**GPT MODULE**

1. **AUTOSAR:**

The GPT driver is part of the microcontroller abstraction layer (MCAL). It initializes and control the internal General Purpose Timer(s) (GPT) of the microcontroller.

The GPT driver provides services and configuration parameters for

* Starting and stopping hardware timers
* Getting timer values
* Controlling time triggered interrupt notifications, if supported by hardware
* Controlling time triggered wakeup interrupts, if supported by hardware

The tick duration of a timer channel depends on channel specific settings (part of GPT driver) as well as on system clock and settings of the clock tree controlled by the MCU module. The tick duration is not limited by this specification.

The GPT driver only generates time bases. Further time based functionality on driver level is covered by other MCAL modules like:

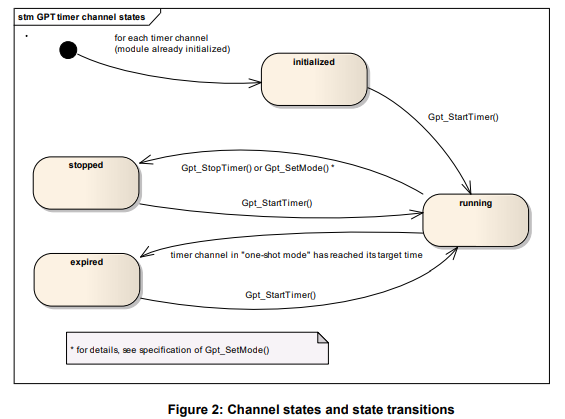
* PWM Driver (driver for pulse width modulation)
* ICU Driver (driver for input capture unit)
* OCU Driver (driver for output compare unit)
* **Dependencies to other modules:**

Module DET[5]

Module MCU[12]

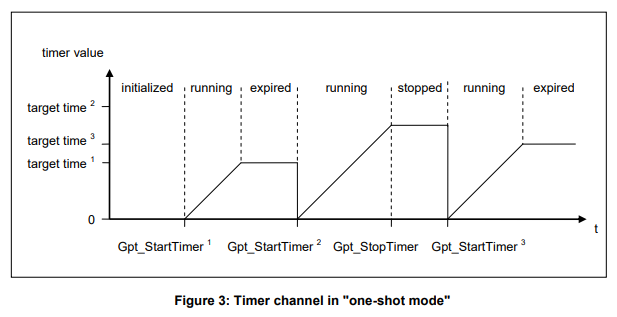
Module EcuM[8]

* **Function specification:**
* **General behavior:** 
  + Starting and Stopping timer:
    - Gpt\_StartTimer : khi start timer thì chọn channel và giá trị target time
    - Gpt\_StopTimer : chỉ cần chọn channel cần stop.

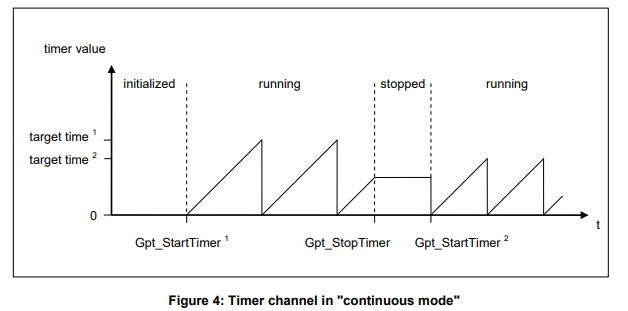


A timer channel can be configured in "one-shot mode"(mode chạy 1 lần) or in "continuous mode"(mode chạy liên tục). Timer channel starts couting at value zero.

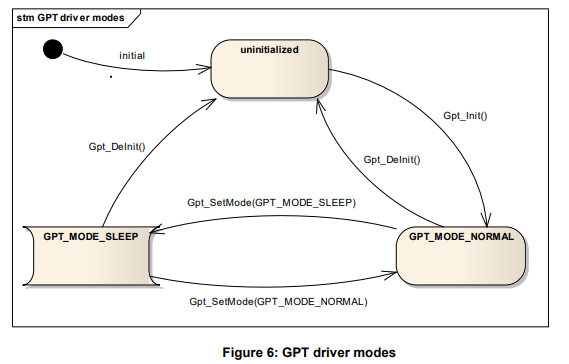
If a timer channel is configured in "one-shot mode": If the timer has reached the target time (timer value = target time), the timer shall stop automatically and maintain its timer value unchanged. The channel state shall change from "running" to "expired".



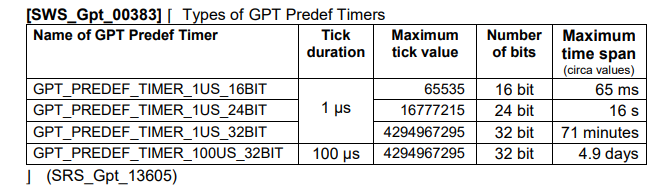
If a timer channel is configured in "continuous mode": If the timer has reached the target time (timer value = target time), the timer shall continue running with the value "0" at next timer tick. So, the time interval of the recurrence is: target time + 1.



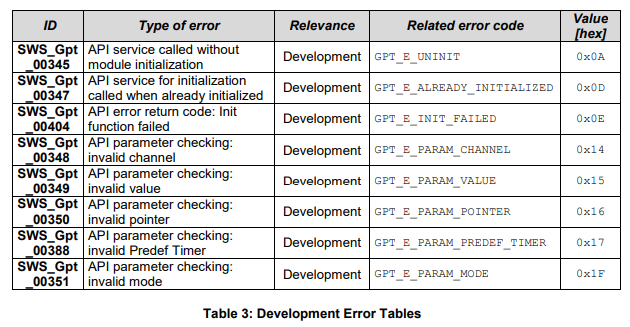
* + Gpt\_GetTimeElapsed && Gpt\_GetTimeRemaining :
    - Gpt\_GetTimeElapsed : lấy thời gian đếm được sau khi start, gọi channel cần lấy.
    - Gpt\_GetTimeRemaining : lấy thời gian còn lại để đạt được target time ( gọi channel cần lấy ).
  + Gpt\_EnableNotification && Gpt\_DisableNotification:
    - Gpt\_EnableNotification : kích hoạt thông báo khi time value = time target.
    - Gpt\_DisableNotification : tắt thông báo khi time value = time target.
* Gpt\_EnableWakeup && Gpt\_DisableWakeup:
* Gpt\_SetMode : Chọn mode hoạt động ( Normal mode hay Sleep mode)



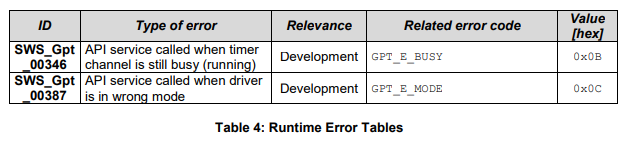
* **GPT Predef Timer :** 
  + A GPT Predef Timer is a free running up counter (user point of view). If the timer has reached the maximum value (max value = 2n -1, n=number of bits), the timer shall continue running with the value "0" at next timer tick.



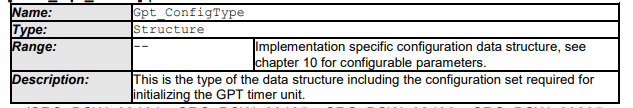
* All enabled GPT Predef Timers run after calling of Gpt\_Init and Gpt\_SetMode(GPT\_MODE\_NORMAL)
* All enabled GPT Predef Timers are stopped by calling of Gpt\_DeInit or Gpt\_SetMode(GPT\_MODE\_SLEEP)
* **Error Classification :** 
  + **Development Error :**



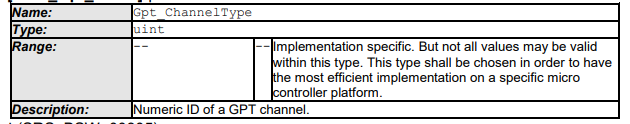
* **Runtime Error:**



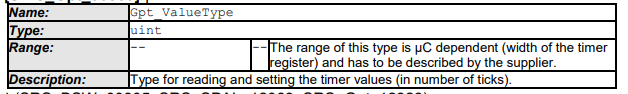
* **API Specification:**
* **Type definition :** 
  + **Gpt\_ConfigType:**



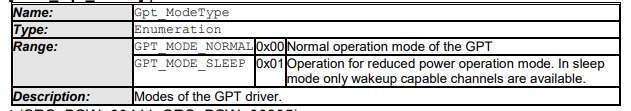
* **Gpt\_ChannelType:**



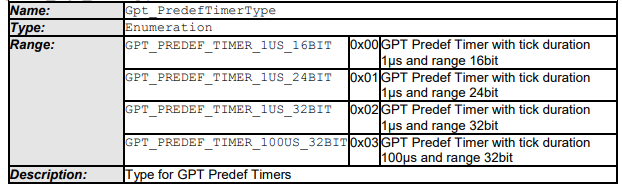
* **Gpt\_ValueType:**



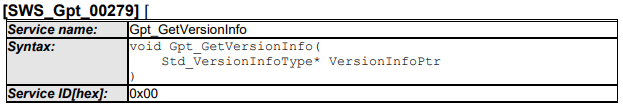
* **Gpt\_ModeType:**

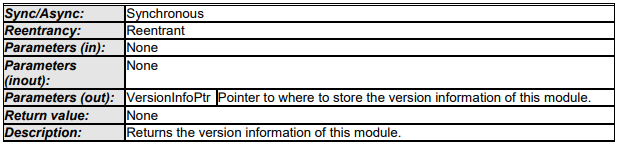


* **Gpt\_PredefTimerType:**



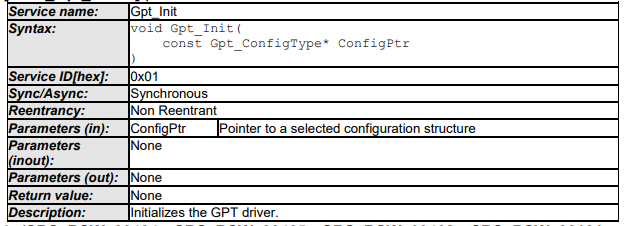
* **Function definition :** 
  + **Gpt\_GetVersionInfo:**





If the parameter VersionInfoPtr is a null pointer, the function Gpt\_GetVersionInfo shall raise the error GPT\_E\_PARAM\_POINTER.

* **Gpt\_Init:**



The function Gpt\_Init shall initialize the hardware timer module according to a configuration set referenced by ConfigPtr.

The function Gpt\_Init shall disable all interrupt notifications, controlled by the GPT driver.

Khởi tạo các thanh ghi :

Module thực hiện chức năng đó có trách nhiệm khởi tạo thanh ghi.

Nếu thanh ghi đó ảnh hưởng đến một số module phần cứng và nếu thanh ghi đó là thanh ghi I/O thì nó sẽ được khởi tạo bởi PORT module , ngược lại nếu không phải thanh ghi I/O thì sẽ được khởi tạo bởi MCU module.

Các thanh ghi có chức năng ghi sẽ được yêu cầu khởi tạo trực tiếp sau khi reset sẽ được khởi tạo bằng code.

Tất cả các thanh ghi khác sẽ được khởi tạo bằng Code.

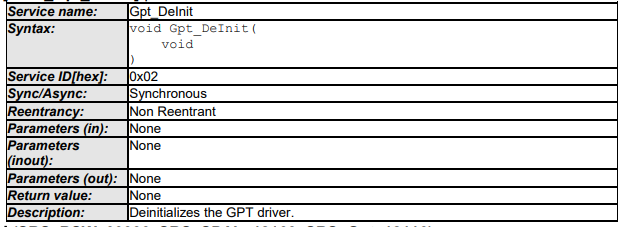
If the GPT driver is not in operation mode "uninitialized", the function Gpt\_Init shall raise the error GPT\_E\_ALREADY\_INITIALIZED.

The function Gpt\_Init shall disable all wakeup interrupts, controlled by the GPT driver.

The function Gpt\_Init shall set the operation mode of the GPT driver to "normal mode".

The function Gpt\_Init shall start all enabled GPT Predef Timers at value “0”.

* **Gpt\_DeInit:**



The function Gpt\_DeInit shall disable all interrupt notifications and wakeup interrupts, controlled by the GPT driver.

The function Gpt\_DeInit shall be pre compile time configurable On/Off by the configuration parameter: GptDeInitApi.

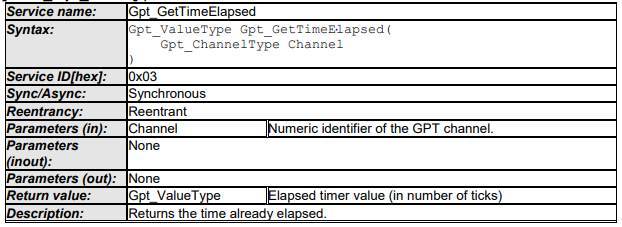
The function Gpt\_DeInit shall set the operation mode of the GPT driver to "uninitialized".

If any timer channel is in state "running", the function Gpt\_DeInit shall raise the runtime error GPT\_E\_BUSY.

If the driver is not initialized, the function Gpt\_DeInit shall raise the error GPT\_E\_UNINIT.

The function Gpt\_DeInit shall stop all enabled GPT Predef Timers.

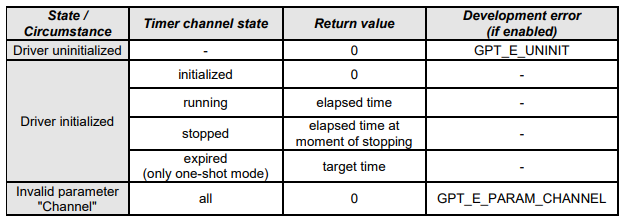
* **Gpt\_GetTimeElapsed:**



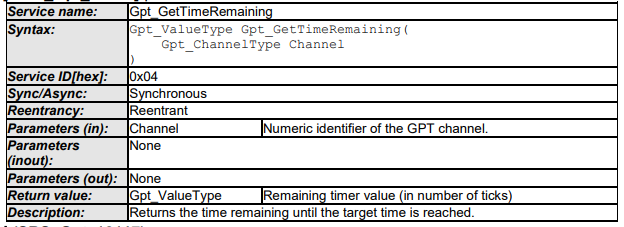
Khi ở chế độ “One-shot Mode” giá trị trả về sẽ là giá trị tính từ thời điểm bắt đầu.

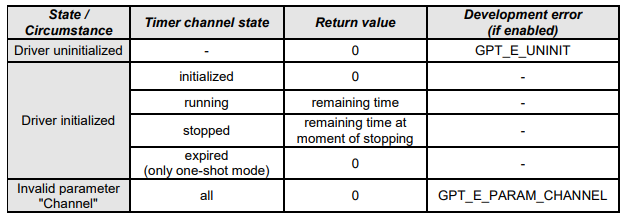
Khi ở chế độ “Continuous Mode” giá trị trả về là giá trị liên quan đến lần lặp lại cuối cùng(giá trị thời gian chạy được ở lần đó) hoặc lần đầu tiên trước khi lặp lại đếm thời gian(thời gian chạy được lần đầu).

Nếu hàm được gọi sau khi gọi hàm khởi tạo thì giá trị trả về là “0” và nếu được gọi ở trong trạng thái “stopped” thì giá trị trả về sẽ là giá trị ngay thời điểm stop đó. Và nếu được gọi ở trạng thái “expired” của chế độ “one-shot mode” thì sẽ trả về giá trị target value.

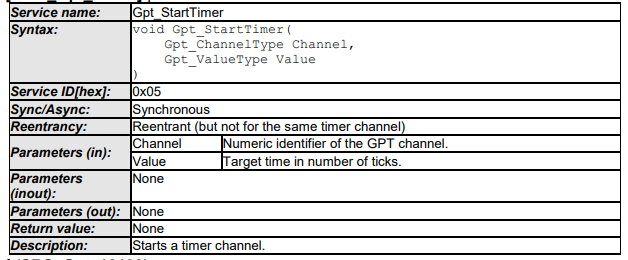


* **Gpt\_GetTimeRemaining:**





* **Gpt\_StartTimer:**



The function Gpt\_StartTimer shall start the selected timer channel with a defined target time.

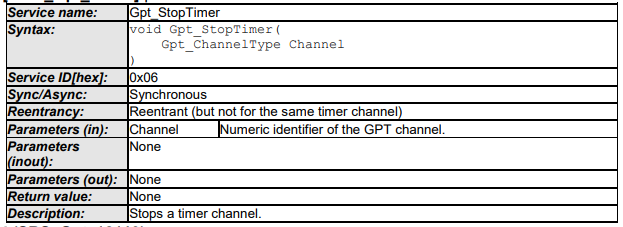
If configured and enabled, an interrupt notification or a wakeup interrupt occurs, when the target time is reached.

If the parameter Channel is invalid (not within the range specified by configuration), the function Gpt\_StartTimer shall raise the error GPT\_E\_PARAM\_CHANNEL.

The function Gpt\_StartTimer shall raise the error GPT\_E\_PARAM\_VALUE if the parameter Value is "0" or not within the allowed range (exceeding the maximum timer resolution). If the driver is not initialized, the function Gpt\_StartTimer shall raise the error GPT\_E\_UNINIT.

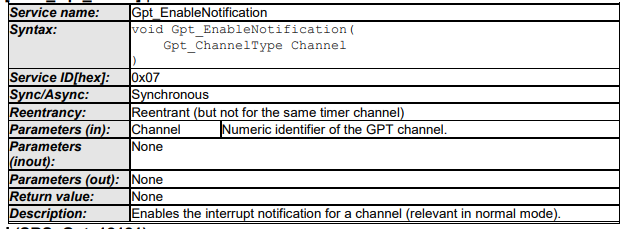
If the function Gpt\_StartTimer is called on a channel in state "running", the function shall raise the runtime error GPT\_E\_BUSY.

* **Gpt\_StopTimer:**



Hàm sẽ chọn timer channel để dừng, trạng thái của kênh đó sẽ được chuyển từ “start”thành “stop”. Nếu hàm được gọi ở trạng thái “stopped”, “expired”, “initialized” thì sẽ ko báo lỗi runtime. Nếu hàm được gọi ở trạng thái chưa khởi tạo “uninitialized” thì sẽ báo lỗi GPT\_E\_UNINIT.

* **Gpt\_EnableNotification:**

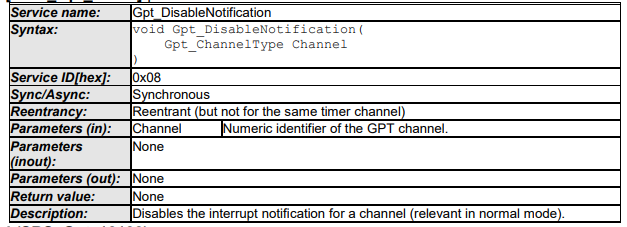


Hàm này chỉ ảnh hưởng đến chế độ “Normal Mode”.

The function Gpt\_EnableNotification shall be pre compile time configurable On/Off by the configuration parameter: GptEnableDisableNotificationApi.

If the driver is not initialized, the function Gpt\_EnableNotification shall raise the error GPT\_E\_UNINIT. If the parameter Channel is invalid (not within the range specified by configuration), the function Gpt\_EnableNotification shall raise the error GPT\_E\_PARAM\_CHANNEL

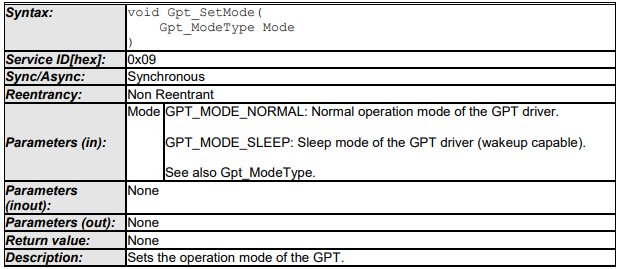
* **Gpt\_DisableNotification:**



Tương tự hàm Gpt\_EnableNotification

* **Gpt\_SetMode:**





The function Gpt\_SetMode is only available if the configuration parameter GptReportWakeupSource is enabled.

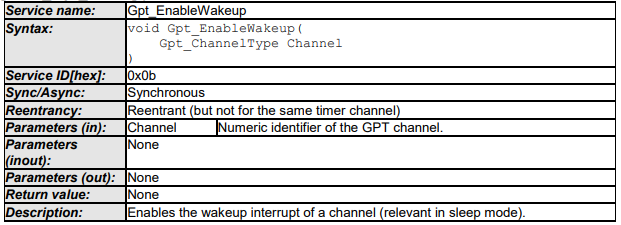
Nếu Mode == GPT\_MODE\_NORMAL thì hàm Gpt\_SetMode sẽ cho phép interrupt notification cho tất cả các channel được cấu hình bằng hàm Gpt\_EnableNotification trước đó. Và hàm Gpt\_SetMode sẽ không reset các Channel đã bị “stopped” trước đó khi vào trạng thái Mode\_Sleep.

Nếu Mode == GPT\_MODE\_SLEEP thì sẽ cho phép sử dụng Wakeup interrupt cho tất cả các Channel nếu được cấu hình trước đó thông qua hàm Gpt\_EnableWakeup. Và tất cả các Channel đang ở trạng thái “running” mà ko được cấu hình Wakeup Interrupt bằng hàm Gpt\_EnableWakeup thì sẽ rơi vào trạng thái “stopped”.

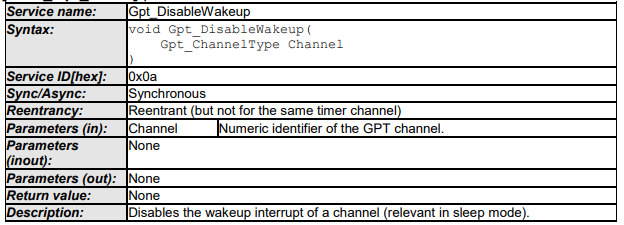
Báo lỗi chưa khởi tạo, Mode is invalid.

Nếu Mode == GPT\_MODE\_NORMAL , mà driver đang ở trạng thái ‘Mode sleep’ thì hàm Gpt\_SetMode sẽ reset GPT Predef Timers at value “0”. Ngược lại, thì hàm Gpt\_SetMode sẽ stop all enabled GPT Predef Timers.

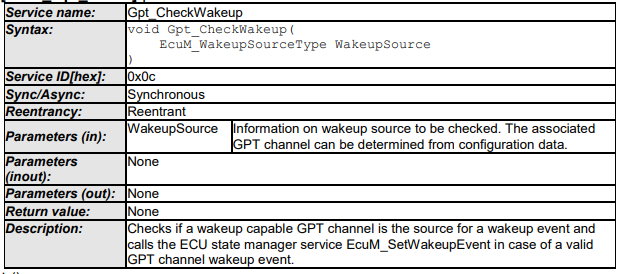
* **Gpt\_EnableWakeup:**



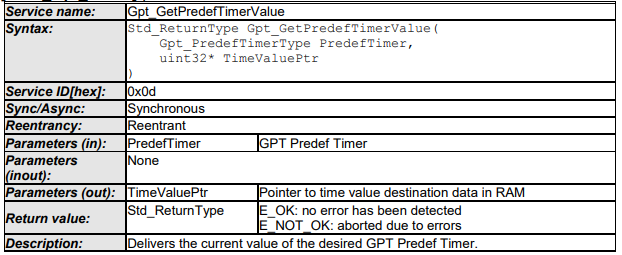
* **Gpt\_DisableWakeup:**



* **Gpt\_CheckWakeup:**



* **Gpt\_GetPredefTimerValue:**



If the timer value of the function Gpt\_GetPredefTimerValue is less than 32 bit (16bit or 24bit timer), the upper bits shall be filled with zero.

Error :

If the GPT driver is not initialized, in “sleep mode” or the GPT Predef Timer is not enabled, the function Gpt\_GetPredefTimerValue shall return E\_NOT\_OK.

If the driver is not initialized, the function Gpt\_GetPredefTimerValue shall raise the error GPT\_E\_UNINIT.

If the parameter PredefTimer is invalid, the function Gpt\_GetPredefTimerValue shall raise the development error GPT\_E\_PARAM\_PREDEF\_TIMER.

If the GPT Predef Timer passed by the parameter PredefTimer is not enabled, the function Gpt\_GetPredefTimerValue shall raise the development error GPT\_E\_PARAM\_PREDEF\_TIMER.

If the driver is in "sleep mode", the function Gpt\_GetPredefTimerValue shall raise the runtime error GPT\_E\_MODE.

If the parameter TimeValuePtr is a null pointer, the function Gpt\_GetPredefTimerValue shall raise the error GPT\_E\_PARAM\_POINTER.