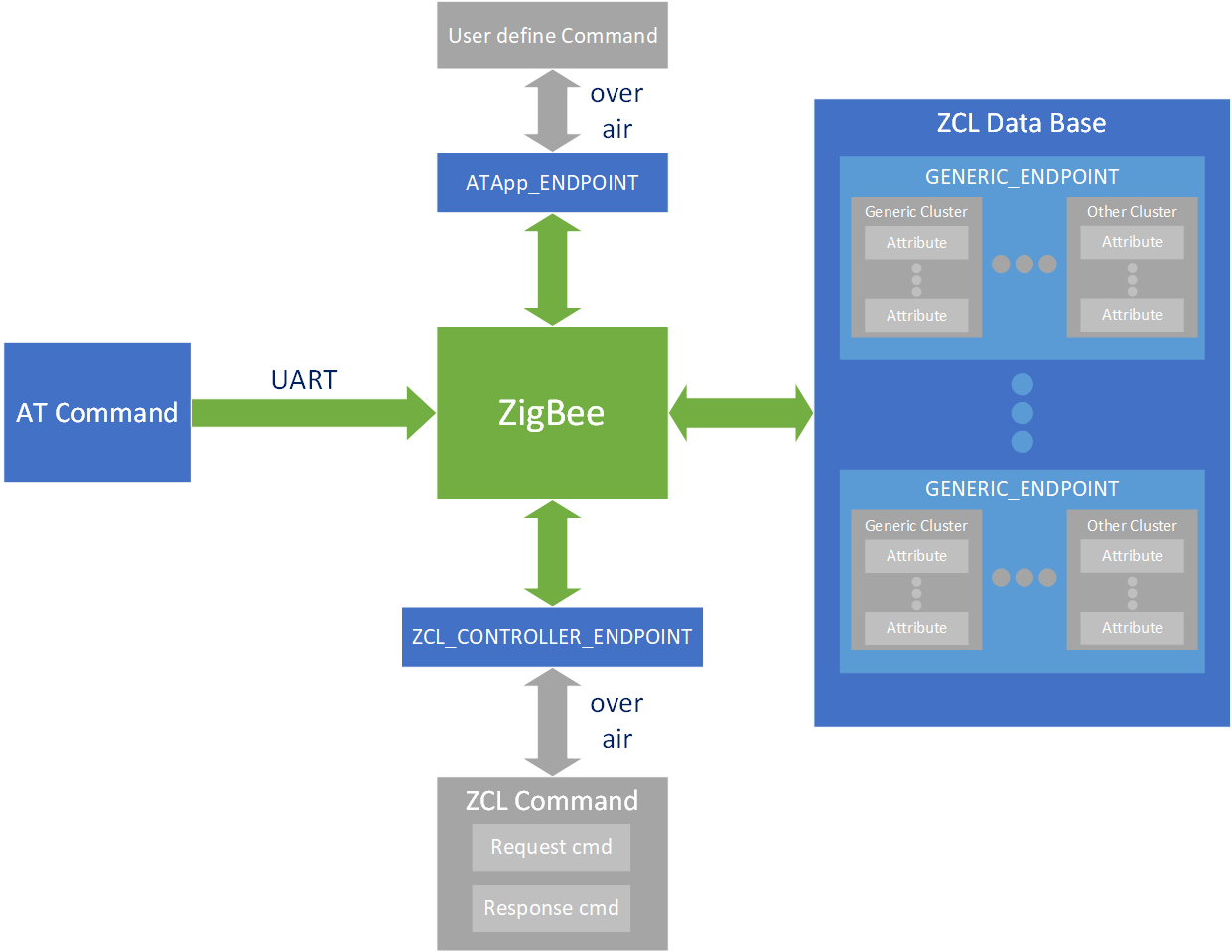
# ZB-AT-CMD

本工程基于Z-Stack 3.0.1。

## 应用架构



架构说明：

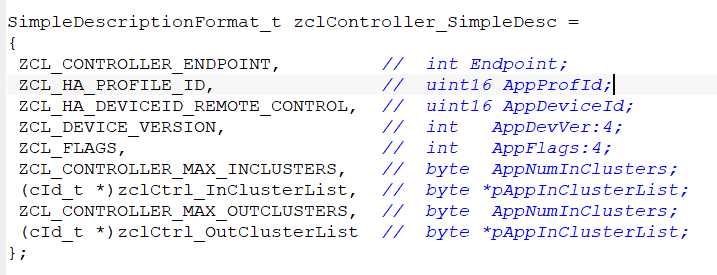
* 用户通过串口给ZigBee下发AT Command。
* 用户自定义的命令（User define command）通过ATApp\_ENDPOINT与ZigBee芯片通信。
* 现已验证的ZCL Command有ReadCmd和WriteCmd。对于Request Command，发送源端点（srcEP）为ZCL\_CONTROLLER\_ENDPOINT，接收目的端点（destAddr. endPoint）为对应ZCL功能的ENDPOINT。
* 对于endpoint的使用定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| Endpoint ID | 1-10 | 11-99 | 100-200 |
| 功能 | Reserve | ZCL | Application |

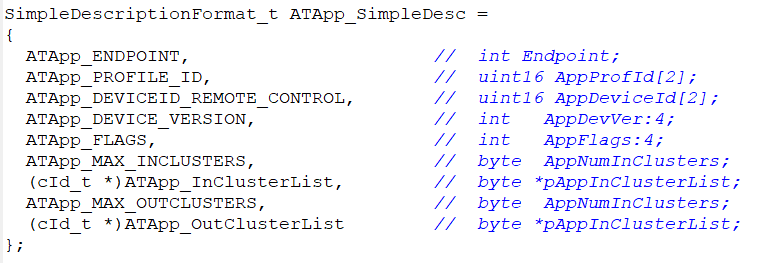
现已使用的endpoint定义如下：

|  |  |
| --- | --- |
| ZCL\_CONTROLLER\_ENDPOINT | 100 (0x64) |
| ATApp\_ENDPOINT | 150 (0x96) |
| GENERIC\_ENDPOINT | 11 (0xB) |

ZCL\_CONTROLLER\_ENDPOINT对应的Simple Description:



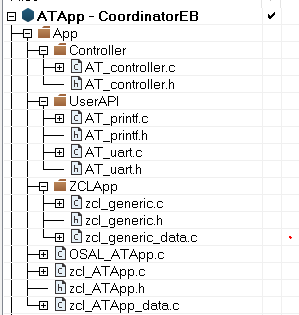
ATApp\_ENDPOINT对应的Simple Description:



该Simple Description使用的profile ID和device ID为用户自定义。

## 文件架构

本次添加的文件包含三个部分：UserAPI、ZCLApp、Controller。如下图所示



* UserAPI组下的源文件有AT\_printf.c、AT\_uart.c。主要用于接收和出力AT Command。
* ZCLApp组下的源文件有zcl\_generic.c、zcl\_generic.h、zcl\_generic\_data.c。

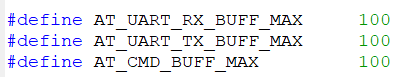
该组下的文件主要用于描述设备可用的ZCL应用。

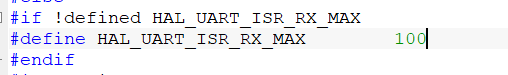
* Controller组下的源文件有AT\_controller.c。主要用于处理ZCL Command和用户定义的Command。

## AT 命令

AT命令通过串口接收，基于状态机进行处理。

串口的缓存大小如下图：





状态机处理过程如下：

