

# Implementierung des Lightway IP Stacks auf Embedded Systemen

Martin

October 29, 2024

# Einführung

- ▶ Einführung in die Implementierung des Lightway IP Stacks auf dem ITS-Board
- ▶ Ziel: Demonstration der Integration und Konfiguration eines TCP/IP-Stacks auf einem ressourcenbeschränkten Mikrocontroller
- ▶ Relevanz: Essentiell für Netzwerkanwendungen in Embedded-Systemen

# Lightway IP Stack: Übersicht

- ▶ Leichtgewichtige TCP/IP-Implementierung, ideal für ressourcenbeschränkte Systeme
- ▶ Unterstützt grundlegende Netzwerkprotokolle: TCP, UDP, ICMP, DHCP
- ▶ BSD-Lizenz ermöglicht freie Nutzung und Anpassung

# Modularer Aufbau des Stacks

- ▶ Modulare Architektur für flexible Anpassungen an verschiedene Hardware-Ressourcen
- ▶ Parameter wie Speicherverbrauch und Verarbeitungsgeschwindigkeit konfigurierbar
- ▶ Praktisch für verschiedene Embedded-Umgebungen ohne Betriebssystem

# Konfigurationsdateien im Detail

- ▶ `cc`: Compiler- und Plattform-spezifische Anpassungen
- ▶ `sys_arch`: Systemabhängige Funktionen wie `sys_now()` für Zeitmessungen
- ▶ `lwip_interface`: Stellt das lwIP-Interface zur Applikation bereit
- ▶ `ethernetif`: Bietet Hardware- und Interrupt-Ansteuerung

# Bare-Metal Implementierung

- ▶ Betrieb ohne Betriebssystem (NoSys), direkte Hardware-Interaktion
- ▶ Konfiguration für Bare-Metal: Kein Thread- oder Task-Management notwendig
- ▶ Notwendig: Speichergröße definieren und Speicher effizient allokalieren

# Netzwerkconfiguration und Initialisierung

- ▶ MAC- und IP-Adresse festlegen
- ▶ Beispiel für IP-Konfiguration:

```
#define LWIP_IPADDR  " 192.168.0.1"  
#define LWIP_NETMASK " 255.255.255.0"  
#define LWIP_GATEWAY " 192.168.0.254"
```

# Initialisierung des Netzwerkinterfaces

- ▶ `ethernetif_init()` Funktion für Hardware-Initialisierung
- ▶ Zuweisung der MAC-Adresse und Start der Hardware-Komponenten
- ▶ Konfiguration von Interrupt-Prioritäten für eine effiziente Kommunikation



# Verbindung mit dem Netzwerkinterface

- ▶ Interface hinzufügen: `netif_add()` und `netif_set_default()`
- ▶ Konfiguration des Netzwerkinterfaces:

```
netif_add(&lwip_netif, &ipaddr, &netmask, &gateway,  
netif_set_default(&lwip_netif);
```

# Interrupt Handling für Ethernet

- ▶ `ethernetif_input()` und `ethernetif_init()` Routinen zur Verarbeitung von Paketen
- ▶ Low-Level-Initialisierung kopiert Datenpakete in eine P-Buff-Struktur zur Weiterverarbeitung
- ▶ Wichtig: Optimierte Nutzung von Ressourcen durch effizientes Interrupt-Management

# Ping-Test und Netzwerkverbindung überprüfen

- ▶ Testmethoden wie Ping, um die Stabilität und Reaktionszeit des Netzwerks zu validieren
- ▶ Regelmäßige Überprüfung des Netzwerkstatus mit `sys_check_timeouts()` und anderen Timer-Funktionen

# Zusammenfassung und Ausblick

- ▶ Erfolgreiche Implementierung des Lightway IP Stacks auf dem ITS-Board
- ▶ Effiziente Bare-Metal-Konfiguration ermöglicht ressourcenschonende Netzwerkkommunikation
- ▶ Weitere Optimierungen und detaillierte Anpassungen an die Hardware für den praktischen Einsatz