

## **5. BASE SOFTWARE OVERVIEW (BSW)**

### **Giới thiệu**

. **Autosar:** Kiến trúc mở cho hệ thống ô tô.

. **Ba lớp phần mềm chính:**

- . Lớp ứng dụng (Application Software - ASW)
- . Môi trường chạy thời gian thực (Runtime Environment - RTE)
- . Phần mềm cơ bản (Basic Software - BSW)

### **Ứng dụng chính**

. **Phân tách phần mềm độc lập với phần cứng và phần mềm cơ bản:**

. RTE tạo lớp trừu tượng hóa giữa phần cứng và phần mềm, giúp phát triển ứng dụng không phụ thuộc vào phần cứng.

. **Tính năng của kiến trúc phần mềm Autosar:**

- . Tăng tính linh hoạt và khả năng tái sử dụng phần mềm.
- . Cho phép phân phối ứng dụng và các mô-đun phần mềm chức năng trên nhiều ECU.
- . Khả năng tích hợp các mô-đun phần mềm từ các nguồn khác nhau.

### **Lớp phần mềm cơ bản (BSW)**

. **Các lớp chính trong BSW:**

- . Lớp dịch vụ (Service Layer)
- . Lớp trừu tượng hóa ECU (ECU Abstraction Layer)
- . Lớp trừu tượng hóa vi điều khiển (Microcontroller Abstraction Layer - MCAL)
- . Driver thiết bị phức tạp: Lớp đặc biệt trong Autosar giúp truy cập phần cứng nhanh hơn khi có các tính năng yêu cầu thời gian xử lý nghiêm ngặt.

### **Tóm tắt về kiến trúc lớp**

. **Lớp ứng dụng:**

- . Phần lớn không phụ thuộc vào phần cứng.
- . Giao tiếp giữa các ứng dụng hoặc từ ứng dụng đến lớp phần mềm cơ bản chỉ thông qua RTE.

### . **Lớp phần mềm cơ bản (BSW):**

- . Chia thành ba lớp chính : lớp dịch vụ, lớp trừu tượng hóa ECU và lớp trừu tượng hóa vi điều khiển và các driver phức tạp.

#### . **Lớp trừu tượng hóa vi điều khiển (MCAL):**

- . Lớp thấp nhất của phần mềm cơ bản.
- . Chứa các driver truy cập trực tiếp vào vi điều khiển, các thiết bị ngoại vi nội bộ và các thiết bị gắn ngoài được ánh xạ bộ nhớ.
- . Cung cấp giao diện tiêu chuẩn cho phần mềm cơ bản, quản lý các thiết bị ngoại vi vi điều khiển và cung cấp các giá trị độc lập với vi điều khiển.

#### **Chi tiết các lớp trong MCAL**

- . **IO Drivers:** . Driver cho ADC, PWM, Digital IO, v.v.
- . **Communication Drivers:** . Driver cho SPI, I2C và các hệ thống giao tiếp trong xe như CAN, LIN.
- . **Memory Drivers:** . Driver cho bộ nhớ trên chip và bộ nhớ gắn ngoài như Flash, EEPROM.
- . **Microcontroller Drivers:** . Driver cho các thiết bị ngoại vi nội bộ như watchdog, và các chức năng truy cập trực tiếp vào vi điều khiển.

#### **Lớp trừu tượng hóa ECU**

- . **Abstraction từ các giá trị điện tử của bất kỳ ECU cụ thể nào:** . Giúp tách biệt phần mềm cao hơn khỏi các phụ thuộc phần cứng dưới.
- . **Các phân nhóm trong lớp trừu tượng hóa ECU:** . Trừu tượng hóa phần cứng IO, giao tiếp, bộ nhớ và các thiết bị gắn trên bo mạch.

#### **Lớp dịch vụ**

- . **Các phân nhóm trong lớp dịch vụ:**
- . Dịch vụ giao tiếp, dịch vụ bộ nhớ và dịch vụ hệ thống.

- . Cung cấp giao diện đồng nhất cho giao tiếp mạng xe và quản lý dữ liệu không bay hơi.
- . Cung cấp các dịch vụ cơ bản cho các mô-đun ứng dụng và phần mềm cơ bản.

### **Driver thiết bị phức tạp**

- . **Điều khiển cảm biến và bộ chấp hành phức tạp:** . Đáp ứng các yêu cầu chức năng và thời gian cho các cảm biến và bộ chấp hành phức tạp.

## **Kết luận**

. **Lợi ích của kiến trúc Autosar:** . Tăng tính linh hoạt, khả năng tái sử dụng phần mềm. . Cho phép phát triển phân tán và nhanh chóng ra thị trường. . Cung cấp khả năng cắm và chạy phần mềm từ các nhà cung cấp khác nhau.

. **Tóm tắt:** . Cuối khóa học, chúng ta sẽ xem lại những lợi ích này một lần nữa để củng cố sự hiểu biết.

Chúng ta sẽ tiến hành khóa học ngay bây giờ và dần khám phá cách Autosar đạt được những điều này.

## 6. AUTOSAR INTERFACES

### Giới thiệu

Autosar chia các giao diện thành ba loại khác nhau: giao diện Autosar, giao diện Autosar chuẩn hóa và giao diện chuẩn hóa.

### Các loại giao diện

#### Giao diện Autosar (Autosar Interface)

- . **Giao diện chung:** Được tạo ra từ các cấu hình.
- . **Vai trò:** Được cung cấp bởi RTE và phục vụ như một giao diện giữa phần mềm ứng dụng hoặc giữa phần mềm ứng dụng và phần mềm cơ bản như trừu tượng hóa phần cứng hoặc driver thiết bị phức tạp.
- . **Ví dụ:** Phần mềm ứng dụng có thể đọc các giá trị đầu vào và ghi các giá trị đầu ra thông qua các giao diện này, sử dụng các API để đọc hoặc ghi.

#### Giao diện Autosar chuẩn hóa (Standardized Autosar Interface)

- . **Giao diện đặc biệt:** Là một loại giao diện Autosar đặc biệt được định nghĩa trước bởi tiêu chuẩn.
- . **Vai trò:** Được sử dụng bởi phần mềm ứng dụng để truy cập vào các dịch vụ được cung cấp bởi các module BSW của lớp dịch vụ, chẳng hạn như trình quản lý trạng thái ECU hoặc trình quản lý chẩn đoán.

#### Giao diện chuẩn hóa (Standardized Interface)

- . **Giao diện định nghĩa trước:** Là một giao diện được định nghĩa trước bởi tiêu chuẩn Autosar dưới dạng API trong ngôn ngữ lập trình C.
- . **Vai trò:** Được sử dụng giữa các module BMW trong một ECU hoặc giữa RTE và hệ điều hành, hoặc giữa RTE và module giao tiếp BSW.
- . **Ví dụ:** . Trong lớp MCAL, Autosar định nghĩa một API để đọc một chân IO gọi là `DIO_ReadChannel`. . Các giao diện chuẩn hóa này có thể được truy cập từ các lớp BSW khác để đọc một chân điều khiển.

### Tóm tắt

- . **Ba loại giao diện:**

. **Giao diện Autosar:** Giao diện chung từ các cấu hình, dùng giữa phần mềm ứng dụng và phần mềm cơ bản.

. **Giao diện Autosar chuẩn hóa:** Giao diện đặc biệt định nghĩa trước bởi tiêu chuẩn, dùng để truy cập vào dịch vụ từ các module BSW.

. **Giao diện chuẩn hóa:** Giao diện API định nghĩa trước bởi Autosar, dùng giữa các module trong ECU hoặc giữa RTE và hệ điều hành hoặc module giao tiếp BSW.

. **Tóm lại:** . Hiểu cơ bản về ba loại giao diện mà Autosar quy định. . Sẽ rõ ràng hơn với các ví dụ trực tiếp ở phần cuối khóa học.

Chúng ta sẽ tiếp tục khóa học với những hiểu biết cơ bản về ba loại giao diện này và đi sâu hơn vào chi tiết sau này.

