

37. CONCLUSION AND SUMMARY

Tổng quan về Buổi Demo

- Cấu hình Cảm biến Nhiệt độ:**
 - Cảm biến nhiệt độ được kết nối với kênh ADC của vi điều khiển.
 - Lớp MCAL được cấu hình cho trình điều khiển ADC với các giao diện chuẩn hóa để bắt đầu chuyển đổi ADC và đọc lại kết quả.
- Lớp Trừu tượng ECU:**
 - Runnable ReadTemperature:**
 - Kích hoạt chuyển đổi ADC và đọc lại kết quả.
 - Chuyển đổi kết quả ADC sang nhiệt độ tương đương và truyền dữ liệu này lên lớp ứng dụng thông qua công provider.
- Lớp Ứng dụng:**
 - Chứa một cổng receiver để đọc dữ liệu nhiệt độ thông qua RTE.
 - Dựa trên các tính toán, nó gọi các hàm server trong lớp trừu tượng ECU để điều khiển H-Bridge (start và stop motor).
- Điều khiển H-Bridge:**
 - Hàm Server trong Trừu tượng ECU:**
 - Các hàm start và stop motor điều khiển quạt làm mát.
 - Những hàm này gọi trình điều khiển MCAL để điều khiển các chân IO kết nối với H-Bridge IC.

Các Điểm Chính từ Buổi Demo

- Lớp ứng dụng và lớp trừu tượng ECU tương tác thông qua các API chuẩn hóa RTE.
- Lớp MCAL cung cấp truy cập phần cứng mức thấp, trong khi các lớp trừu tượng xử lý logic cấp cao hơn.
- Tệp ARXML thành phần kết nối các thành phần phần mềm khác nhau bằng cách sử dụng các kết nối assembly.

Lợi ích Thực tế của Autosar

- Thay đổi Vi điều khiển:**
 - Chỉ cần cập nhật lớp MCAL cho vi điều khiển mới.
 - Phần mềm còn lại không bị ảnh hưởng, đơn giản hóa quá trình tái phát triển.
- Thay đổi H-Bridge IC:**
 - Chỉ cần cập nhật logic bên trong các hàm start và stop của lớp trừu tượng ECU.
 - Mã ứng dụng không thay đổi, đảm bảo tác động tối thiểu.
- Ưu điểm của Autosar:**
 - Xử lý Độ phức tạp:** Phần mềm được chia thành các lớp khác nhau, cho phép phát triển song song mà không có sự phụ thuộc lẫn nhau.
 - Plug and Play:** Phần mềm ứng dụng và phần cứng được cách ly thông qua các lớp RTE và BSW, tạo điều kiện thay thế phần cứng dễ dàng.
 - Tái sử dụng:** Các thành phần ứng dụng chung có thể được tái sử dụng trong các dự án khác nhau.

- **Phát triển Song song:** Cho phép các nhà cung cấp phát triển các thành phần phần mềm đồng thời, tăng tốc quá trình phát triển.
- **Đổi mới và Cạnh tranh:** Khuyến khích phát triển các thành phần phần mềm tốt nhất, vì chúng có thể dễ dàng tích hợp và thay thế.

Kết luận

- Phương pháp Autosar cho phép xử lý hiệu quả việc phát triển phần mềm ô tô phức tạp.
- Kiến trúc phân lớp đảm bảo tác động tối thiểu của thay đổi phần cứng đối với phần mềm tổng thể, giảm chi phí tái phát triển.
- Khuyến khích tái sử dụng và phát triển song song, thúc đẩy đổi mới và cạnh tranh trong ngành phần mềm ô tô.

Ghi chú Cuối cùng

- Hãy đăng câu hỏi của bạn trong phần Hỏi & Đáp hoặc yêu cầu các giải thích bổ sung nếu cần.
- Phản hồi của bạn rất quý giá.
- Cảm ơn bạn đã tham gia và chúc bạn thành công trong tương lai với Autosar và phát triển phần mềm ô tô.