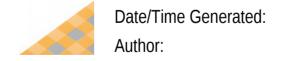
# **Diagram Report**

Version •



24.10.2022 14:36:20 Thomas Batt

EA Repository:



## **Table of Contents**

1.1.1 Architecture_Overview diagram	3
1.1.1.1 Clock_Observer_Application diagram	4
1.1.1.2 Counter diagram	5
1.1.1.3 Clock diagram	6
1.1.1.4 Clock_Source diagram_	7
1.1.1.5 Common diagram	8
1.1.1.6 User_Exception diagram	9
1.1.1.6.1.1 Device_Driver_Abstraction diagram	10
1.1.1.6.1.1.1.1 NVIC_Dispatcher diagram	11
1.1.1.6.1.1.1.2 TIMER diagram	14
1.1.1.7 Exercise_1: Classes, Objects and Relations diagram	16
1.1.1.8 Exercise_2: Namespaces and Multiple Inheritance diagram	17
1.1.1.9 Exercise_3: Exception Handling and Dynamic Memory Management diagram	18
1.1.1.10 Exercise_4: Observer Pattern: Template Class, RTTI and Smart Pointer diagram	19
1.1.1.11 Exercise_5: Observer Pattern: Standard Library Container and Template Exception Cl	lass diagram
	20
1.1.1.12 Exercise_6: Callback, Hardware Driver and Interrupt diagram	21

## 1.1.1 Architecture\_Overview diagram

Package diagram in package 'Architecture'

Architecture\_Overview Version 1.0 Thomas Batt created on 02.11.2021. Last modified 02.11.2021

Name: Architecture\_Overview
Package: Architecture
Version: 1.0
Author: Thomas Batt

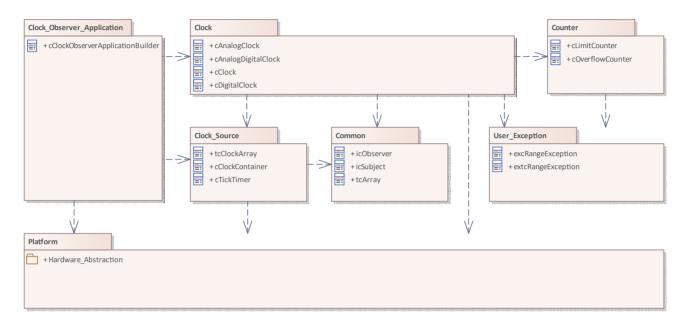


Figure 1: Architecture\_Overview

### 1.1.1.1 Clock\_Observer\_Application diagram

Class diagram in package 'Clock Observer Application'

Clock\_Observer\_Application
Version 1.0
Thomas Batt created on 02.11.2021. Last modified 03.11.2021

Name: Clock\_Observer\_Application Package: Clock\_Observer\_Application

Version: 1.0 Author: Thomas Batt

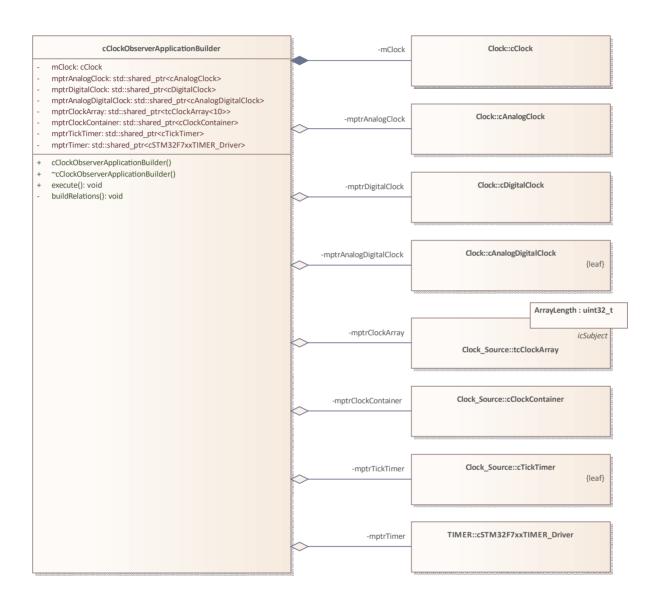


Figure 2: Clock\_Observer\_Application

#### 1.1.1.2 Counter diagram

Class diagram in package 'Counter'

Counter Version 1.0 Thomas Batt created on 02.11.2021. Last modified 03.11.2021

Name: Counter
Package: Counter
Version: 1.0
Author: Thomas Batt

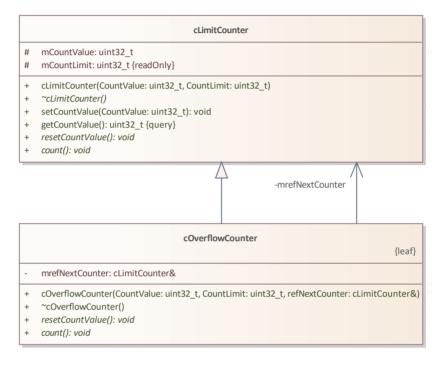


Figure 3: Counter

#### 1.1.1.3 Clock diagram

Class diagram in package 'Clock'

Clock Version 1.0 Thomas Batt created on 02.11.2021. Last modified 18.03.2022

Name: Clock
Package: Clock
Version: 1.0
Author: Thomas Batt

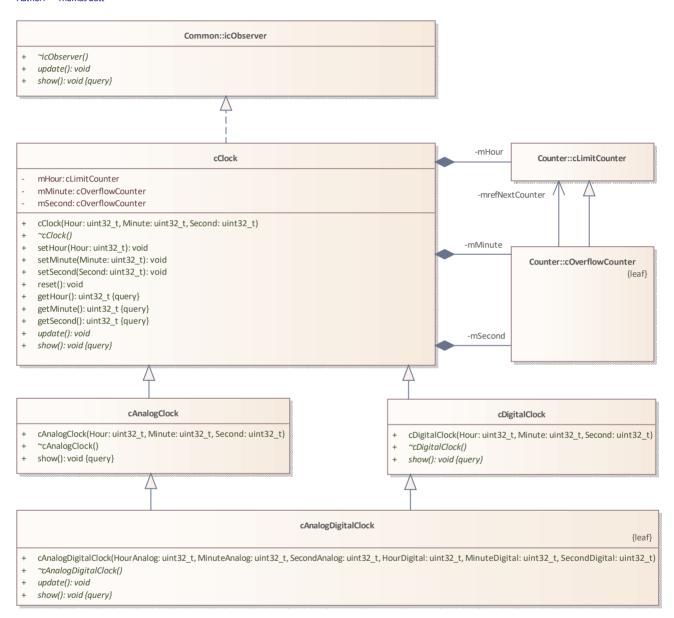


Figure 4: Clock

#### 1.1.1.4 Clock\_Source diagram

Class diagram in package 'Clock Source'

Clock\_Source
Version 1.0
Thomas Batt created on 02.11.2021. Last modified 25.02.2022

Name: Clock\_Source
Package: Clock\_Source
Version: 1.0
Author: Thomas Batt

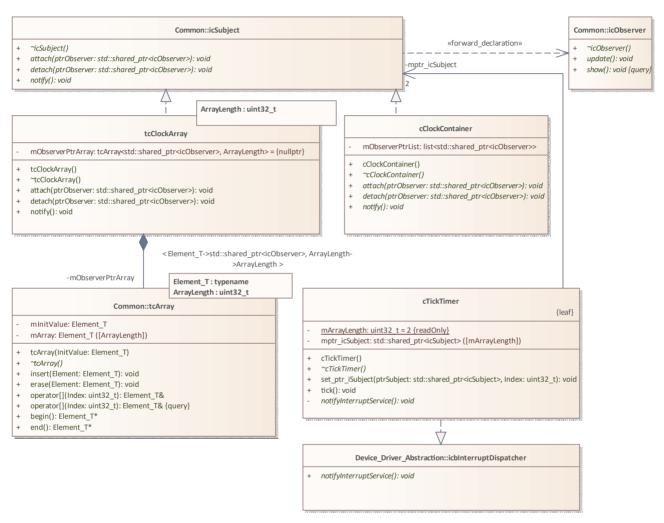


Figure 5: Clock Source

#### 1.1.1.5 Common diagram

Class diagram in package 'Common'

Common Version 1.0 Thomas Batt created on 02.11.2021. Last modified 25.02.2022

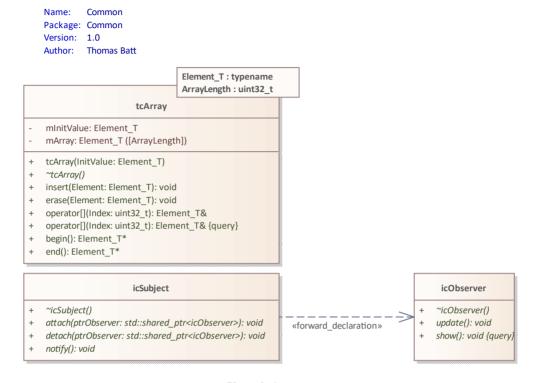


Figure 6: Common

#### 1.1.1.6 User Exception diagram

Class diagram in package 'User Exception'

User\_Exception
Version 1.0
Thomas Batt created on 02.11.2021. Last modified 03.11.2021

Name: User\_Exception
Package: User\_Exception
Version: 1.0
Author: Thomas Batt

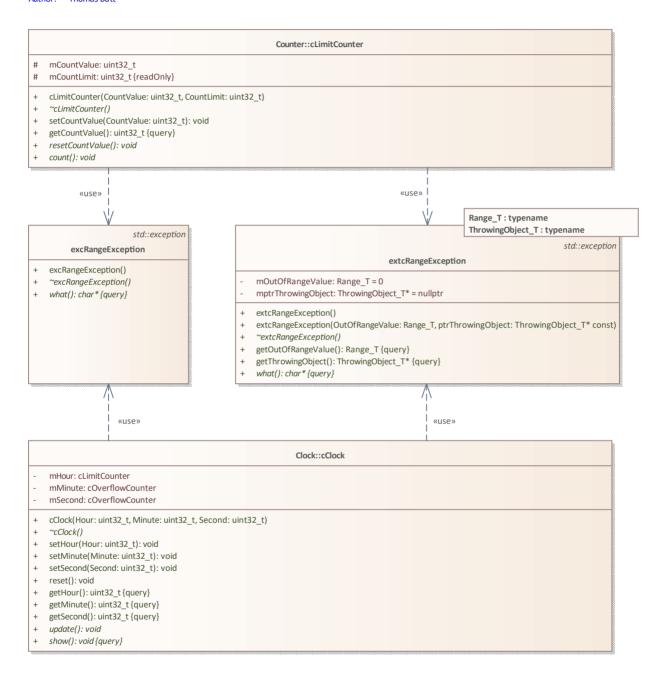


Figure 7: User\_Exception

### 1.1.1.6.1.1 Device\_Driver\_Abstraction diagram

Class diagram in package 'Device\_Driver\_Abstraction'

Device\_Driver\_Abstraction
Version 1.0
Thomas Batt created on 02.11.2021. Last modified 02.11.2021

Name: Device\_Driver\_Abstraction
Package: Device\_Driver\_Abstraction

Version: 1.0 Author: Thomas Batt

icbInterruptAcknowledge + notifyInterruptAcknowledge(): void +



Figure 8: Device\_Driver\_Abstraction

#### 1.1.1.6.1.1.1 NVIC\_Dispatcher diagram

Class diagram in package 'NVIC\_Dispatcher'

NVIC\_Dispatcher Version 1.0
Thomas Batt created on 02.11.2021. Last modified 02.11.2021

Name: NMC\_Dispatcher Package: NMC\_Dispatcher Version: 1.0 Author: Thomas Batt





schammer allows
Memory (Control Number 2)
Me

setumidación kternatáriotty kijáset + 0 kternatáriotty kijáset + 0 kternatáriotty kijáset + 0 kternatáriotty 1 = 1 kternatáriotty 2 = 2 kternatáriotty 2 = 2 kternatáriotty 3 = 3 kternatáriotty 4 + 4 kternatáriotty 4 + 4 kternatáriotty 4 + 5 kternatáriotty 4 + 5 kternatáriotty 4 + 6 kternatáriotty 4 kter

Figure 9: NVIC\_Dispatcher

#### 1.1.1.6.1.1.1.2 TIMER diagram

Class diagram in package 'TIMER'

TIMER
Version 1.0
Thomas Batt created on 02.11.2021. Last modified 02.11.2021

Name: TIMER
Package: TIMER
Version: 1.0
Author: Thomas Batt

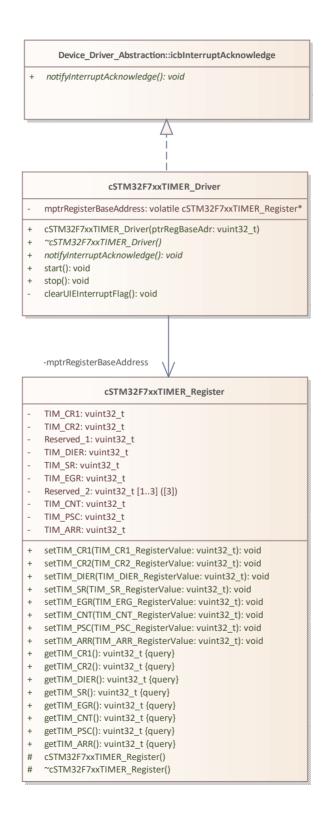


Figure 10: TIMER

### 1.1.1.7 Exercise\_1: Classes, Objects and Relations diagram

Class diagram in package 'Exercise 1'

Exercise\_1: Classes, Objects and Relations

Version 1.0
Thomas Batt created on 02.11.2021. Last modified 29.11.2021

Name: Exercise\_1: Classes, Objects and Relations Package: Exercise\_1

Version: 1.0
Author: Thomas Batt

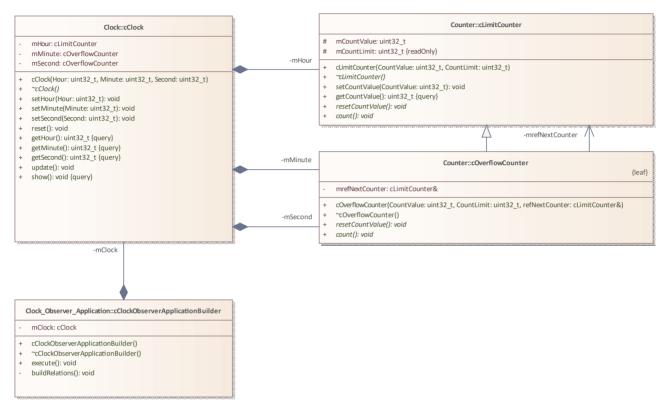


Figure 11: Exercise\_1: Classes, Objects and Relations

### 1.1.1.8 Exercise\_2: Namespaces and Multiple Inheritance diagram

Class diagram in package 'Exercise 2'

Exercise\_2: Namespaces and Multiple Inheritance
Version 1.0
Thomas Batt created on 02.11.2021. Last modified 29.11.2021

Name: Exercise\_2: Namespaces and Multiple Inheritance
Package: Exercise\_2
Version: 1.0
Author: Thomas Batt

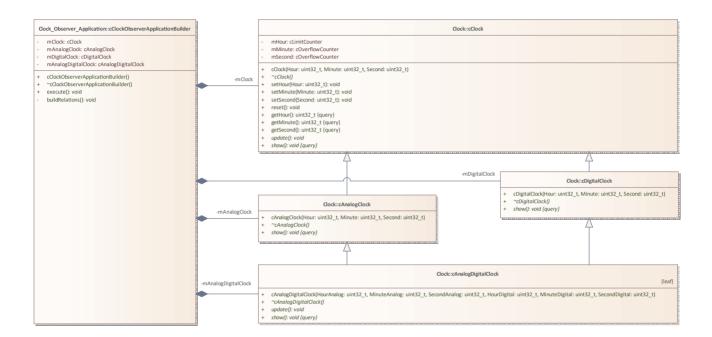


Figure 12: Exercise\_2: Namespaces and Multiple Inheritance

# **1.1.1.9** Exercise\_3: Exception Handling and Dynamic Memory Management diagram

Class diagram in package 'Exercise 3'

Exercise\_3: Exception Handling and Dynamic Memory Management
Version 1.0
Thomas Batt created on 02.11.2021. Last modified 29.11.2021

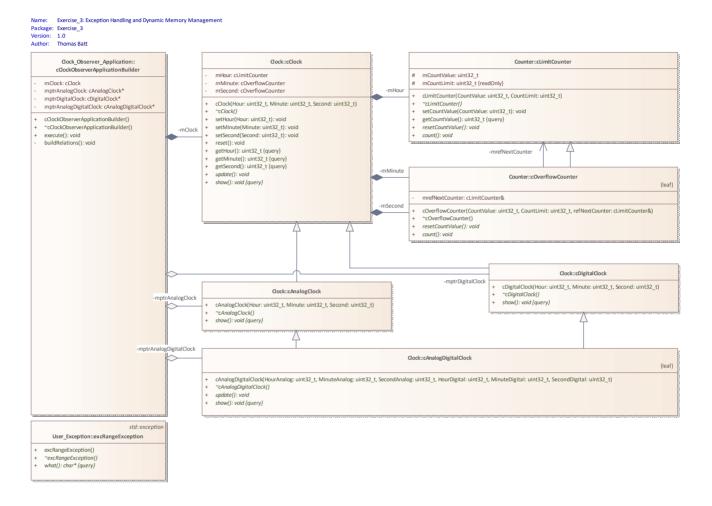


Figure 13: Exercise\_3: Exception Handling and Dynamic Memory Management

# 1.1.1.10 Exercise\_4: Observer Pattern: Template Class, RTTI and Smart Pointer diagram

Class diagram in package 'Exercise 4'

Exercise\_4: Observer Pattern: Template Class, RTTI and Smart Pointer

Version 1.0

Thomas Batt created on 02.11.2021. Last modified 25.02.2022

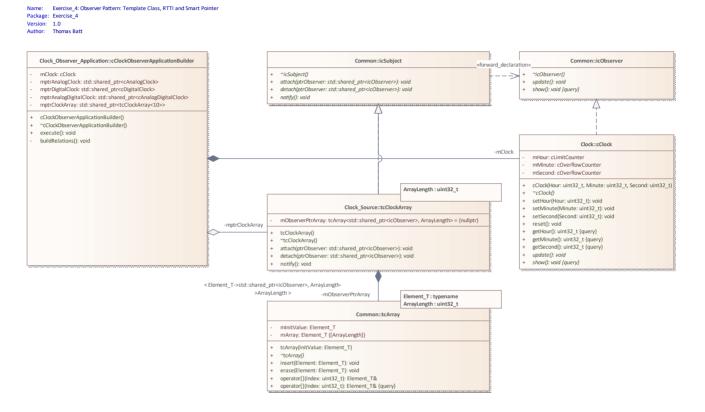


Figure 14: Exercise\_4: Observer Pattern: Template Class, RTTI and Smart Pointer

# 1.1.1.11 Exercise\_5: Observer Pattern: Standard Library Container and Template Exception Class diagram

Class diagram in package 'Exercise 5'

Exercise\_5: Observer Pattern: Standard Library Container and Template Exception Class

Version 1.0

Thomas Batt created on 02.11.2021. Last modified 25.02.2022

Name: Exercise\_5: Observer Pattern: Standard Library Container and Template Exception Class Package: Exercise\_5

Author: Thomas Batt Clock\_Observer\_Application::cClockObserverApplicationBuilder Common::icSubject Common::icObserver mClock: cClock ~icObserver() update(): void show(): void {query} rver: std::shared\_ptr<icObserver>): void rver: std::shared\_ptr<icObserver>): voia impuriangecoci. sci.:shared\_ptr<cli>ptralicIlocks td::shared\_ptr<cli>ptrIloidicolocks td::shared\_ptr<cAnalogDigitalClocks td::shared\_ptr<cAnalogDigitalClocks td::shared\_ptr<cClockArray=10>> mptrClockArray=10>> mptrClockArray=10>> mptrClockContainer: std::shared\_ptr<cClockContainer> Clock::cClock ArrayLength: uint32 t Clock Source::cClockContainer mObserverPtrList: list<std::shared\_ptr<icObserver>> cClockContainer()
"cClockContainer()
"tClockContainer()
attach[ptr:Observer: std::shared\_ptr<icObserver>): void
attach[ptr:Observer: std::shared\_ptr<icObserver>): void
notify(): void Element\_T : typename ArrayLength : uint32\_t Common::tcArray Range\_T : typename ThrowingObject\_T : typename mOutOfRangeValue: Range\_T = 0 mptrThrowingObject: ThrowingObject\_T\* = nullptr extcRangeException() extcRangeException(OutOfRangeValue: Range T, ptrThrowingObject: ThrowingObject T\* const) extchangetxeption()
getOutOfRangeValue(): Range\_T {query}
getThrowingObject(): ThrowingObject\_T\* {query}
what(): char\* {query}

Figure 15: Exercise\_5: Observer Pattern: Standard Library Container and Template Exception Class

#### 1.1.1.12 Exercise 6: Callback, Hardware Driver and Interrupt diagram

Class diagram in package 'Exercise 6'

Exercise\_6: Callback, Hardware Driver and Interrupt

Version 1.0

Thomas Batt created on 02.11.2021. Last modified 25.02.2022

Name: Exercise\_6: Callback, Hardware Driver and Interrupt Package: Exercise\_6 Version: 1.0 Author: Thomas Batt Clock\_Observer\_Application::cClockObserverApplicationBuilder Common::icSubject Common::icObserve Clock\_Ubsetver\_nepro---mclock: cClock
mptrDigitalClock: std::shared\_ptr<CAnalogClock>
mptrDigitalClock: std::shared\_ptr<ColigitalClock>
mptrDigitalClock: std::shared\_ptr<CAnalogDigital
mptrClockArray-std::shared\_ptr<CAnalogDigital
mptrClockArray-std::shared\_ptr<CalcackContainer>
mptrTickKrimer: std::shared\_ptr<CalcAckContainer>
mptrTickTimer: std::shared\_ptr<CalcAckContainer>
mptrTimer: std::shared\_ptr<CalcAckTimer>
mptrTimer: std::shared\_ptr<CalcActTimer>
mptrTimer: std::shared\_ptr</a>
mptrTimer: std::shared\_ptr</a>
mptr ArrayLength : uint32\_t Clock\_Source::tcClockArray Element\_T : typename ArrayLength : uint32\_t -mObserverPtrArray Clock\_Source::cTickTimer mArrayLength: uint32\_t = 2 {readOnly} mptr\_icSubject: std::shared\_ptr<icSubject cTickTimer()
~cTickTimer() Clock\_Source::cClockContainer set\_ptr\_iSubject(std::shared\_ptr<icSubject>, uint32\_t): void tick(): void Clock::cClock notifyInterruptService(): void -mptrCallbackObject NVIC\_Dispatcher::cSTM32F7xxNVIC\_Dispatcher TIMER::cSTM32F7xxTIMER Driver 
$$\label{limit} \begin{split} & mptrCallbackObject: icbinterruptDispatcher* ([InterruptVectorNumberAmount]) \\ & mptrAcknowledgeObject: icbinterruptAcknowledge* ([InterruptVectorNumberAmount]) \\ \end{split}$$
mptrRegisterBaseAddress: volatile cSTM32F7xxTIMER\_Register\* CSTM32F7xxTIMER\_Driver(vuint32\_t)
~CSTM32F7xxTIMER\_Driver()
notifyinterruptAcknowledge(): void
start(): void
stop(): void
dearUiEinterruptFlag(): void init(): void int[]; void registerinterruptCallback(icbinterruptDispatcher\* const, interruptVectorNumber\_t); void urregisterinterruptCallback(interruptVectorNumber\_t); void urregisterinterruptCallback(interruptVectorNumber\_t); void urregisterinterruptAcknowledge(interruptVectorNumber\_t); void urregisterinterruptAcknowledge(interruptVectorNumber\_t); void urregisterinterruptAcknowledge(interruptVectorNumber\_t); void

Figure 16: Exercise\_6: Callback, Hardware Driver and Interrupt