WINDOWS DRIVER FOUNDATION - WDF

This file consists of a very **basic, and working understanding of WDF (Window Driver Foundation)** for this project. These are the **major and high-level points** to look out for in this README.

1.) What is WDF?
2.) Some Examples from the current code.

NOTE: This README only gives a working understanding of WDF for **this** project, however if you are curious about WDF you may look it up here: https://learn.microsoft.com/en-us/windows-hardware/drivers/wdf/using-the-framework-to-develop-a-driver

What is WDF?

It is a set of software tools and libraries that Microsoft provides to **help developers write device drivers for Windows operating systems**. We can say that WDF is a **Framework.** WDF consists of several features inbuilt with it:

- A consistent object model for device (driver) objects, queues, and other driver components.
- A model which consists of both Synchronous and Asynchronous I/O.
- A device-specific pnp (plug and play) model to make it easier to add a new device easily.
- Tools and Utility for developers to debug and troubleshoot drivers.

WDF has two layers consisting in it:

- KMDF: Kernel-mode Driver Framework - More privileged
- UMDF: User-mode Driver Framework - Less privileged

Some Examples from the current code

- **`WDF_DEVICE_INIT_CONTEXT:`** A struct to initialize the properties of a device driver.
- **`EVT_WDF_DEVICE_D0_ENTRY:`** Event is raised when the driver device is put to D0 state.