

ステップ5…無線LAN接続

中林 智之

リスト1 無線LANアクセス・ポイントに接続するsrc/wifi.rs

```

use std::{net::Ipv4Addr, time::Duration};

use anyhow::bail;
use esp_idf_hal::peripheral;
use esp_idf_svc::netif::{EspNetif, EspNetifWait};
use esp_idf_svc::{eventloop::*, wifi::*};

use embedded_svc::wifi::*;

use anyhow::Result;
use log::*;

/// Connect to a Wi-Fi access point.
/// Returns initialized Wi-Fi stack on success.
///
/// - `ssid`: Your Wi-Fi name
/// - `pass`: Your Wi-Fi password
///
/// Note that ESP32-C3 does not support the 5GHz
band, please use a WiFi with active 2.4GHz band.
pub fn connect<'a>(
    modem: impl peripheral::Peripheral<
        P = esp_idf_hal::modem::Modem,
        > + 'static,
    ssid: &'a str,
    pass: &'a str,
) -> Result<Box<EspWifi<'a>>> {
    // 1. Initialize the Wi-Fi stack ← ①
    let sysloop = EspSystemEventLoop::take()?;
    let mut wifi = Box::new(EspWifi::new(
        modem,
        sysloop.clone(),
        None,
    )?);

    info!("Wifi created, about to scan");

    // 2. Scan our Wi-Fi access point ← ②
    let ap_infos = wifi.scan()?;

    // 3. If our access point found, configure
the channel ← ③
    let ours =
        ap_infos.into_iter().find(|a| a.ssid == ssid);
    let channel = if let Some(ours) = ours {
        Some(ours.channel)
    } else {

        None
    };

    // 4. Configure the Wi-Fi stack as a Wi-Fi station ← ④
    wifi.set_configuration(&Configuration::Client(
        ClientConfiguration {
            ssid: ssid.into(),
            password: pass.into(),
            channel,
            ..Default::default()
        },
    )?);

    // 5. Wait until Wi-Fi stack is operating ← ⑤
    wifi.start()?;
    if !WifiWait::new(&sysloop)?
        .wait_with_timeout(Duration::from_secs(20), || {
            wifi.is_started().unwrap()
        }) {
        bail!("Wifi did not start");
    }

    // 6. Connect to our access point and wait until
network interface get an IP address ← ⑥
    wifi.connect()?;
    if !EspNetifWait::new::<EspNetif>(
        wifi.sta_netif(),
        &sysloop,
    )?
        .wait_with_timeout(
            Duration::from_secs(20),
            || {
                wifi.is_connected().unwrap()
                && wifi.sta_netif().get_ip_info().unwrap().ip
                != Ipv4Addr::new(0, 0, 0, 0)
            },
        ) {
        bail!("Wifi did not connect or did not receive a
            DHCP lease");
    }

    info!("Wifi connected");

    Ok(wifi)
}

```

本章ではESP32の無線LANを有効化し、std::netを使ってネットワーク通信するRustのコードを紹介します。

ビルド用のソースコードはGitHubのstd-netディレクトリ下にあります。また、本誌に載せきれなかったリストA～リストH、図A～図Dは本誌ウェブ・ページに掲載しています。

<https://interface.cqpub.co.jp/2305rust2/>

<筆者のGitHub>

<https://github.com/tomoyuki-nakabayashi/interface202305-c3-std-rust>