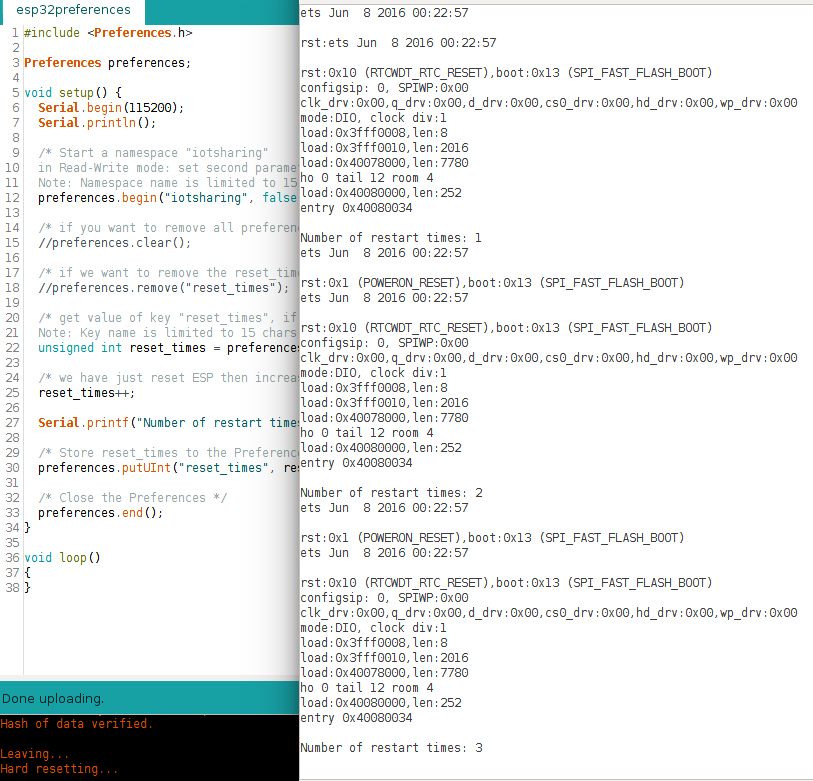
# **[Demo 23: How to use Preferences to backup Arduino ESP32 data in main flash memory when power is off](http://www.iotsharing.com/2017/06/how-to-use-preferences-to-backup-data-in-main-flash-when-power-off.html)**

**1. Introduction**  
- This library use a portion of main flash memory to store data (beside this way we can use sdcard to store data [Demo 7: How to use Arduino ESP32 to store data to sdcard](http://www.iotsharing.com/2017/05/how-to-use-arduino-esp32-to-store-data-to-sdcard.html" \t "http://www.iotsharing.com/2017/06/_blank)).  
- Data is stored under key-value pair. Key is 15 ASCII characters length and must be unique. Value can be:  
+ integer types  
+ variable length binary data (blob)  
- Data type of writing and reading value must be matched.  
- Our Preferences has the structure like this:  
namespace {  
    key: value  
}  
where namespace is to avoid key collisions (in case we have 2 keys with same name, we use namespace to separate them). E.g:  
section1 {  
    counter: value  
}  
section2 {  
    counter: value  
}  
Here, we have a duplicate key named counter but in different namespaces.  
- We create a demo for this tutorial: storing number of restart times when user press reset button. and print it on Terminal.  
**2. Hardware**  
This library uses main flash memory then no need extra hardware.   
**3. Software**  
- To test the program just press the Reset button and see reset\_times is increased.

|  |
| --- |
| #include <Preferences.h>  /\* create an instance of Preferences library \*/  Preferences preferences;  void setup() {  Serial.begin(115200);  Serial.println();  /\* Start a namespace "iotsharing"  in Read-Write mode: set second parameter to false  Note: Namespace name is limited to 15 chars \*/  preferences.begin("iotsharing", false);  /\* if you want to remove all preferences under opened namespace uncomment it \*/  //preferences.clear();  /\* if we want to remove the reset\_times key uncomment it \*/  //preferences.remove("reset\_times");  /\* get value of key "reset\_times", if key not exist return default value 0 in second argument  Note: Key name is limited to 15 chars too \*/  unsigned int reset\_times = preferences.getUInt("reset\_times", 0);  /\* we have just reset ESP then increase reset\_times \*/  reset\_times++;  Serial.printf("Number of restart times: %d\n", reset\_times);  /\* Store reset\_times to the Preferences \*/  preferences.putUInt("reset\_times", reset\_times);  /\* Close the Preferences \*/  preferences.end();  }  void loop()  {  } |

1. **Result**

[](https://3.bp.blogspot.com/-l4J89YnBhIs/WUvEH8ec8uI/AAAAAAAAEFs/VUz2JFsAV-saTTWXDLMEnKW_mcvREFRgQCLcBGAs/s1600/esp32preferences.png)

**Figure: Press Reset button to see restart times**