

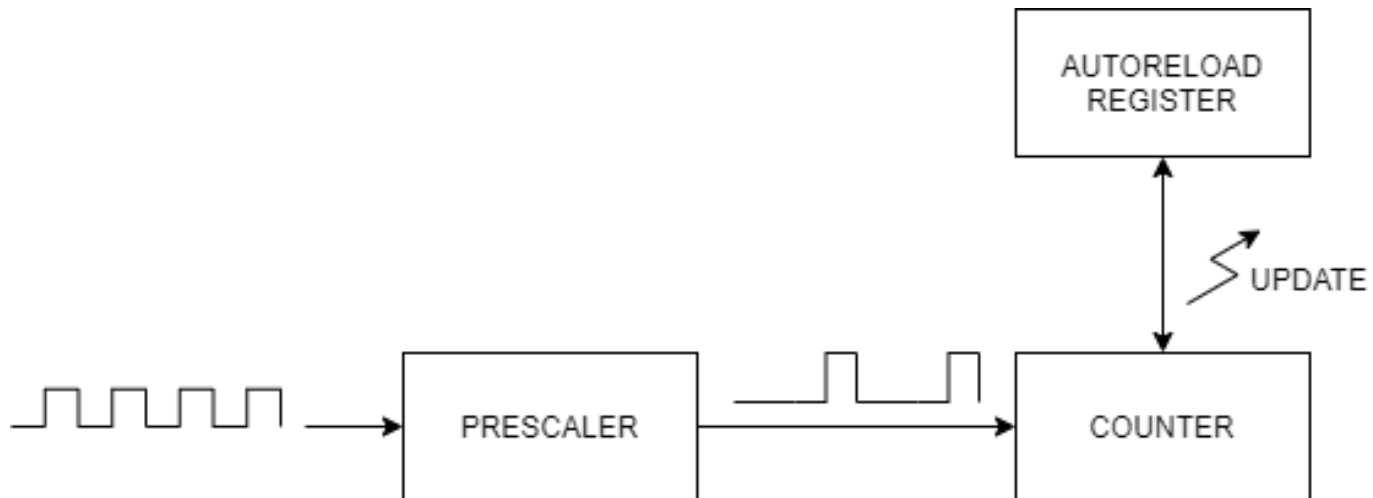
Časovníki

ORS

Časovníki (timers)

- STM32F4 & časovníki
 - 2 naprednejša časovníka (advanced)
 - TIM1, TIM8
 - **10 splošno-namenskih časovníkov (general-purpose)**
 - TIM2, TIM3, **TIM4**, TIM5, TIM9, TIM10, TIM11, TIM12, TIM13, TIM14
 - 2 osnovna časovníka (basic)
 - TIM6, TIM7
- Osnovna ura časovníkov
 - Odvisna od ure vodila
 - APB1: TIM1, 8, 9 in 12
 - APB2: preostali

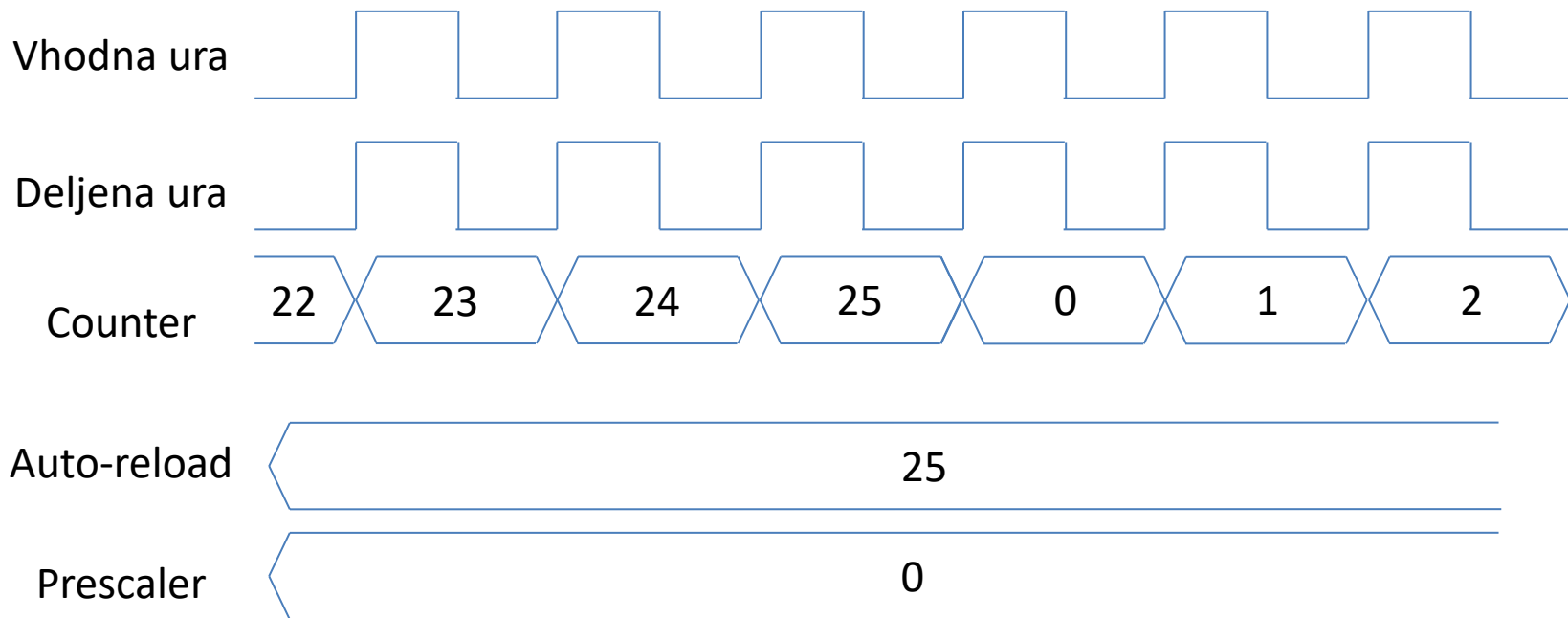
Osnovna funkcija



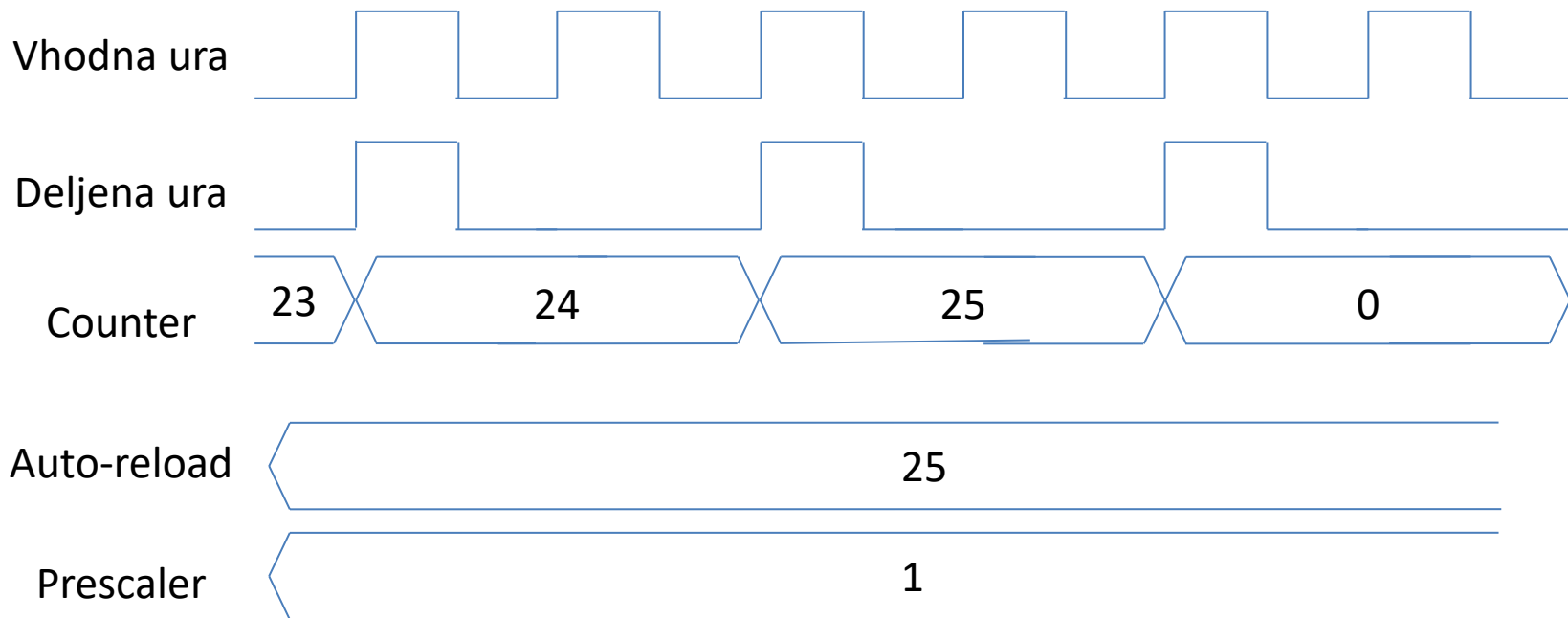
Osnovna funkcija

- Štetje v **counter** registru
 - 16 biten register
 - štetje navzgor, navzdol ali navzgor & navzdol
- Frekvenca štetja je odvisna od izbire vhodne ure in nastavitve **prescaler**-ja
 - prescaler je delilnik ure
 - vrednost registra = 0 -> ni deljenja
 - vrednost registra = 1 -> polovična frekvenca ure
 - vrednost registra = 2 -> frekvenca ure zmanjšana na tretjino
- Štetje se resetira, ko prištejemo do vrednosti, ki je zapisana v **auto-reload** registru (ARR)

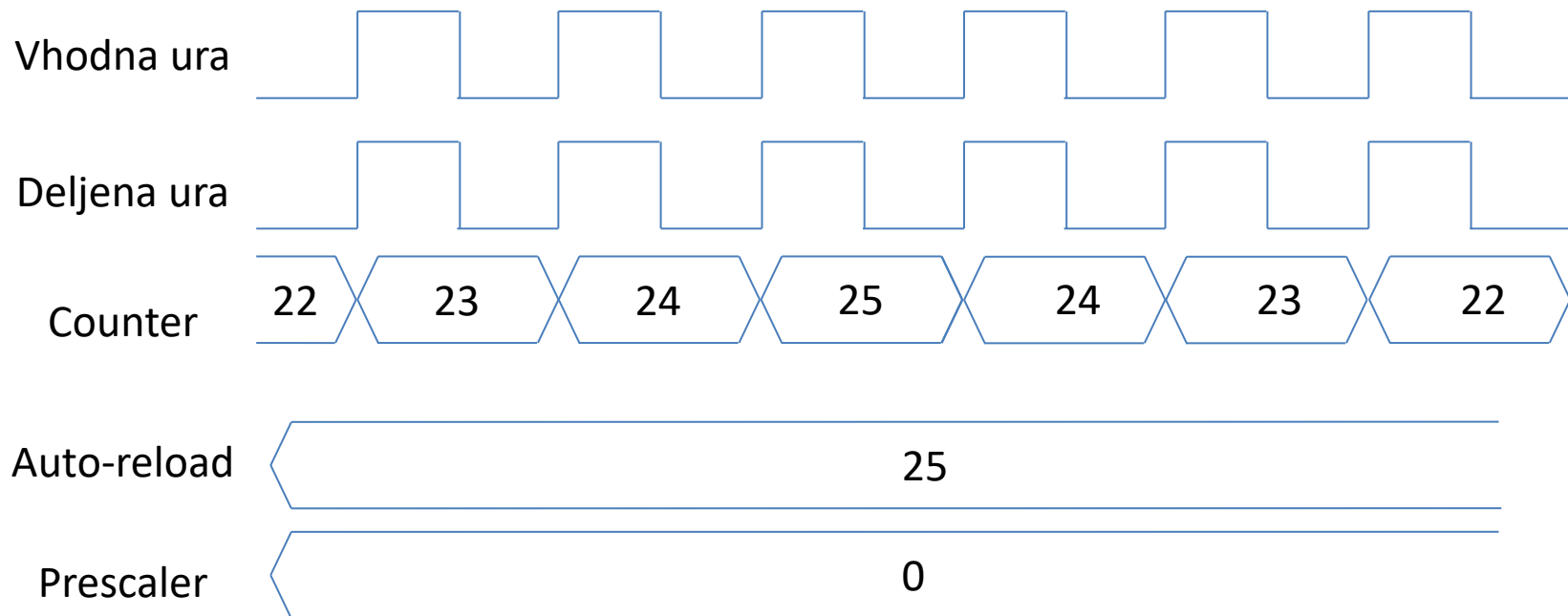
Časovni diagram – štetje navzgor



Časovni diagram – štetje navzgor



Časovni diagram štetje navzgor & navzdol



Inicializacija časovnika (na primeru TIM4)

1. Vklopimo uro časovnika

```
__HAL_RCC_TIM4_CLK_ENABLE();
```

2. Ustvarimo init strukturo in določimo vrednosti

```
TIM_HandleTypeDef timer4;
```

```
timer4.Instance = TIM4;
```

```
timer4.Init.CounterMode = TIM_COUNTERMODE_UP;
```

```
timer4.Init.Period = 1000 - 1;
```

```
timer4.Init.Prescaler = 16000 - 1;
```

```
HAL_TIM_Base_Init(&timer4);
```

3. Zaženemo časovnik

```
HAL_TIM_Base_Start(&timer4);
```


Zastavice

- Branje stanja zastavic

```
__HAL_TIM_GET_FLAG(&timer,zastavica)
```

- Brisanje zastavice

```
__HAL_TIM_CLEAR_FLAG(&timer, zastavica);
```

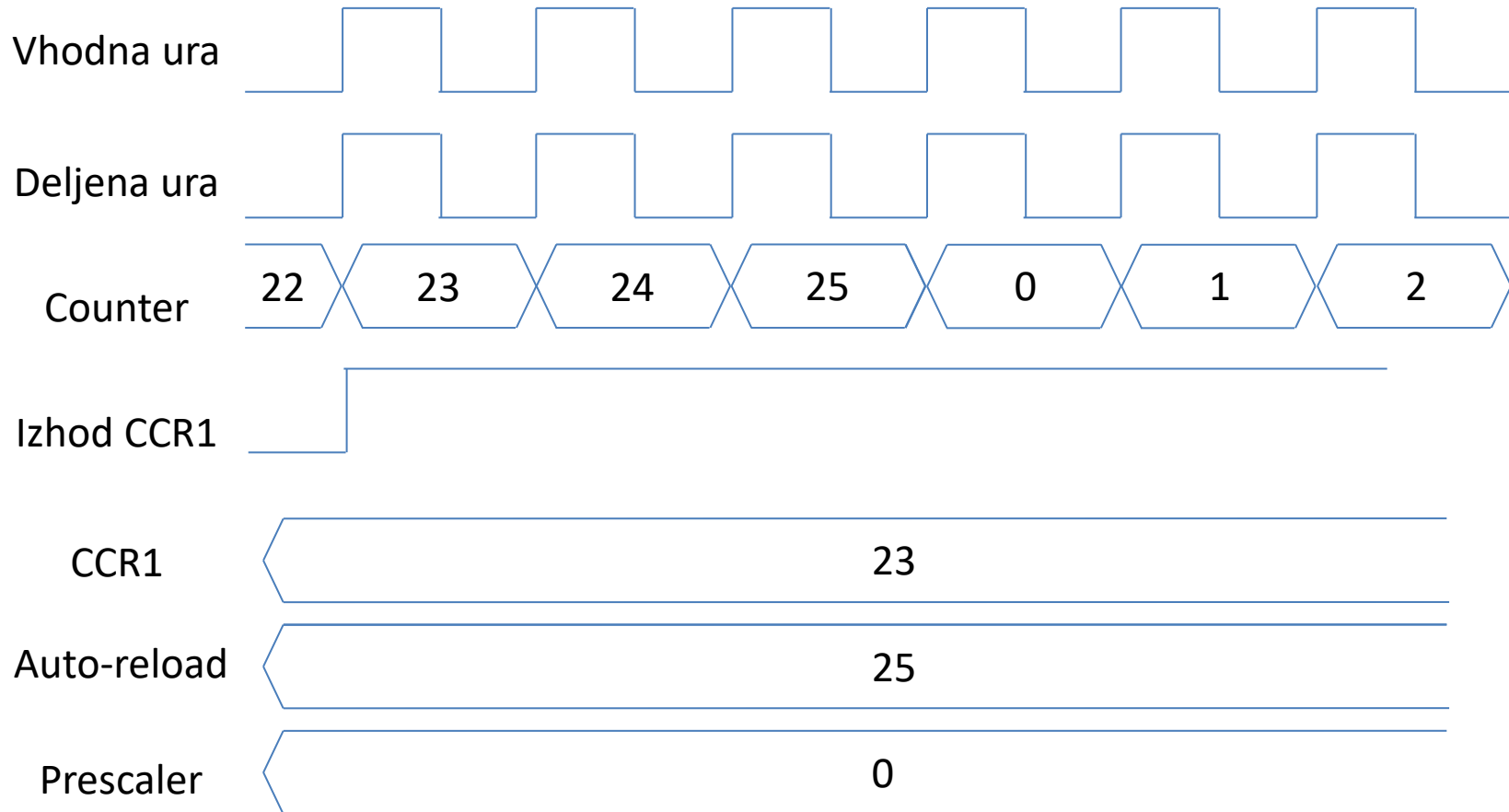
- Primer zastavice:

```
TIM_FLAG_UPDATE
```

Compare Channels

- Vsak časovnik ima 4 Compare kanale
 - vsakemu Compare kanalu nastavimo vrednost Compare registra (CCR1,CCR2,CCR3,CCR4)
 - ko **counter** doseže vrednost CCRx se kanal "odzove"
 - vsak kanal ima določen tudi GPIO pin
- Odziv je lahko
 - sprememba stanja na GPIO pinu
 - prekinitev
 - proženje akcij (npr. proženje ADC konverzije)
 - postavljanje zastavice
 - ...

Output Compare - primer



Inicializacija časovnika (na primeru TIM4)

```
TIM_HandleTypeDef timer4;  
timer4.Instance = TIM4;  
timer4.Init.CounterMode = TIM_COUNTERMODE_UP;  
timer4.Init.Period = 1000 - 1;  
timer4.Init.Prescaler = 16000 - 1;  
HAL_TIM_OC_Init(&timer4);
```

Inicalizacija za Output Compare

```
TIM_OC_InitTypeDef OC_kanal;  
OC_kanal.OCMode      = TIM_OCMODE_TOGGLE;  
OC_kanal.OCPolarity  = TIM_OCPOLARITY_HIGH;  
OC_kanal.Pulse       = 500;  
HAL_TIM_OC_ConfigChannel(&timer4, &OC_kanal, TIM_CHANNEL_1);
```

Nastavitev GPIO Pinov

```
__HAL_RCC_GPIOD_CLK_ENABLE();  
GPIO_InitTypeDef init_structure;  
  
init_structure.Pin = GPIO_PIN_12;  
init_structure.Mode = GPIO_MODE_AF_PP;  
init_structure.Pull = GPIO_NOPULL;  
init_structure.Speed = GPIO_SPEED_FREQ_LOW;  
init_structure.Alternate = GPIO_AF2_TIM4;  
  
HAL_GPIO_Init(GPIOD, &init_structure);
```

Ostale pomembnejše funkcije

- Vklon časovnika

```
HAL_TIM_OC_Start(&timer, TIM_CHANNEL_1);
```

- Branje stanja zastavic

```
__HAL_TIM_GET_FLAG(&timer, zastavica)
```

- Brisanje zastavice

```
__HAL_TIM_CLEAR_FLAG(&timer, zastavica);
```

- Možne zastavice:

TIM_FLAG_UPDATE -> counter je bil resetiran (dosežen ARR)

TIM_FLAG_CC1 -> counter je dosegel vrednost CC reg. kanala 1

TIM_FLAG_CC2 -> counter je dosegel vrednost CC reg. kanala 2

TIM_FLAG_CC3 -> counter je dosegel vrednost CC reg. kanala 3

TIM_FLAG_CC4 -> counter je dosegel vrednost CC reg. kanala 4

... (glej knjižnico časovnika)

Ostale pomembnejše funkcije

- Nastavi vrednost Counter registra

```
__HAL_TIM_SET_COUNTER(&timer, 1000);
```

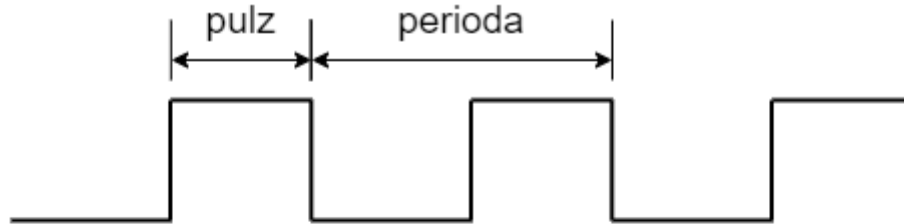
- Ponastavi vrednost ARR registra

```
__HAL_TIM_SET_AUTORELOAD(&timer, 1000);
```

- Nastavi vrednost CC registra kanalov

```
__HAL_TIM_SET_COMPARE(&timer, TIM_CHANNEL_1, 1000);
```


PWM



$$\text{duty cycle} = \frac{\text{pulz}}{\text{perioda}}$$

$$\text{perioda} = \frac{1}{f}$$

Inicalizacija za PWM

```
TIM_HandleTypeDef timer4;
```

```
timer4.Instance = TIM4;
```

```
timer4.Init.CounterMode = TIM_COUNTERMODE_UP;
```

```
timer4.Init.Period = 100;
```

```
timer4.Init.Prescaler = 16000-1;
```

```
HAL_TIM_PWM_Init(&timer4);
```

```
TIM_OC_InitTypeDef OC_kanal;
```

```
OC_kanal.OCMode = TIM_OCMODE_PWM1;
```

```
OC_kanal.OCPolarity = TIM_OCPOLARITY_HIGH;
```

```
OC_kanal.Pulse = 50;
```

```
HAL_TIM_PWM_ConfigChannel(&timer4, &OC_kanal, TIM_CHANNEL_1);
```

Inicalizacija za PWM

```
TIM_OC_InitTypeDef OC_kanal;  
OC_kanal.OCMode      = TIM_OCMODE_PWM1;  
OC_kanal.OCPolarity  = TIM_OCPOLARITY_HIGH;  
OC_kanal.Pulse       = 500;  
HAL_TIM_PWM_ConfigChannel(&timer4, &OC_kanal, TIM_CHANNEL_1);
```