK *If Failure:* FAIL	Param1: Bluetooth device address	-	*Seek the authenticated device (address: 12:34:56:ab:cd:ef) in the pair list* AT+FSAD=1234,56,abcdef\r\n *If Success*: OK *If Failur*e: FAIL
Get the authenticated device count from the pair list	AT+ADCN?	+ADCN:Param1 OK	Param1: Authenticated Device Count	-	*AT+ADCN?* +ADCN:0 OK
Get the Bluetooth address of Most Recently Used Authenticated Device	AT+MRAD?	+MRAD:Param1 OK	Param1: the Bluetooth address of Most Recently Used Authenticated Device	-	*AT+MRAD?* +MRAD:0:0:0 OK

Get the work status of Bluetooth module	AT+STATE?	+STATE:Param1 OK	Param1: work status of module: “INITIALIZED” -> initialized status “READY” -> ready status “PAIRABLE” -> pairable status “PAIRED” -> paired status “INQUIRING” -> inquiring status “CONNECTING” -> connecting status “CONNECTED” -> connected status “DISCONNECTED” -> disconnected status “NUKNOW” -> unknown status	-	AT+STATE? +STATE:INITIALIZED OK
Initialize the SPP profile lib	AT+INIT	*If Success:* OK *If Failure:* FAIL	None	-	*AT+INIT\r\n* OK
Inquire Bluetooth device	AT+INQ	+INQ:Param1, Param2, Param3 ... OK	Param1: Bluetooth address Param2: Device type Param3: RSSI signal intensity	-	*AT+INIT\r\n* OK *AT+IAC=9e8b33\r\n* OK *AT+CLASS=0\r\n* OK *AT+INQM=1,3,48\r\n* OK *AT+INQ\r\n* +INQ:2:72:D2224,3E0104,FFBC +INQ:1234:56:0,1F1F,FFC1 +INQ:1234:56:0,1F1F,FFC0 OK
Cancel Bluetooth device	AT+INQC	OK	-	-	-
Set pair	AT+PAIR=Param1, Param2	*If Success:* OK *If Failure:* FAIL"	Param1: Bluetooth address of remote device Param2: limited time of connection (second)	-	*Make pair with the remote Bluetooth device(address:12:34:56:ab:cd:ef), the limited time is 20s:* *AT+PAIR=1234,56,abcdef,20\r\n* OK
Connect device	AT+LINK=Param1	*If Success:* OK *If Failure:* FAIL"	Param1: Bluetooth address of remote device	-	*Connect with the remote Bluetooth device (address: 12:34:56:ab:cd:ef):* *AT+FSAD=1234,56,abcdef\r\n* OK *AT+LINK=1234,56,abcdef\r\n* OK

Disconnection	AT+DISC	*If Successful Disconnection*: +DISC:SUCCESS OK *If Lose the connection*: +DISC:LINK_LOSS OK *If No SLC connection* +DISC:NO_SLC O*K* *If Disconnection Timeout* +DISC:TIMEOUT OK *If Disconnection Error* +DISC:ERROR OK	None	-	-
Enter to energy mode	AT+ENSNIFF=Param1	OK	Param1: Bluetooth address of device	-	-
Exit energy mode	AT+EXSNIFF=Param1	OK	Param1: Bluetooth address of device	-	-