

Checking Capabilities

Afero supports multiple ASR systems-on-chips, which may have different capabilities inherently or by virtue of the version of firmware active at runtime. In order to help you create MCU code flexible enough to tolerate such different environments, afLib3 provides a function to enable MCU to query ASR to check if a specific capability is offered.

af_lib_asr_has_capability()

Description

Check the current ASR/firmware for the availability of specified capability.

Syntax

```
af_lib_error_t af_lib_asr_has_capability(af_lib_t *af_lib, uint32_t
af_asr_capability)
```

Parameters

af_lib	Pointer to the active afLib instance.

af_asr_capability	A given capability to test.
-------------------	-----------------------------

Possible capability values currently include:

- AF_ASR_CAPABILITY_MCU_OTA - ASR supports MCU OTA.

Returns

An [afLib result code](#).



It can take non-zero time for ASR to communicate a capability to afLib. If this function is called too early it will return the AF_ERROR_BUSY error code, in which case it should be tried again "at a later time."

If you need an explicit determination of that "later time," it is marked by the time when you can successfully call `af_lib_get_attribute()` for attribute

AF_ATTRIBUTE_ID_ASR_CAPABILITIES, and retrieve it via the `attrEventCallback()`, in the AF_LIB_EVENT_ASR_GET_RESPONSE event.

Example

```
include "af_lib.h"
#include "af_module_commands.h"
#include "af_module_states.h"
#include "arduino_spi.h"
#include "arduino_transport.h"

af_lib_t *af_lib;
af_lib_error_t capabilityResult = AF_SUCCESS;
bool stopChecking = false;

void setup() {
    // Initialize afLib
    af_transport_t* arduinoSPI = arduino_transport_create_spi(CS_PIN);
    af_lib = af_lib_create_with_unified_callback(attrEventCallback, arduinoSPI);
    arduino_spi_setup_interrupts(af_lib, digitalPinToInterrupt(INT_PIN));
}
```

```

}

void loop() {
  capabilityResult = af_lib_asr_has_capability(af_lib, AF_ASR_CAPABILITY_MCU_OTA);
  if (!stopChecking){
    switch (capabilityResult) {
      case AF_SUCCESS:
        Serial.println("MCU OTA is SUPPORTED!!!");
        stopChecking = true;
        break;
      case AF_ERROR_NOT_SUPPORTED:
        Serial.println("MCU OTA is NOT SUPPORTED!!!");
        stopChecking = true;
        break;
      case AF_ERROR_BUSY:
        Serial.println("It's too early to know if MCU OTA is supported!!!");
        break;
    }
  }

  // Give afLib processing time by calling af_lib_loop(af_lib) in sketch's loop()
  af_lib_loop(af_lib);
}

```

➞ **Next:** [Console Logging](#)

Updated September 25, 2019