

用户数据标准 V1.0 (2019)

1、概述

据我所知，当前市场及网络上尚未有关于串口助手类软件的数据标准，为设计通用、方便、好用的同类软件，特此制定本标准。本标准主要定义规范的文件组织形式及其主要关键字的定义。主要应用于串口助手的用户数据发送部分。依据此标准生成一个用户定义的数据文件（XXX.XML），串口助手软件按照本标准解析文件即可得到用户的数据结构及其详细数据。此后即可应用于测试目的。也是为了方便软件的使用。同时本标准作为软件数据的规范起辅导及约束作用。

2、应用场景

2.1、测试工程师

按照本标准编辑好需要的测试命令串口文件，或者直接使用别人提供的命令串标准文件（模块提供商提供），打开软件，打开串口，选择对应的数据文件（XXX.XML）直接点击对应的命令按钮，直接发送命令即可着手测试。

2.2、模块供应商

按照本标准，编辑自家模块的测试命令文件，随模块一起提供给客户，方便客户使用即方便自己的销售。同样也方便自己的技术支持工作

2.3、开发工程师

研发工程师开发产品免不了要测试，使用网络上的串口助手软件，总有些限制，比如命令行数量不够用，命令行修改难，不通用。遇到新的模块，供应商不提供测试验证方法或者文件，需要自行编写验证命令，比如 AT 指令之类的模块，假如此类模块供应商按照本标准写好其 AT 测试命令文件，并提供给你，是不是可以节省很多时间。答案是肯定的。

3、需求

3.1、命令行条数→无限，即想要多少就多少条

3.2、命令行数据含义自定义→这是必须的，每条命令自定义一个名称，名称显示到按钮上。

3.3、数据方式形式：十六进制及字符模式

3.4、命令行自动发送：支持

3.5、支持帧格式：帧头，帧尾，帧长 1 字节 2 字节可选，帧前填充，帧后填充，启用的开个选择。其中帧前帧后填充支持不少于 50 个字符（即发送数据帧时在其前后填充这两个串）。

4、约定与约束

4.1、文件格式约定：全部数据按照 XML 标准设计文件元素及其节点，有关 XML 文件知识请网络查询。

4.2、文件名称：文件名称自由定义，取有意义的名称即可，文件后缀必须为.XML。

4.3、命令行分页：按照需求，需要无限的条目数量，必须将命令行分页显示，每页最少 20 条，提供首页，尾页，上一页，下一页导航，每页支持一个名称（可自定义）

4.4、关于自动发送：空行不自动发送，自动发送以页为单位，即同一页的命令行均会自动发送，不对名一行设置自动发送。

4.5、直白的表述

文件中选项的元素名称使用直白的单词，元素字符节点使用 YES /NO 之类的直白形式，不使用 1 代表 TRUE,0 代表 FALSE 之类的。当然文档内部主要使用小写字母，限制使用大写字母。

5、关键字

Userdata—>代表用户数据

Setting--->代表用户数据的设置节点

Pages—>代表用户数据的命令行页（总的节点名称）

Page->代表用户数据中一个具体的页，由 ID 标识具体页，由 name 标识页名称

Cmdline->标识某一个命令行 由 ID 标识命令行编号，命令行下有三个子元素，分别为 name 命令行名称，显示到按钮上，hex 由其字符节点内部标识是否按照十六进制数据的形式发送，text 即为要发送的数据内容。当启用设置时会将数据包含到一个数据帧中发送。

Headfill->表示帧前填充串

Endfill->表示帧尾填充串

Onlyfillend—>表示命令行发送时仅填充“帧尾填充串”，适应用 AT 指令集需要回车换行的应用场景。

6、标准文档样式



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <userdata name="cmdline" version="1.0">
3   <setting id="1">
4     <used>yes</used>
5     <len>1</len>
6     <headfill>123456</headfill>
7     <head>5A</head>
8     <end>5A</end>
9     <endfill>7890AB</endfill>
10    <onlyfillend>yes</onlyfillend>
11  </setting>
12  <pages>
13    <page id="1" name="test" atuotx="no">
14      <cmdline id="1">
15        <name>num1</name>
16        <hex>yes</hex>
17        <text>12 34 56</text>
18      </cmdline>
19      <cmdline id="2">
20        <name>num2</name>
21        <hex>yes</hex>
22        <text>12 34 56</text>
23      </cmdline>
24      <cmdline id="3">
25        <name>num3</name>
26        <hex>yes</hex>
27        <text>12 34 56</text>
28      </cmdline>
29    </page>
30  </pages>
31 </userdata>
```

7、文件内容

7.1、顶行必须为<?xml version="1.0" encoding="utf-8"?>

7.2、首节点必须为<userdata name="cmdline" version="1.0">，软件解析时首先分析首节点是否为“userdata”，其名称必须为“cmdline”，版本号作为解析文件其他元素的依据，目前固定为 1.0

7.3、次节点

目前 1.0 版本仅支持两个次级节点分别是：<setting id="1">和<pages>一个是设置节点，一个是页节点；设置节点有 id 属性，目前固定为 1 表示第一个设置节点，保留以

后扩展使用。后面是页节点，为复数，表示有很多页。其下一级才是具体的页数据。软件已经此可知道用户数据文件有多少个页。

7.4、设置节点

```
3  <setting id="1">
4    <used>yes</used>
5    <len>1</len>
6    <headfill>123456</headfill>
7    <head>5A</head>
8    <end>5A</end>
9    <endfill>7890AB</endfill>
10   <onlyfillend>yes</onlyfillend>
11 </setting>
```

由上图可见，设置节点有 7 个元素，均是用于控制帧结构的，当 onlyfillend 为 yes 时发送忽略数据帧结构，即这个缘分优先级最高，主要用于 AT 命令末尾插入回车换行的作用，其他正常使用设置为 “no” 即可启用正常的数据帧通信。也可以在命令行中定义自己的帧结构，只是重复了些帧头帧尾之类的数据；帧长可以使用 1 个字节或者 2 个字节表示；帧头与帧尾只能是一个字节。可输入帧前与帧后填充串，其为空则无填充。填充长度支持不少于 50 个字符。

7.5、数据页总览

```
12  <pages>
13    <page id="1" name="test1" atuotx="yes">
30    <page id="2" name="test2" atuotx="no">
47    <page id="3" name="test3" atuotx="no">
64  </pages>
```

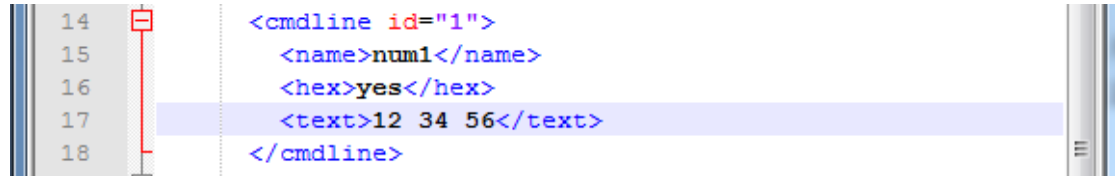
由图可见，此文件有 3 个数据页，ID 表示其页码，不能有重复，名称可以任意，自动发送可以配置。总的数据页数不限制，因此可以实现无限~~无限~~的命令条目数据。

7.6、数据页

```
13  <page id="1" name="test1" atuotx="yes">
14    <cmdline id="1">
15      <name>num1</name>
16      <hex>yes</hex>
17      <text>12 34 56</text>
18    </cmdline>
19    <cmdline id="2">
20      <name>num2</name>
21      <hex>yes</hex>
22      <text>12 34 56</text>
23    </cmdline>
24    <cmdline id="3">
25      <name>num3</name>
26      <hex>yes</hex>
27      <text>12 34 56</text>
28    </cmdline>
29  </page>
```

由图可见数据页内部有三个命令行，命令行有 ID 表示该命令行的编号，不能重复，一个页内部只能有 20 个命令行，多于 20 的命令行不被解析（具体软件实现可以适当增加一页的行数）

7.7、命令行



由图可见命令行除自身的 ID 属性之外，包含三个元素，分别为 name 命令行名称（显示于按钮上）hex 命令行属性：yes/no 是/否十六进制，text 实际的命令数据串。

2019/1/24