IDF (Interrupt Driven Framework) Platform Default Port

This folder provides a default implementation for IDF related XF classes. You can use these classes to construct an IDF based XF.

If these classes do not suit your needs, they can be reimplemented for your platform. Create an additional folder in the 'port' folder and implement there the classes you need for your platform.

Available Default IDF Port Classes

Class name	File location	Define to set
XFEventQueueDefault	xf/port/default-idf/eventqueue-default.cpp	USE_XF_EVENT_QUEUE_DEFAULT_IDF_IMPLEMENTATION
XFMutexDefault	xf/port/default-idf/mutex-default.cpp	USE_XF_MUTEX_DEFAULT_IDF_IMPLEMENTATION

If you need more information about the classes mentioned above, please have a look into their header files and the doxygen comments in code.

Example *config/xf-config.h* File

Following you will find an example giving you a basic idea which define to set in the application specific *config/xf-config.h* file.

The *IDF Stm32Cube* port uses quite all default implementations:

```
// Defines to set to use the IDF Stm32Cube port
#define USE_XF_DEFAULT_IMPLEMENTATION                                   1
#define USE_XF_DISPATCHER_DEFAULT_IMPLEMENTATION                        1
#define USE_XF_TIMEOUTMANAGER_DEFAULT_IMPLEMENTATION                    1
#define USE_XF_RESOURCE_FACTORY_DEFAULT_IMPLEMENTATION                  1
#define USE_XF_MUTEX_DEFAULT_IDF_IMPLEMENTATION                         1
#define USE_XF_EVENT_QUEUE_DEFAULT_IDF_IMPLEMENTATION                   1

#define USE_XF_PORT_IDF_STM32CUBE_PORT_FUNCTIONS_IMPLEMENTATION         1

#include "default-idf/eventqueue-default.h"
#ifdef __cplusplus
    using XFEventQueue = XFEventQueueDefault;
#endif // __cplusplus
```