

AI1276-401H SDK

(V0.0.2)

**History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Description** | **Author** |
| V0.0.1 | 2016/05/25 | First Draft | Chia-Chang Wu |
| V0.0.2 | 2016/07/12 | Add Flash Usage | Chia-Chang Wu |
|  |  |  |  |
|  |  |  |  |

**Introduction**

Provide the description and programming sample of AI1276-401H SDK.

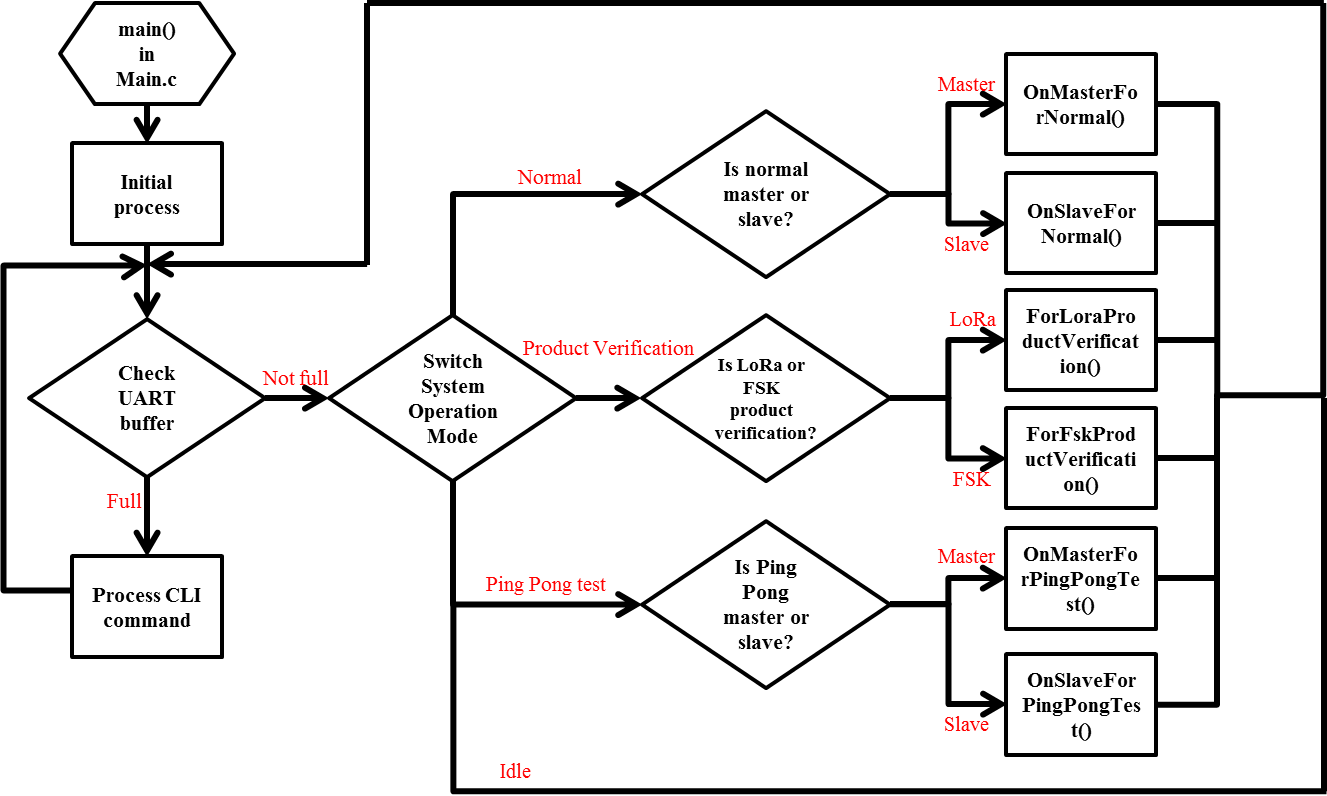
**Project File**

* The project file “STM32F401VCTx.uvproj” would be in the folder “STM32F401VCTx\MDK\”.



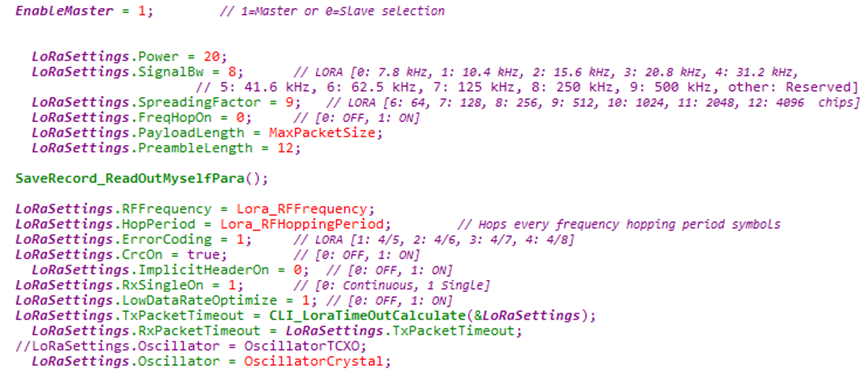
**Flow Chart**

* The flow chart of main() in Main.c



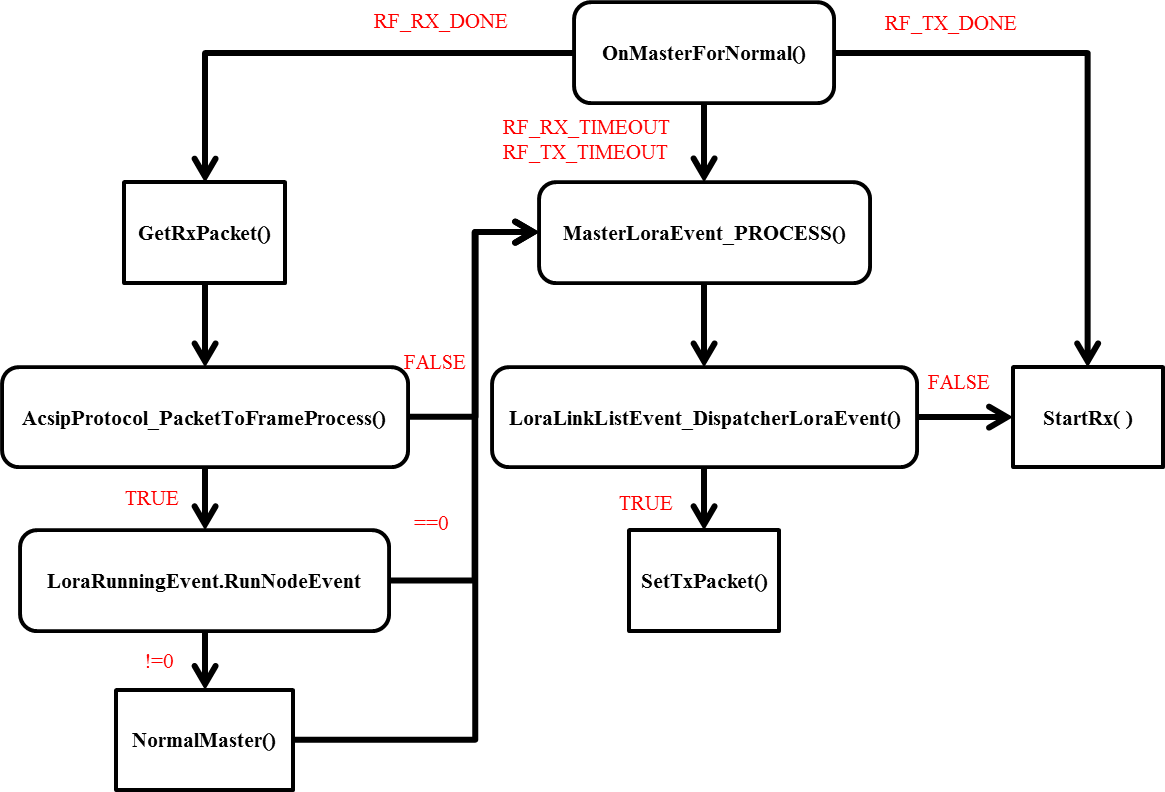
**Default LoRa Settings**

* The default LoRa settings are in the function “LoraPara\_LoadAndConfiguration()” of “USER\Main.c”.

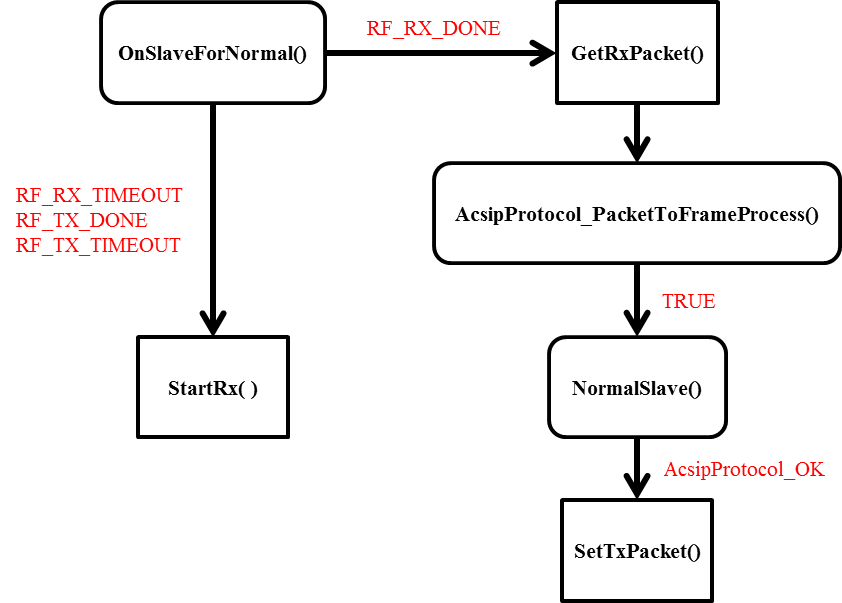


**System Operation Mode**

* Normal
  + If user doesn’t use CLI command to change System Operation Mode, Normal mode would be the default.
  + Normal Master



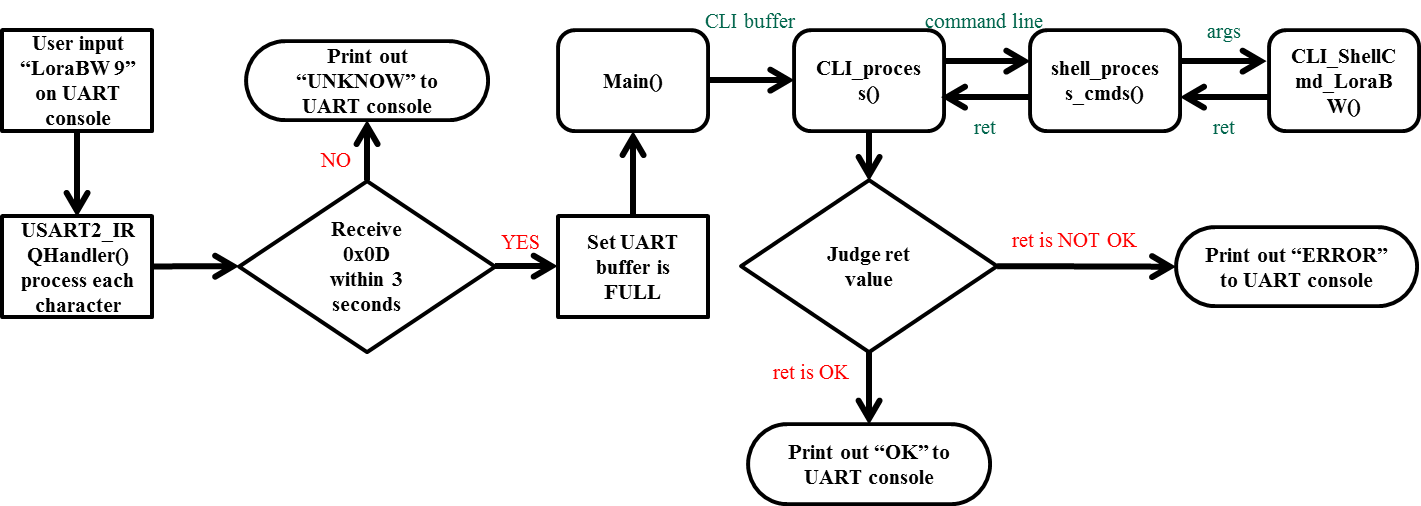
* + Normal Slave



* + CLI Command Flow Sample for Normal Mode:
    1. COM port: 38400 baud rate, N-8-1.
    2. LoraSystemMode inNormal.
    3. For Master: LoraMode MASTER.
    4. For Slave: LoraMode SLAVE.
    5. LoraStartWork ENABLE.
    6. For Master: LoraJoinNode “Slave address”.
    7. If the response is not OK, please refer to “CLI\_ShellCmd\_LoraJoinNode()” in “Cli.c” for further information.
* Product Verification
  + FSK product verification: Verify RF modulation performance.
  + LoRa product verification: Verify LoRa performance by sending and receiving specific packet.
* Ping Pong Test
  + This mode is for demonstration. The master send packet with “PING” string. The slave response packet with “PONG” string when it receives the packet from master.

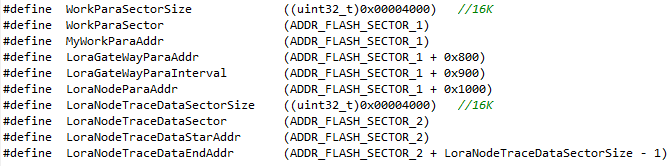
**CLI Command Flow Chart**

* Input “LoraBW 9” to change the Bandwidth to 500kHz for example:



**Flash Usage for LoRa Parameters and Data**

* The functions of storing LoRa parameters and data to flash memory are in the file “USER\STM32F401VCx\Save\_record.(h/c)”.
* The usage for LoRa parameters and data is defined in “USER\STM32F401VCx\Save\_record.h”.



**Programming Sample: LED Blink**

* LED on/off by controlling specific GPIO.
* LED blinking pattern can be set by the parameters of LED blink function.
* There are three parameters of LED blink function: onTime, offTime and duration.



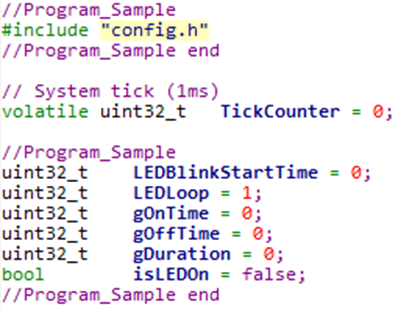
* Add LED Blink function to “USER\board.(h/c)” like the picture below:
  + In board.h:



* + In board.c:



* Add global variables and include “config.h” for printing debug message to console in board.c:



* Test LED Blink function by calling this function in while loop of Main.c.

