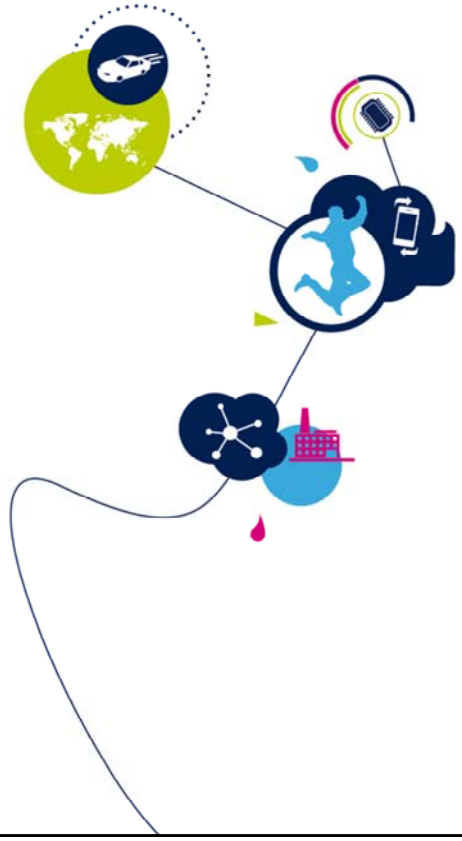
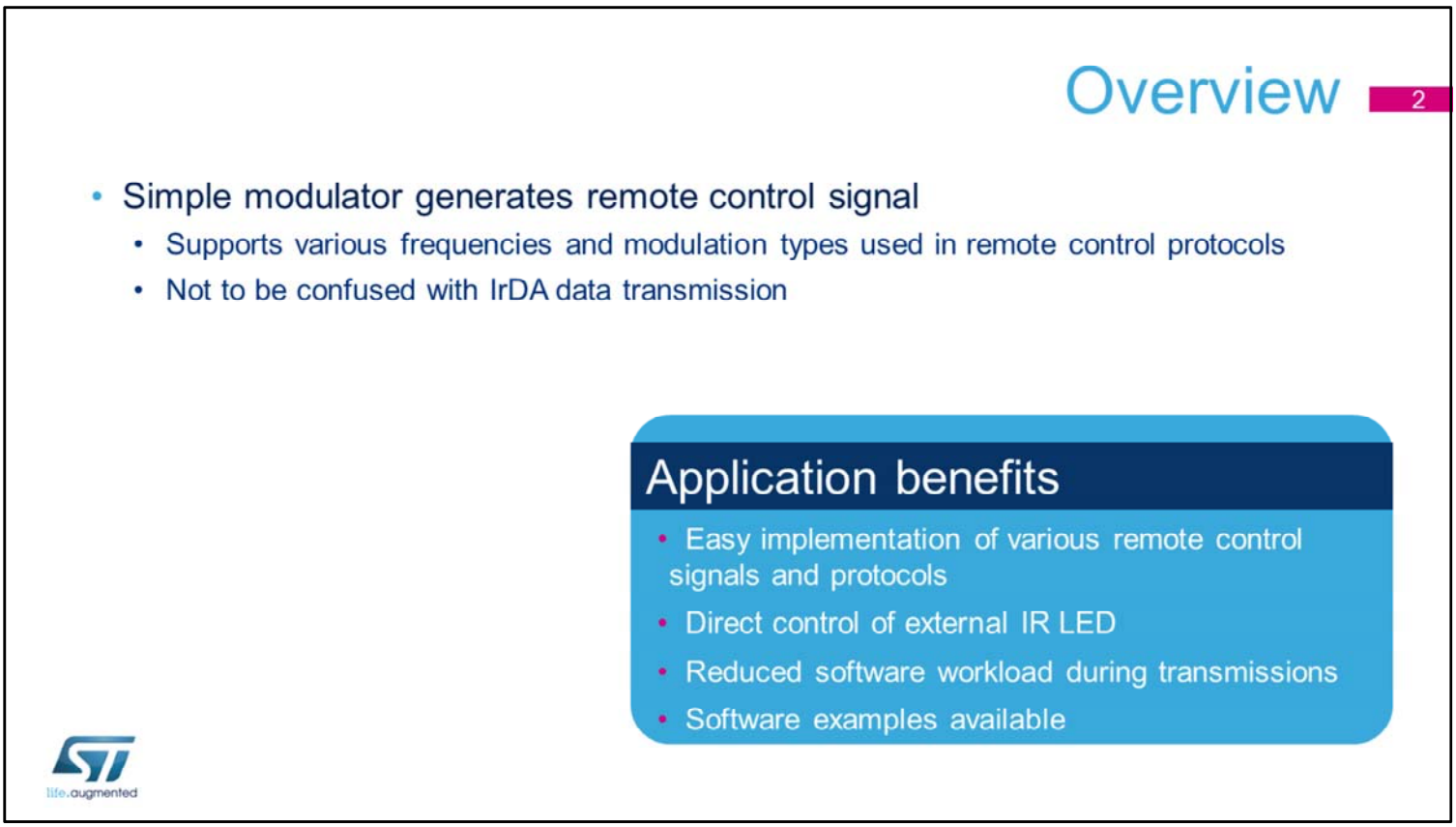
# STM32G4 – IRTIM

Infrared Interface
Revision 1.0



life.augmented

Hello, and welcome to this presentation of the STM32 InfraRed Interface (IRTIM).
It covers the main features of this peripheral, which is used to generate an infrared remote control signal.



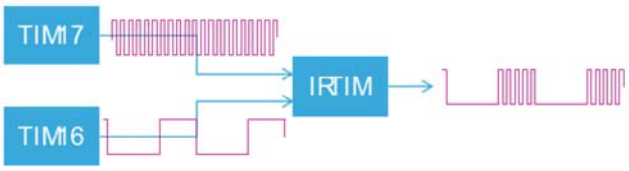

The Infrared Timer peripheral facilitates the generation of infrared remote control protocols, used in many consumer devices such as TV sets, audio systems, air conditioning units, etc.
The Infrared Timer provides a simple modulator to generate the remote control signal, using Timers 16 and 17 to generate the carrier frequency and modulation signal.
The user can configure a wide range of carrier frequencies and modulations to facilitate the implementation of any remote control protocol.

# Key features

3

## Flexible & simple

- Signal generation driven by hardware
  - Configurable carrier frequency
- Protocol flow controlled by software
  - Flexibility to support required protocol




### Application benefits

- Direct drive of IR LED, no additional components needed
- Supports RC5, RC6, RCA, SIRC, …
- Flexible software-generated modulation control to adopt various protocols
- Low software overhead

The Infrared Timer provides hardware support to generate remote control signals. The carrier frequency is generated autonomously by the timer, while the modulation waveform is controlled by software. This allows flexibility to support any required infrared remote control protocol. The Infrared Timer automatically combines the carrier frequency and the modulation waveform into a signal controlling the infrared LED that transmits IR control signals to the controlled device.
The application does not require external transistors, as the infrared LED can be driven directly by the GPIO pin.
Many remote control protocols including RC5, RC6, SIRC and others can be implemented and supported due to flexible and simple modulation control.
The CPU workload is limited to the control of the modulation signal only.

# Main differences with STM32F0 and STM32G0



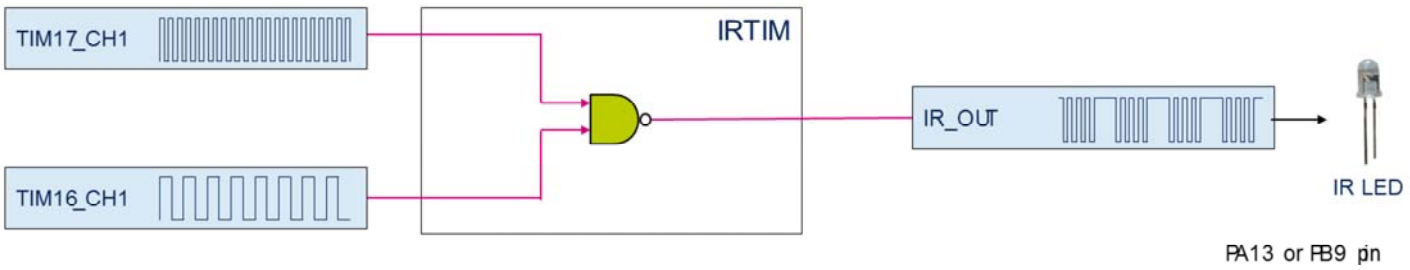
| Functionality | STM32G4<br>STM32L4 | STM32G0 | STM32F0 |
|---|---|---|---|
| IR_OUT polarity selection | NO | YES, through a control bit in SYSCFG | NO |
| Carrier signal | TIM17 | | |
| Modulation envelope | TIM16 | TIM16 or USART1 or USART4 | |

life.augmented

This slide highlights the differences between the STM32F0, the STM32G0 and STM32G4 IRTIM modules. The STM32L4 and STM32G4 support the same IRTIM module.

## IRTIM block diagram

5




- To generate the infrared remote control signals, the IR interface must be enabled and TIM16 channel 1 (TIM16_OC1) and TIM17 channel 1 (TIM17_OC1) must be properly configured to generate correct waveforms

life.augmented

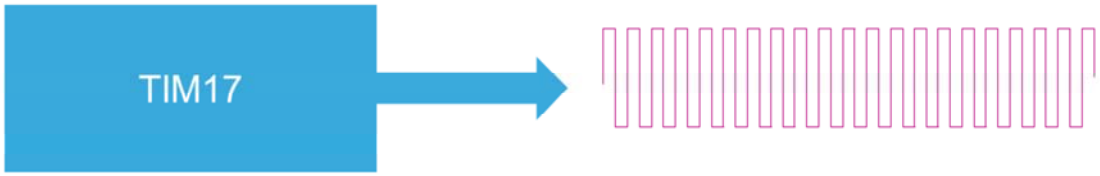
An infrared interface (IRTIM) for remote control is available on the device.
It can be used with an infrared LED to perform remote control functions.
The activation of this function is done through the GPIO alternate function register by enabling the related alternate function bit.
TIM17 is used to generate the high frequency carrier signal, while TIM16 generates the modulation envelope.
The IR output signal can be driven on GPIOs PA13 or PB9.

# Carrier frequency generator



## Compatible with any remote control protocol

- Carrier frequency is generated by TIM17
  - Typical frequency in the range of 34 – 40 kHz
  - Timer configured to PWM mode with 50% duty cycle
  - No interrupt is needed
  - Refer to timer section for detailed description of timer configuration registers




life.augmented

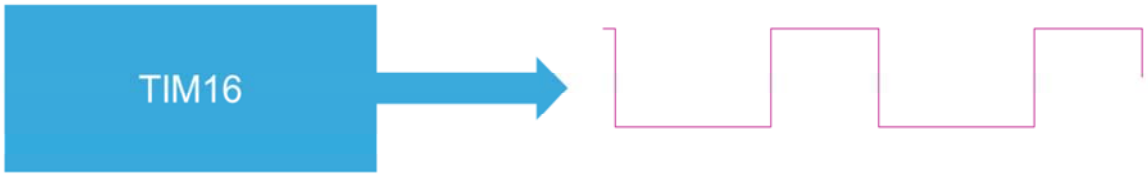
Timer 17 generates the carrier frequency for the remote control protocol used in the application.
The carrier frequency can be configured to any frequency needed by the chosen protocol, including typical frequencies in the range of 34 to 40 kilohertz.
This is done by configuring Timer 17 into PWM mode with a 50% duty cycle.
Once the timer is started, it does not require additional software control including interrupts.
For a detailed description on how to configure Timer 17, please refer to the timer section in the reference manual.

# Modulation waveform generator

7

**Compatible with any remote control protocol**

- Modulation signal is generated by TIM16
  - Typical pulse duration in the range of 100 µs to 100 ms
  - Timer configured to Output Compare mode
  - Output compare interrupt is used to control data flow
  - Refer to timer section for detailed description of timer configuration registers




life.augmented

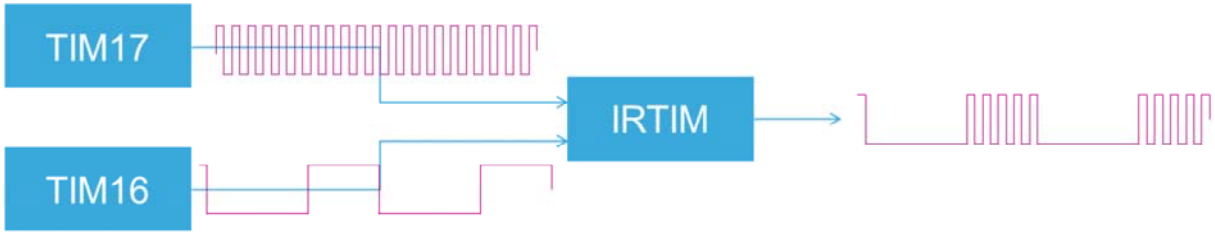
Timer 16 generates the modulation waveform for the remote control protocol used in the application.
The Timer is configured in output compare mode, using the output compare interrupt to generate pulses representing a logical 0 or 1 and control the modulation of the data flow.
For a detailed description of how to configure Timer 16, please refer to timer section in the reference manual.

# IRTIM modulator



## Compatible with any remote control protocol

- IRTIM modulator is a simple gate
- Output signal is created by gating carrier frequency from TIM17 by a modulation signal from TIM16
  - Synchronization with TIM17 is used to avoid spikes and glitches




The Infrared Timer modulator is a simple gate, gating the carrier frequency from Timer 17 by a modulation waveform from Timer 16.
To avoid spikes and glitches on the output waveform, the Infrared Timer gate is synchronized with the carrier frequency from Timer 17.
All standard IR pulse modulation modes can be obtained by programming TIM16 and TIM17 output compare channels.

# IR_OUT pin

9

- IRTIM output (IR_OUT pin) enabled in GPIO_AFR register
- IR_OUT pin supports direct IR LED drive
- The infrared function is output on the IR_OUT pin
  - The high sink LED driver capability (only available on the PB9 pin) can be activated through the I2C_PB9_FMP bit in the SYSCFG_CFGR1 register
    - Up to 20 mA

life.augmented

The Infrared Timer output is an alternate GPIO feature, configurable in the GPIO_AFR registers.
The IR_OUT pin can directly drive the infrared LED, especially on pin PB9, where the high sink driver capability is supported.
This feature can be activated through the system configuration register.
GPIO PA13 does not support this option.

| Interrupt event | Description |
|---|---|
| TIM16 | Used to control modulation flow |

No interrupt is associated directly with the Infrared Timer, however, the Timer 16 interrupt is used to control the modulation of the output signal.

# Low-power modes



| Mode | Description |
|---|---|
| Run | Active |
| Sleep | Active<br>➢ Peripheral interrupts cause the device to exit Sleep mode. |
| Low-power run | Active |
| Low-power sleep | Active<br>➢ Peripheral interrupts cause the device to exit Low-power sleep mode |
| Stop 0/Stop 1 | Frozen<br>➢ Peripheral registers content is kept |
| Standby | Powered-down<br>➢ The peripheral must be reinitialized after exiting Standby mode |
| Shutdown | Powered-down<br>➢ The peripheral must be reinitialized after exiting Standby mode |

life.augmented

The Infrared Timer can be active only in Run and Sleep modes.
In all other low-power modes (including Stop, Standby and Shutdown modes), the Infrared Timer must be disabled.

# Related peripherals

12

- Refer to additional trainings linked to this peripheral:
  - Timers (TIM16 and TIM17 configuration)
  - GPIO (IRTIM_OUT configuration)
  - System configuration (SYSCFG_CFGR1 register)
  - Interconnect matrix (TIM16 and TIM17 connection)

life.augmented

The listed peripherals influence Infrared Timer behavior. Please refer to the additional peripheral trainings for complete information.