



众山DTU脚本编程手册

众联万物 智慧未来
我们用心创造

目录

前 言.....	3
版权声明	3
版本信息.....	3
相关文档.....	3
一、众山 DTU 自动采集脚本简介.....	4
二、脚本详解.....	4
2.1 脚本格式.....	4
2.2 脚本指令详解.....	5
2.2.1 @E 命令回传指令.....	5
2.2.2 @C 执行命令指令 (HEX 格式)	5
2.2.3 @CS 执行命令指令 (字符格式)	6
2.2.4 @D 延时控制指令.....	6
2.2.5 @W 脚本执行期间接收中心下发指令.....	6
2.2.6 @A 接收应答指令.....	7
2.2.7 @S 立即上报指令.....	7
2.2.8 @H 数据标识指令 (HEX 格式)	8
2.2.9 @STR 数据标识指令 (字符格式)	8
2.2.10 @DOx 继电器开关控制指令.....	8
2.2.12 @GPS 采集 GPS 定位指令.....	9
2.2.13 @TS 获取 NTP 时间指令.....	11
2.2.14 定时脚本.....	11
2.2.15 @PUB json 发布脚本.....	11
2.2.16 @获取设备 CSQ 指令.....	12
2.2.17 @获取设备 ICCID 指令.....	12
2.2.18 @获取设备 IMEI 指令.....	12
2.2.19 @获取设备 IMSI 指令.....	12
2.2.20 @LBS 基站定位.....	12
2.2.21 @WIFI 定位.....	13
三、应用案例.....	13

前 言

感谢您使用成都众山科技有限公司提供的DTU产品。

本手册主要介绍众山 DTU 本地脚本编程指令。

适用型号：LTE-xxx、LTE-xxx-P 4G DTU

版权声明

本手册版权属于成都众山科技有限公司，任何人未经我公司书面同意复制将承担相应法律责任。

版本信息

文档名称：众山DTU脚本编程手册

版本：1.06

修改日期： 2021 年 7 月 16 日

相关文档

《众山 LTE-xxx 4G DTU 用户手册》

《众山 LTE-xxx 4G DTU 串口及远程控制协议》

一、众山DTU自动采集脚本简介

在很多应用中，用户设备是被动仪表如 Modbus 设备，需要发送命令进行数据采集，众山 DTU 针对这类应用开发了一种脚本语言，用户可以使用脚本语言自定义用户设备的采集行为。

众山 DTU 支持用户自定义脚本实现自动采集外接传感器、仪表数据，无需用户再单独增加控制器。通过配置 DTU 的脚本指令实现 DTU 定时自动采集，用户只需知道外接仪表、传感器的采集流程，然后通过编写脚本指令即可让 DTU 按照用户的流程自动采集、上传数据。脚本实现了基本的开关控制、开关量状态上报、延时控制、指令下发、插入自定义报文头、是否接收仪表应答、等功能，能满足大部分仪表的采集需求。脚本可以周期执行，或者通过中心下发 SHELL 指令临时执行一次脚本。

此文档定义的脚本指令主要是为了实现 DTU 自动采集功能，在某种情况下，用户可能需要数据中心远程控制采集或者干预一些开关动作，众山 DTU 也支持用户的数据中心通过“SHELL:+脚本”或者“AT+SHELL=”的形式发送给 DTU，DTU 收到此数据会临时执行一次数据中心下发的脚本来实现远程采集和控制。如数据中心下发：SHELL:@DO1=1@C=0102@DO1=0，DTU 收到 SHELL:开始的数据不会透传给 DTU 串口，而是执行一次脚本。数据中心也可以下发 STARTSHELL 让 DTU 立即启动本地脚本执行。

众山 DTU 具有的所有控制功能和 Modbus 协议都可以用在脚本的@C 命令中，让脚本不仅实现采集外接仪表的功能，还能实现控制 DTU 自己。

脚本执行和程序一样，从开始顺序执行，有循环指令时才会改变执行顺序，直到脚本执行完毕退出。默认情况下，在1次脚本执行中，收到的DTU响应或外接串口仪表响应数据时在脚本执行完毕才会打包上传，用户可以使用@S=1在脚本执行中用户需要的地方立即上传当前接收到的数据。

二、脚本详解

2.1 脚本格式

@cmd=value

@: 脚本头，每一条脚本指令都使用@开始。

cmd: 指令，为 1 或多个字符、数字组成的字符串，不区分大小写，下面详解每一条脚本指令的含义。

value: 指令动作，指定指令需要执行的值。

2.2 脚本指令详解

2.2.1 @E 命令回传指令

指令	名称	格式	含义	默认值
@E	命令回传	@E=1 @E=0	控制@C 和@O 定义的命令是否需要回传中心 @E=1 表示回传 @E=0 表示不回传	@E=0

E: 命令回传，此脚本指令控制@C 和@O 脚本指令定义的采集命令是否需要回传给中心，value 为 1 表示需要回传，value 为 0 表示不回传。不使用此指令时默认为不回传。此指令为整个脚本全局生效指令，直到脚本结束。如果有些采集指令需要回传有些不需要回传只需要在相应的采集指令前加入此指令做相应的控制即可。

示例：@E=1 回传采集命令

@E=0 不回传采集命令

2.2.2 @C 执行命令指令（HEX 格式）

指令	名称	格式	含义
@C	执行命令	@C=HEX @C=HEX+V1	定义执行命令，等效于中心下发数据，命令输入采用 HEX 格式

C: 执行命令，@C 定义的命令 DTU 首先会判断是不是 AA55 开始的控制自己的指令（参考控制协议手册），如是，则执行命令的功能，DTU 响应的数据会进入发送队列，如不是，DTU 则会输出命令到串口作为外接仪表的采集命令。当命令为 Modbus 指令时，可以在命令最后使用 V1 让 DTU 自动计算 CRC 校验并跟在命令的结尾，省去了用户需要自己计算校验的麻烦。此脚本指令定义的采集命令内容为可见的 HEX 字符。

示例：Modbus 采集指令：01 03 00 01 00 01 12 34

脚本表示为：@C=0103000100011234@D=1

采集外接传感器的 Modbus 指令并自动计算 CRC 校验：64 03 00 11 00 01

脚本表示为：@C=640300110001V1@D=1

命令执行或者下发后，DTU 自己的响应命令或者仪表返回的数据，DTU 放入发送队列，根据定义的发送控制适当的时候进行上传。

注：因为传感器应答需要一定时间，所以@C 指令后面必须加个@D 延时等待传感器应答

比如@C=010300000001V1@D=1

2.2.3 @CS 执行命令指令（字符格式）

指令	名称	格式	含义
@CS	执行命令	@CS=STR	定义执行命令，等效于中心下发数据，字符格式，功能同 @C，不支持自动校验。字符内容不能出现 '@' 和 '=' 字符，其他字符都支持

CS：此指令功能和 @C 相同，@C 命令输入的是 HEX 格式命令，@CS 输入的命令采用文本格式，适合于用户仪表为字符通讯协议的应用，免去用户转换成 HEX 格式命令的麻烦。

2.2.4 @D 延时控制指令

指令	名称	格式	含义
@D	延时控制	@D=X @D=XY	在此处 DTU 会等待指令所定义的延时，然后再继续执行脚本。纯数字(X)单位默认为秒；也可以数字(X)+单位(Y)的形式，支持的单位有： 'S'：秒 'M'：分钟 'H'：小时 'm'：毫秒

D：延时控制，此脚本指令用于控制延时，DTU 开关仪表电源后等待仪表稳定的延时或者发送采集命令后等待应答的延时控制。value 为数字，可以是小数（比如 0.1，表示延时 0.1 秒），单位默认为秒；也可以为数字+单位，支持的单位有 S(秒)，M(分)，H(时)，m(毫秒)。

示例：@D=1S(等待 1 秒)

@D=0.5(等待 0.5 秒)

@D=2M(等待 2 分钟)

@D=100m(等待 100ms)

2.2.5 @W 脚本执行期间接收中心下发指令

指令	名称	格式	含义
@W	中心下发指令 延时控制	@W=X @W=XY	当脚本执行期间，同时需要接收服务器下发的指令，可以用 @W 指令延时。格式和 @D=指令一样，支持的单位有： 'S'：秒 'M'：分钟 'H'：小时 'm'：毫秒

当脚本执行期间，服务器下发指令，指令会被丢弃。如果想让 DTU 在脚本执行期间也可以响应服务

器下发的指令，可以使用@W 指令，在@W 延时期间，DTU 将服务器下发指令发送到串口，接收传感器返回的数据并上报到服务器。如果没有收到中心下发的指令，@W 则不延时。指令格式和@D 相同，支持的单位有：S(秒)，M(分)，H(时)，m(毫秒)。

注：@W 指令必须用在@D 指令之后

示例：@C=010300000001V1@D=1@W=1@C=010300010001V1@D=1@W=1

2.2.6 @A 接收应答指令

指令	名称	格式	含义	默认值
@A	接收应答	@A=1 @A=0	在某些命令用户不需要关心仪表的返回值时可以使用@A=0 关闭接收仪表响应，在需要接收的时候再用@A=1 打开	@A=1

A： 接收应答，此脚本指令用于控制 DTU 是否接收仪表的应答，value 为 1 表示接收应答，value 为 0 表示不接收应答。不使用此指令时默认为接收应答。此指令为整个脚本全局生效指令，直到脚本结束，如果有些采集指令需要接收应答有些不需要接收应答只需在相应的采集命令前加入此指令做相应的控制即可。

示例：@A=0 不接收后面仪表的应答数据

小提示：在有些需要开关外接传感器电源的应用场合，此指令的灵活使用可以避免传感器开电期间串口数据的不稳定性，如传感器在开电期间可能会产生无用的数据或乱码输出给DTU，此时，用户可以在开电之前使用@A=0关闭DTU接收传感器数据，然后开电，等待传感器稳定，然后再@A=1打开接收，最后发送命令，这样可以过滤掉传感器上电期间串口不稳定性导致的多余数据。比如：

@A=0@DO1=1@D=2S@A=1@C=xxxxxx@D=2S

2.2.7 @S 立即上报指令

指令	名称	格式	含义	默认值
@S	立即上报	@S=1	在进行多次采集或者多仪表采集时，可以使用@S=1 命令强制 DTU 进行一次当前接收数据的发送，如果不使用@S=1 DTU 默认会脚本执行完毕才进行发送	@S=0

S： 立即上报，此脚本指令用于控制 DTU 立即上报仪表数据。value 为 1 表示立即上报，默认情况下 DTU 执行完 T 指令控制的采集次数后上报（T 默认为 1，每次脚本执行结束都上报），用户在需要分开上报多个仪表数据时或者 1 个仪表的多条数据时，可以在发送采集命令后，DTU 已经收到仪表应答时

插入@S=1 来立即上报之前的数据。

示例：@S=1 立即上报

2.2.8 @H 数据标识指令（HEX 格式）

指令	名称	格式	含义
@H	数据标识	@H=HEX	定义固定上传的数据，用于用户在不同的数据前使用@H 指令插入固定的标识数据，DTU 遇到@H 指令时直接向发送缓冲区的当前位置插入@H 后面定义的数据，数据输入格式为 HEX 格式

H: 数据标识，此脚本指令用于在采集的仪表数据前加入用户自定义的数据标识或用户便于解析的特征数据等，比如在进行多个仪表数据采集时加入分隔符便于区分不同的仪表数据，value 为可见的 HEX 字符，比如 123 表示为 313233。如需要多个标识数据需要在需要的位置使用@H 插入。

示例：@H=DD01 @H=DD02

2.2.9 @STR 数据标识指令（字符格式）

指令	名称	格式	含义
@STR	数据标识	@STR=STR	定义固定上传的数据，用于用户在不同的数据前使用@STR 指令插入固定的标识数据，DTU 遇到@STR 指令时直接向发送缓冲区的当前位置插入@STR 后面定义的数据，数据格式采用字符格式输入，字符内容不能出现 '@' 和 '=' 字符，其他字符都支持

STR: 数据标识，此脚本指令用于在采集的仪表数据前加入用户自定义的数据标识或用户便于解析的特征数据等，比如在进行多个仪表数据采集时加入分隔符便于区分不同的仪表数据，value 为可见的文本字符。如需要多个标识数据需要在需要的位置使用@STR 插入。

示例：@STR=123

2.2.10 @DOx 继电器开关控制指令

指令	名称	格式	含义
@DO1 @DO2	DO 开关控制 DO 状态上报	@DO1=0 @DO1=1	此指令用户控制 DTU 的 DO1~DO2 继电器，或者上报 DO1~DO2 的状态

DO: DO 开关控制, 此脚本指令用于控制 DTU 的 DO1~DO2。value 为 1 表示开, value 为 0 表示关。只有带有 DI DO 的 DTU 才具有此功能, 并且不同的型号 DO 个数可能不一样, 用户可以查看产品手册了解该型号支持的 DO 个数。@DO1=0 表示让 DTU 断开 DO1 继电器, @DO1=1 表示让 DTU 闭合 DO1 继电器, @DO1=? 表示让 DTU 上报 DO1 继电器的状态, 状态为 1 个字节, ‘0’ 表示断开, ‘1’ 表示闭合。此指令经常用于用户在采集仪表数据时给仪表上电和关电的动作。

示例: @DO1=1 打开开关 1

@DO1=0 关闭开关 1

2.2.11 @DIx 状态上报指令

指令	名称	格式	含义
@DI1 @DI2	DI 状态上报	@DI1=?	此指令用于读取 DI1~DI2 口的状态

DI: 状态上报, value 的值为: ?。@DI1=? DTU 会上报 DI1 的状态, 状态为 1 个字节, ‘0’ 表示低电平, ‘1’ 表示高电平。

示例: @DI1=? 获取 DI1 的状态

2.2.12 @GPS 采集 GPS 定位指令

指令	名称	格式	含义
@GPS	GPS 定位信息 采集	@GPS=*	采集 DTU 所有 GPS 定位信息, 发送到服务器
		@GPS=UART_*	采集 DTU 所有 GPS 定位信息, 发送到串口
		@GPS=GGA	采集标准的 GGA 信息, 发送到服务器
		@GPS=UART_GGA	采集标准的 GGA 信息, 发送到串口
		@GPS=RMC	采集标准的 RMC 信息, 发送到服务器
		@GPS=UART_RMC	采集标准的 RMC 信息, 发送到串口
		@GPS=GSA	采集标准的 GSA 信息, 发送到服务器
		@GPS=UART_GSA	采集标准的 GSA 信息, 发送到串口

@GPS	GPS 定位信息 采集	@GPS=JW	采集经纬度信息，发送到服务器
		@GPS=UART_JW	采集经纬度信息，发送到串口
		@GPS=WJ	采集经纬度信息，发送到服务器
		@GPS=UART_WJ	采集经纬度信息，发送到串口
		@GPS=WJ	采集经纬度信息，发送到服务器
		@GPS=UART_WJ	采集经纬度信息，发送到串口
		@GPS=TL	TLink 平台专用脚本，格式为 GPS,2236.70368,N,11350.37840,E,\$

注：此脚本只适用于 GPS 版本的 4G DTU。

示例：@GPS=GGA

返回的报文格式为标准的 GPGGA 格式，解析如下：

\$GPGGA,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,M,<10>,M,<11>,<12>*xx

<1> UTC 时间，格式为 hhmmss.sss；

<2> 纬度，格式为 ddmm.mmmm(第一位是零也将传送)；

<3> 纬度半球，N 或 S(北纬或南纬)

<4> 经度，格式为 dddmm.mmmm(第一位零也将传送)；

<5> 经度半球，E 或 W(东经或西经)

<6> 定位质量指示，0=定位无效，1=定位有效；

<7>使用卫星数量，从 00 到 12(第一个零也将传送)

<8>水平精确度，0.5 到 99.9

<9>天线离海平面的高度，-9999.9 到 9999.9 米 M 指单位米

<10>大地水准面高度，-9999.9 到 9999.9 米 M 指单位米

<11>差分 GPS 数据期限(RTCMSC-104)，最后设立 RTCM 传送的秒数量

<12>差分参考基站标号，从 0000 到 1023(首位 0 也将传送)。

当 DTU 返回报文

\$GNGGA,040201.000,3027.82284,N,10359.55214,E,1,07,1.4,451.1,M,0.0,M,,*7D 时，

具体解析格式可以参考标准的 GGA 格式。

2.2.13 @TS 获取 NTP 时间指令

指令	名称	格式	含义
@TS	NTP 时间	@TS=1 @TS=2 @TS=3 @TS=4	@TS=1 表示往服务器发送 HEX 格式的时间戳 @TS=2 表示往服务器发送字符串格式的时间戳 @TS=3 表示往串口发送 HEX 格式的时间戳 @TS=4 表示往串口发送字符串格式的时间戳

注：只有当 DTU 连上网以后才能使用 NTP 时间功能

示例：@TS=1 往服务器发送 HEX 格式时间戳：2020032805100227

@TS=4 往串口发送字符串格式时间戳：2020/03/28,05,10:05:30

2.2.14 定时脚本

指令	名称	格式	含义
支持所有 @ 脚本指令	定时脚本	见示例	当设备连接上网络后，通过获取时钟，可以在定时脚本参数中配置时间和对应的脚本命令，就能在设置的时间执行指定的脚本

注：G1.14 及以上版本支持定时脚本功能

在参数中配置的格式为：年/月/日,星期,时:分:秒，然后用#接需要执行的脚本，用 | 分割每个时间需要执行的脚本。

示例：

2020/03/30,01,12:00:00#@C=010300000001840A@D=1|2020/03/30,01,13:00:00#@C=0203000000018439

表示 2020/03/30 12 点执行@C=010300000001840A 指令，2020/03/30 13 点执行@C=0203000000018439

也可以用%d 匹配任意数字，这样会使脚本更加灵活，例如：

2020/03/30,01,12:%d%d:00#@C=010300000001840A@D=1

表示 2020/03/30 12 点的每一分钟执行一次@C=010300000001840A

2.2.15 @PUB json 发布脚本

指令	名称	格式	含义
@PUB	json 发布脚本	@PUB=1 @PUB=0	当用户使用 Modbus 转 json 网关版本时，可以在 Lua 代码里面使用此脚本控制是否立即发布数据到服务器

注：只有 Modbus 转 json 网关版本才支持此脚本

2.2.16 @获取设备 CSQ 指令

指令	名称	格式	含义
@CSQ	获取信号强度	@CSQ=?	在脚本当前位置插入信号强度，格式为十进制的实际信号强度

示例：@CSQ=? 如果信号强度为 27，则上报的数据为 27

2.2.17 @获取设备 ICCID 指令

指令	名称	格式	含义
@ICCID	获取 ICCID	@ICCID=?	在脚本当前位置插入 SIM 卡的 ICCID，格式为可见字符串

示例：@ICCID=?

2.2.18 @获取设备 IMEI 指令

指令	名称	格式	含义
@IMEI	获取 IMEI	@IMEI=?	在脚本当前位置插入设备的 IMEI 码，格式为可见字符串

示例：@IMEI=?

2.2.19 @获取设备 IMSI 指令

指令	名称	格式	含义
@IMSI	获取 IMSI	@IMSI=?	在脚本当前位置插入设备的 IMSI 码，格式为可见字符串

示例：@IMSI=?

2.2.20 @LBS 基站定位

指令	名称	格式	含义
@LBS	获取基站定位信息	@LBS=JW @LBS=WJ	获取设备基站定位的经纬度，@LBS=JW 表示经度在前，纬度在后；@LBS=WJ 表示纬度在前，经度在后。 格式为：经度，纬度（纬度，经度） 如果定位失败，返回 0,0

示例：@LBS=JW

2.2.21 @WIFI 定位

指令	名称	格式	含义
@WIFI	获取 WIFI 定位信息	@WIFI=JW @WIFI=WJ	获取设备 WIFI 定位的经纬度，@WIFI=JW 表示经度在前，纬度在后；@WIFI=WJ 表示纬度在前，经度在后。 格式为：经度，纬度（纬度，经度） 如果定位失败，返回 0,0

注：只有 LTE-659-P，LTE-658 支持此指令

示例：@WIFI=JW

三、应用案例

用户通过 485 采集两个仪表的数据，Modbus 地址分别为 01 和 02，寄存器地址为 0000 和 0002，采集的寄存器数据量为 1 和 2。

配置的脚本为@H=AA01@C=010300000001V1@D=1@H=AA02@C=020300020002V1@D=1

添加个@H 头用于区分两个仪表的数据。