

EEL7030 - Microprocessadores



LCS

Laboratório de
Comunicações
e Sistemas
Embarcados

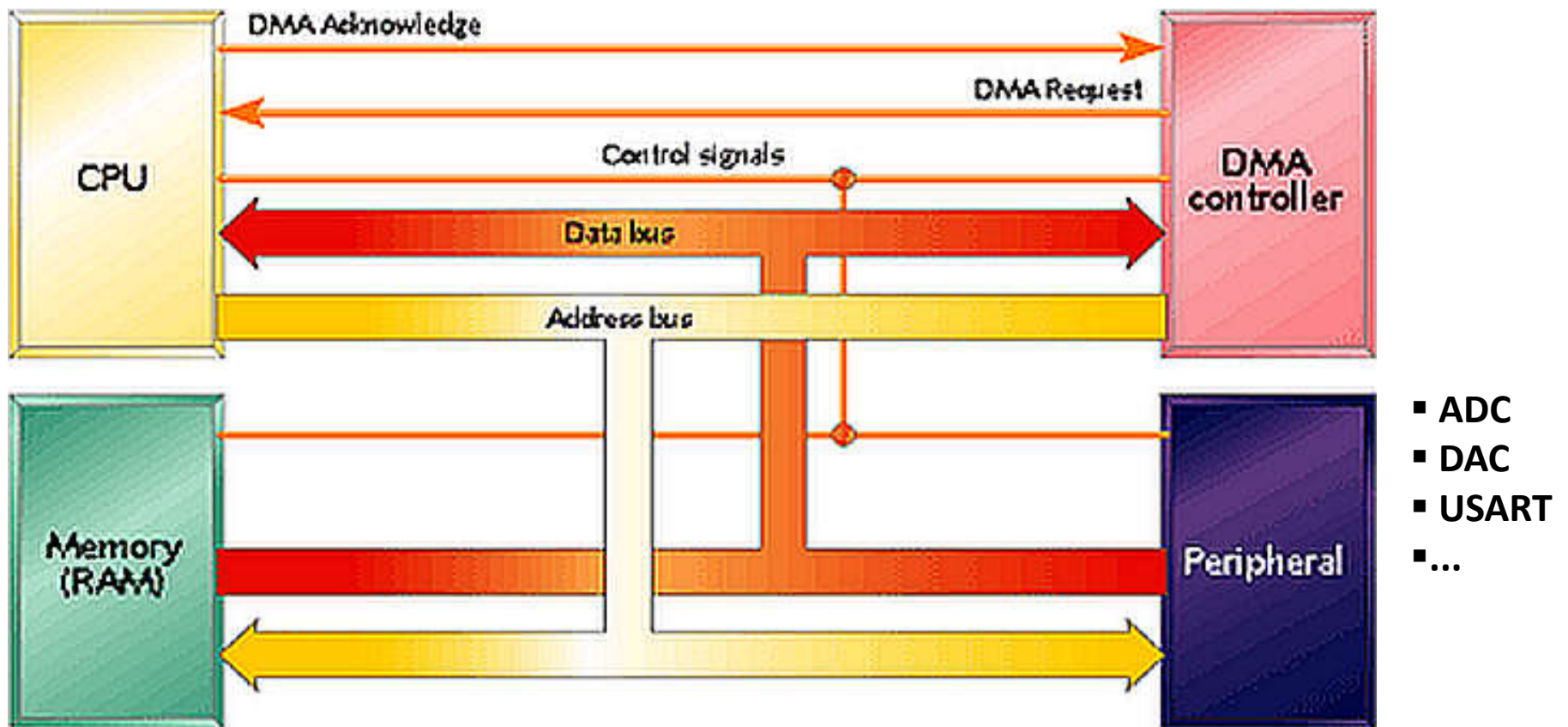
Prof. Raimes Moraes

EEL - UFSC

Direct Memory Access (DMA)

Periférico que acessa a memória RAM e outros periféricos, independentemente da UCP. Útil, por exemplo para transferir dados entre a UART e a memória, sem a necessidade de intervenção da UCP. Assim, o DMA permite que a CPU ocupe-se de outras tarefas mais críticas do que a lenta transferência de dados, melhorando o desempenho do microcontrolador.

OBS: Em arquiteturas mais antigas, DMA e UCP compartilham os mesmos barramentos como mostrado na figura abaixo. ARM permite que DMA e UCP acessem SRAM simultaneamente.



Exemplo 6 - Configurar USART com DMA

☐ Objetivo:

- Configurar UART no STM32CubeMX
- Gerar código e utilizar funções HAL
- Transmitir/Receber dados empregando DMA

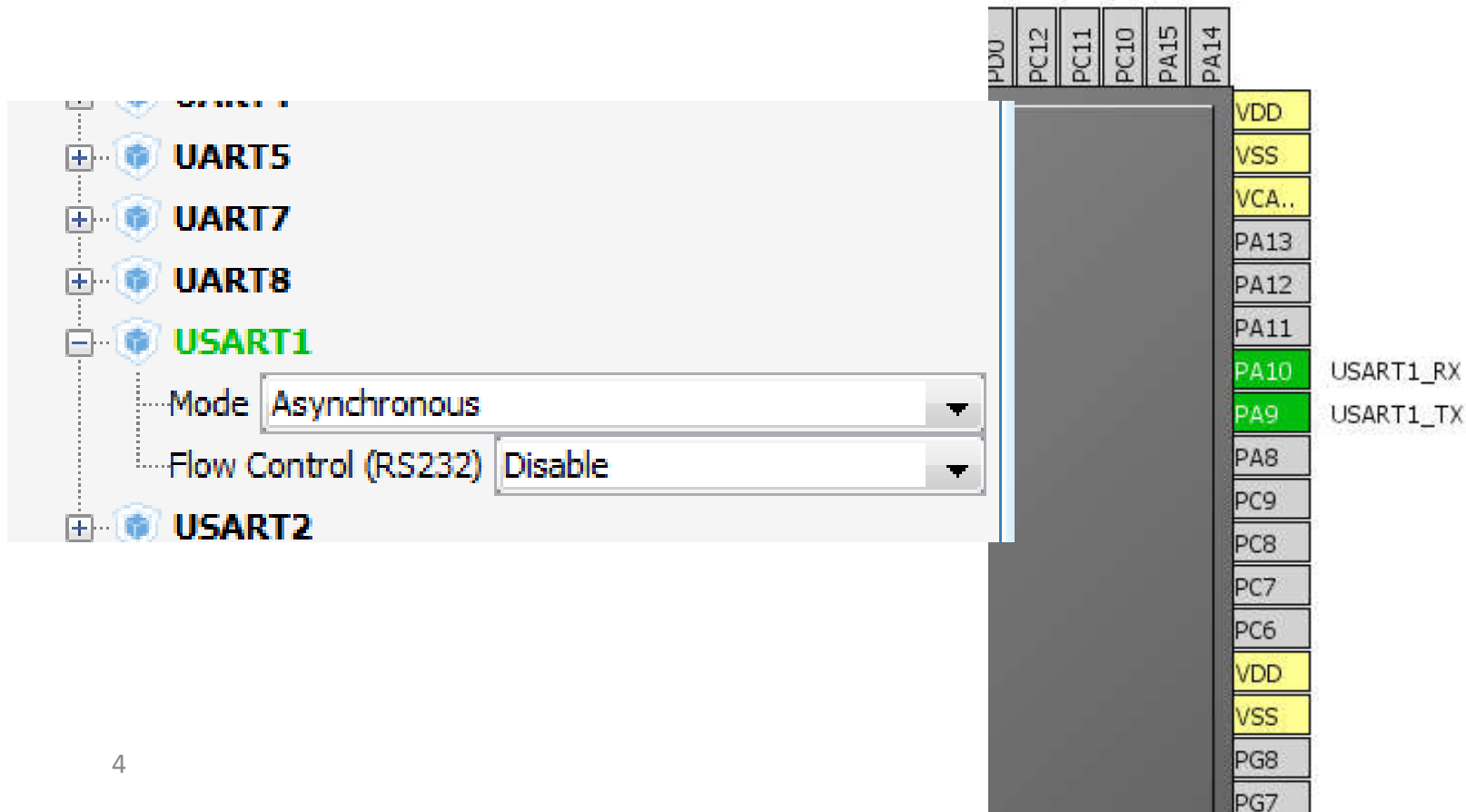
☐ Crie projeto no STMCubeMX:

1. *New Project*

2. No MCU Selector: STM32F4 (series); STM32F429/439 (lines); LQFP144 (package); selecione: STM32F429ZITx - **OBS: Selecionar crystal em RCC e configurar clocks; Selecionar SWD para Debug em SYS**

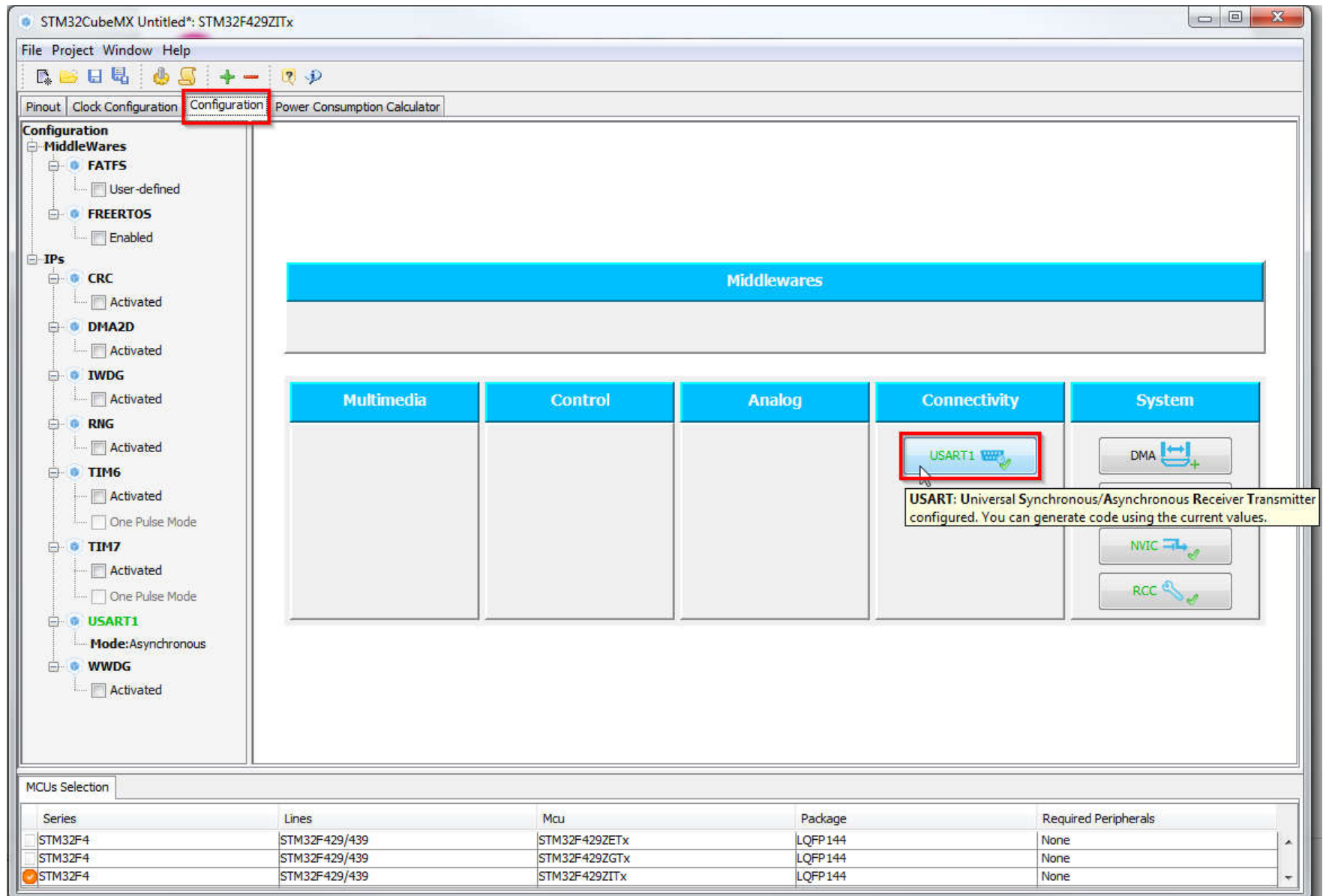
Exemplo 6 - Configurar USART com DMA

3. Seleccione USART1 e seleccione modo assíncrono
4. Clique nos pinos PA9 e PA10 e seleccione USART1_TX e USART1_RX, respectivamente



Exemplo 6 - Configurar USART com DMA

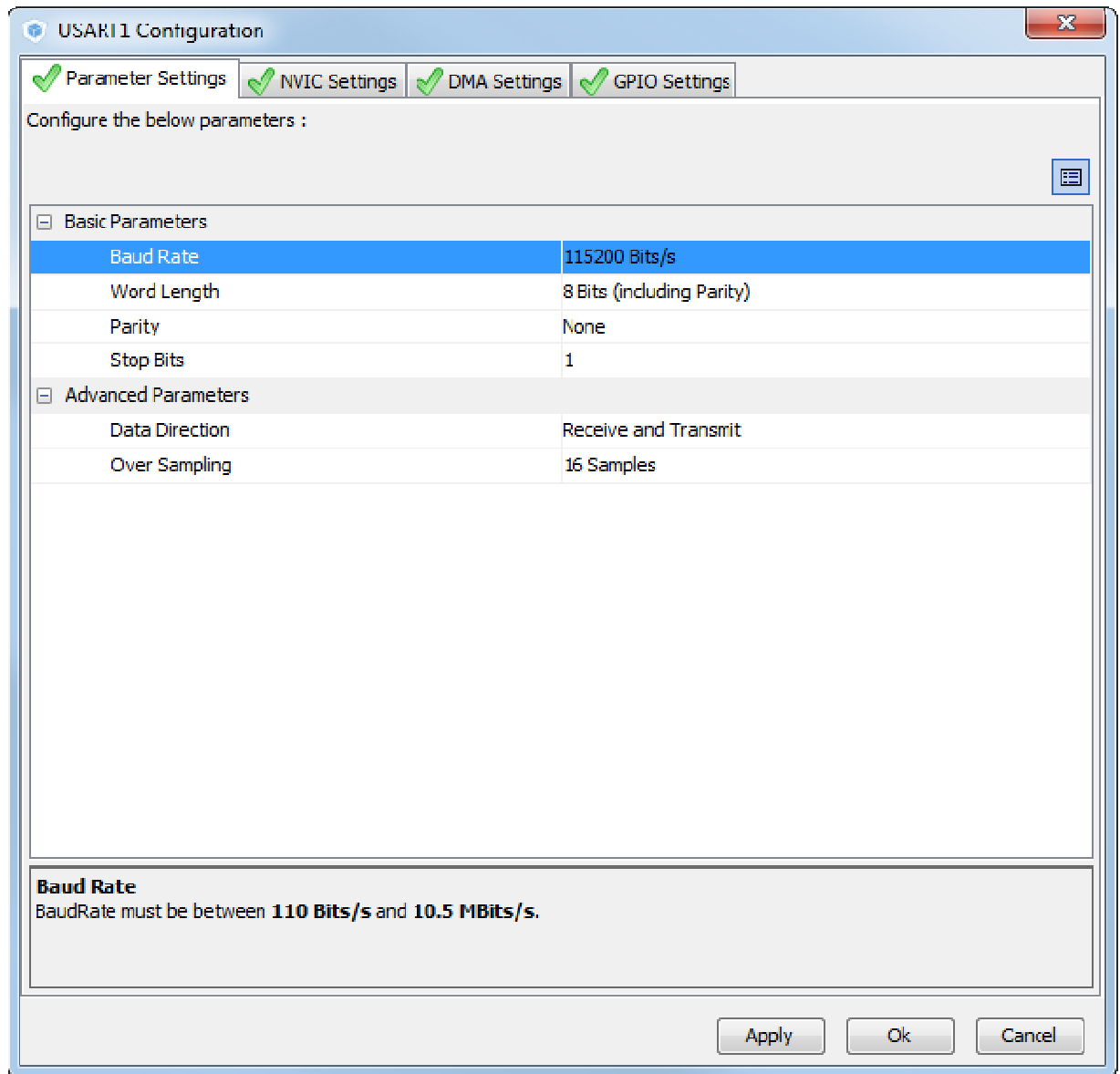
❑ Configure o USART1 => Tab>Configuration>Connectivity>USART1



Exemplo 6 - Configurar USART com DMA

5. Configure a USART1:

- *Baud Rate*
- *Word Length*
- *Parity*
- *Stop Bits*
- *Data Direction*
- *Over Sampling*



The image shows a 'USART1 Configuration' dialog box with four tabs: 'Parameter Settings' (active), 'NVIC Settings', 'DMA Settings', and 'GPIO Settings'. Each tab has a green checkmark icon. The main area is titled 'Configure the below parameters :'. It contains two expandable sections: 'Basic Parameters' and 'Advanced Parameters'. The 'Basic Parameters' section is expanded, showing a table with the following settings:

Parameter	Value
Baud Rate	115200 Bits/s
Word Length	8 Bits (including Parity)
Parity	None
Stop Bits	1

The 'Advanced Parameters' section is also expanded, showing a table with the following settings:

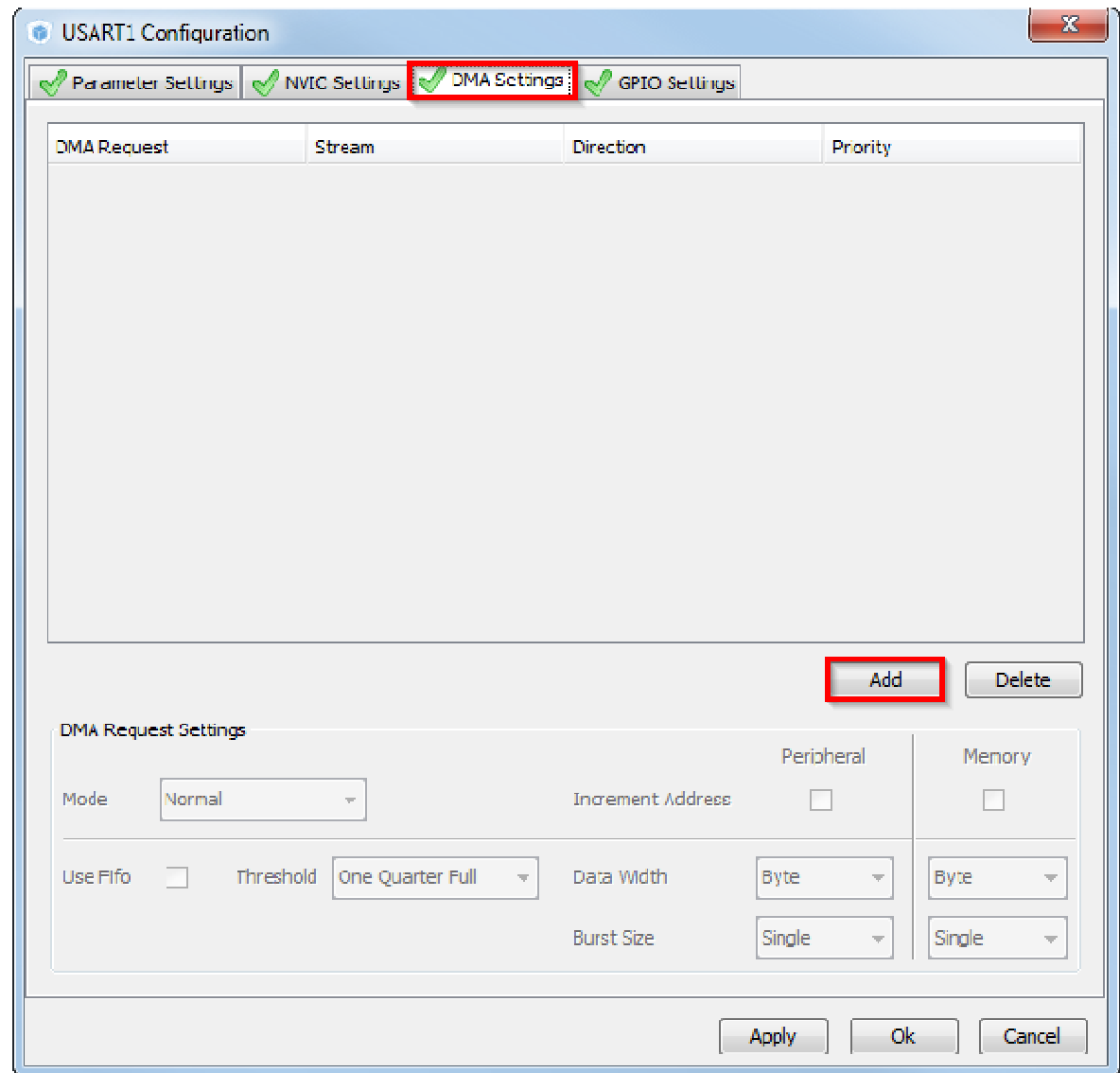
Parameter	Value
Data Direction	Receive and Transmit
Over Sampling	16 Samples

At the bottom of the dialog, there is a 'Baud Rate' section with a warning: 'BaudRate must be between 110 Bits/s and 10.5 MBits/s.' The bottom right corner contains three buttons: 'Apply', 'Ok', and 'Cancel'.

Exemplo 6 - Configurar USART com DMA

6. Configurar USART para utilizar DMA:

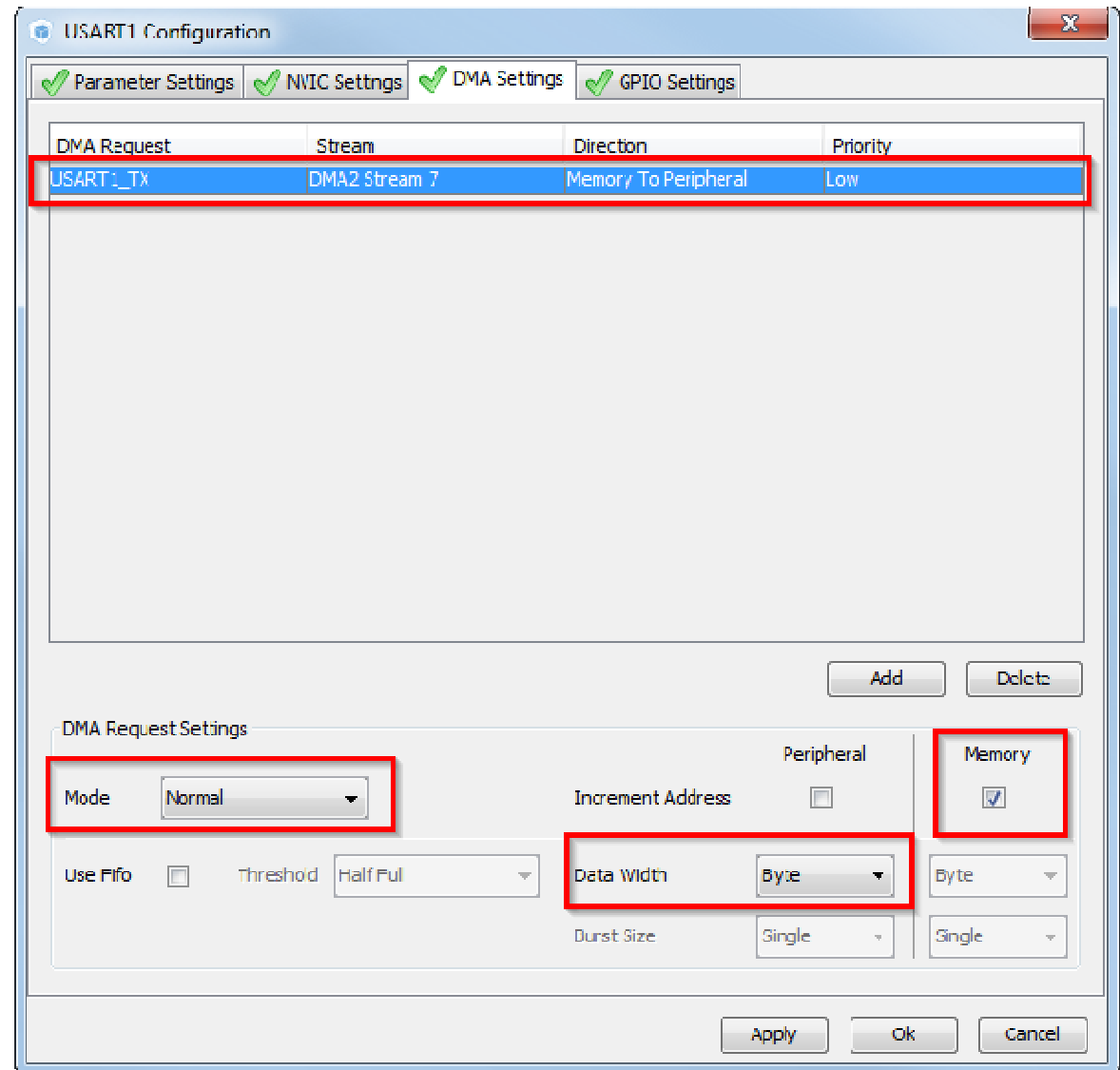
- Tab-> *DMA Settings*
- Clique em *Add*



Exemplo 6 - Configurar USART com DMA

7. Configurar USART para transmitir, selecionando :

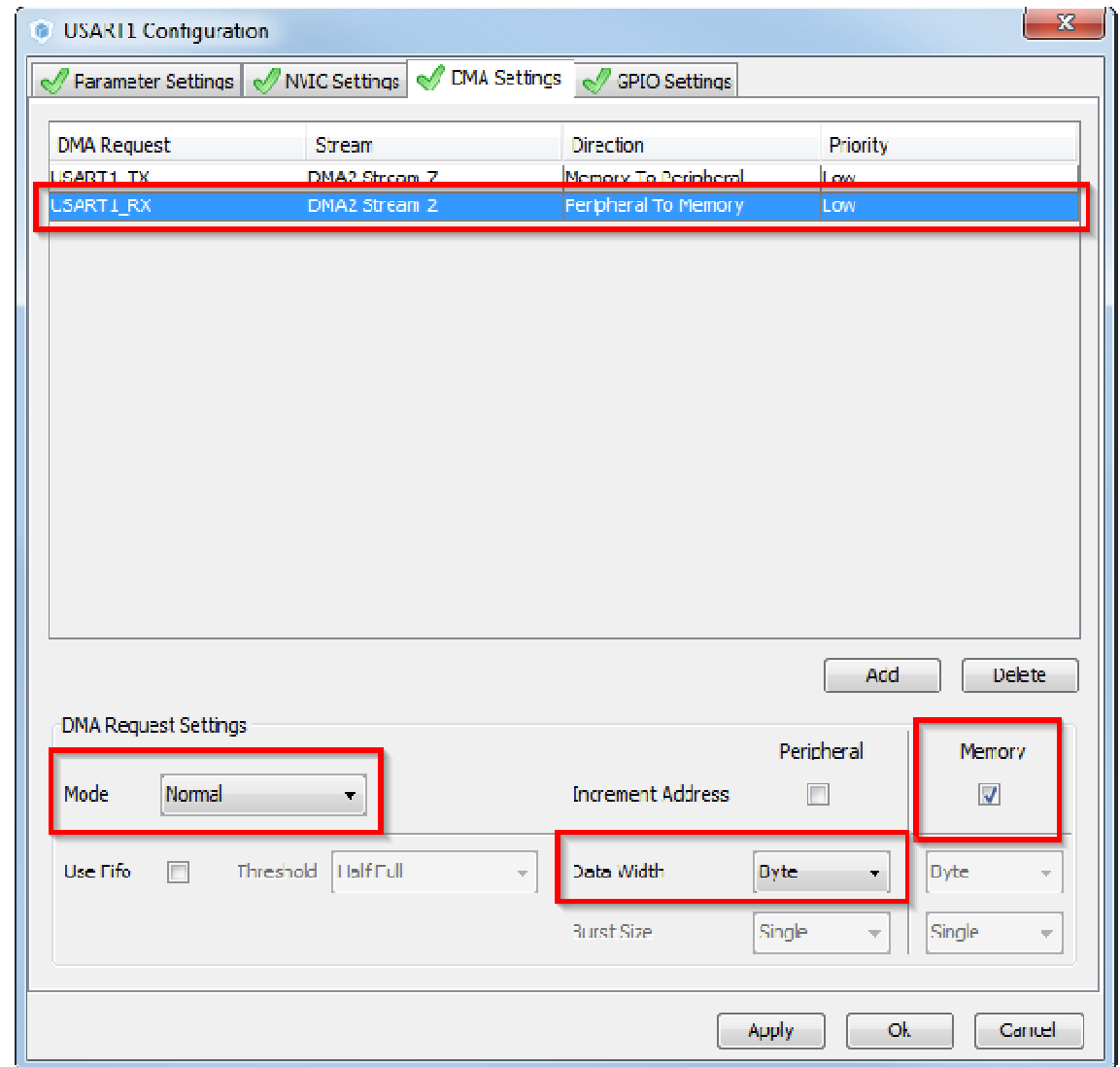
- *USART1_TX*
- *Direction: Memory to Peripheral*
- *Mode: Normal*
- *Increment Address: Memory*
- *Data Width: Byte*



Exemplo 6 - Configurar USART com DMA

8. Configurar USART para receber, selecionando :

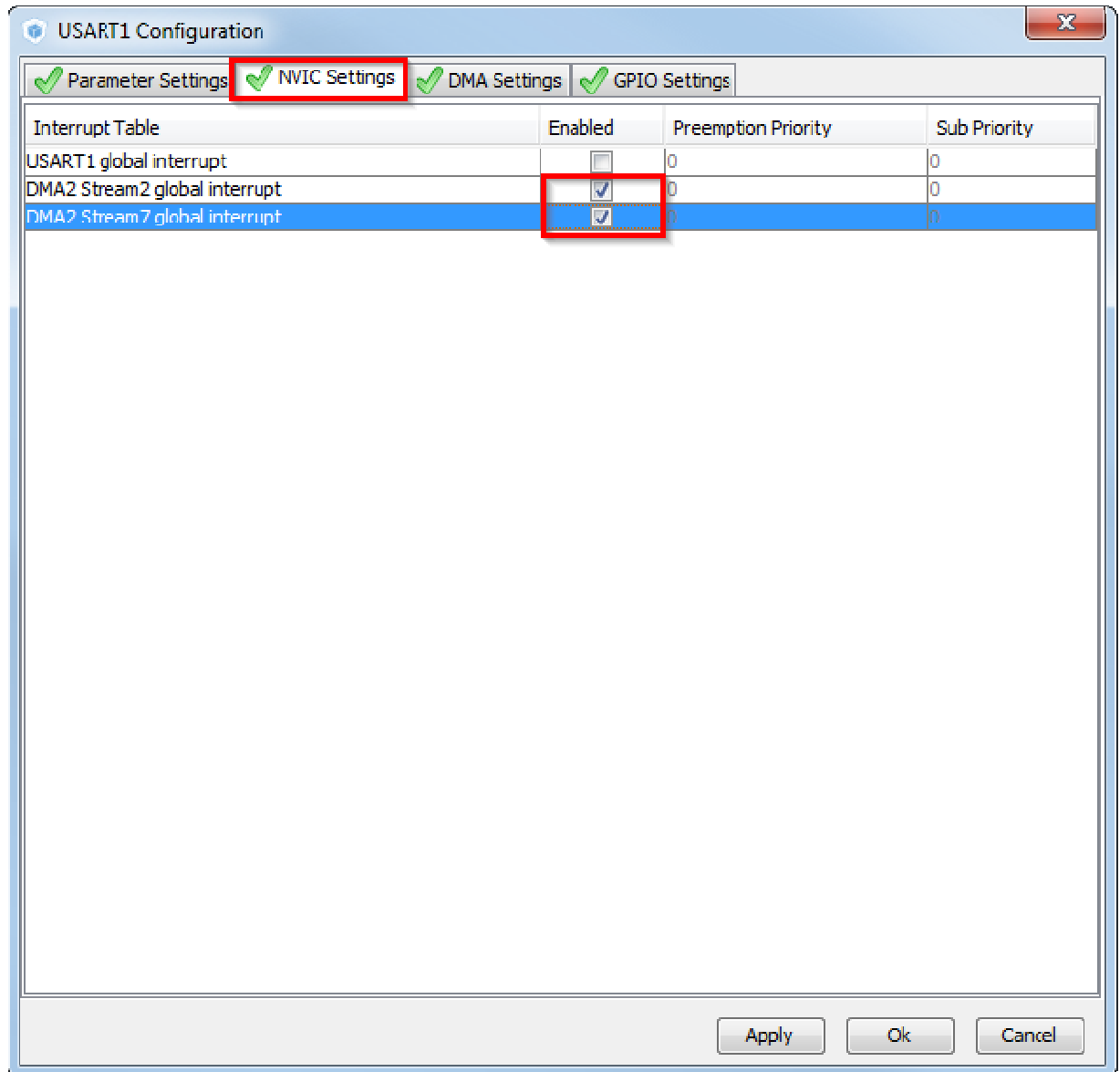
- *ADD*
- *USART1_RX*
- *Direction: Peripheral to Memory*
- *Mode: Normal*
- *Increment Address: Memory*
- *Data Width: Byte*



Exemplo 6 - Configurar USART com DMA

9. Configurar registradores para gerar interrupção:

- Tab-> *NVIC Settings*
- Habilite interrupções do DMA2 para USART1
- Clique em Ok



Exemplo 6 - Configurar USART com DMA

10. Forneça dados para geração do código:

- Menu -> Project -> Project Settings
- Digite *Project name*
- Digite *Project location*
- Escolha *toolchain* (MDK-ARM V5)
- Clique em Ok

11. Gere template para código:

- Menu -> Project -> Generate Code
- *Open Project*

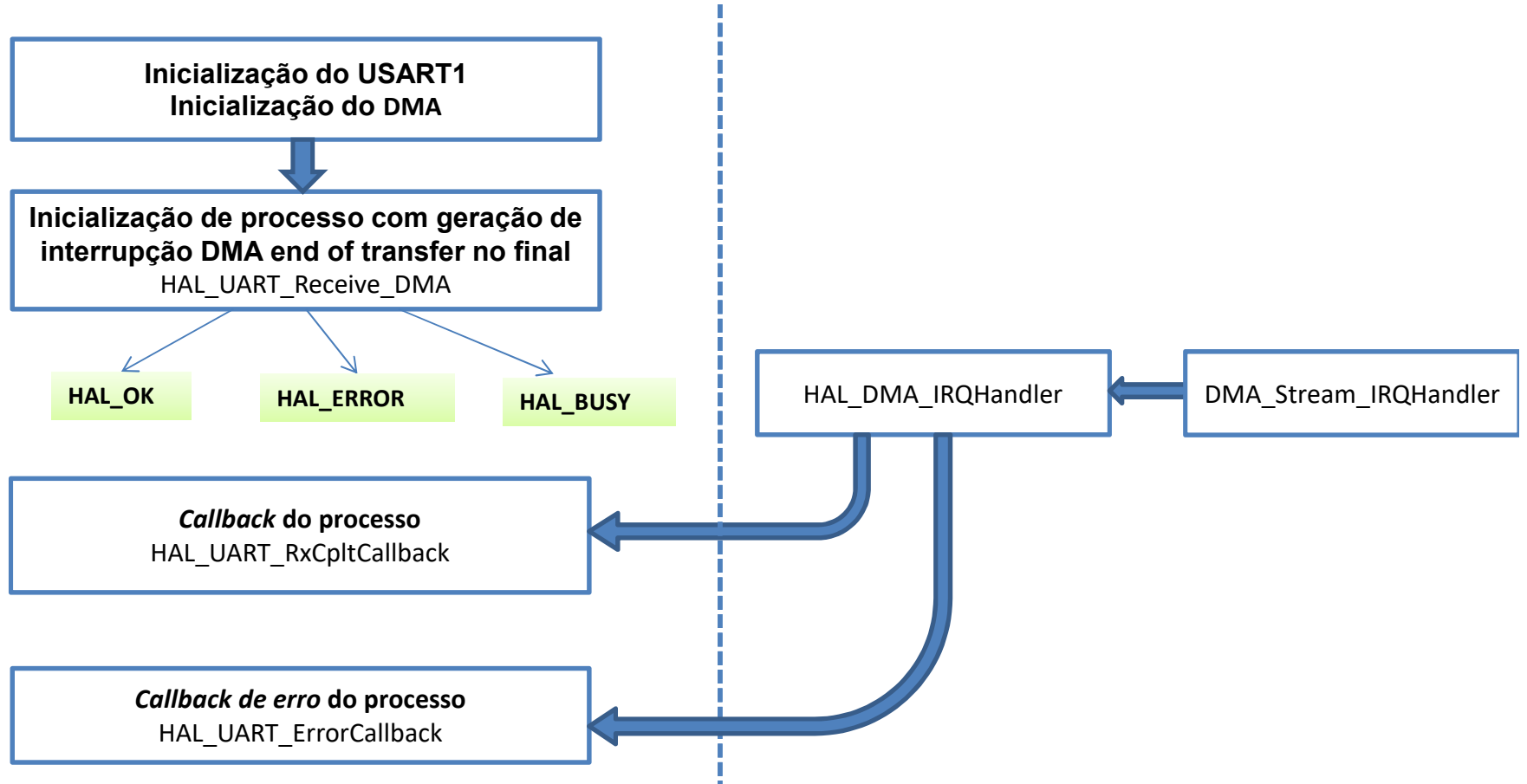
The screenshot shows the 'Project Settings' dialog box with the 'Project' tab selected. The 'Code Generator' sub-tab is also visible. The following fields are filled out:

- Project Name:** First
- Project Location:** C:\Users\Raines\STM\Exemplos\ (with a 'Browse' button next to it)
- Toolchain Folder Location:** C:\Users\Raines\STM\Exemplos\First\
- Toolchain / IDE:** MDK-ARM V5 (selected from a dropdown menu)
- Mcu Reference:** STM32F429ZITx
- Firmware Package Name and Version:** STM32Cube FW_F4 V1.9.0

At the bottom right, there are 'Ok' and 'Cancel' buttons.

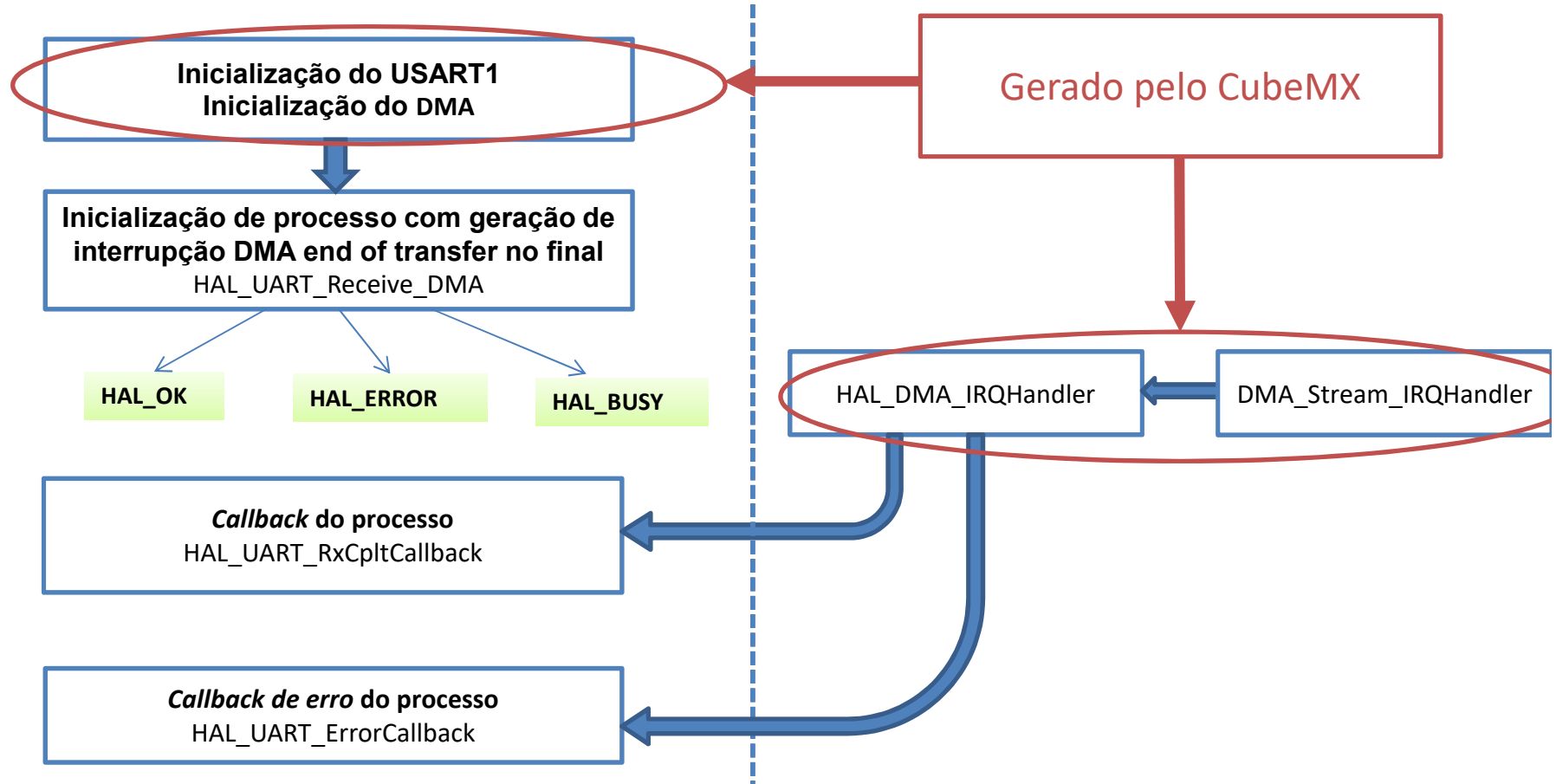
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



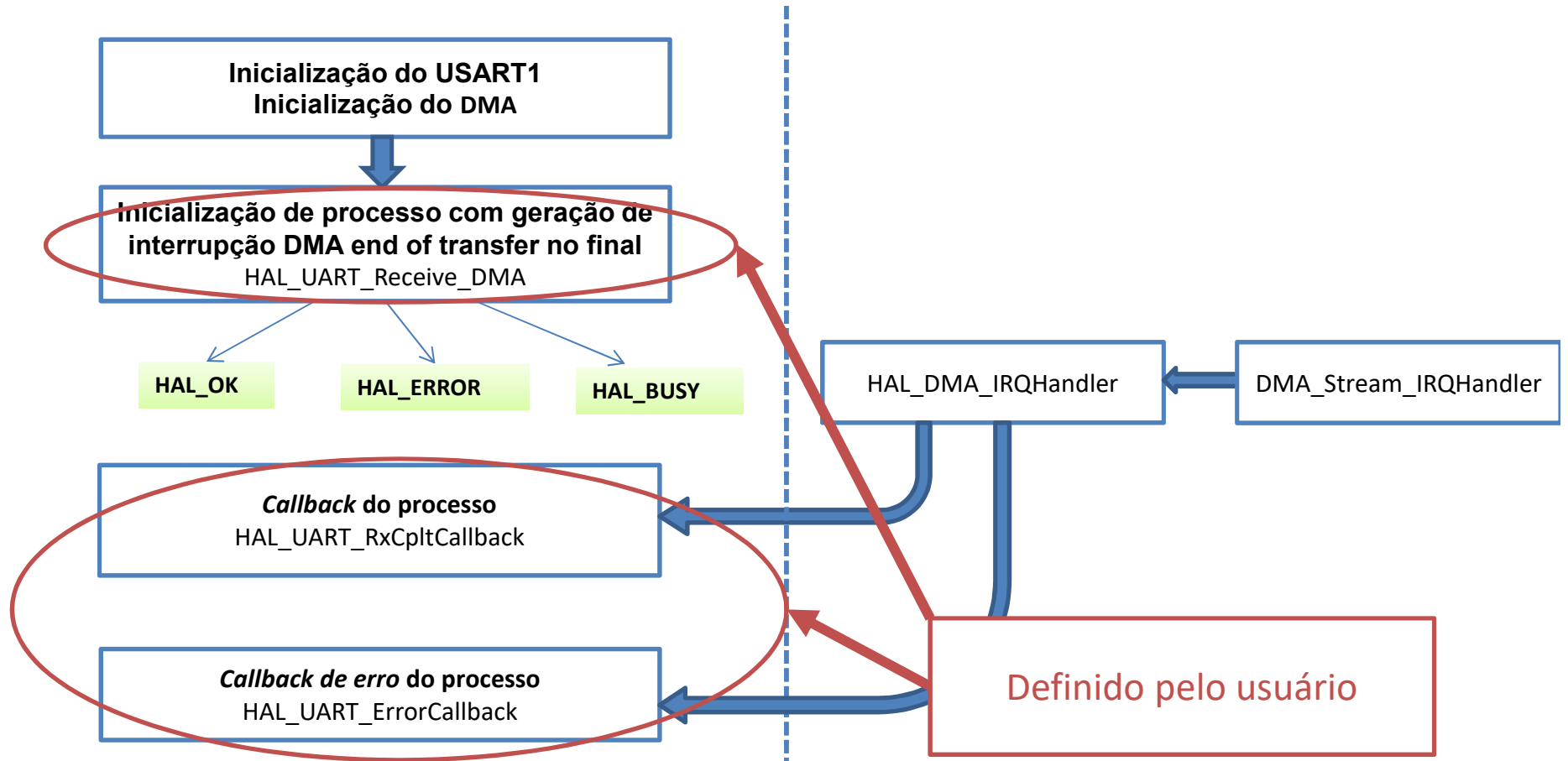
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



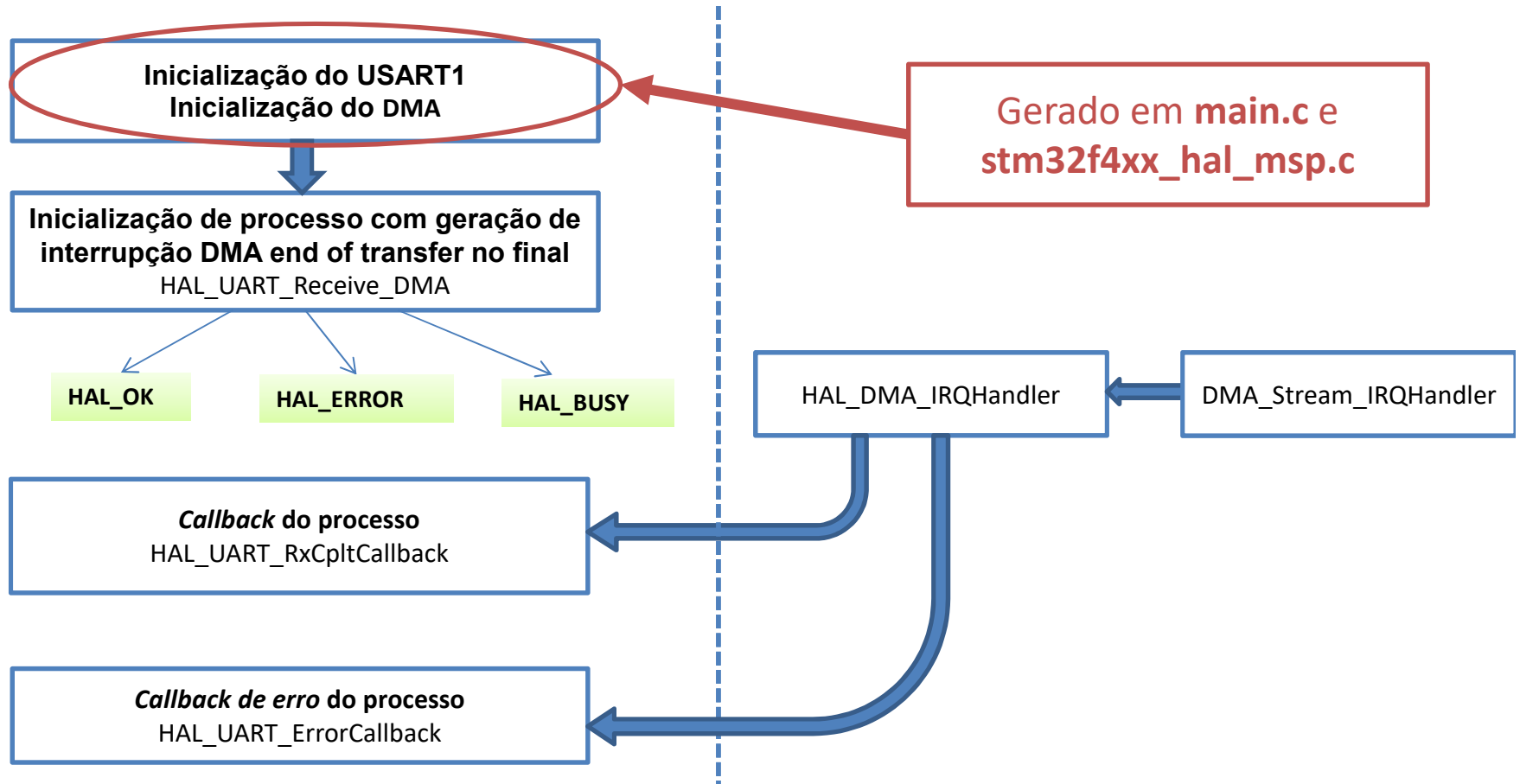
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



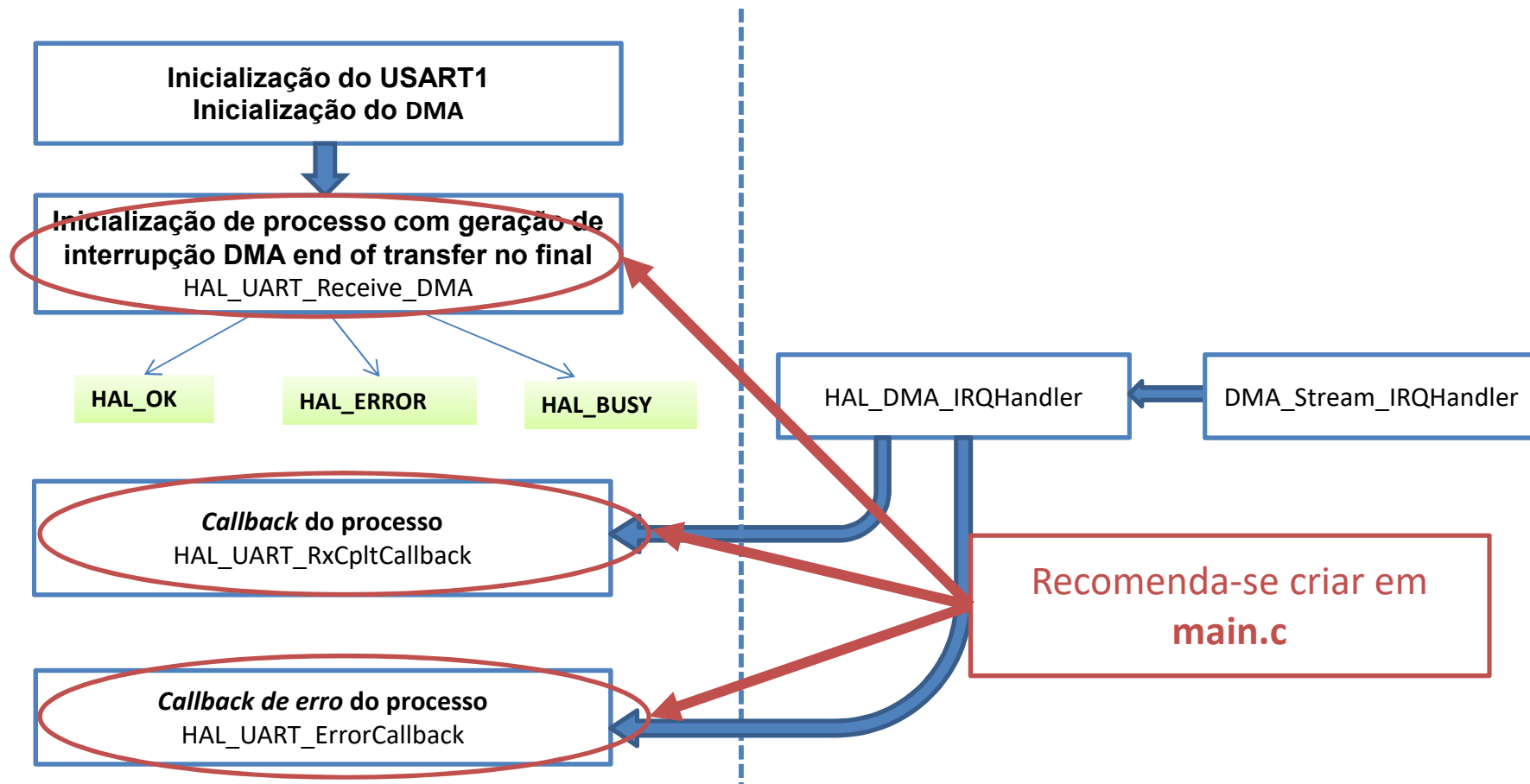
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



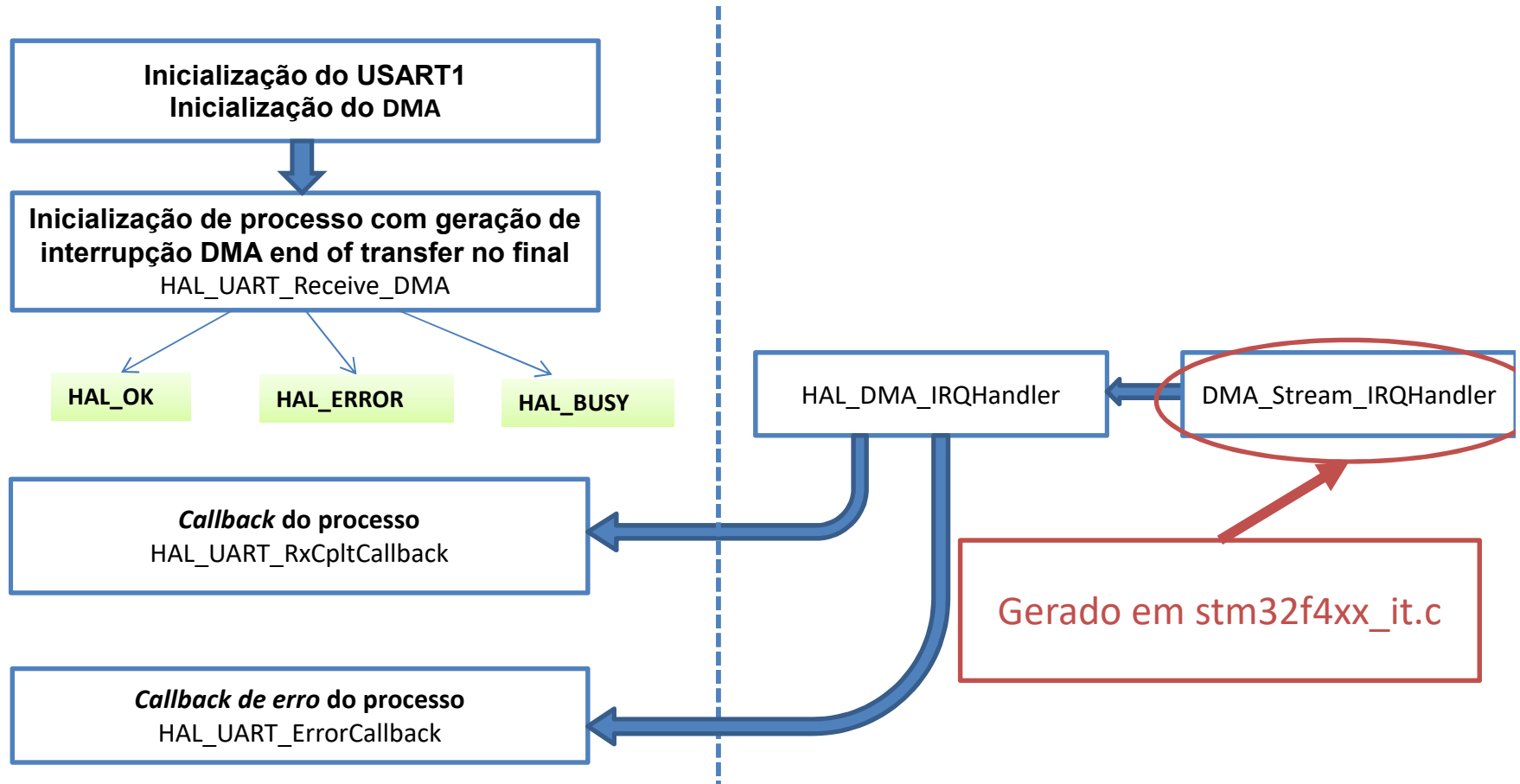
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



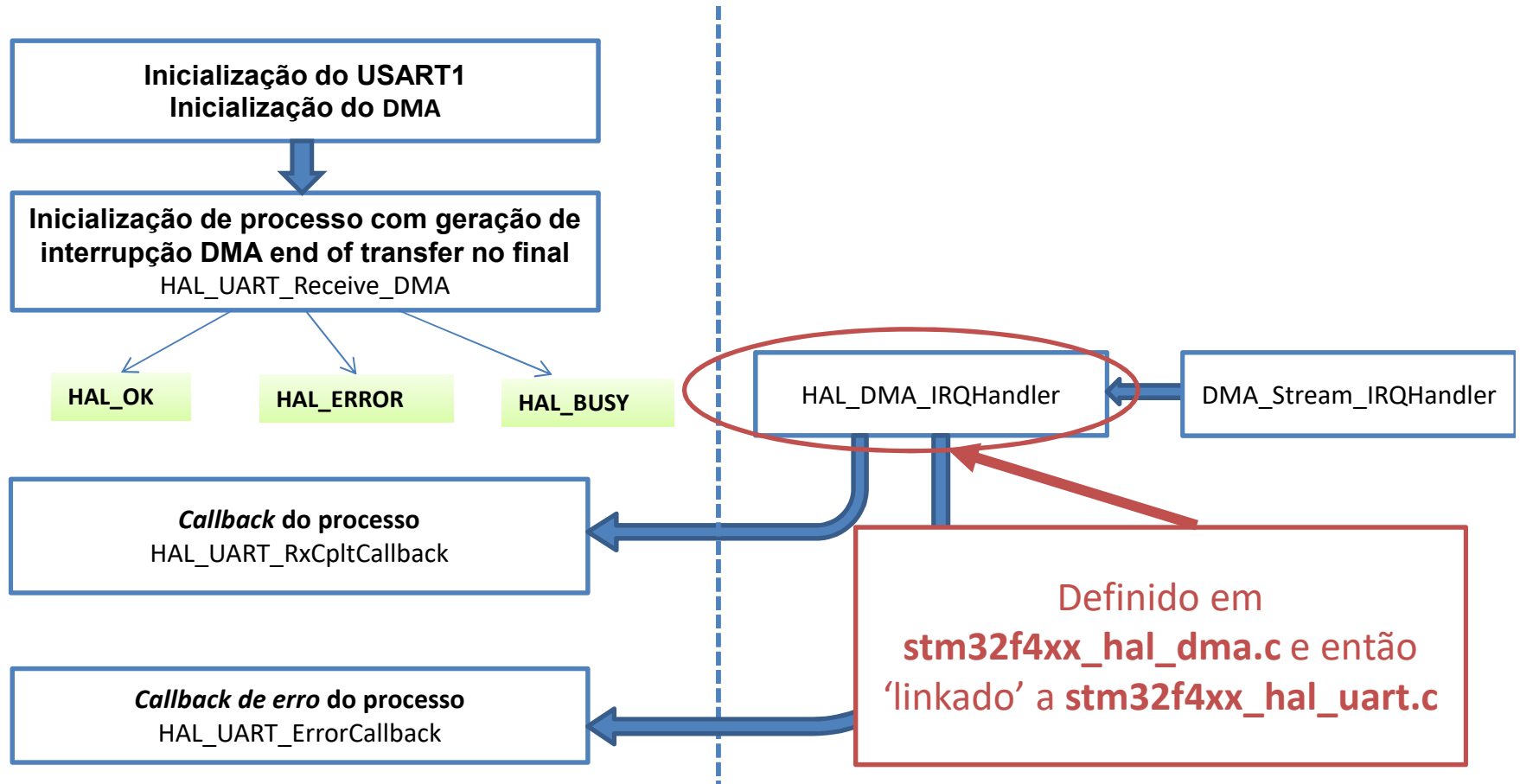
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



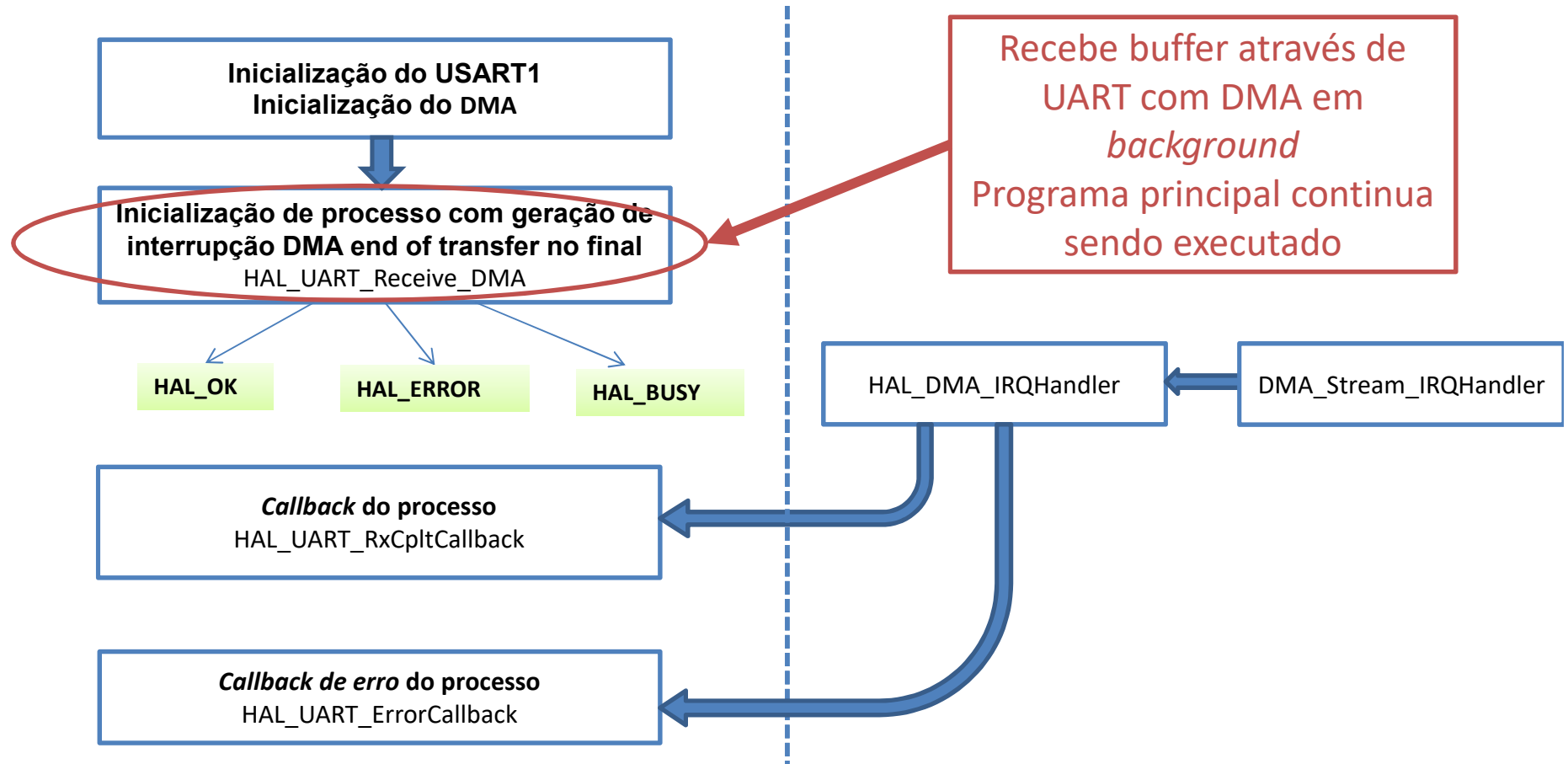
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



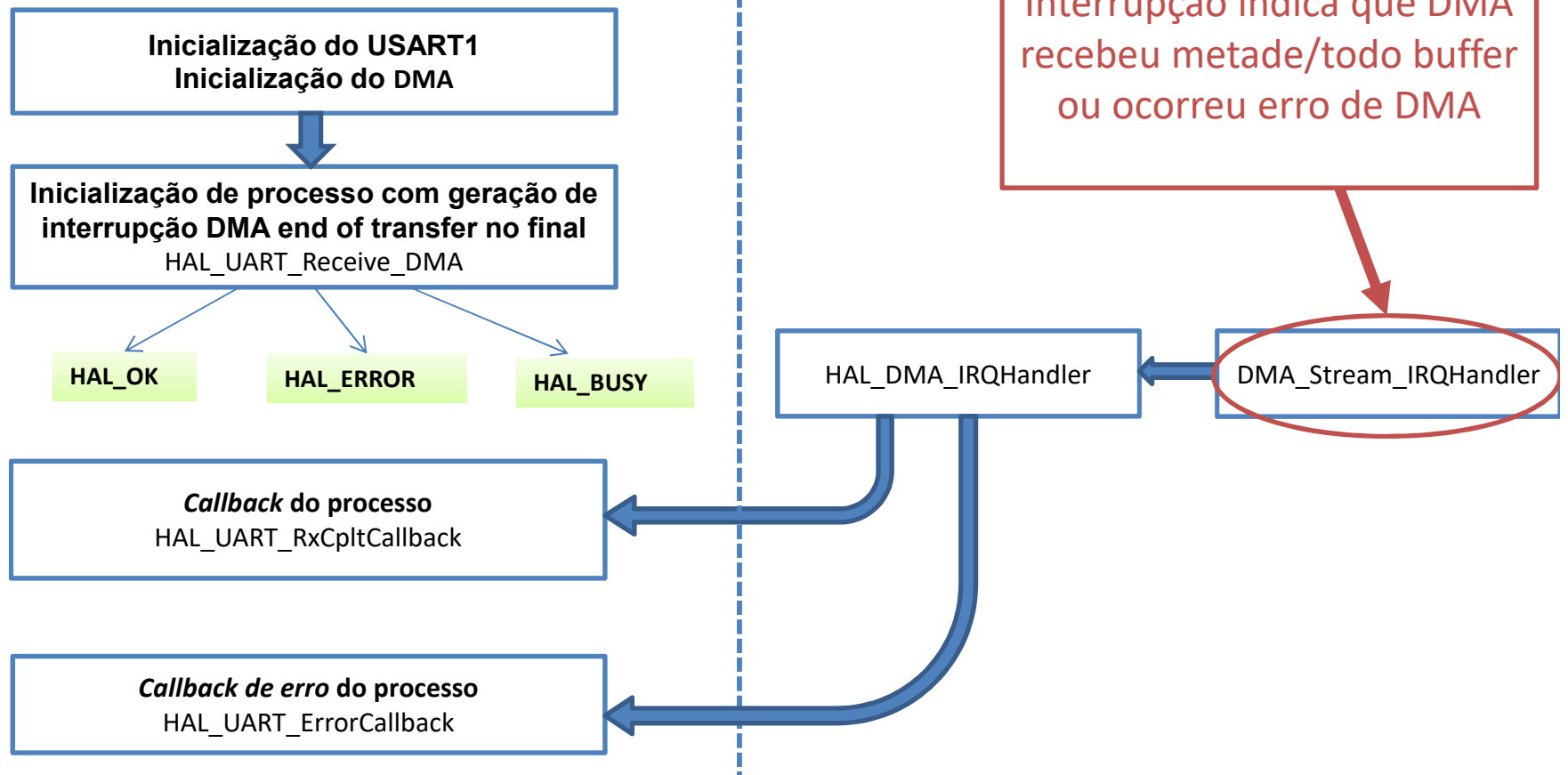
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



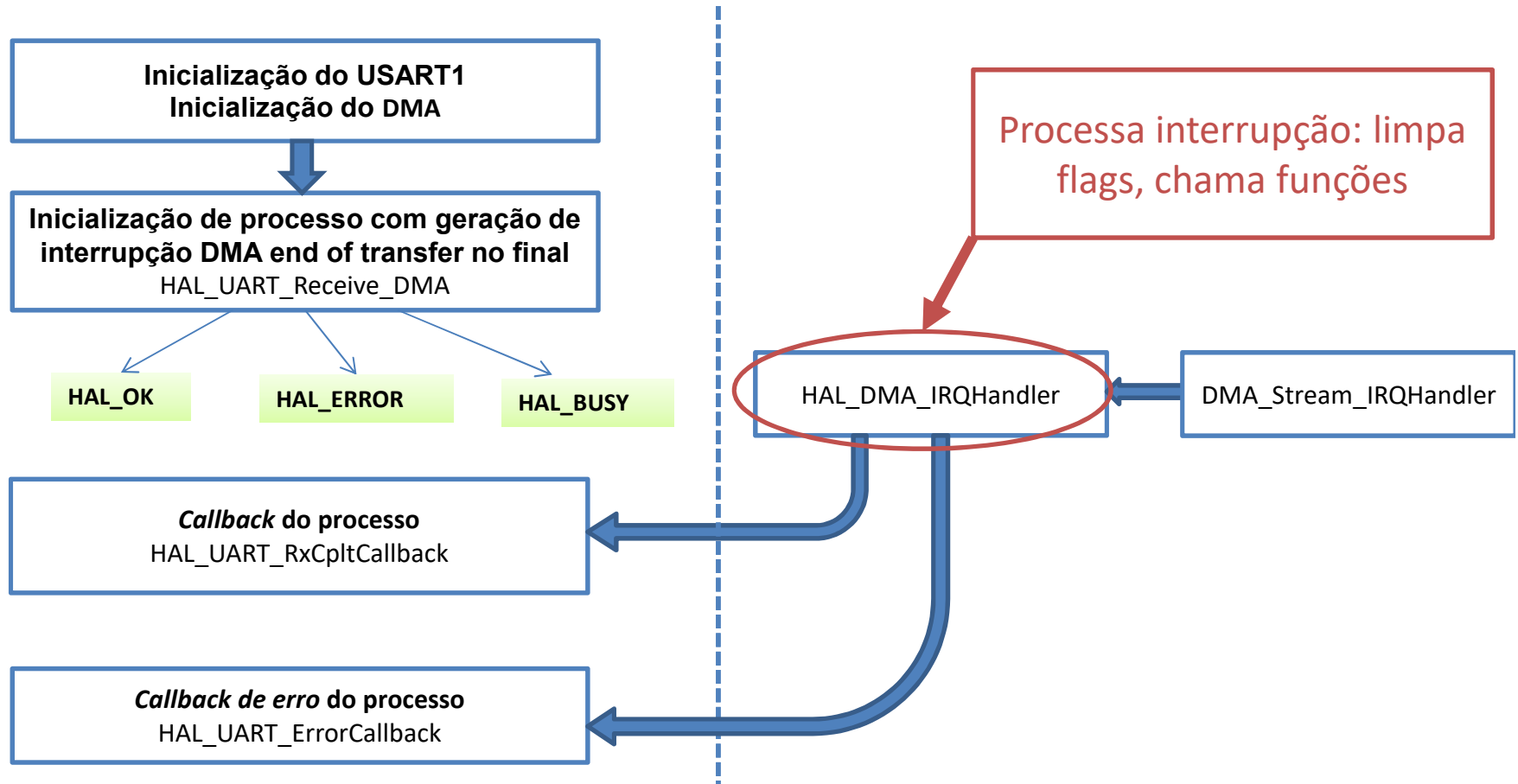
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



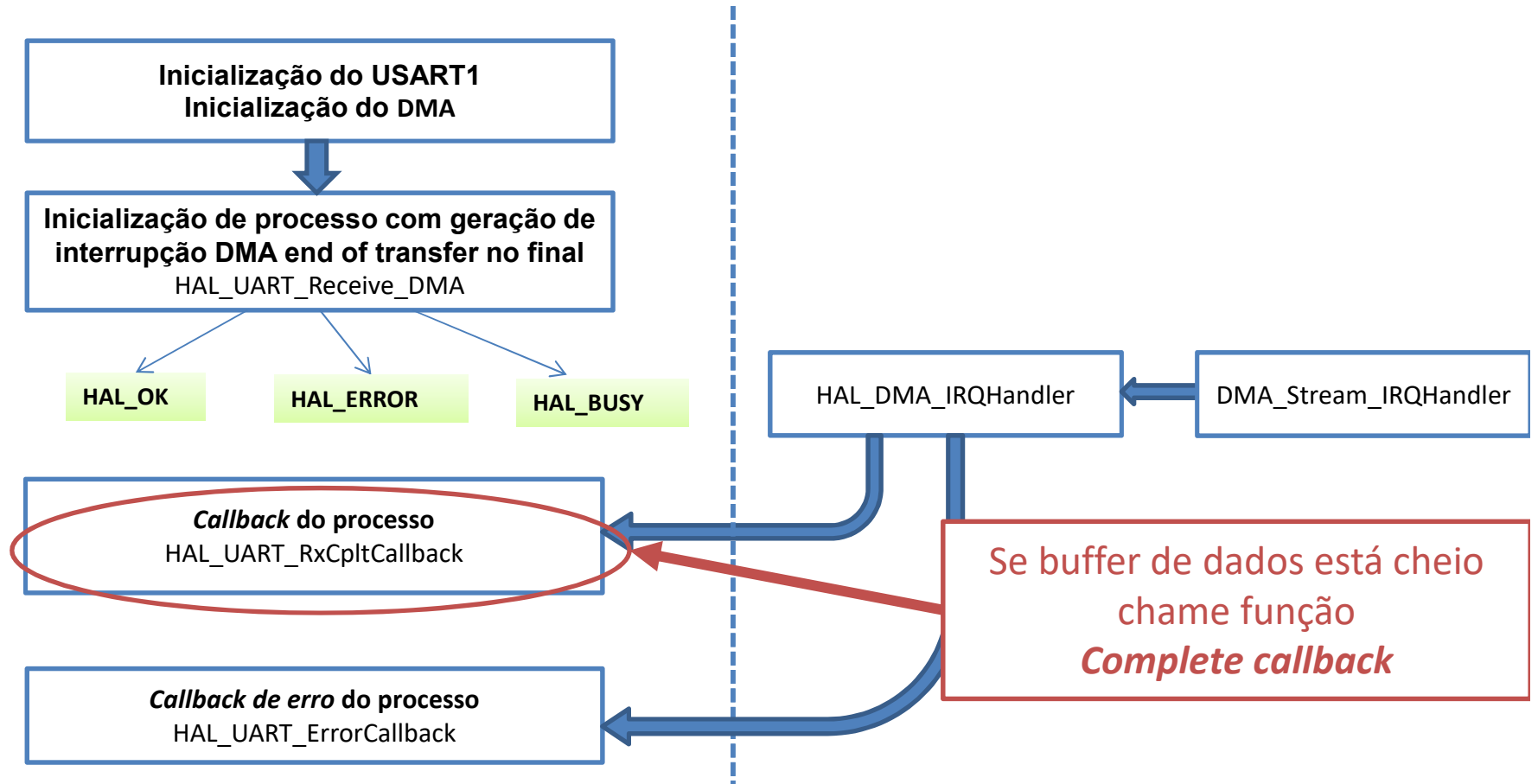
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



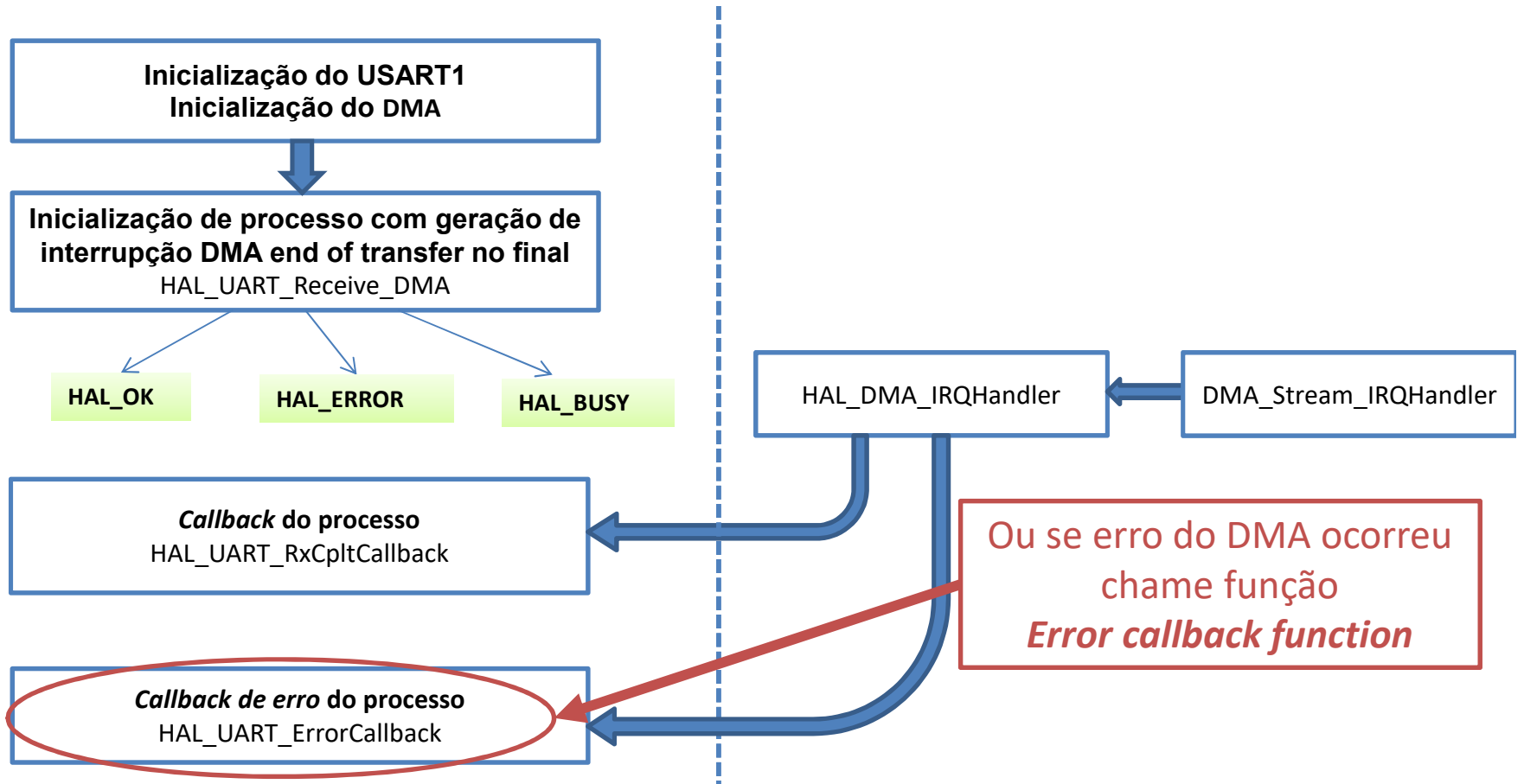
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



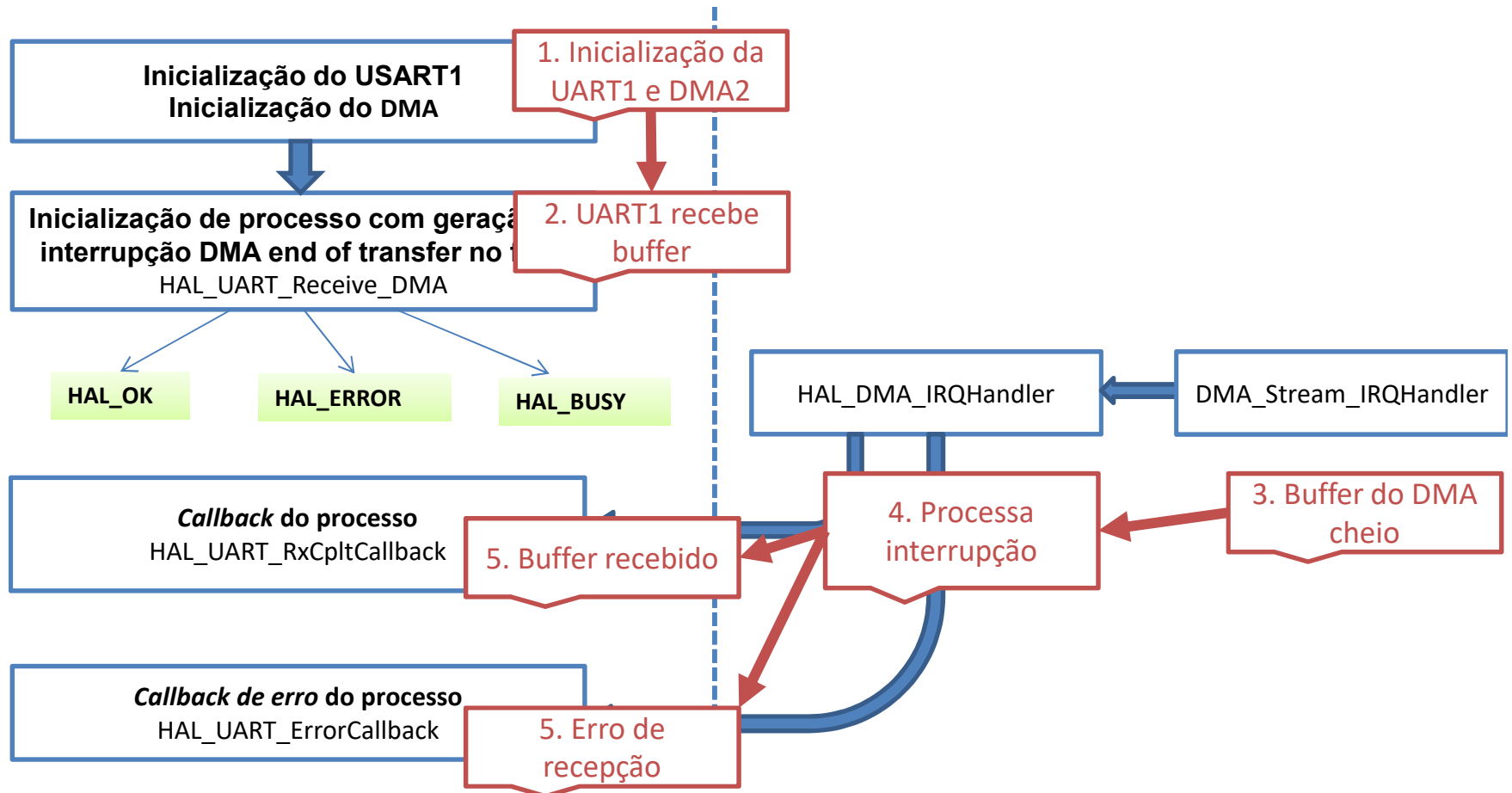
Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



Exemplo 6 - Configurar USART com DMA

HAL Library UART COM DMA Rx



Exemplo 6 - Configurar USART com DMA

12. Abra o projeto no Keil

13. Deve-se inserir funções em **main.c** (em **Application User**) entre os rótulos
/ USER CODE BEGIN 2 */* e */* USER CODE END 2 */*

- Para a transmissão, utilize a função:

```
HAL_UART_Transmit_DMA(UART_HandleTypeDef *huart, uint8_t *pData,  
uint16_t Size);
```

- Para a recepção, utilize a função:

```
HAL_UART_Receive_DMA(UART_HandleTypeDef *huart, uint8_t *pData,  
uint16_t Size);
```

Exemplo 6 - Configurar USART com DMA

14. Criar vetores de transmissão e recepção:

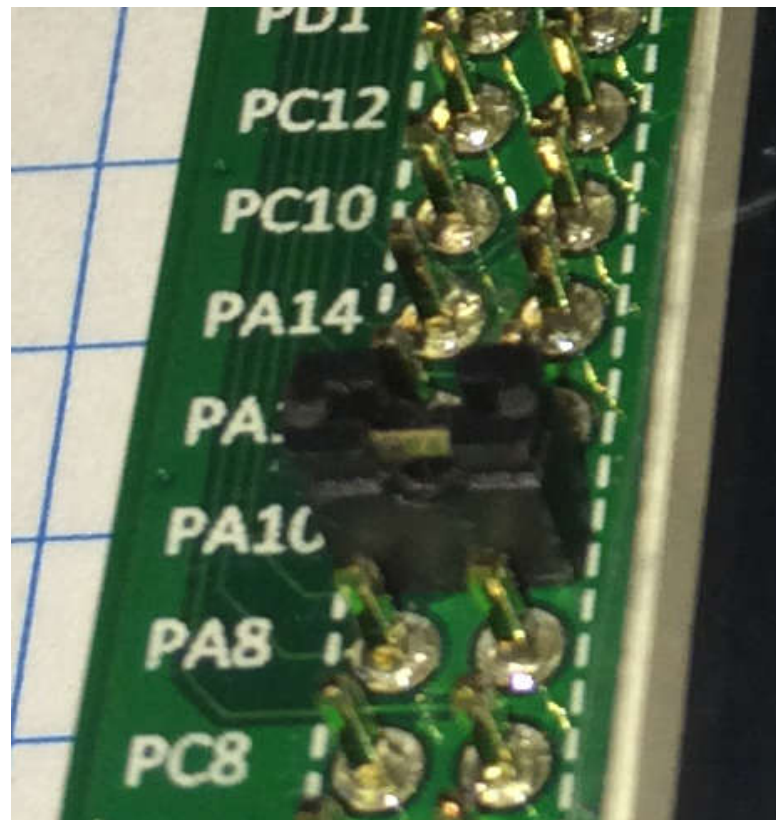
```
/* USER CODE BEGIN 0 */  
uint8_t tx_buff[]={0,1,2,3,4,5,6,7,8,9};  
uint8_t rx_buff[10];  
/* USER CODE END 0 */
```

15. Inicializar funções do DMA para transmissão e recepção:

```
/* USER CODE BEGIN 2 */  
HAL_UART_Receive_DMA(&huart1,rx_buff,10);  
HAL_UART_Transmit_DMA(&huart1,tx_buff,10);  
/* USER CODE END 2 */
```

Exemplo 6 - Configurar USART com DMA

16. Conecte pino PA9 ao PA10 para transmitir e receber dados na mesma interface



Exemplo 6 - Configurar USART com DMA

17. Testar função de *callback*, inserindo breakpoint no NOP para verificar final de recepção de buffer

```
/* USER CODE BEGIN 4 */  
void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart)  
{  
    __NOP(); //check if we receive all data  
}  
/* USER CODE END 4 */
```

18. Compile o programa e grave-o no kit
19. Verifique no Keil através de *debug*, preenchimento do buffer de recepção (rx_buff).