STM32G4 Technique Training

- General-purpose input/output interface
- DMA/DMAMUX
- GPTIMER

life.augmented

STM32G4 - GPIO

life.augmented

- Same as F3/L4 with a one difference:
  - In F3/L4, when the GPIO is configured in analog mode, the pull up/pull down are disabled by hardware.

MODE(i) [1:0]	OTYPE(i)	OSPEED(i) [1:0]		PUPD(i) [1:0]		I/O configuration	
01	0	SPEED [1:0]		0	0	GP output	PP
	0			0	1	GP output	PP + PU
	0			1	0	GP output	PP + PD
	0			1	1	Reserved	
	1			0	0	GP output	OD
	1			0	1	GP output	OD + PU
	1			1	0	GP output	OD + PD
	1			1	1	Reserved (GP output OD)	
10	0	SPEED [1:0]		0	0	AF	PP
	0			0	1	AF	PP + PU
	0			1	0	AF	PP + PD
	0			1	1	Reserved	
	1			0	0	AF	OD
	1			0	1	AF	OD + PU
	1			1	0	AF	OD + PD
	1			1	1	Reserved	
00	x	x	x	0	0	Input	Floating
	x	x	x	0	1	Input	PU
	x	x	x	1	0	Input	PD
	x	x	x	1	1	Reserved (input floating)	
11	x	x	x	0	0	Input/output	Analog
	x	x	x	0	1	Reserved	
	x	x	x	1	0	Input/output	Analog, PD
	x	x	x	1	1	Reserved	

  - In G4, it becomes possible to enable/disable the pull down, so the combination PUPD = 10 is no more reserved. The pull up remains disabled by hardware.

Special considerations for I/O pins

- **Only debug pins remain in AF mode under reset**

- During and after reset, the alternate functions are not active
  - I/O ports default to analog mode
  - Saves current consumption during and after reset (Schmitt trigger is off)

- Only JTAG/SWD debug pins remain in AF pull-up/pull-down configuration
  - PA13: JTMS/SWDIO
  - PA14: JTCK/SWCLK
  - PA15: JTDI
  - PB3: JTDO
  - PB4: NJTRST

Special considerations for I/O pins

- **+ one I/O if boot0 is based on option bit**

- PB8 pin can be used as:
  - Boot0 pin
  - Or as standard GPIO

  Depending of the nSWBOOT0 option bit
  - nSWBOOT0 = 0 → Boot0 taken from an option bit nBOOT0, so PB8 is available.
  - nSWBOOT0 = 1 → Boot0 taken from pin, so PB8 is not available.

- The nSWBOOT0 option bit production value will be “0” so that having PB8 available by default.

Special considerations for I/O pins

- **NRST shared with PG10**

- PG10 pin can be used as:
  - NRST
  - Or as standard GPIO

  Depending on the NRST_MODE option bits

- The NRST_MODE option bits production value will be “11” corresponding to Input/Output reset pin NRST (Legacy) → PG10 is not available.

STM32G4 – DMA/DMAMUX

- Direct memory access controller (DMA)
- DMA request multiplexer (DMAMUX)

life.augmented

Main Differences with STM32F3

- The DMA Controller is similar to the one implemented in the STM32F3, but with additional DMAMUX

	STM32F3	STM32G4
DMA	2 DMAs DMA1: 7 channels DMA2: 5 channels	2 DMAs DMA1: up to 8 channels DMA2: up to 8 channels
DMA Features	Same	
DMAMUX	No	Yes

DMAMUX

- DMAMUX is a DMA request multiplexer/router
- DMAMUX provides a programmable routing of any of the 16 DMA (hardware) requests from any peripheral request
- Additionally, there are 4 request generators
  - Software can configure a DMA request to be generated by the DMAMUX itself, upon a trigger input. The following are programmable:
    - The trigger selection: EXTI0…EXTI15, LPTIM1OUT, or any of the 4 generated DMAMUX events
    - The trigger event: rising edge, falling edge or either edge
    - The number of generated DMA requests upon the trigger event
  - There is a trigger overrun flag & interrupt in order to alert the software when the number of generated DMA requests have not been completed before a next trigger event

STM32G4 DMA & DMAMUX implementation

DMAMUX features	DMAMUX
Number of peripheral requests	115
Number of request generators	4
Number of triggers	21
Number of synchronizations	21
Number of output DMA requests	16 (12 In G431)

DMA features	DMA1
Number of channels	8 (6 in G431)

DMA features	DMA2
Number of channels	8 (6 in G431)

DMAMUX Interrupt

Interrupt event	Description
Request generator trigger overrun	Set when there is a detected trigger input overrun before that the programmed number of DMA requests created by the DMAMUX request generator has been completed
Request multiplexer synchronization overrun	Set when there is a detected synchronization input overrun before that a programmed number of transmitted DMA requests or generated DMAMUX events has been completed

GPTIMER v4.0

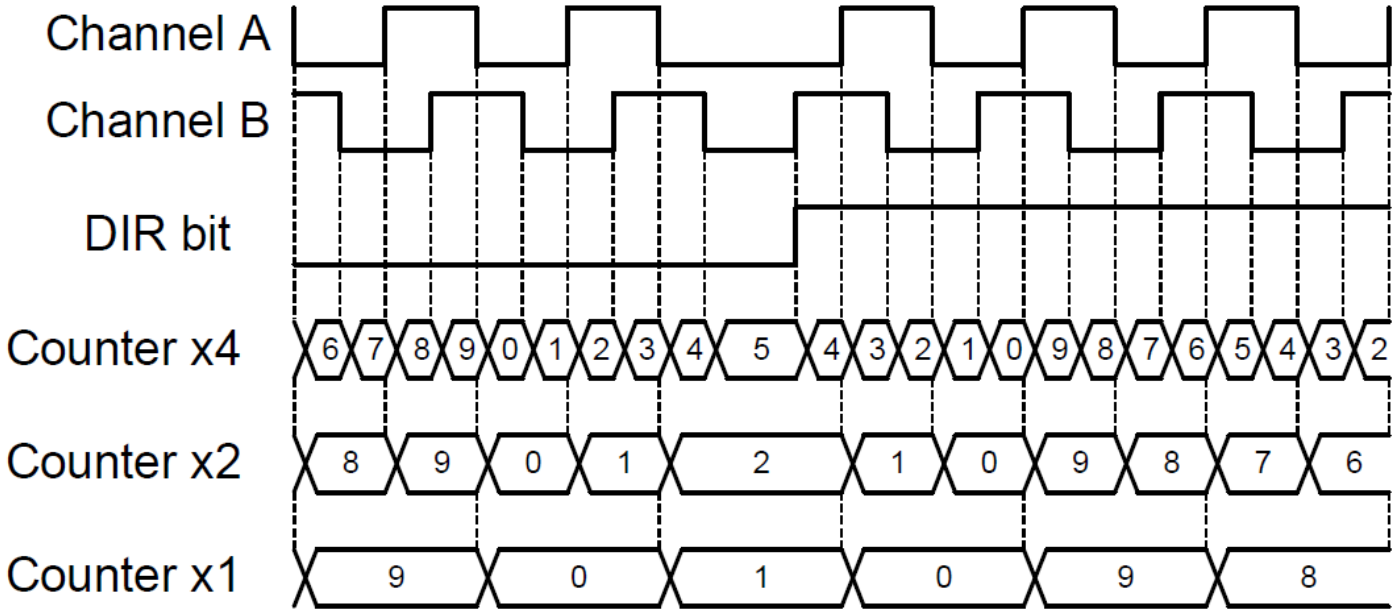
- New features

Summary

- Motor Control + Digital potentiometers
  - Better management of incremental encoder sensors
    - 2 new protocols supported
    - Hardware management of the Index (Z) input
- Power conversion
  - Improvements on deadtime generator
  - Higher (average) resolution using hardware dithering
- General purpose
  - Higher (average) resolution using hardware dithering
  - New slave mode (gated+reset)
- SoC level
  - Higher flexibility interconnect
  - Bidirectional break

New quadrature encoder counting modes

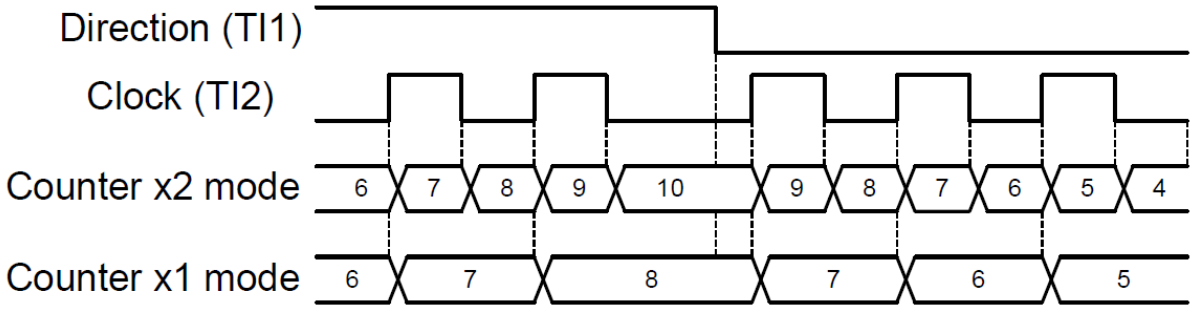
- x1 encoding mode added (x2 and x4 mode already supported today)
  - CPU burden decreased for high RPM applications



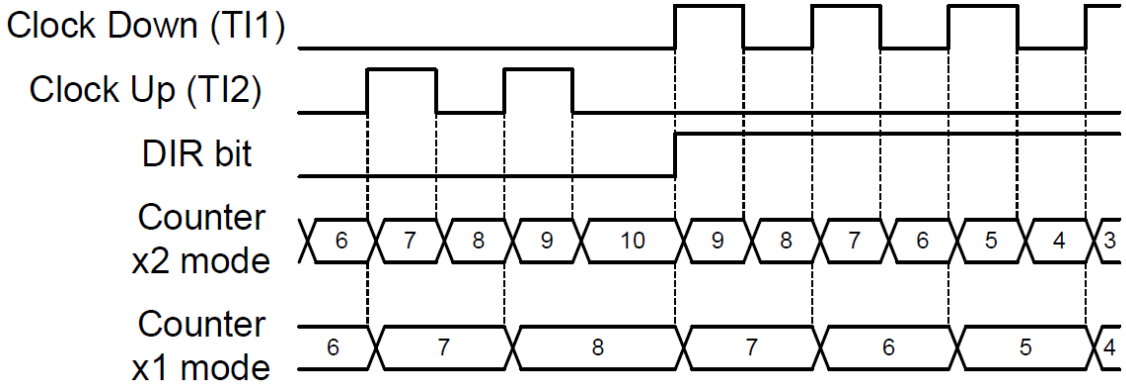
- Slave mode selection preload allows run-time encoder mode update

2 new encoder modes

- Clock plus direction

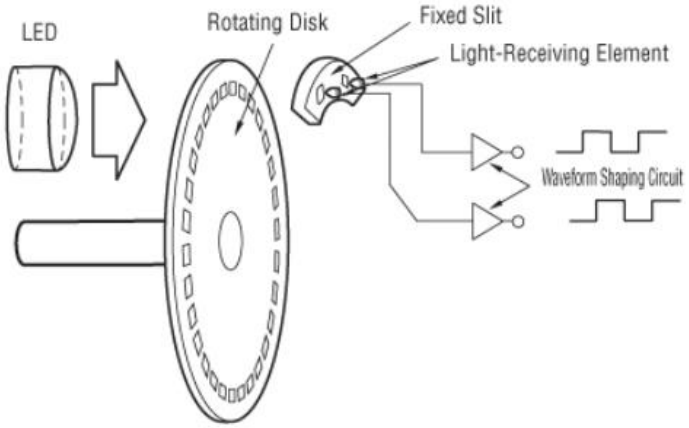


- Directional clock



Encoder

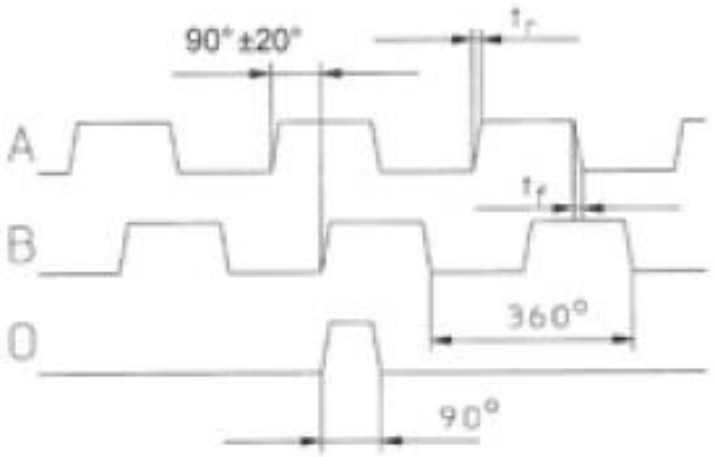
- Can be incremental or absolute for motor drives



- Incremental encoders are cheaper, but do not provide a reference position at power-up

Encoder index

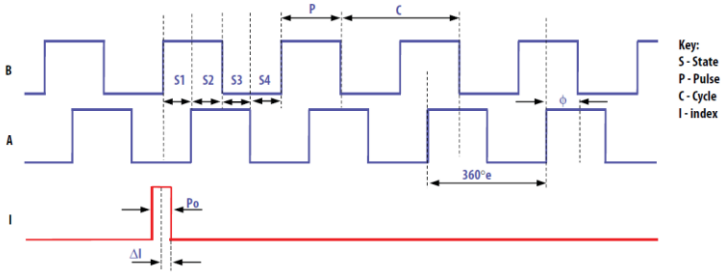
- In addition to the two channels A and B, an Index (also-called zero) pulse is available, which occurs once per revolution and is usually used for the reference run (zero point calibration) of a machine



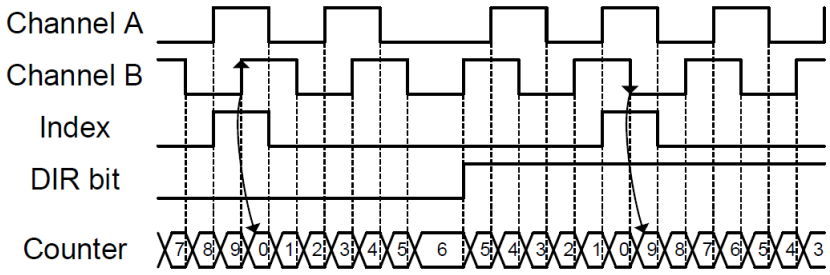
- So far possible this was possible using external interrupts and software, this additional CPU burden

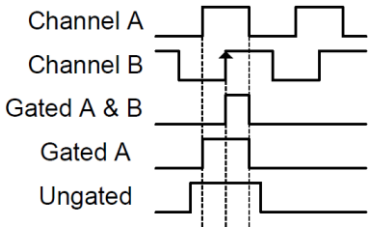
Hardware support of encoder's Index

- Supports all encode modes (quadrature mode below)



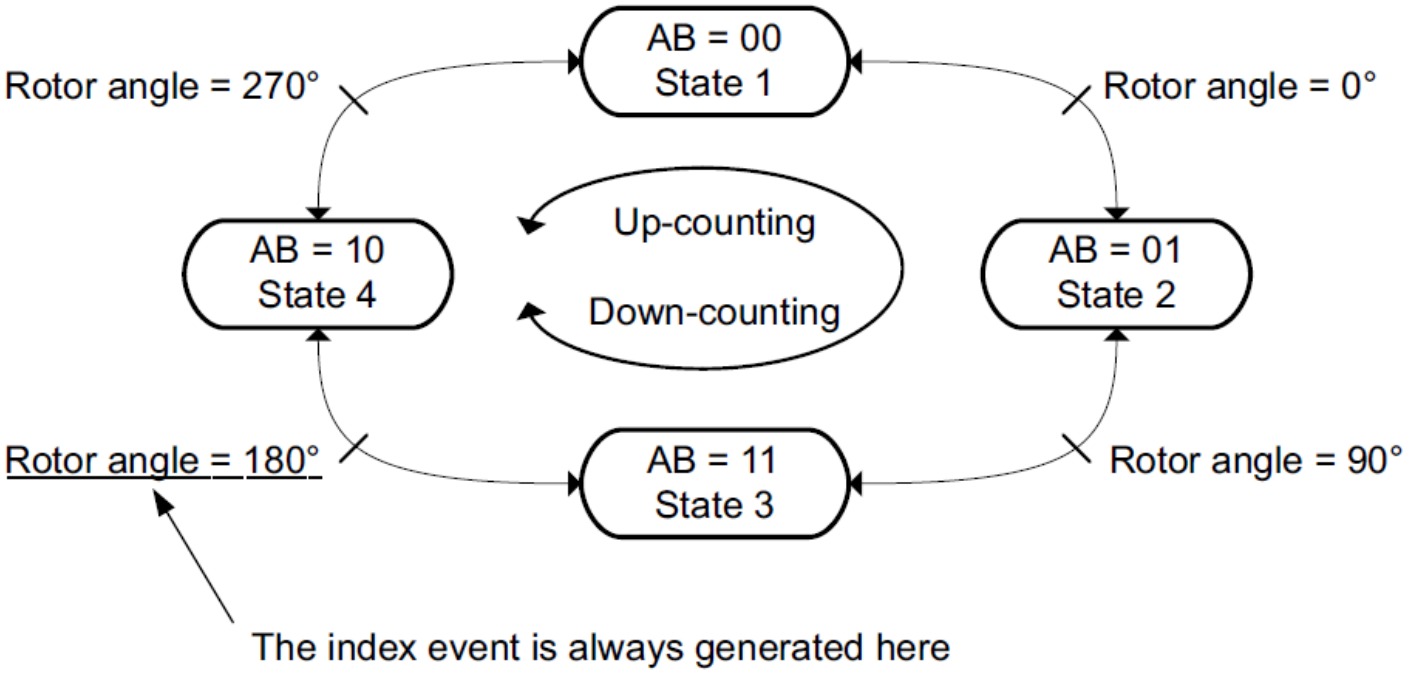
- The index resets the counter (with any gating option)





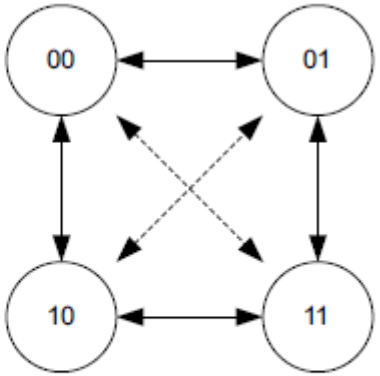
Index detection

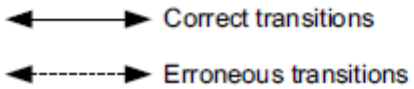
- 2 bits (IPOS[1:0]) are defining on which state is the index detected
- The Index detection is done differently depending on counting direction to ensure symmetrical operation during speed reversal:
  - The counter is reset during up-counting (DIR bit = 0).
  - The counter is set to TIMx_ARR when down counting.



Encoder Error detection

- Quadrature phase error detection
  - In case of forbidden transition (e.g. 01 to 10)

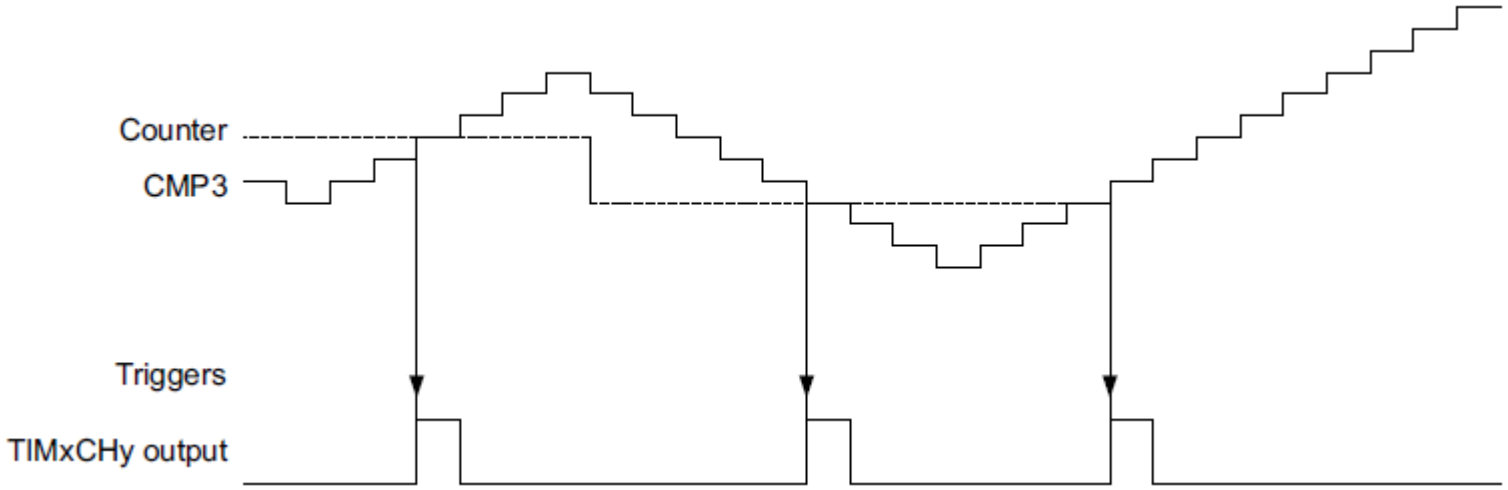




- Position counter error detection
  - If the counter value is incremented from TIMx_ARR to 0 or decremented from 0 to TIMxARR value without any index event, this will be reported as an Index position error
  - Indicates some pulses were added / missed due to external noise, wiring issues, dusty sensor,…

Other encoder related functions

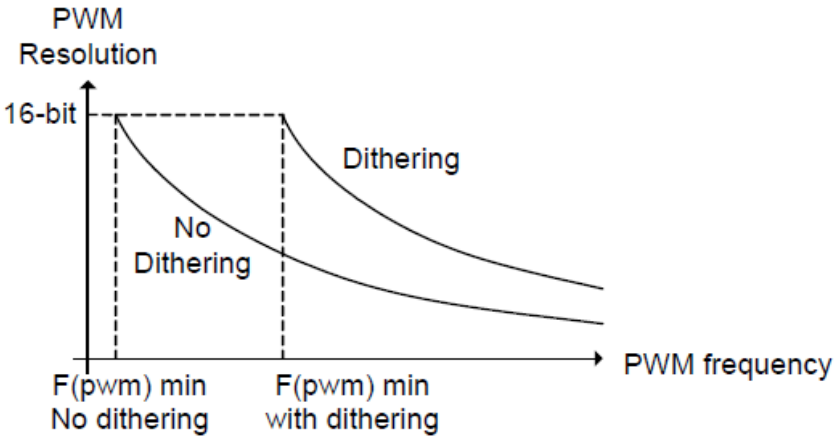
- Position compare sync event (e.g. generate a pulse when rotor position = 90°)
  - An output compare channel generates a pulse on match (Width programmable, as well as polarity)

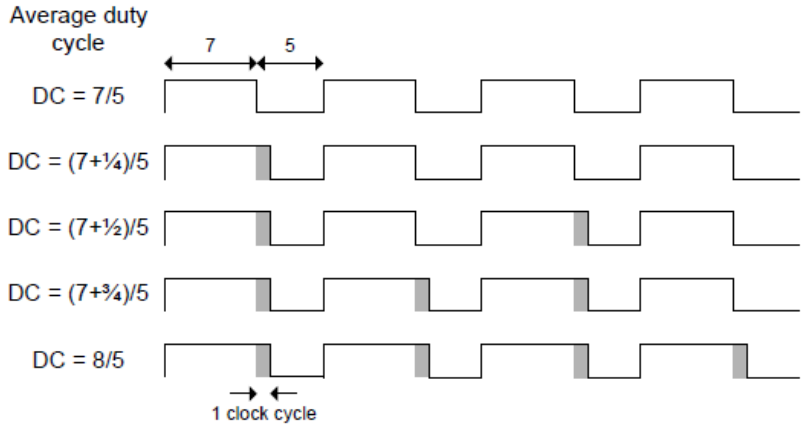


- Direction change interrupt added
  - When the DIR status bit is toggling

Higher resolution with hardware dithering

- Basic idea is to have additional 4-bits to go from 6.66ns (@170MHz) down to the sub-ns range (6.66/16 ns = 367ps average)
- Applicable to both period and duty cycle, adding +0/+1 clock cycle depending on programming

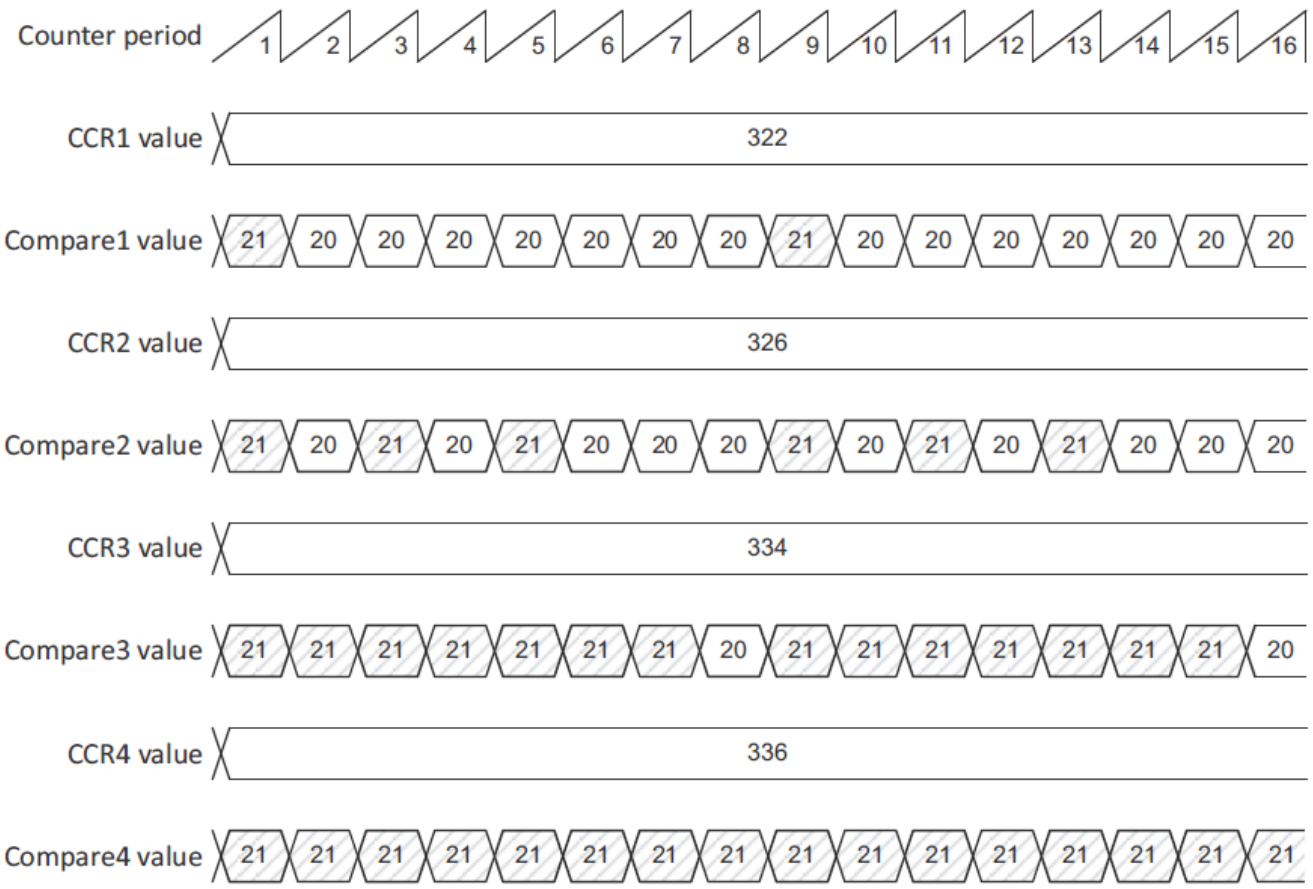




See AN4507 and the X-CUBE-PWM-DITHR for software implementation

16-cycles dithering pattern

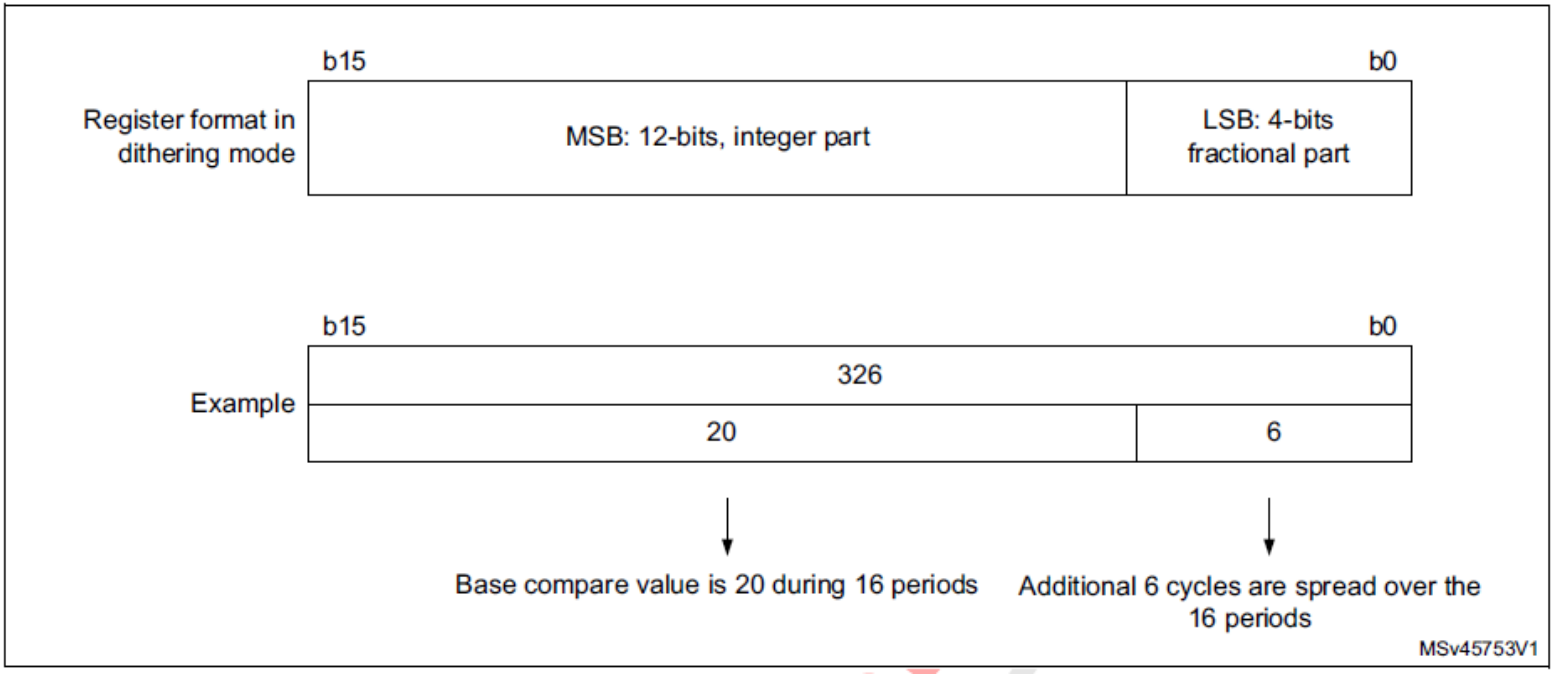
- 322 = 16x20 + 2
- 326 = 16x20 + 6
- 334 = 16x20 + 15
- 336 = 16x21



Dithering mode, practically

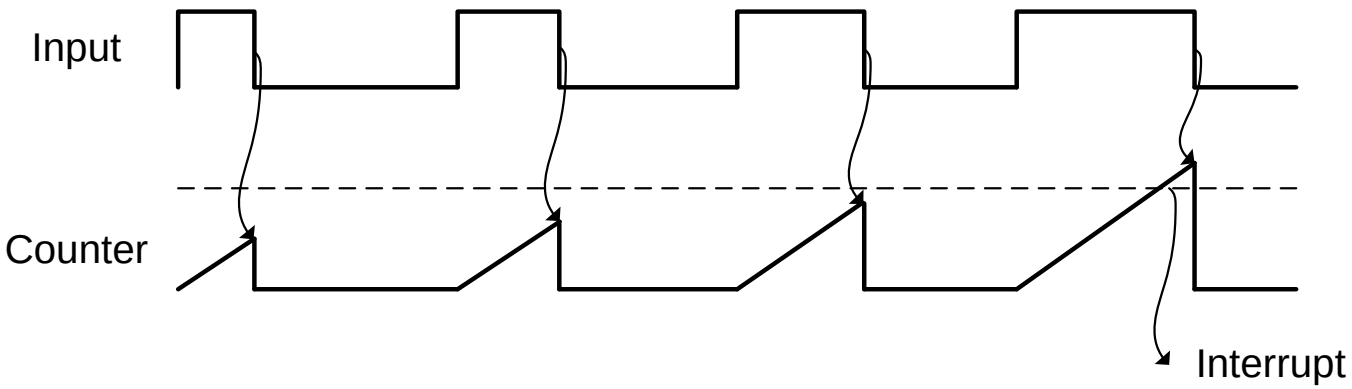
- The 12.4 format
  - The integer part ranges from 0 to 4095, the fractional from 0 to 0.9375 (with 15 steps of 0.0625)

Figure 297. Data format and register coding in dithering mode



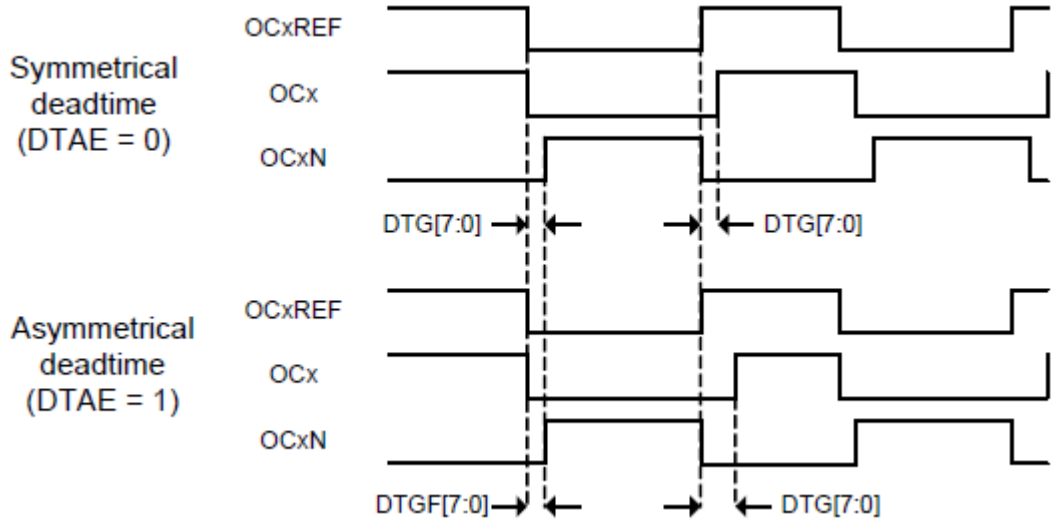
Combined gated + reset mode

- Useful to detect a duty cycle above a given threshold, without having interrupts systematically
  - The timer counts during the high level and is reset on the falling edge. If the counter exceeds a compare value, an interrupt is issued
- Use case: when monitoring a fly-back controller, issues an interrupt in case of control out of expected conditions (indicating a imminent power failure)
  - When the mains disappears, the SMPS controller will increase the duty cycle to compensate the input voltage drop. Detecting an out-of-bound duty cycle condition will allow an early power failure detection.

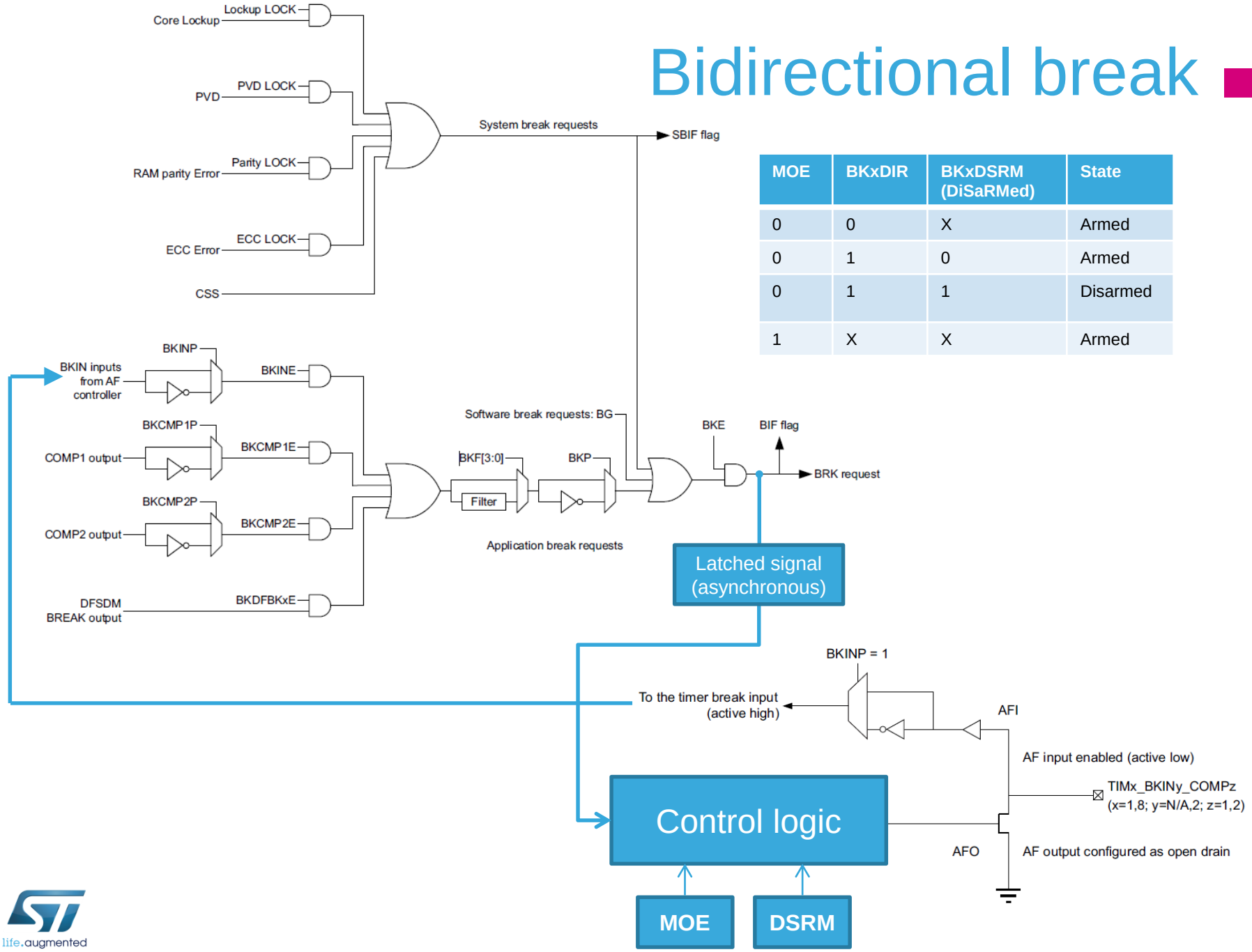


Deadtime

- Asymmetrical: new register with enable bit for supporting both legacy mode (symmetrical) and the new mode
  - For applications having asymmetrical gate driver / optocouplers propagation delays
- Shadow register for on-the fly deadtime update (adaptive deadtime schemes)



Bidirectional break



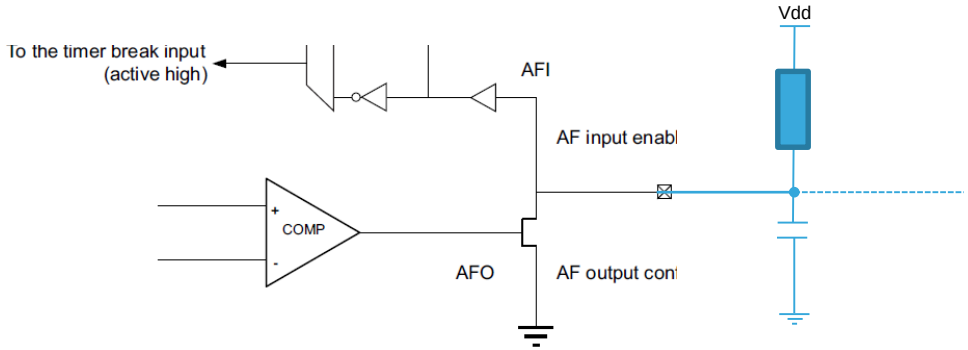
MOE	BKxDIR	BKxDSRM (DiSaRMed)	State
0	0	X	Armed
0	1	0	Armed
0	1	1	Disarmed
1	X	X	Armed

Functionality

- The break input can be disarmed only in bidirectional mode, when MOE is reset:

MOE	BKxDIR	BKxDSRM (DiSaRMed)	State
0	0	X	Armed
0	1	0	Armed
0	1	1	Disarmed
1	X	X	Armed

- The purpose is to be able to have the input voltage back to high-state, whatever the time constant on the output (depending on pull-up resistor and potential filtering capacitor



Orthogonality in timer interconnect

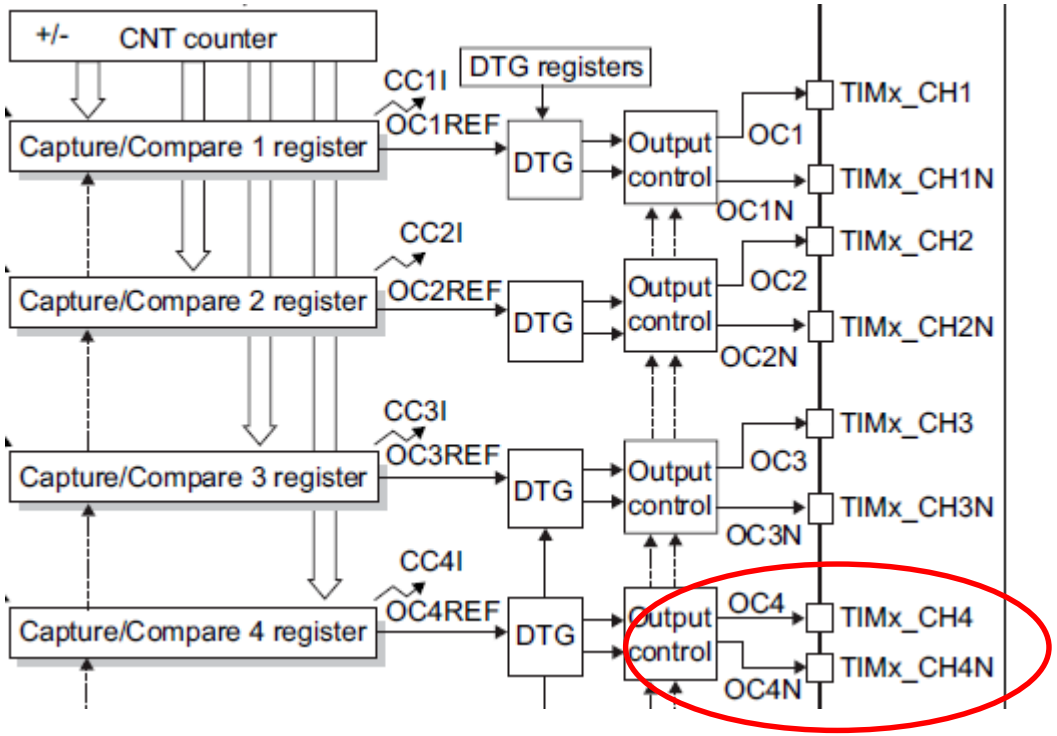
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM15	TIM20
ITR0		TIM1 TRGO	TIM1 TRGO	TIM1 TRGO	TIM1 TRGO	TIM1 TRGO	TIM1 TRGO	TIM1 TRGO
ITR1	TIM2_TRGO		TIM2 TRGO	TIM2 TRGO	TIM2 TRGO	TIM2 TRGO	TIM2 TRGO	TIM2 TRGO
ITR2	TIM3 TRGO	TIM3 TRGO		TIM3 TRGO	TIM3 TRGO	TIM3 TRGO	TIM3 TRGO	TIM3 TRGO
ITR3	TIM4 TRGO	TIM4 TRGO	TIM4 TRGO		TIM4 TRGO	TIM4 TRGO	TIM4 TRGO	TIM4 TRGO
ITR4	TIM5 TRGO	TIM5 TRGO	TIM5 TRGO	TIM5 TRGO		TIM5 TRGO	TIM5 TRGO	TIM5 TRGO
ITR5	TIM8 TRGO	TIM8 TRGO	TIM8 TRGO	TIM8 TRGO	TIM8 TRGO		TIM8 TRGO	TIM8 TRGO
ITR6	TIM15 TRGO	TIM15 TRGO	TIM15 TRGO	TIM15 TRGO	TIM15 TRGO	TIM15 TRGO		TIM15 TRGO
ITR7	TIM16 OC	TIM16 OC	TIM16 OC	TIM16 OC	TIM16 OC	TIM16 OC	TIM16 OC	TIM16 OC
ITR8	TIM17 OC	TIM17 OC	TIM17 OC	TIM17 OC	TIM17 OC	TIM17 OC	TIM17 OC	TIM17 OC
ITR9	TIM20 TRGO	TIM20 TRGO	TIM20 TRGO	TIM20 TRGO	TIM20 TRGO	TIM20 TRGO	TIM20 TRGO	

	TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM20
ETR1	Input pins	Input pins	Input pins	Input pins	Input pins	Input pins	Input pins
ETR2	COMP1	COMP1	COMP1	COMP1	COMP1	COMP1	COMP1
ETR3	COMP2	COMP2	COMP2	COMP2	COMP2	COMP2	COMP2
ETR4	COMP3	COMP3	COMP3	COMP3	COMP3	COMP3	COMP3
ETR5	COMP4	COMP4	COMP4	COMP4	COMP4	COMP4	COMP4
ETR6	COMP5	COMP5	COMP5	COMP5	COMP5	COMP5	COMP5
ETR7	COMP6	COMP6	COMP6	COMP6	COMP6	COMP6	COMP6
ETR8	COMP7	COMP7	COMP7	COMP7	COMP7	COMP7	COMP7

	TIM1	TIM2	TIM3	TIM8	TIM15	TIM16	TIM17	TIM20
Ocref_clr0	COMP1	COMP1	COMP1	COMP1	COMP1	COMP1	COMP1	COMP1
Ocref_clr1	COMP2	COMP2	COMP2	COMP2	COMP2	COMP2	COMP2	COMP2
Ocref_clr2	COMP3	COMP3	COMP3	COMP3	COMP3	COMP3	COMP3	COMP3
Ocref_clr3	COMP4	COMP4	COMP4	COMP4	COMP4	COMP4	COMP4	COMP4
Ocref_clr4	COMP5	COMP5	COMP5	COMP5	COMP5	COMP5	COMP5	COMP5
Ocref_clr5	COMP6	COMP6	COMP6	COMP6	COMP6	COMP6	COMP6	COMP6
Ocref_clr6	COMP7	COMP7	COMP7	COMP7	COMP7	COMP7	COMP7	COMP7
Ocref_clr7								

4th complementary pair in TIM1/TIM8/TIM20

- A 4th complementary pair is now available, with the TIMx_CH4N pin
  - The programming is identical to the other TIMx_CHyN pins
- This allows to control 2 full-bridges with a single advanced timer



life.augmented

Summary, by application fields

- Motor Control + Digital potentiometers
  - Better management of incremental encoder sensors
    - 2 new protocols supported
    - Hardware management of the Index (Z) input
- Power conversion
  - Improvements on deadtime generator
  - Higher (average) resolution using hardware dithering
    - Finer regulation capability
- General purpose
  - Higher (average) resolution using hardware dithering
    - Typically for DAC emulation
  - New slave mode (gated+reset)

Releasing Your Creativity

/STM32 @ST_World community.st.com stm32g4-online-training

www.st.com/STM32G4

life.augmented