Technical Summary: STM32F303RE ADC+DMA Implementation (Task 2)

Project Background

This project involved implementing a data acquisition system on the STM32F303RE microcontroller to sample analog signals from three sensors using ADC with DMA, then transmit the readings via UART at regular intervals.

Implementation Challenges

During development, significant challenges were encountered with the ADC and DMA integration. Despite following standard initialization procedures, the system consistently halted at the HAL_ADC_Start_DMA() function call. Initial debugging revealed that the ADC's DMA handle was NULL after initialization, suggesting an issue with the HAL library's handling of the STM32F3 architecture.

Technical Solution

After consulting with external resources (datasheet and claude chat bot), I determined that the STM32F3 series has specific ADC and DMA requirements not fully addressed by the standard HAL implementation. The solution involved:

1. Implementing direct register configuration for the ADC and DMA peripherals
2. Explicitly enabling the ADC12 clock domain with `__HAL_RCC_ADC12_CLK_ENABLE()`
3. Manually linking the DMA to the ADC using `__HAL_LINKDMA()`
4. Configuring the DMA for circular mode operation with the appropriate data alignment

This approach bypassed the limitations in the HAL library while maintaining the remainder of the HAL infrastructure for other peripherals. I am not sure why such trivial code did not work using HAL library but I just couldn’t make it work so I had to bypass HAL library for this specific function HAL_ADC_Start_DMA().

ADC Sampling Frequency

The ADC is configured with a clock prescaler of PCLK/4 and a sampling time of 7.5 cycles for each channel. With the STM32F303RE running at 72MHz and PCLK at 36MHz, this yields an

ADC clock of 9MHz. For a 12-bit resolution, each conversion takes approximately 15 ADC clock cycles (7.5 for sampling plus conversion time), resulting in:

- Individual channel sampling rate: ~600 kHz
- Three-channel sequence sampling rate: ~200 kHz

This means each channel is sampled approximately 200,000 times per second, far exceeding the requirement of 1 kHz per channel. The actual sampling rate could be slightly lower due to DMA overhead, but remains well above the specified requirements.

Outcome

The final implementation successfully samples all three ADC channels at a rate exceeding 1kHz per channel. The system transmits formatted data over UART every 100ms, handles button interrupts for timestamp resets, and provides visual feedback via LED toggling.

This project demonstrates the importance of understanding hardware-specific details when working with complex peripherals, particularly when standard library abstractions may not fully account for architectural differences between microcontroller families.

Source code can be found here:
https://github.com/RonNeter/ms_adc_dma_task