# Troubleshooting Guide

## Common Issues and Solutions

### 1. “fatal error: zephyr_image_info.h: No such file or directory”

**Cause**: This error occurs when building single-core applications that try to include sysbuild-generated headers.

**Solution**: The project now automatically detects build mode and conditionally includes these headers:
- Single-core builds: Headers are NOT included
- Multi-core (sysbuild) builds: Headers are included

**Verification**: Check that your source files have:

```
#ifdef CONFIG_OPENAMP_SYSBUILD
#include <zephyr_image_info.h>
#endif
```

### 2. Rust Compilation Errors

**Cause**: Missing Rust target or incorrect Rust configuration.

**Solution**:

```
# Install the required target
rustup target add thumbv7em-none-eabihf

# Verify installation
rustup target list --installed | grep thumbv7em-none-eabihf
```

### 3. Build Mode Detection Issues

**Cause**: CMakeLists.txt not properly detecting sysbuild mode.

**Solution**: The detection logic checks for:

```
if(DEFINED SYSBUILD_PROJECT OR DEFINED SB_CONFIG_REMOTE_BOARD)
    set(CONFIG_OPENAMP_SYSBUILD ON)
else()
    set(CONFIG_OPENAMP_SYSBUILD OFF)
endif()
```

### 4. IPC Service Not Ready

**Cause**: Device tree configuration or timing issues.

**Solution**:
- Ensure proper device tree configuration for your board
- Add delays in initialization if needed
- Check that both cores are properly configured for IPC

### 5. Memory Configuration Issues

**Cause**: Insufficient memory allocation for OpenAMP.

**Solution**: Adjust memory settings in `prj.conf`:

```
CONFIG_IPC_SERVICE_STATIC_VRINGS_MEM_SIZE=16384
CONFIG_HEAP_MEM_POOL_SIZE=16384
CONFIG_MAIN_STACK_SIZE=2048
```

## Build Command Reference

### Single Core Builds

```
# CM7 only
west build -b mimxrt1160_evk/mimxrt1166/cm7 samples/open_amp_rust/cm7 -p always

# CM4 only
west build -b mimxrt1160_evk/mimxrt1166/cm4 samples/open_amp_rust/cm4 -p always
```

### Multi-Core Build

```
# Both cores with sysbuild
west build -b mimxrt1160_evk/mimxrt1166/cm7 --sysbuild samples/open_amp_rust/cm7 -p al-
ways
```

## Debugging Tips

1. **Check build logs**: Look for Rust compilation messages and CMake configuration output
2. **Verify Kconfig**: Use `west build -t menuconfig` to check configuration options
3. **Test incrementally**: Start with single-core builds before attempting multi-core
4. **Check device tree**: Ensure IPC nodes are properly configured for your board

## Environment Setup

### Required Tools

- Zephyr SDK
- West build tool
- Rust toolchain with embedded target
- CMake 3.20+

### Environment Variables

```
export ZEPHYR_BASE=/path/to/zephyr
export ZEPHYR_SDK_INSTALL_DIR=/path/to/zephyr-sdk
```

# Advanced Configuration

## Custom Rust Dependencies

Edit `rust_lib/Cargo.toml` to add dependencies:

```
[dependencies]
cortex-m = "0.7"
nb = "1.0"
```

## Memory Layout Customization

Modify linker scripts or device tree overlays as needed for your specific memory requirements.

## Performance Tuning

- Adjust Rust optimization level in `Cargo.toml`
- Tune IPC buffer sizes in `prj.conf`
- Configure appropriate log levels for production builds