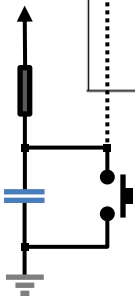
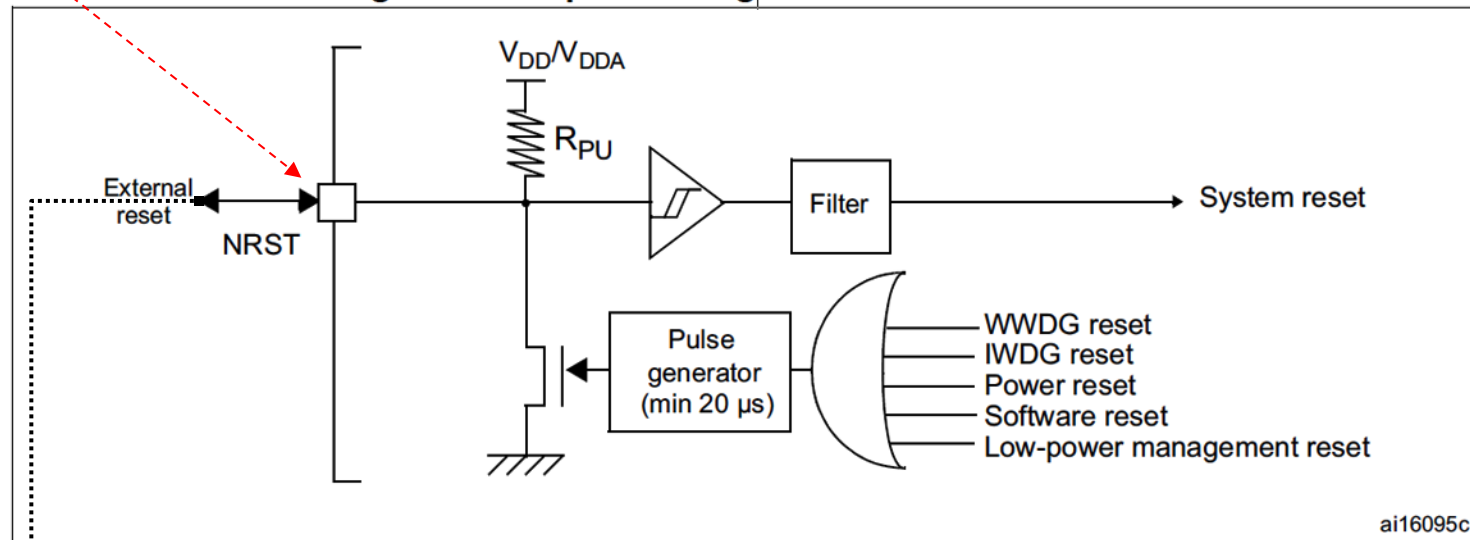


- Núcleo de PROCESSAMENTO do ARM (hardware)
 - Básico de programação (Interrupções, Pilha, Funções)
 - Conjunto de Instruções (ISA = Instruction Set Architecture)
 - Programação ASSEMBLY para ARM Cortex
-
- RESET
 - Watchdog timers
 - Temporizadores (timers de propósito geral e específicos)
 - PWM (saída analógica ainda que usando níveis lógicos digitais)
 - GPIO (General Purpose Input / Output)

- μProcessadores e SoC geralmente podem ser resetados de várias formas. No ARM:

**Pino no chip
NRST
(externo)**

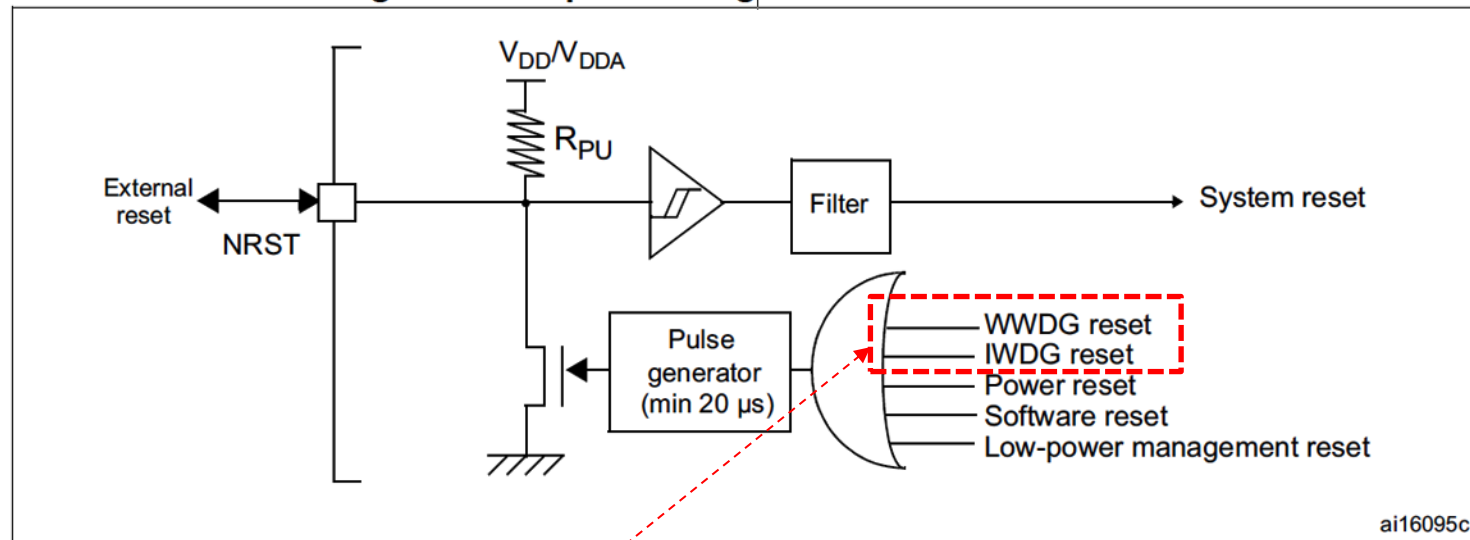
Figure 7. Simplified diagram of the reset circuit



Circuito na placa...

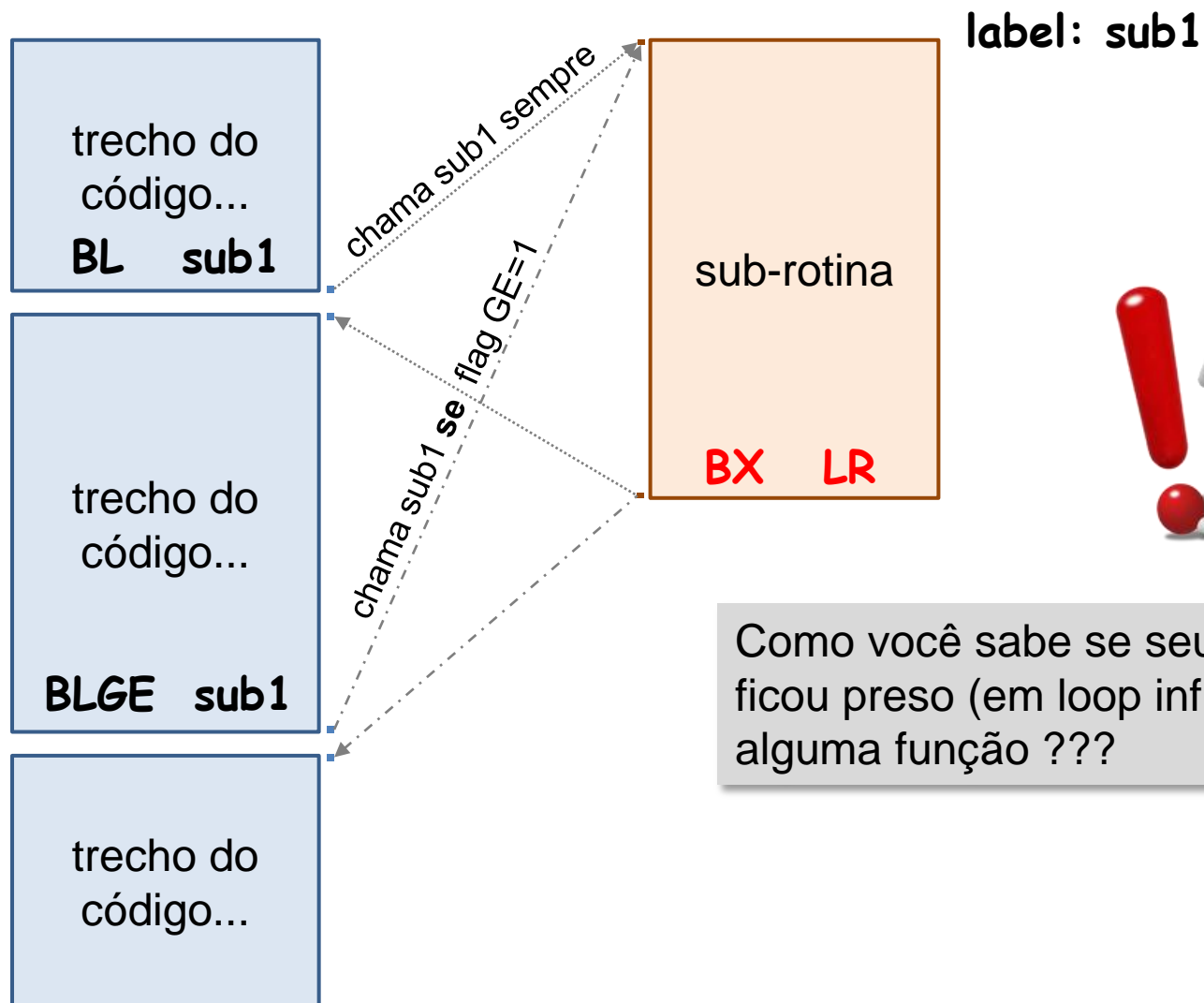
- μ Processadores e SoC geralmente podem ser resetados de várias formas. No ARM:

Figure 7. Simplified diagram of the reset circuit



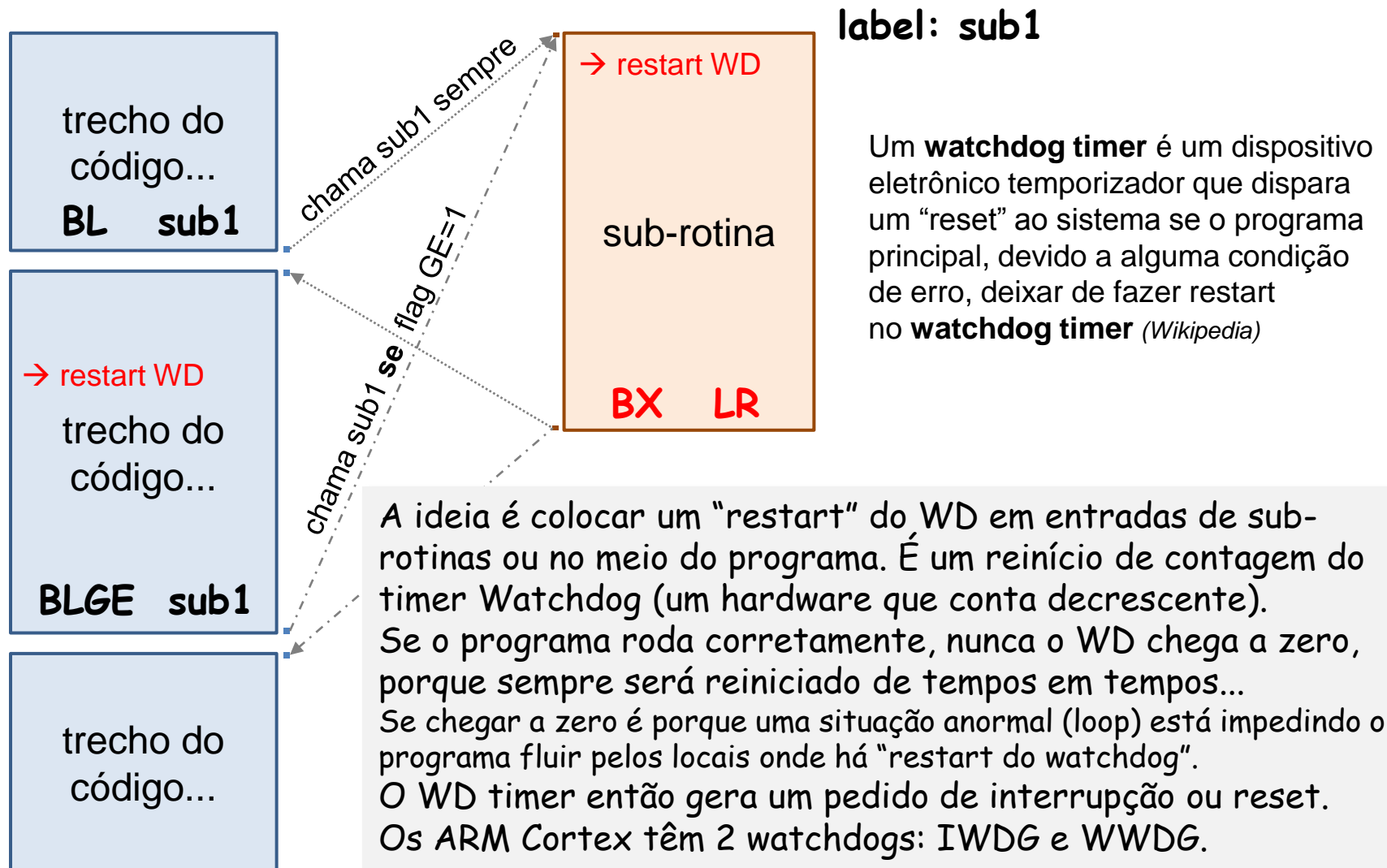
Watchdog timers!

- Recordando o conceito de sub-rotina (ou chamada de função)



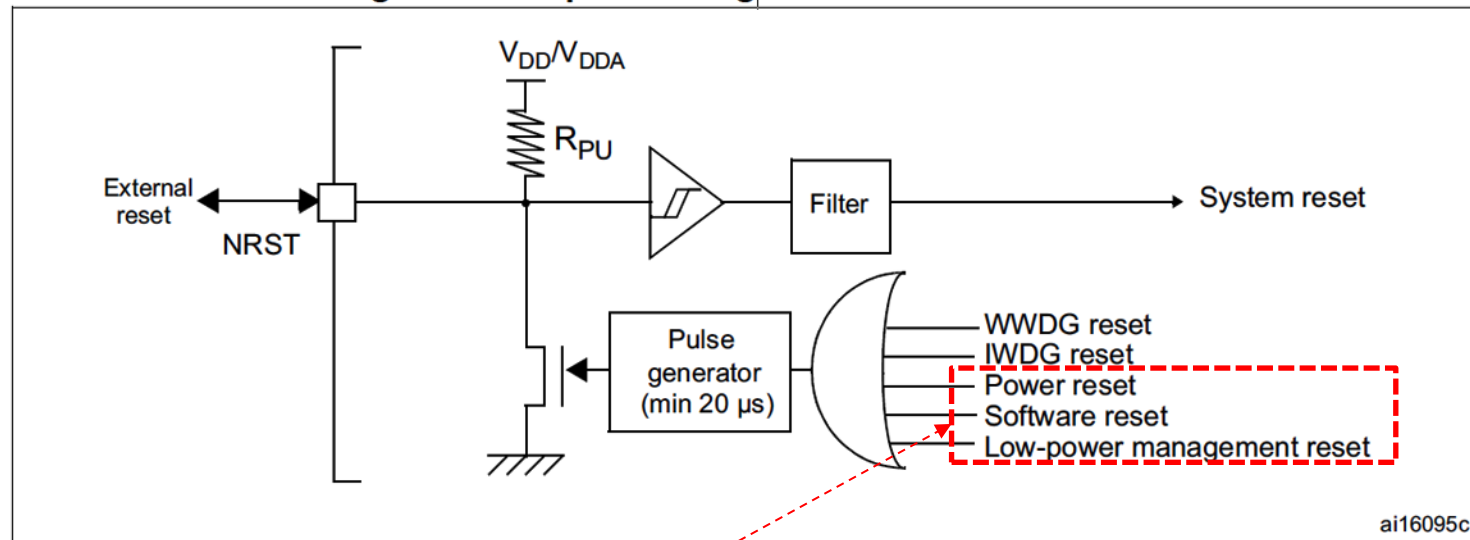
Como você sabe se seu software ficou preso (em loop infinito) em alguma função ???

- Recordando o conceito de sub-rotina (ou chamada de função)



- μ Processadores e SoC geralmente podem ser resetados de várias formas. No ARM:

Figure 7. Simplified diagram of the reset circuit



Outras formas de reset !

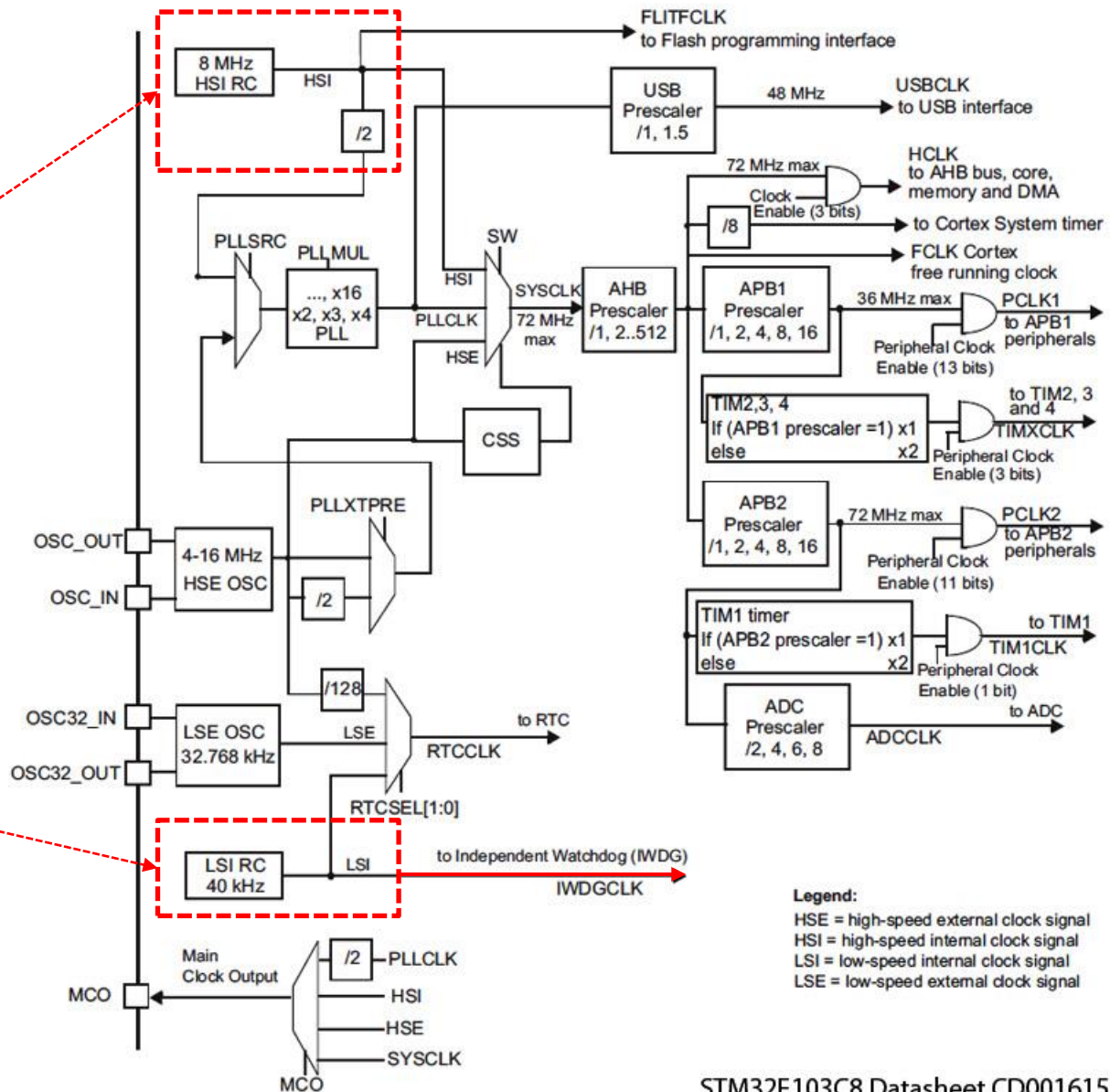
➤ µP e SoC podem ser programados para operar em frequências desejadas...

HSI RC (8MHz)
High-Speed Internal clk

ARM Cortex-M3 têm 2 geradores de clk internos

LSI RC (40KHz)
Low-Speed Internal clk

Figure 2. Clock tree



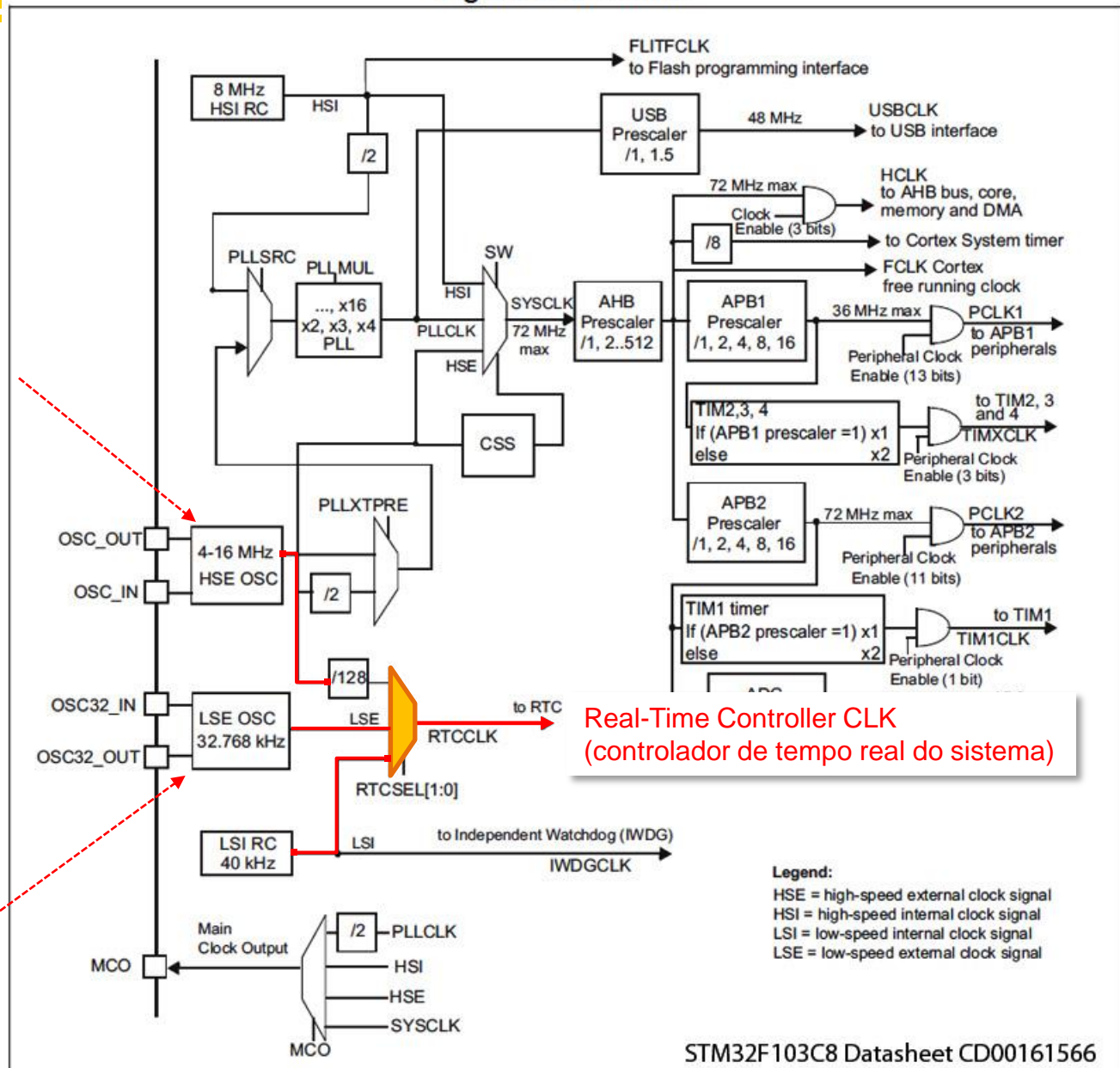
❖ Sistemas de CLOCK dos processadores ARM

➤ µP e SoC podem ser programados para operar em frequências desejadas...

oscilador externo
(cristal 4...16 MHz)
High-Speed External clk

oscilador externo
(cristal 32.768 KHz)
Low-Speed External clk

Figure 2. Clock tree



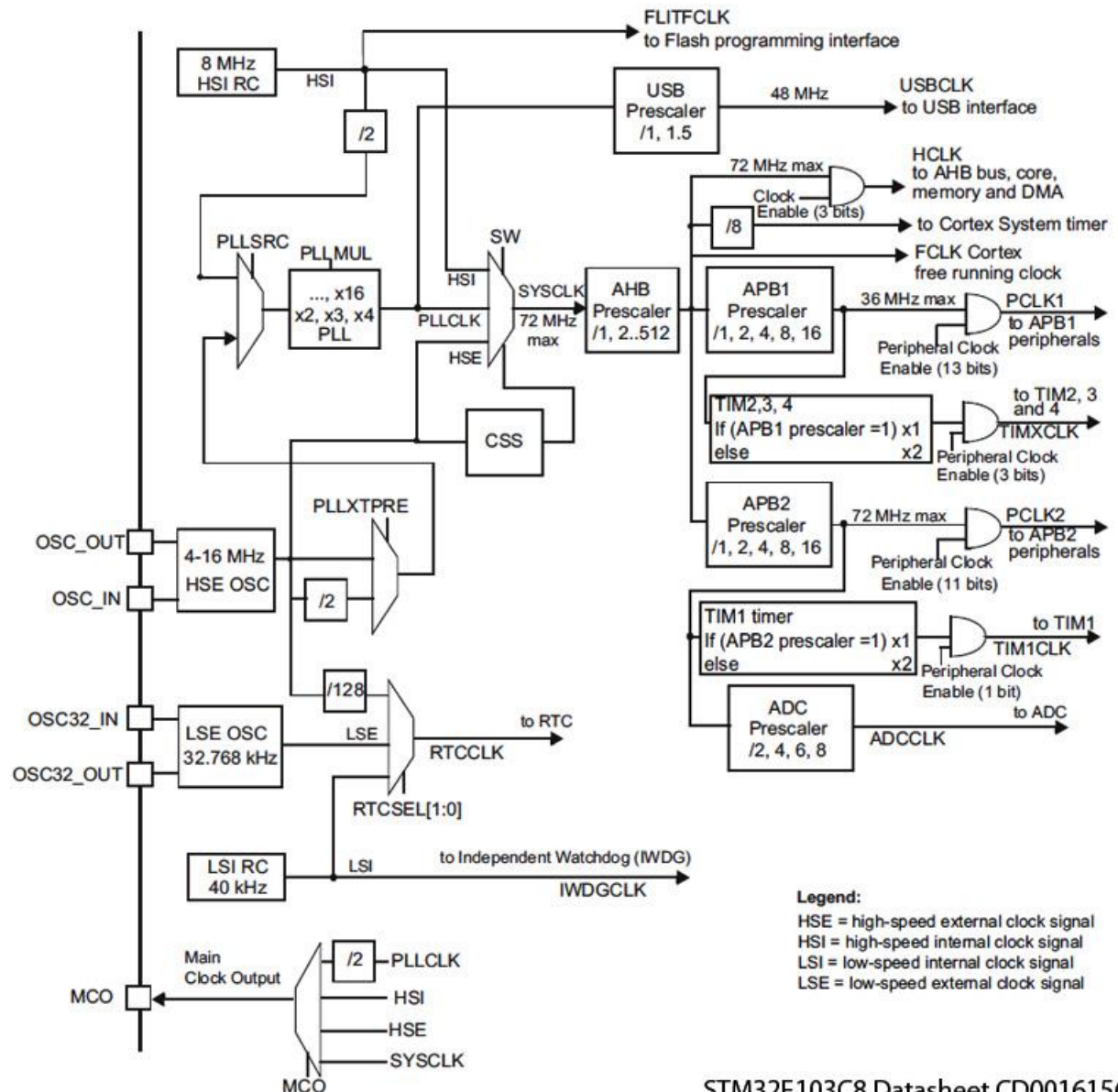
➤ µP e SoC podem ser programados para operar em frequências desejadas...

Multiplicador de frequência e PLL

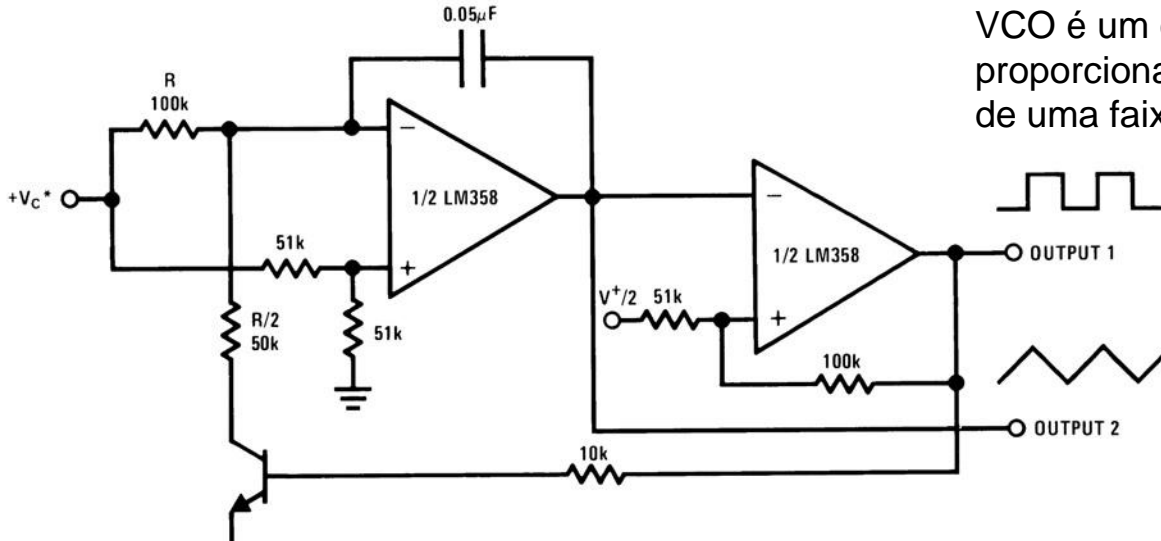


*Multiplicador?
Pepe... o que?*

Figure 2. Clock tree



Voltage Controlled Oscillator (VCO)

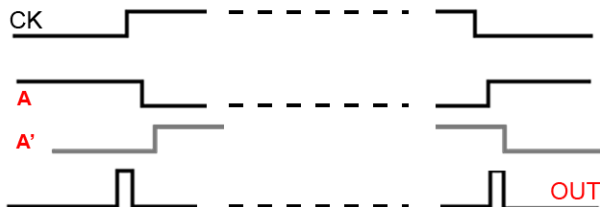
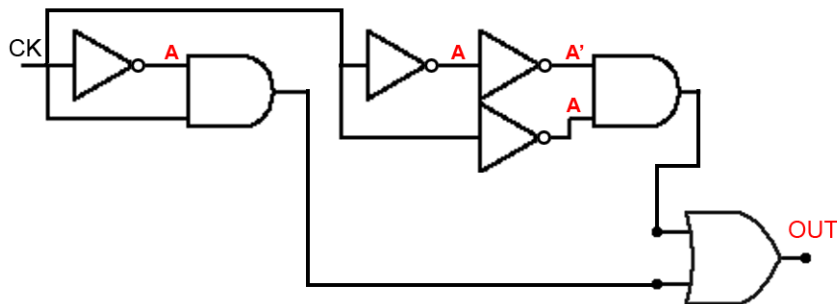


Conceito 1: VCO (*Voltage-Controlled Oscillator*)

VCO é um circuito que gera uma frequência de saída proporcional à tensão de controle de entrada, dentro de uma faixa pré-determinada:

Por exemplo:

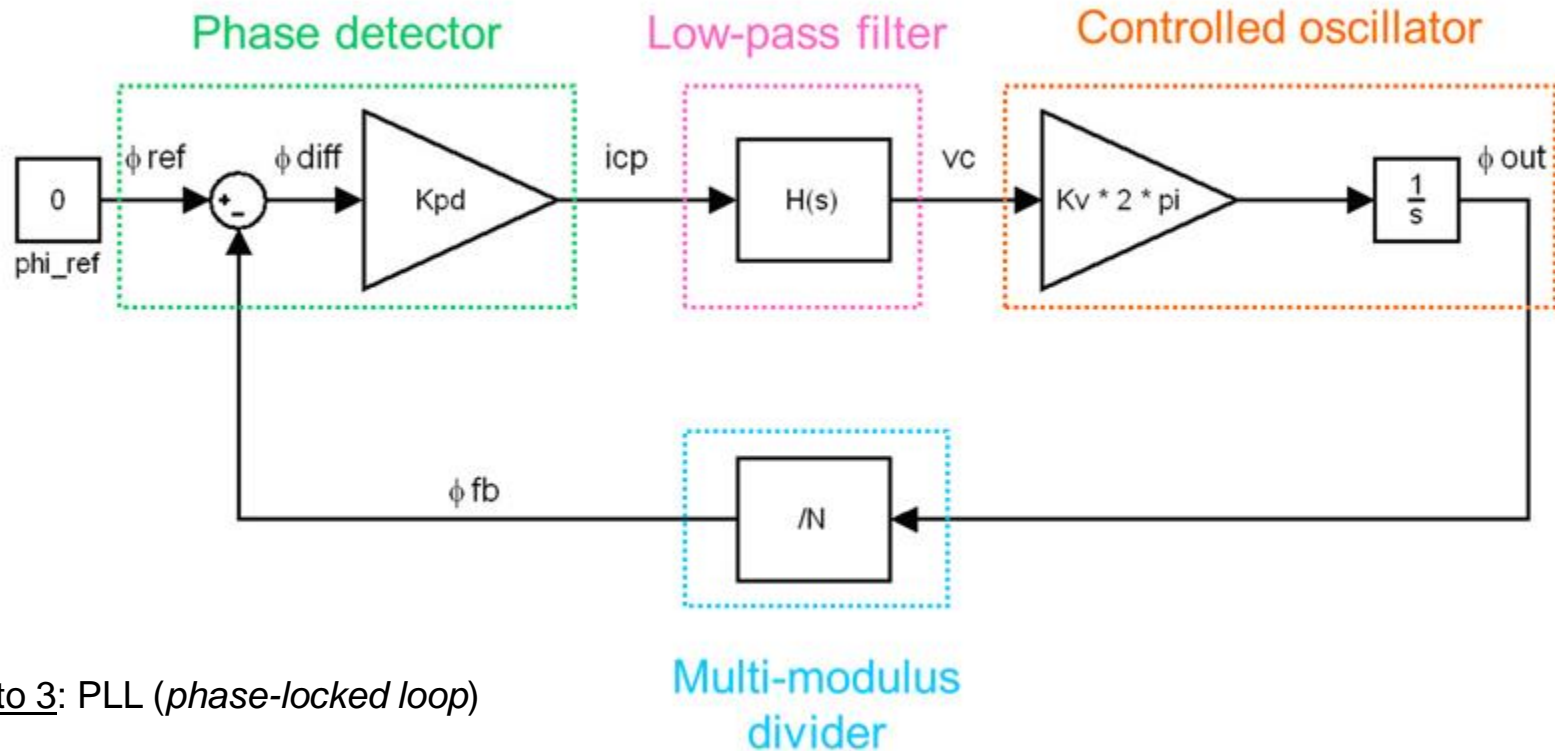
Vin	Freq
1 V	20 MHz
...	...
4 V	80 MHz



Conceito 2: Multiplicador de frequência...

Circuito que multiplica (geralmente base 2) o clock de entrada gerando frequências mais altas.

Ex: Duplicador de frequência simples, gera um pulso de saída tanto na subida quanto na descida do CK.



Conceito 3: PLL (phase-locked loop)

Circuito que gera um sinal de saída cuja FASE está relacionada com a fase do sinal de entrada.

O circuito geralmente tem um VCO e um detector de fase que compara a saída do VCO com uma frequência de referência. Manter a fase de entrada e saída no “passo de bloqueio” implica manter as frequências de entrada e saída estáveis (iguais ou múltiplas uma da outra).

Exemplo: se (1) **phi_ref** = sinal de **clock** em 8MHz e (2) o VCO opera em 72MHz, então (3) o módulo divisor que gera ϕ_{fb} deve dividir a saída do VCO por “9”.

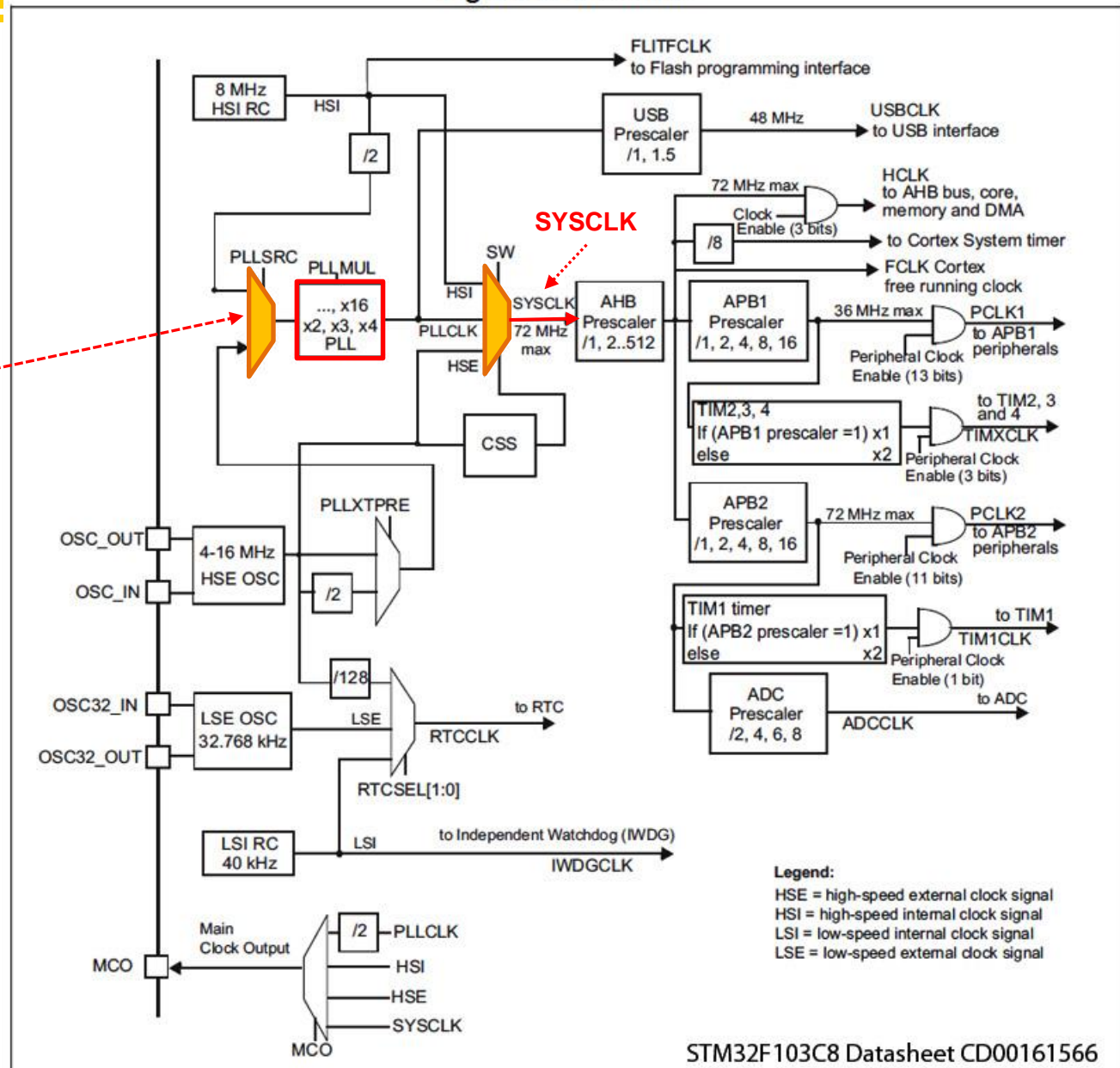
Ou seja, um sinal de 8MHz serve de referência para o VCO do circuito PLL gerar 72 MHz de forma estável.

❖ Sistemas de CLOCK dos processadores ARM

➤ µP e SoC podem ser programados para operar em frequências desejadas...

Multiplicador de frequência e PLL

Figure 2. Clock tree



- μ P e SoC podem ser programados para operar em frequências desejadas...

CLOCK GATES

(as que temos que programar para habilitar os clocks nos periféricos) !

Figure 2. Clock tree

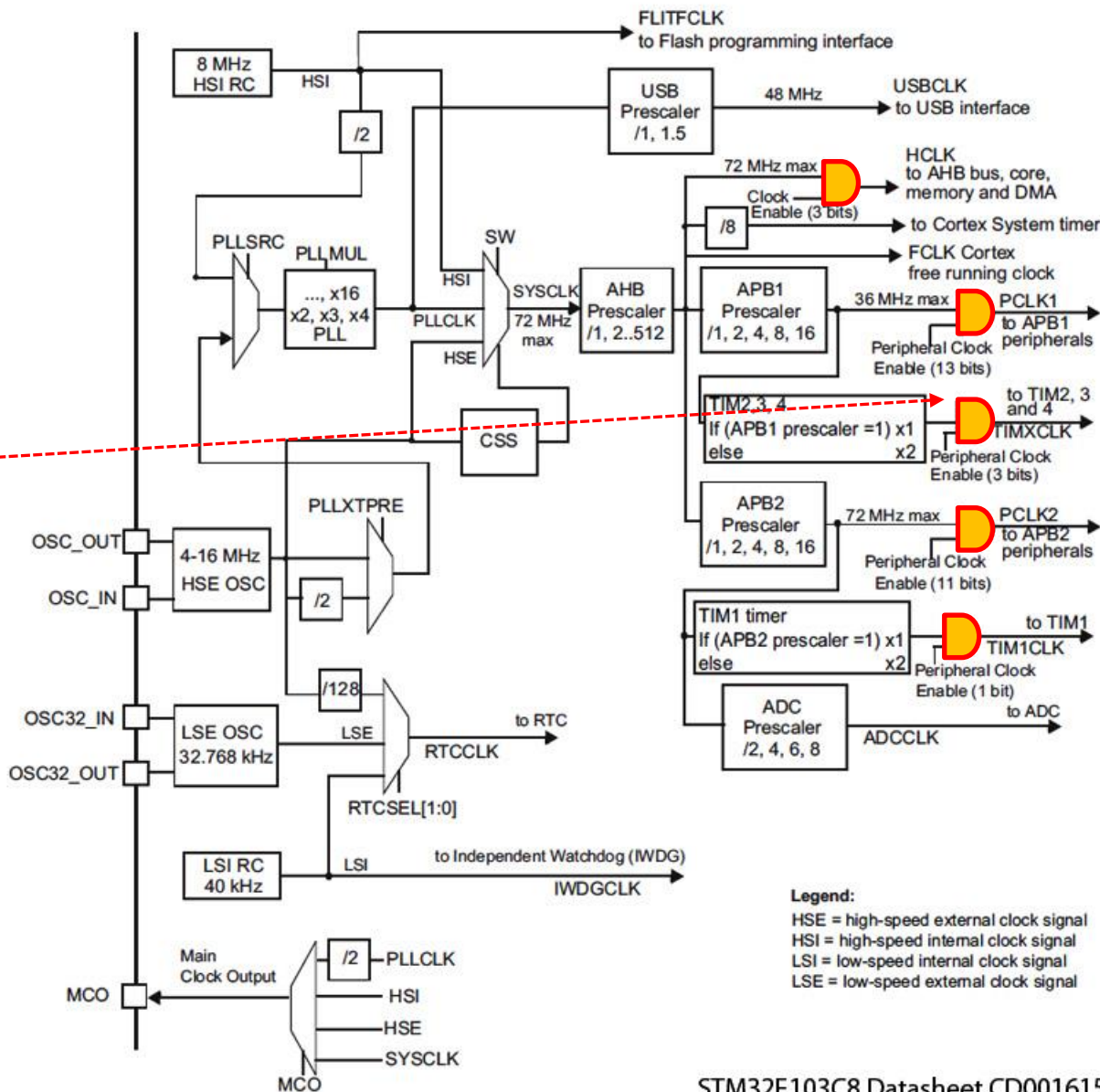
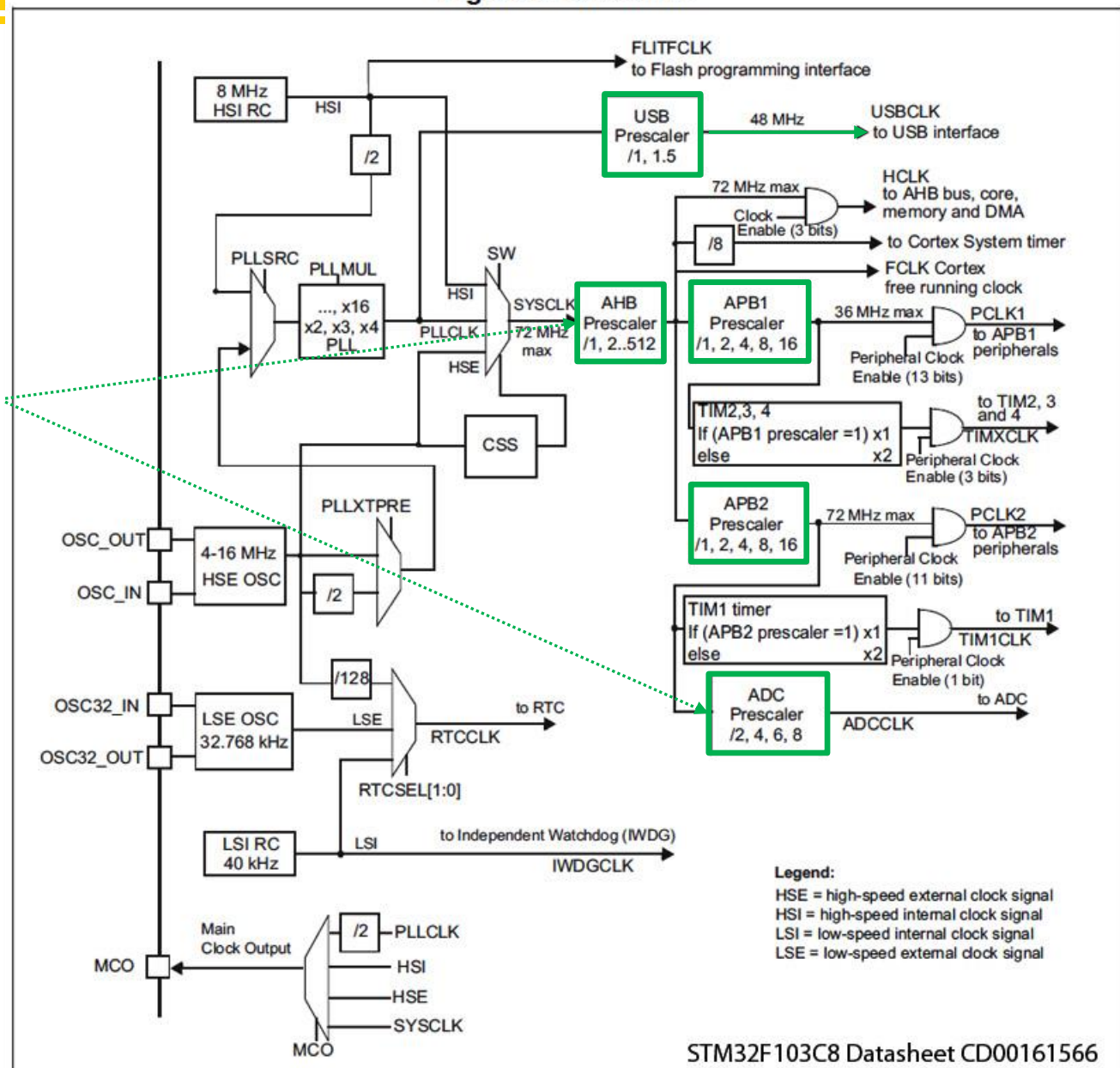


Figure 2. Clock tree

➤ μ P e SoC podem ser programados para operar em frequências desejadas...

PRESCALERS

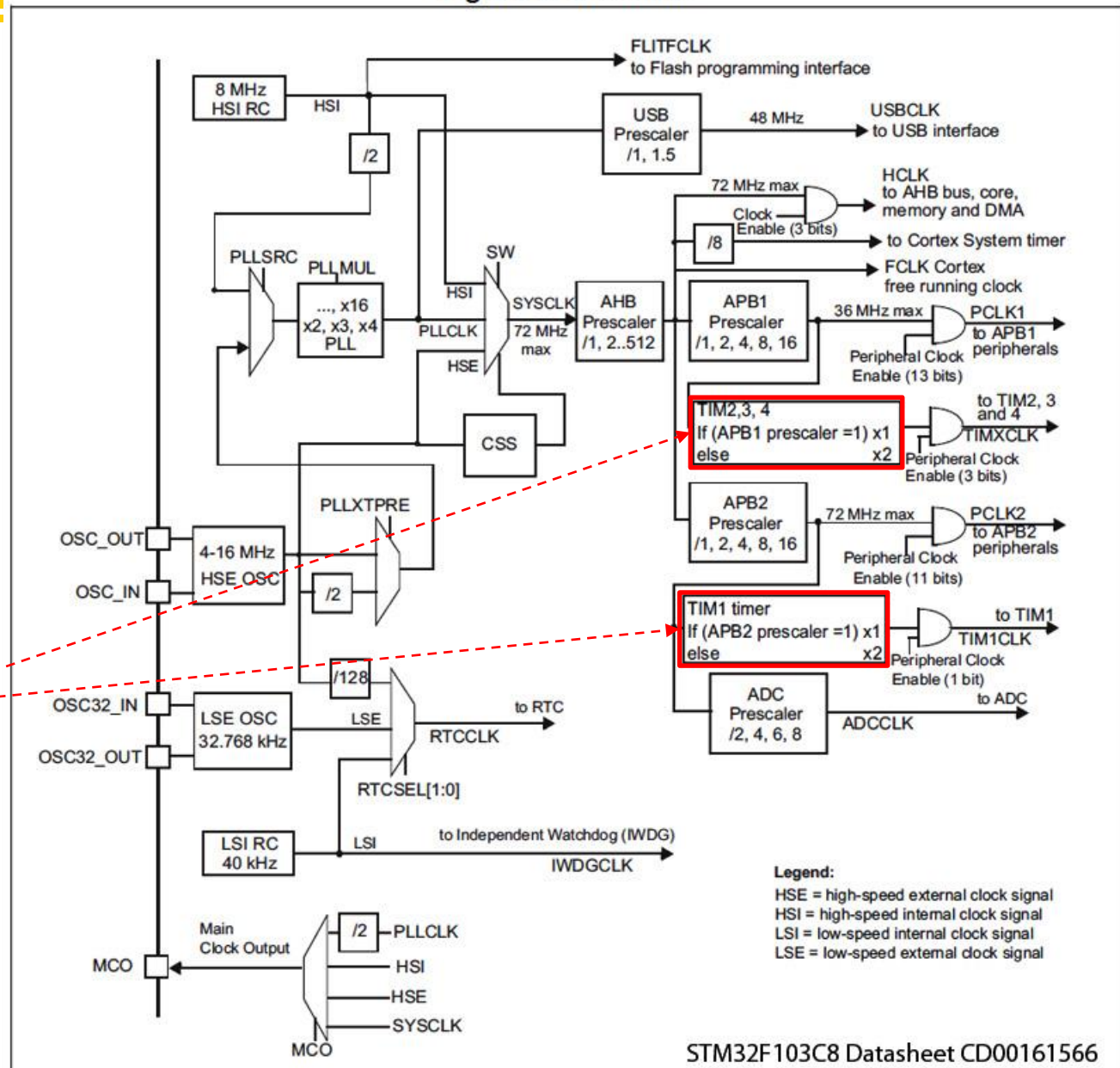
(divisores do clock *SYSCLK* para adequar a frequência para os timers, ADC, etc...)



➤ μ P e SoC podem ser programados para operar em frequências desejadas...

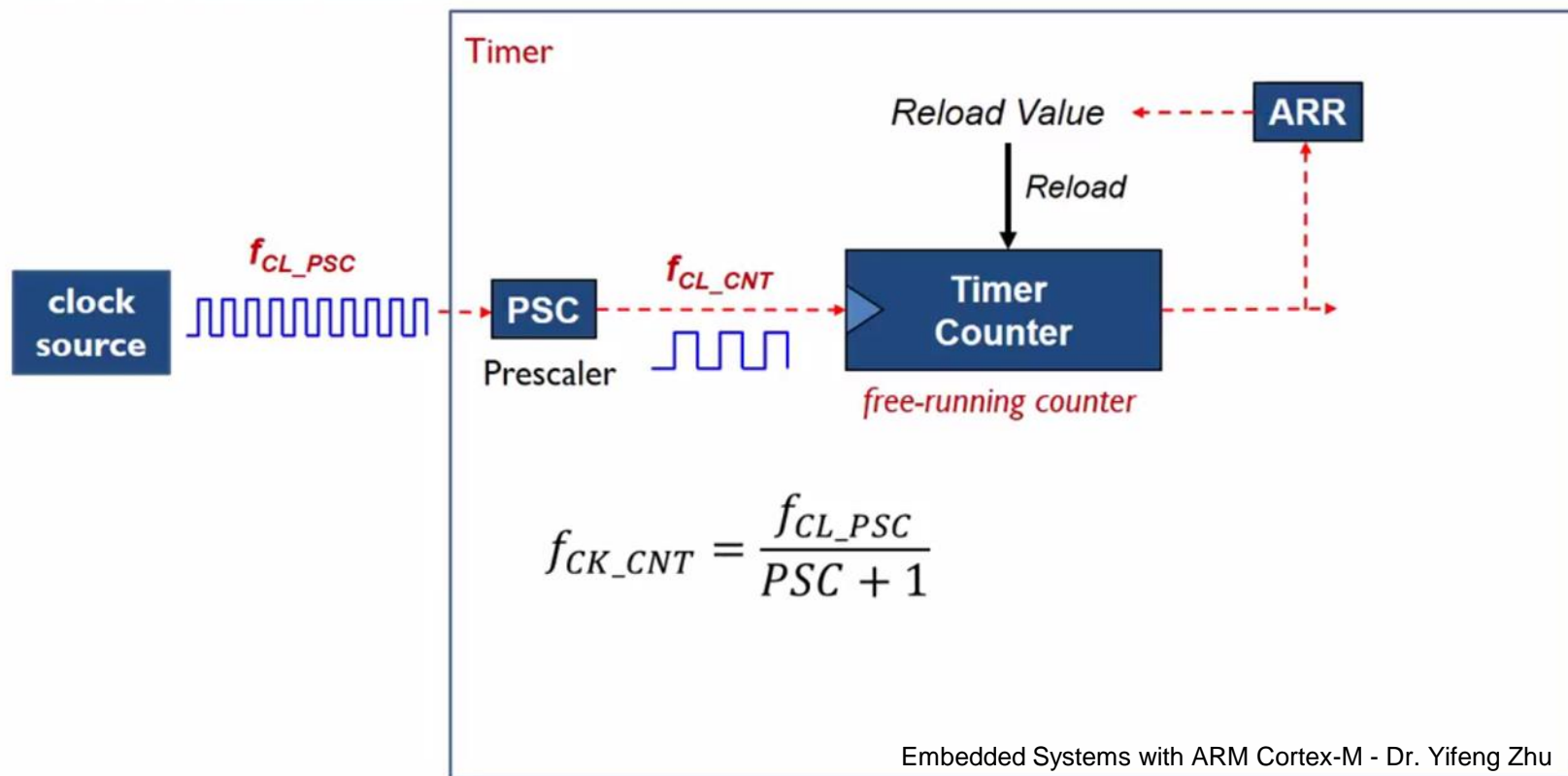
TIMERS
(nosso próximo assunto...)

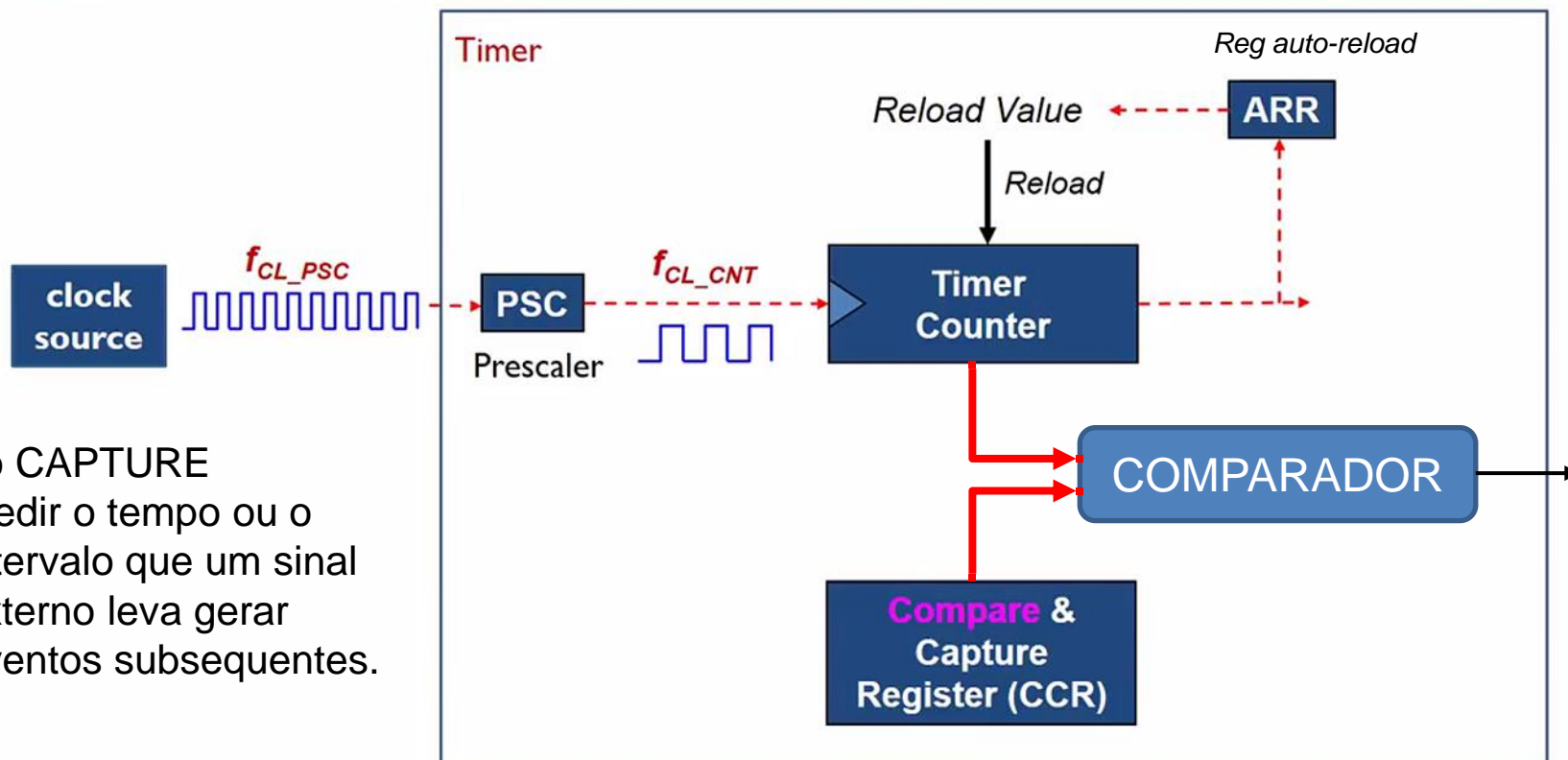
Figure 2. Clock tree



- timers são contadores que operam paralelo à CPU gerando bases temporais para o sistema.
- ARM tem até 16 timers de uso geral (STM32F103C8 tem 4 timers)
- são independentes e podem contar UP, DOWN ou UP-DOWN
- SysTick é um timer dedicado 24 bits (*já usamos para criar base de tempo 1 ms*)

Esquema básico de um timer no Cortex-M:



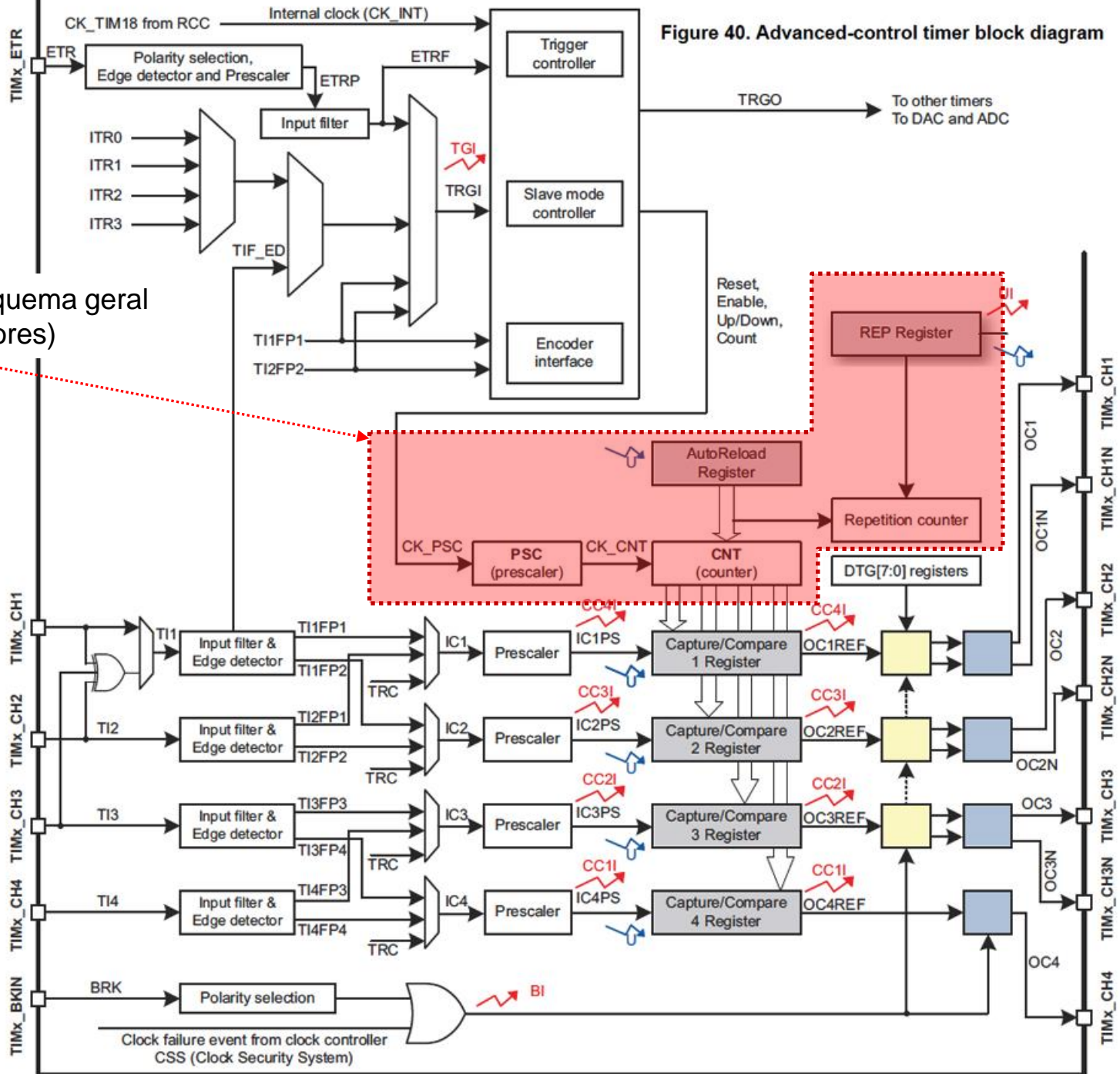
**Modo CAPTURE**

- Medir o tempo ou o intervalo que um sinal externo leva gerar eventos subsequentes.

Modo COMPARAÇÃO (compara o *timer counter* com o valor de um registrador CCR):

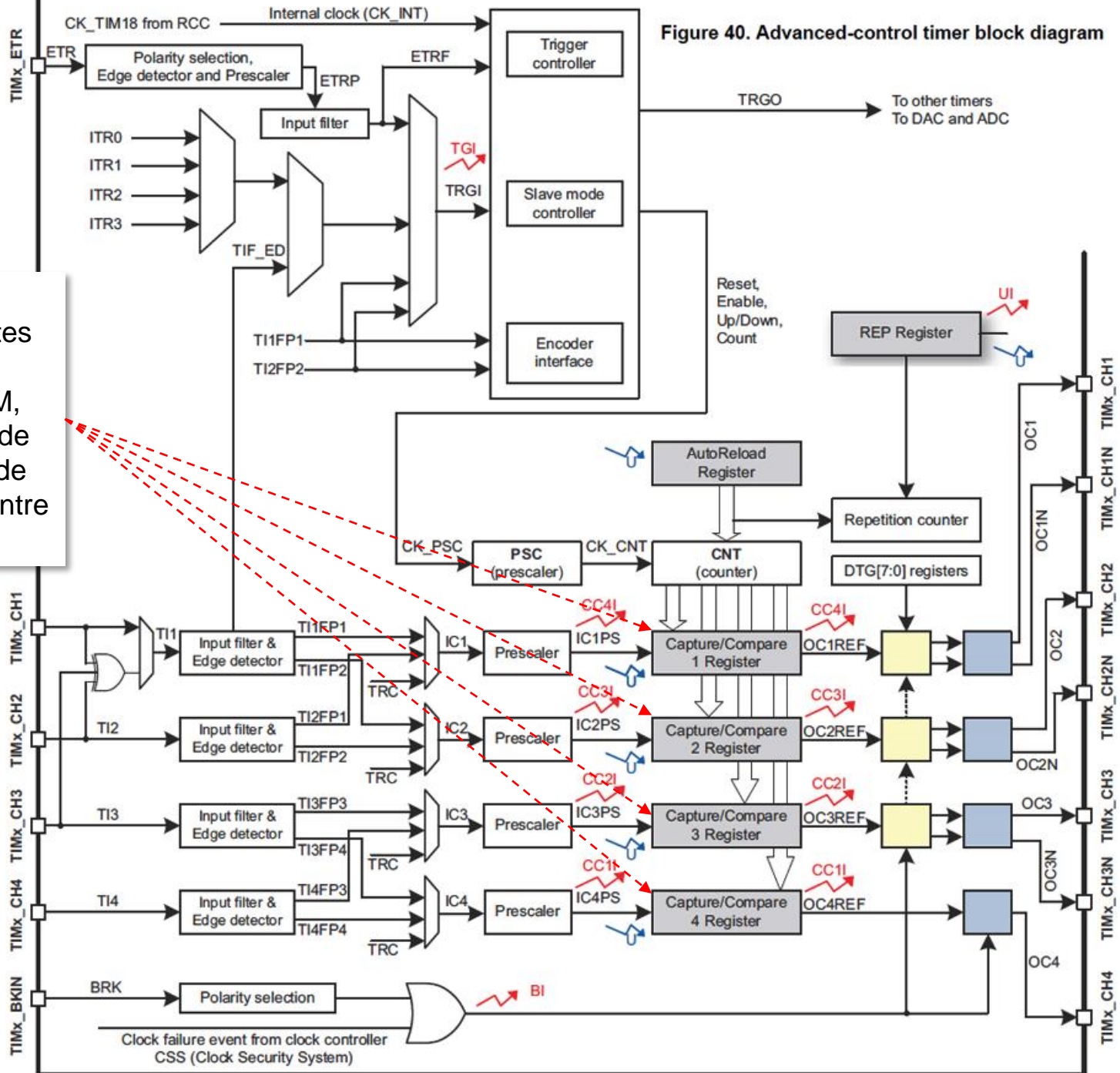
- usado para gerar interrupções
- usado para saída PWM (veremos PWM adiante...)

Figure 40. Advanced-control timer block diagram



O que vimos no esquema geral
(slide anteriores)

Cada timer tem 4 **canais** independentes que podem fazer 4 comparações (PWM, int, etc) ou captura de tempo de um sinal de entrada (intervalo entre eventos, etc)...



PWM tem sido usado como **saída analógica** em SoC e em μ Controladores.

PWM: Gera sinais com relação temporal entre níveis lógicos '1' e '0' controlados e bem definidos.

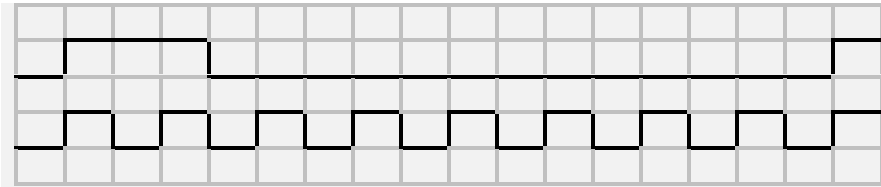
Ciclo de trabalho (**duty cycle**):
% tempo do sinal 'alto' (1) em
relação ao sinal 'baixo' (0)
Onda quadrada: ciclo de 50%

Usos comuns: controlar tensão
média para o dispositivo elétrico:

- Mais simples que conversor DC-DC ou conversor D-A;
- Velocidade de motor DC, dimmers

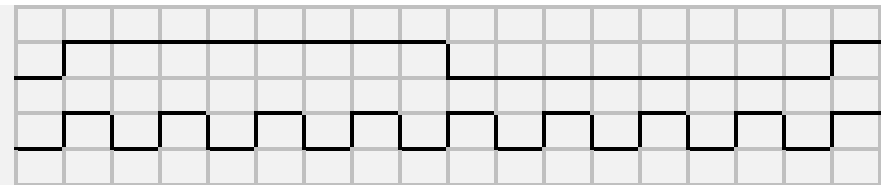
Outro uso: **comandos codificados temporalmente**, onde o receptor usa temporizador para decodificar sinal recebido.

pwm_out
clk



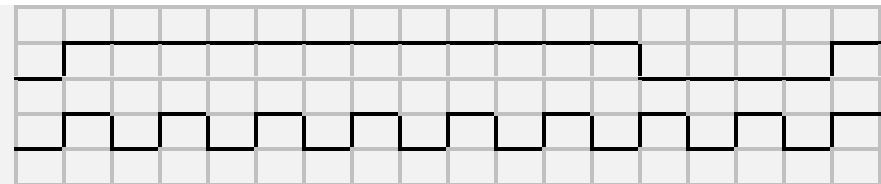
25% duty cycle – average pwm_o is 1.25V

pwm_out
clk



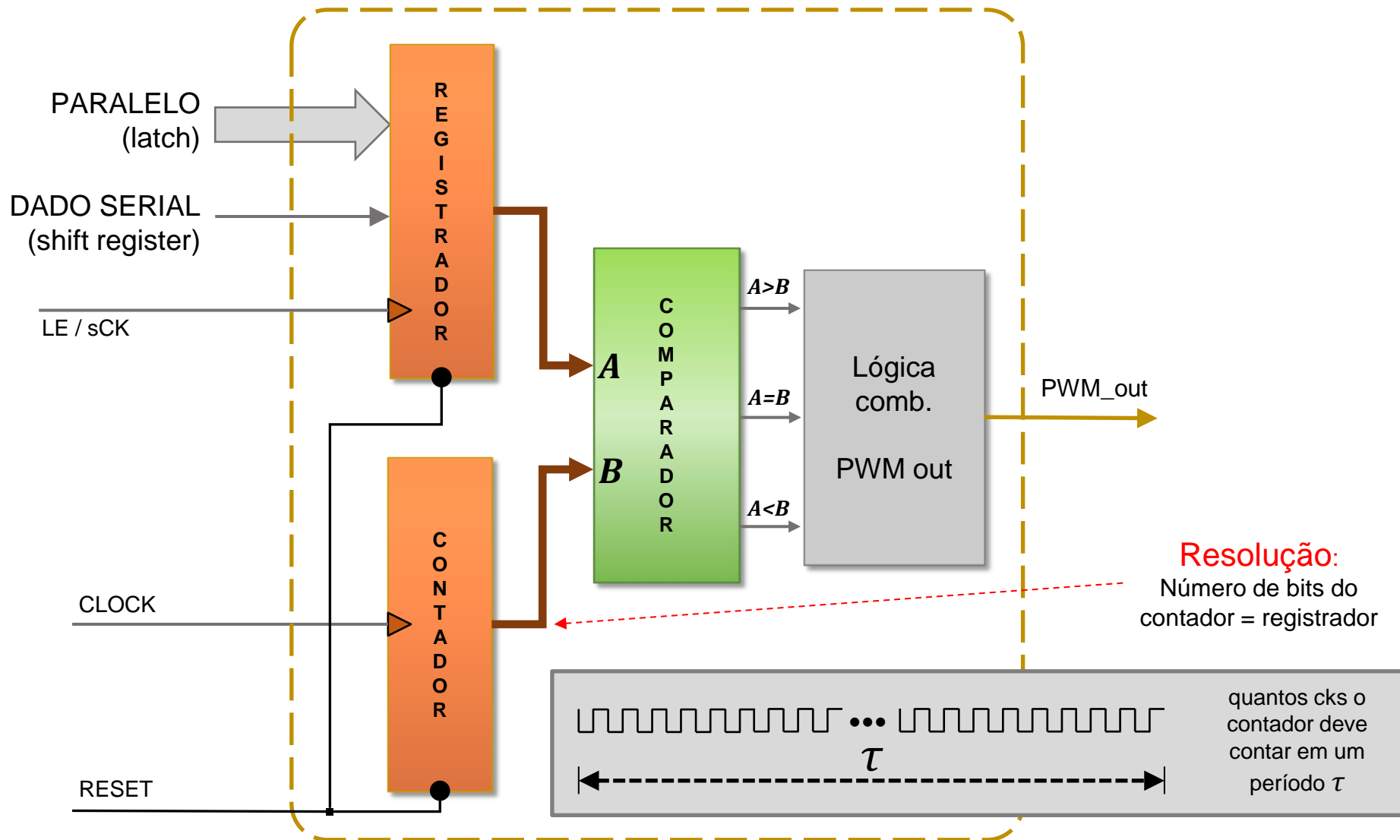
50% duty cycle – average pwm_o is 2.5V.

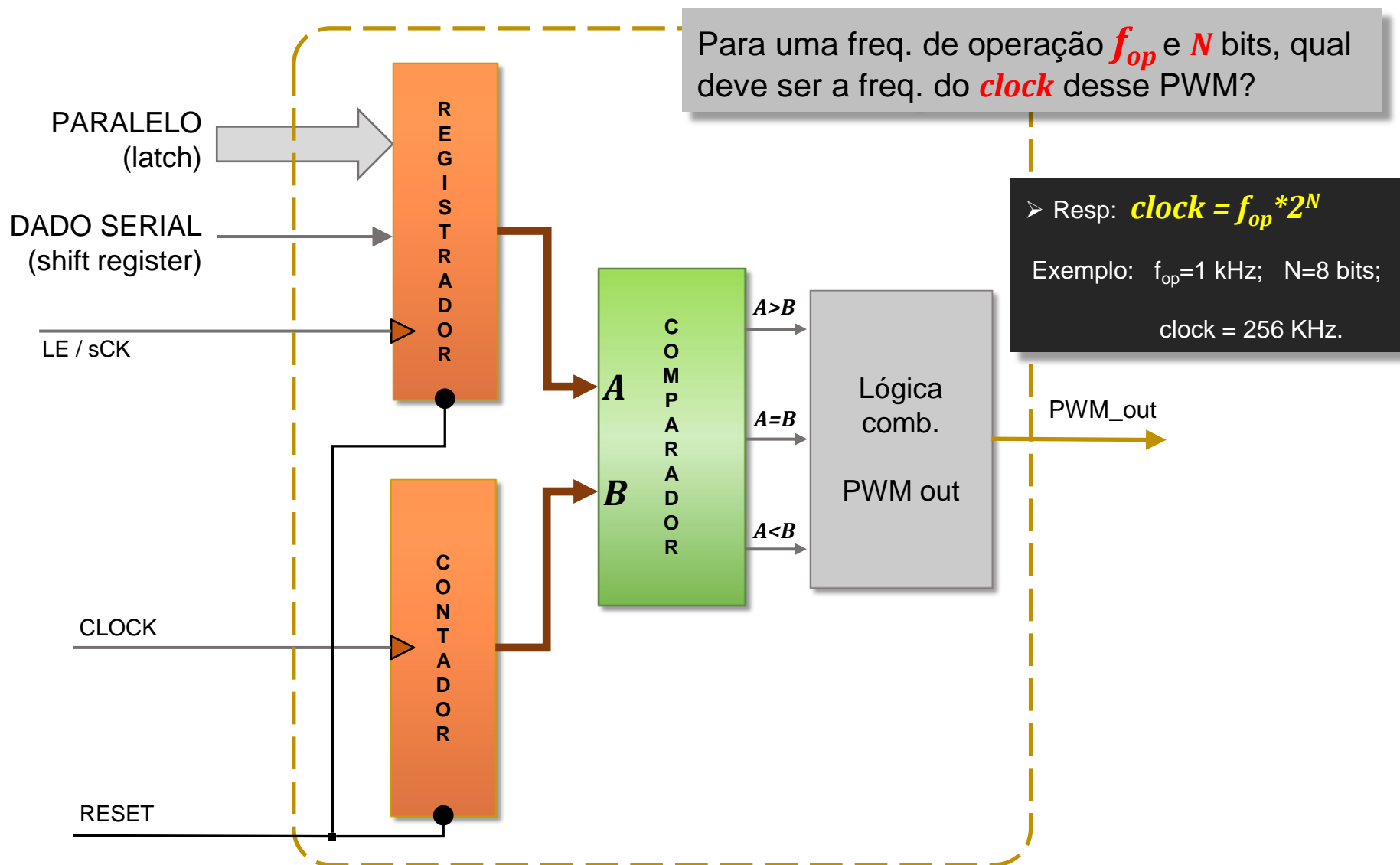
pwm_out
clk



75% duty cycle – average pwm_o is 3.75V.

Construindo um *PWM Digital* – (processador pwm)

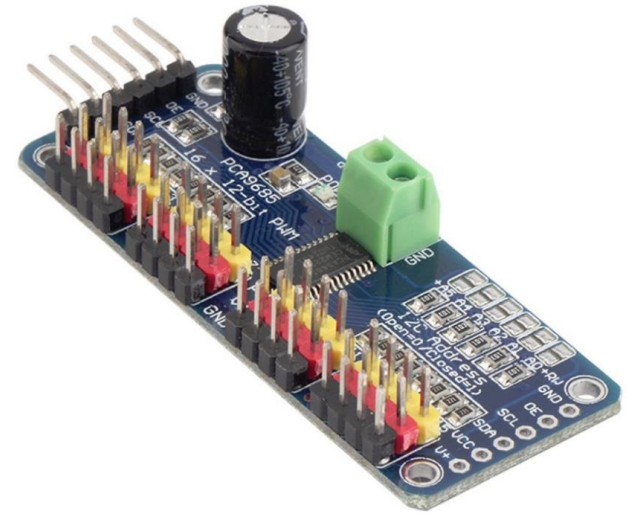
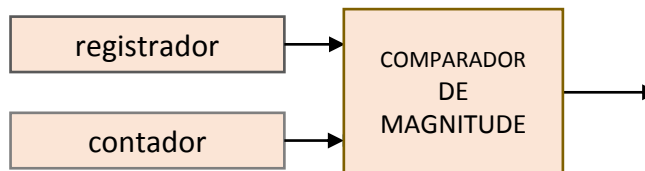
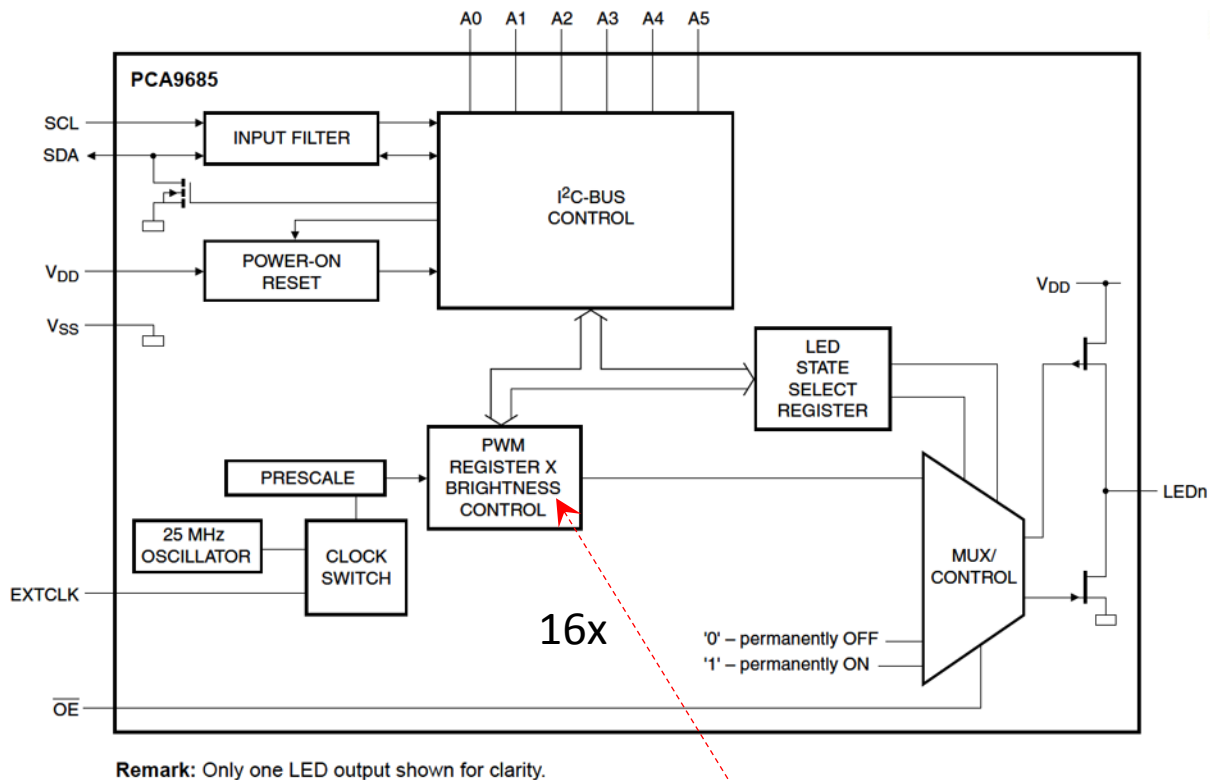


Construindo um *PWM Digital* – (processador pwm)

PCA-9685:

16-channel, 12-bit PWM Fm+ I²C-bus LED controller

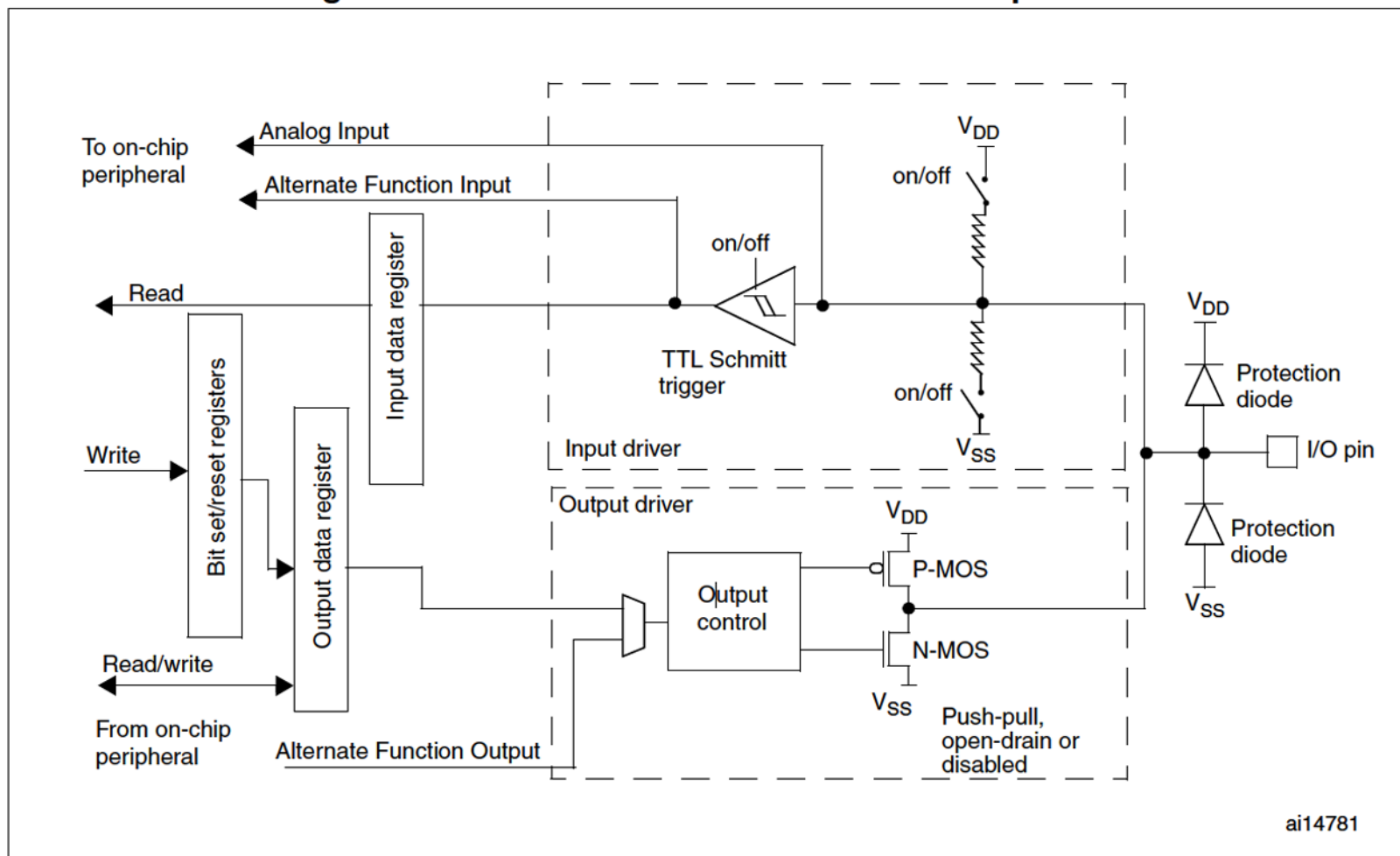
(custo Digikey: US\$ 1,06 p/ 2500 peças)



Placa Adafruit:
(custo: ~US\$ 15 por placa)
(genérica chinesa: US\$ 8)

- Como μControladores têm mais funções que pinos de I/O os fabricantes atribuem mais de uma função a cada pino. A funcionalidade do pino deve ser programada.

Figure 13. Basic structure of a standard I/O port bit



- Como μControladores têm mais funções que pinos de I/O os fabricantes atribuem mais de uma função a cada pino. A funcionalidade do pino deve ser programada.

Cada pino de GPIO no ARM Cortex pode ser configurado nos modos:

- Input floating
 - Input pull-up
 - Input-pull-down

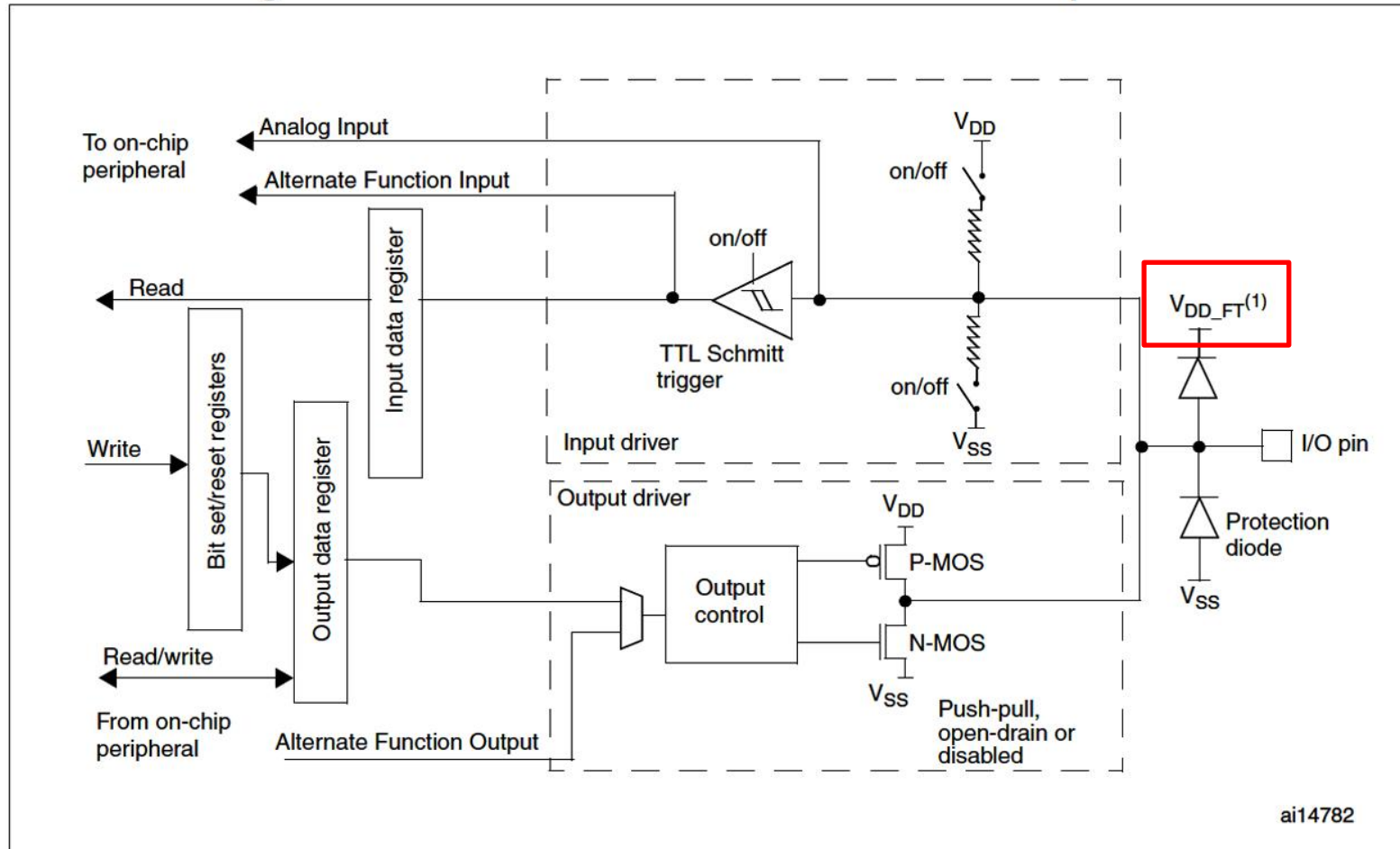
 - (Input) Analog

 - Output open-drain
 - Output push-pull

 - Alternate function push-pull
 - Alternate function open-drain
-
- ✓ Cada bit de porta de I/O é livremente programável, porém, os registros de porta de I/O devem ser acessados como palavras de 32 bits (*não são permitidos acessos de meia palavra ou de byte*).
 - ✓ Durante e logo após o reset as funções alternativas ficam INATIVAS e as portas de I/O são configuradas no modo de *entrada flutuante*.

- Alguns pinos do CI são tolerantes à excitação de 5V (lógica de 5V)...

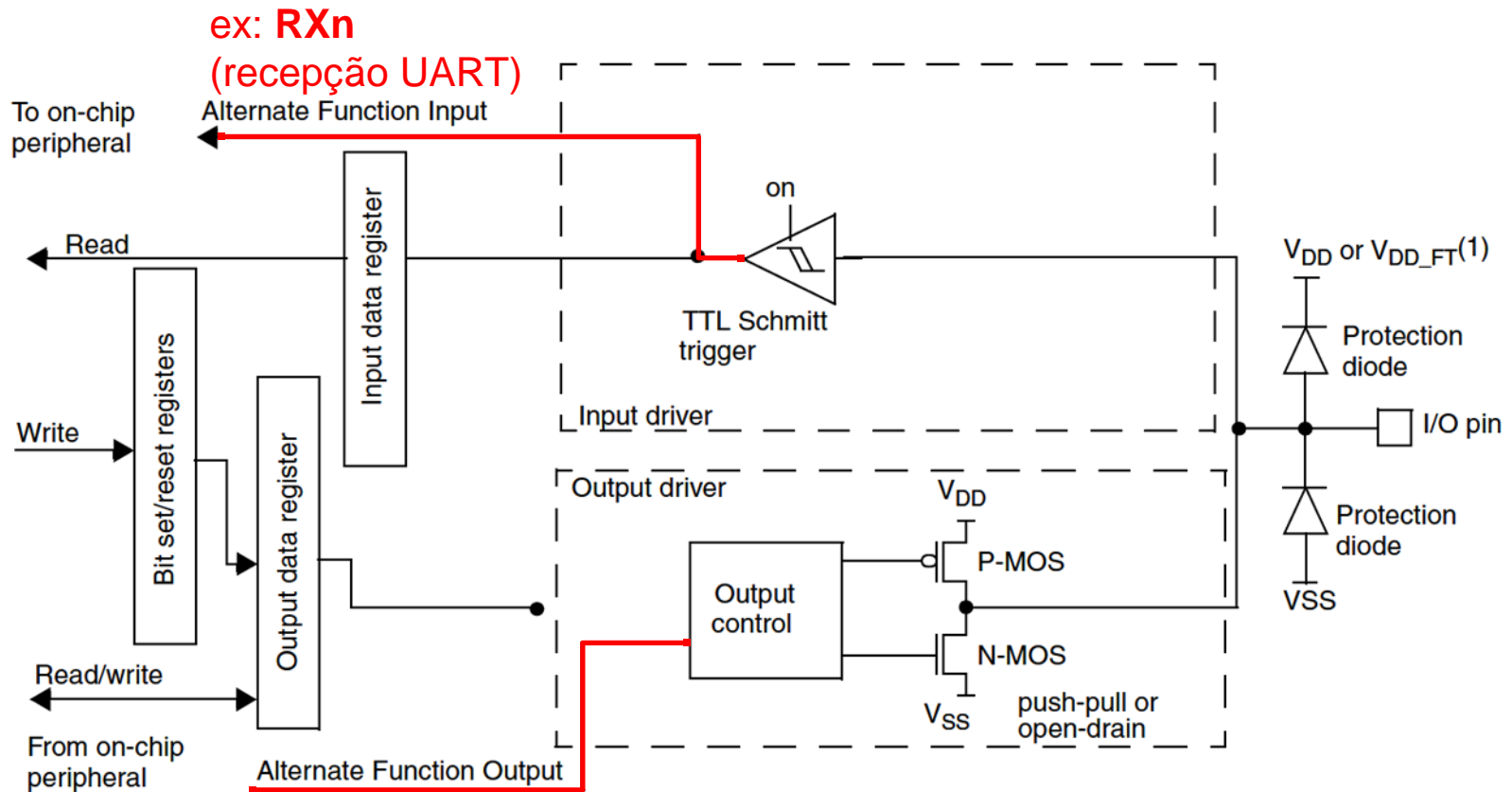
Figure 14. Basic structure of a five-volt tolerant I/O port bit



1. V_{DD_FT} is a potential specific to five-volt tolerant I/Os and different from V_{DD} .

- Quando os pinos não são usados como GPIO devem ser configurados p/ modo de funções alternativas (exemplo: saída PWM – referenciada como **AFIOs**)

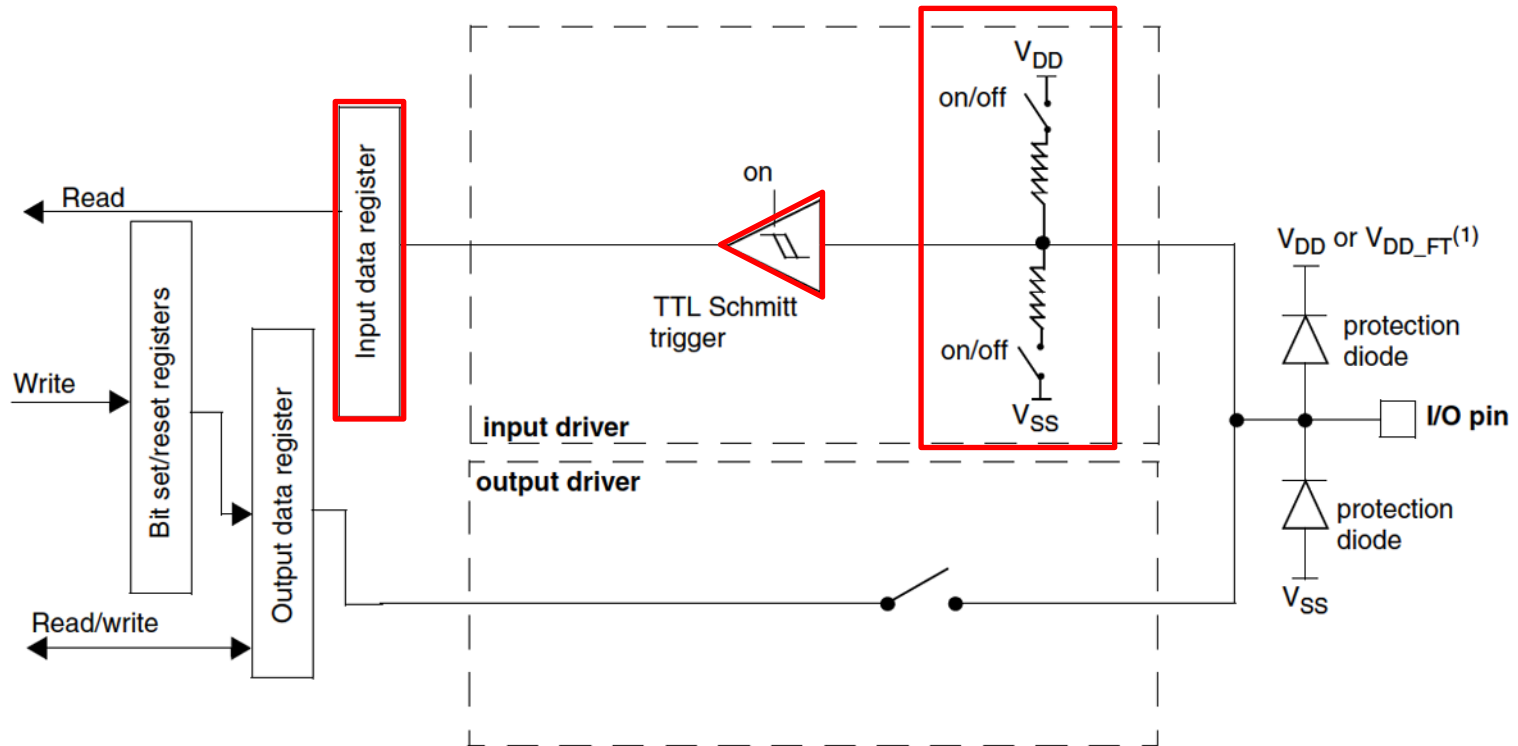
Figure 17. Alternate function configuration



ex: **PWM**
(um dos 4 canais de um TIMEx)

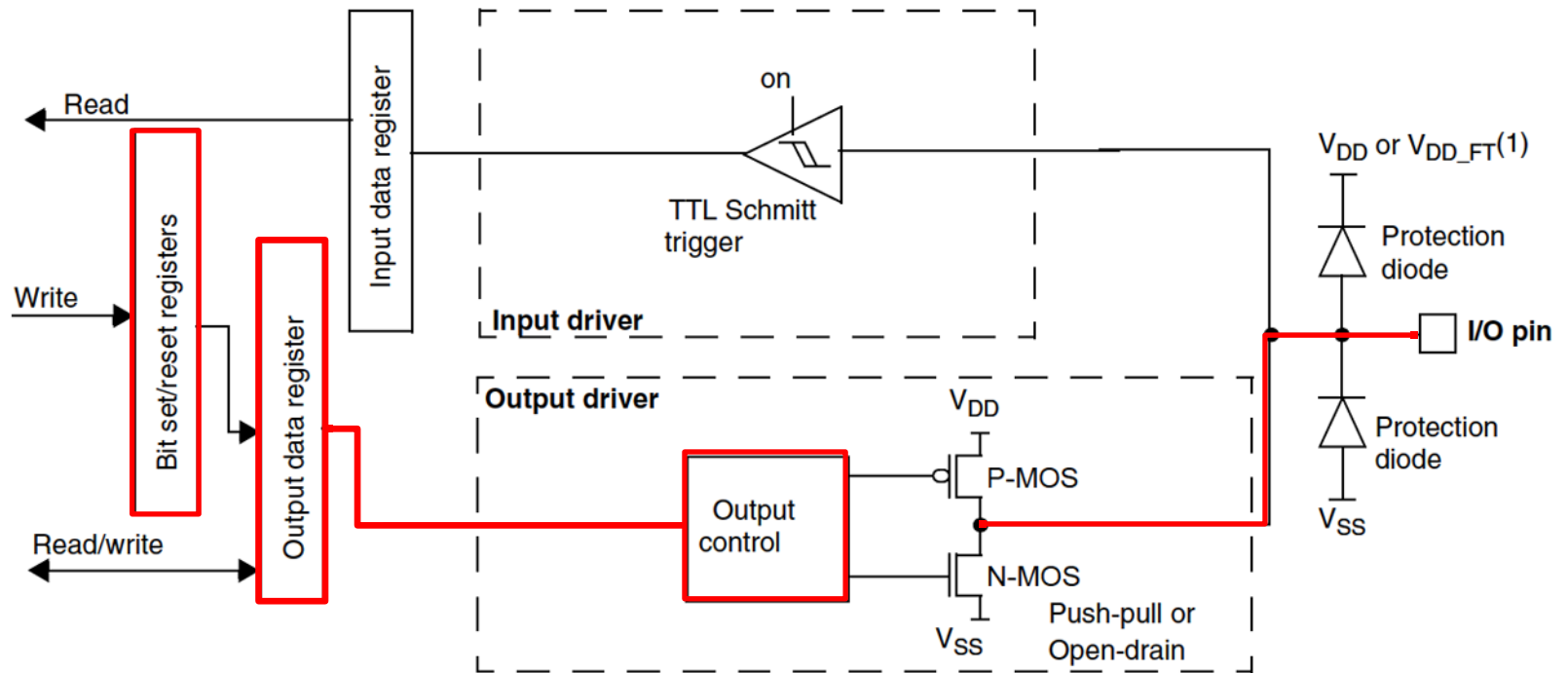
- Esta é uma estrutura básica para INPUT digital...

Figure 15. Input floating/pull up/pull down configurations



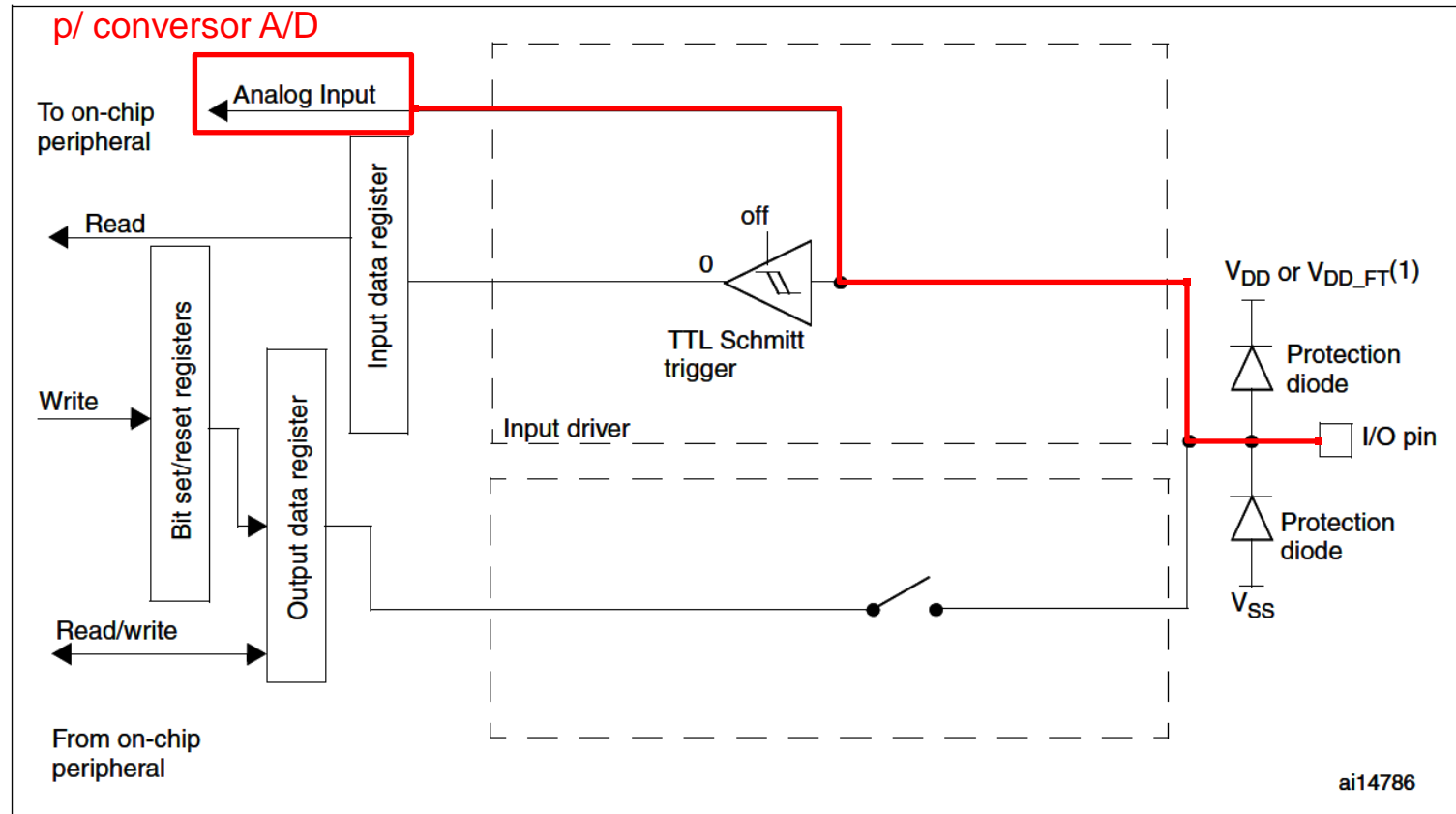
- Esta é uma estrutura básica para OUTPUT digital...

Figure 16. Output configuration



- Esta é uma estrutura básica para OUTPUT digital...

Figure 18. High impedance-analog configuration



- Circuito de RESET
- Conceito de Watchdog timers
- Circuito de CLOCK do ARM – programador de clock
- Duplicadores de frequência, divisores de frequência, e PLL
- Timers – como são estruturados e configurados os timers do ARM Cortex
- PWM – pulse width modulation (uma saída analógica dos μC e SoC)
- Configuração dos pinos de entrada e saída do CI (GPIOs)
- Configuração dos pinos para SAÍDA, ENTRADA, função ALTERNATIVA e entrada ANALÓGICA.