

RDK-B

- RDK-B Emulator
 - RDK-B Emulator Build Guide
 - RDK-B Emulator Yocto 2.2 - Morty
 - RDK-B Emulator Build Instructions
 - Host Setup
 - Building
 - RDK-B Emulator Yocto 3.1 - Dunfell
 - RDK-B Emulator Dunfell Build Instructions
 - Host Setup
 - Building
 - RDKB Emulator: Morty to Dunfell migration
 - Error handling
 - RDK-B Emulator User Guide
 - RDK-B Emulator Users Guide
 - Document Status
 - Table of Contents
 - Introduction
 - Purpose of this Document
 - Abbreviations and Acronyms
 - Overview of RDK Emulator
 - Supported Build Types
 - Build Setup Instructions
 - Setting up the Host Environment
 - Downloading Source Code & Building
 - Bringing up the Emulator on Virtual Box
 - STEP 1: Create your new VM Instance:
 - STEP 2: Configure your new VM Instance:
 - STEP 3: Run Emulator in Vbox :
 - STEP 4: Lanching Emulator in WebUI :
 - Launching WebUI App
 - RDK-B Environment Setup
- RDK-B Raspberry Pi
 - RDK-B R-Pi Build guide
 - RDK-B R-Pi Yocto 2.2 - Morty
 - RDK-B (Raspberry Pi 3 B+) Build and Setup Manual - Router Profile
 - Introduction
 - Build Instructions and Flashing Procedure
 - Hardware and Setup Information
 - RDK-B Bootstrap Sequence
 - Router Profile Test Bed using RPI
 - Router Functionality Use Case - Internet connectivity for Wired and Wireless Client(s)
 - Lists of Supported Features
 - Supported Features User manual
 - Additional Links
 - Yocto Modification for RDK B+ Support
 - RDK-B Raspberypri - Host SetUp and Build Instructions
 - Host Setup
 - Yocto Build Steps
 - Build Steps
 - Flashing Procedure
 - RDKB RPI Setup Information
 - Required Hardware
 - USB to Serial Debug converter Link and set up
 - Dongles Information Links
 - RDK-B R-Pi Yocto 3.1 - Dunfell
 - RDKB: Extensible SDK support on Raspberypri Yocto 3.1 dunfell build
 - Yocto Build
 - Host Machine Setup
 - Yocto workspace setup
 - eSDK Installation
 - Host Configuration
 - Installation
 - Build from eSDK
 - Environment Setup
 - New Components
 - Add Component
 - Modify Component
 - Edit Component's recipe
 - Build Components
 - Deploy binaries
 - Build Image
 - Known Issues
 - Reference ticketJIRA11deff04-0380-3a3d-a916-0849d4e573f7REFPLTB-2916
- RDK-B (Raspberry Pi) Build and Setup Manual - Router Profile
 - Introduction
 - Build Instructions and Flashing Procedure

- Hardware and Setup Information
 - Router Profile Test Bed using RPI
 - Router Functionality Use Case - Internet connectivity for Wired and Wireless Client(s)
 - Lists of Supported Features
 - Supported Features User manual
 - Additional Links
 - RPI 3B/3B+ Model Reference Platform
 - Introduction
 - Host Setup
 - Repo Setup
 - Yocto Build Steps
 - Flashing Procedure
 - Validated Functionalities
 - RPI 4B Model Reference Platform
 - Host Setup
 - Repo Setup
 - Yocto Build Steps
 - Flashing Procedure
 - Validated Functionalities
 - RPI 4B Model Reference Platform: Extender build instructions
 - Host Setup
 - Building
 - Hardwares
 - RPI 4B Model Reference Platform with 64bit Arch
 - Host Setup
 - Repo Setup
 - Yocto Build Steps
 - Validated Functionalities
 - OEM Side Changes Required
 - For DBUS Enabling support
 - System Testing
 - E2E Validation Test Results
 - Performance comparison metrics between RPi4 64 bit Vs 32 bit
 - RDK-B R-Pi Yocto 4.0 - Kirkstone
 - Kirkstone RDK-B (Raspberry Pi 4) Build and Setup Manual - Router Profile
 - kirkstone RDK-B RPi4 32 bit
 - Introduction
 - Host Setup
 - Repo Setup
 - Yocto Build Steps
 - Flashing Procedure
 - Validated Functionalities
 - Code Commits
 - Kirkstone (Yocto 4.0) - Epic Story:
 - Jira Tickets :
 - Technical support:
 - Kirkstone RDK-B RPi4 64 bit (Both User Space & Kernel Space)
 - Introduction
 - Host Setup
 - Repo Setup
 - Yocto Build Steps
 - Validated Functionalities
 - Code Commits
 - Kirkstone (Yocto 4.0) - Epic Story:
 - Jira Tickets :
 - Technical support:
 - RDKB: Extensible SDK support on Raspberrypi Yocto 4.0 Kirkstone build
 - Yocto Build
 - Host Machine Setup
 - Yocto workspace setup
 - eSDK Installation
 - Host Configuration
 - Installation
 - Build from eSDK
 - Environment Setup
 - New Components
 - Add Component
 - Modify Component
 - Edit Component's recipe
 - Build Components
 - Deploy binaries
 - Build Image
 - Known Issues
 - Reference ticketJIRA11deff04-0380-3a3d-a916-0849d4e573f7REFPLTB-2916
- RDK-B Boot time data plot
 - April_2019_RPI_BootTimeData
 - 2019-04-30_RPI
 - December_2018_RPI_BootTimeData

- 2018-12-28_RPI
- January_2019_RPI_BootTimeData
 - 2019-01-31_RPI
- July_2019_RPI_BootTimeData
 - 2019-07-31_RPI
- June_2019_RPI_BootTimeData
 - 2019-06-27_RPI
- June_2019_RPI_Performance_Analysis_On_BootTimeData
 - 2019-06-14_RPI_InputData_Scripts_Analysis
- March_2019_RPI_BootTimeData
 - 2019-03-01_RPI
 - 2019-03-31_RPI
- May_2019_RPI_BootTimeData
 - 2019-05-16_RPI
 - 2019-05-24_RPI
 - 2019-05-24_RPI_LXC_Bootdata
- May_2019_RPI_Performance_Analysis_On_BootTimeData
 - 2019-05-21_RPI_InputData_Scripts_Analysis
 - 2019-05-29_RPI_InputData_Scripts_Analysis
- October_2018_RPI_BootTimeData
 - 2018-10-01_RPI
 - 2018-10-02_RPI
 - 2018-10-03_RPI
- September_2018_RPI_BootTimeData
 - 2018-09-28_RPI
- Features in RDK-B Reference Platform
 - List of Supported Features in RDKB Reference Platform
- Kirkstone (Yocto 4.0) Migration - RDK-B
 - Benefits of Kirkstone:
 - RPI4 Kirkstone build instructions
 - RPI4 32-bit build
 - RPI4 64-bit build
 - Additional information for building a project:
 - Sanity Test cases :
 - Code Commits
 - Kirkstone (Yocto 4.0) - Epic Story:
 - Jira Tickets :
 - Technical support:
- OneWifi Integration in RDK-B Raspberry Pi4
 - Scope
 - Build Instructions
 - RPI4 32bit
 - RPI4 64bit
 - Supported Wi-Fi Chipsets
 - Feature Enhancements
 - Test Results:
 - Supported Dongles
 - References
 - EPIC/User Stories
 - Porting guide:
 - Contact us
 - OneWifi Sanity Test Cases
 - Scope
 - Test Cases
 - Porting Guide - OneWifi
 - Scope:
 - Target Audience:
 - Prerequisites:
 - Supported platforms:
 - Supported yocto version:
 - High level architecture:
 - OneWiFi Thread Management, Inter Thread Communication and Data Handling
 - Core Functional Blocks/Subsystem
 - Core Thread Software Architecture
 - Operational Modes
 - Router Mode
 - Extender Mode
 - Flow Diagram/Pseudo Code
 - Message Sequence Diagrams
 - Epic details:
 - Approach followed:
 - Step by step procedure:
 - Systemd Service file:
 - Create platform layer for mediatek platform, please consider raspberry pi as reference
 - Manifest entries:
 - Layers to consider:
 - Bug fixing or feature enhancement:
 - Flags defined in Onewifi:
 - Bulk atomic HAL apis for common configuration

- Stats implementation:
 - wifi database:
 - 64 bit build support:
 - Difference Between CcspWifiAgent and OneWiFi Apply settings
 - CcspWifiAgent:
 - Onewifi
 - Debugging tips:
 - Wi-Fi 7 segment:
- Speedtest Integration in RDK-B
 - Scope:
 - Component used:
 - Ticket details:
 - Testing :
- DAC demonstration on Broadband using Dobby & Sample App in Local WebUI:
 - Introduction:
 - Architecture:
 - UI Low level Design:
 - Integration in RPI3:
 - Sample Button on local WebUI:
 - Demo video:
 - References:
 - Sample Button on RDKB Local WebUI for DAC Support
 - Introduction
 - Build Procedure
 - Create an OCI image for a sample application using meta-dac-sdk(Ex: helloworld-test)
 - Source Code Creation
 - Demo DAC > Sample DAC
 - DAC
 - Testing Procedure
 - Troubleshoot
 - Demo Video
 -
- Avahi mDNS support in RDK-B Reference Platform
 - Scope:
 - End-End flow:
 - Build instructions:
 - Code changes:
 - Recipe path:
 - Additional configurations:
 - Jira Ticket:
 - Testing:
 - Host Side Tools on Mac Book:
 - References & RPI4 32bit image:
 - Logs:
- RPI4 Extender - reference implementation
 - Introduction - RDK Extender
 - Scope of the work
 - High level Design / Architecture
 - RPI4 Extender build(32-bit) instructions:
 - OpenSync version details:
 - Sanity Test cases :
 - Known issues/Observations:
 - References :
 - User Stories/EPIC
 - Technical support
- RDK-B CPUProcAnalyzer
 - Introduction
 - Scope of the work
 - High level Design / Architecture
 - Sanity Testing
 - 1. How to Configure:
 - 2. How to view results:
 - 3. Visualizer:
 - References
 - EPIC/User Stories
- OpenSync 4.4 with OneWiFi
 - Introduction
 - Scope of the work
 - Target Audience
 - High level Design / Architecture
 - Target Build Variant
 - Identified task list
 - Check point
 - Dependencies
 - Validation
 - Risks
 - Flow Diagram/Pseudo Code
 - Sequence Diagrams
 - Build Instructions

- Supported device
 - Use Cases
 - Sanity Testing
 - References
 - EPIC/User Stories
- RdkCellularManager-MM
 - Introduction
 - Scope of the work
 - High level Design / Architecture
 - Use Cases
 - Sanity Testing
 - References
 - EPIC/User Stories
- RDK-B Router Profile
 - Target audience
 - Objective
 - Target platforms that support router profile
 - High Level Architecture
 - Build instructions
 - Open Reference platform
 - Raspberry Pi 4
 - SoC Reference platform
 - Mediatek Wi-Fi 6 device
 - Features supported
 - Test Bed
 - User Manuals
 - References
- Telemetry 2.0 support for RDKB RPI -User Manual
 - Introduction
 - RDKB Telemetry Components
 - Environment Setup
 - 2.1. Build procedure
 - 2.2. Pre- requisites for enabling Telemetry2_0 (Version : 2) - Single profile
 - Xconf server
 - Change Application to stb in top right corner of the window
 - Create Formula
 - Create Device Settings
 - Create Upload Repository
 - Create LogUpload Settings
 - Test Page
 - Create Permanent Profiles
 - Create Targeting Rules
 - Test Page
 - JSON Response
 - RAW data Response
 - RAW Data Response of T1
 - RPI(target) devices Response's
 - DCMresponse Text Data
 - RPI device Logs
 - JSON Report
 - Configuration file
 - Log File
 - Log upload onto HTTPS Server
 - JSON file upload
 - Telemetry Version : 2.0.1 - Multiprofile
 - Pre- requisites for enabling Telemetry2_0
 - Telemetry 2.0 Report Profiles
 - T2 DataModel
 - DMCLI SET Command for T2 Report Profile
 - Dmcli Get
 - RPI Device Logs
 - cJSON Report
 - Log Upload
 - Telemetry 2.0 ProfileMsgPack
 - Sample JSON Profile
 - Convert JSON into base-64
 - dmcli set
 - dmcli Get
 - profiles.msgpack
 - Telemetry 2 Log data
 - cJSON Report
 - Log upload
 - Report Profiles Validation through WebPA Commands
 - ReportProfilesMsgPack
 - Webpa Set
 - Webpa Get
 - Device Logs
 - Limitations
 - References

- **RDK-B Boot Time Data Measurement**
 - **Boot Time**
 - **Build Procedure – To enable systemd boot measurement tools**
 - **Generating boot-chart in rpi4**
 - **systemd-analyze**
 - **systemd-bootchart**
- **Tiny RDKB support for RPI Reference Platform**
 - **Introduction**
 - **Design**
 - **Comparison of Image Size**
 - **RDKB RPI Memory Comparison**
 - **Initial Boot-up**
 - **RAM Usage On Initial Boot-up**
 - **CPU Utilization On Initial Boot-up**
 - **Rootfs Size on Storage Disk On Initial Boot-up**
 - **After one hour from Image Boot-up**
 - **RAM Usage On After One Hour**
 - **CPU Utilization On After One hour**
 - **Rootfs Size on Storage Disk On After One hour**
 - **Linux Commands for Memory Usage**
 - **Initial Boot-up - Sample Output from fresh image**
 - **One Hour after from Image Boot-up - Sample Output**
 - **Limitation**
 - **References**
 - **Jira**
 - **Gerrit Links**
- **TR-069 Support for RDKB RPI Reference Platform**
 - **Introduction**
 - **GenieACS Server and Client Rpi communication**
 - **GenieACS Server**
 - **Test Procedure**
 - **Client RPI set-up**
 - **GenieACS Server Login & Initial check**
 - **Limitations**
- **TR-069 Get/Set Data Parameters Support through GenieACS Server for RDKB RPI Reference Platform**
 - **Basic GenieACS Server and RPI communication**
 - **GetParameter Values/Names**
 - **SetParameter Values/Names**
 - **Boolean Value**
 - **String Names/values**
 - **Limitations**
- **FwUpgradeManager Support in RDKB RPI - Broadband - User Manual - 2021 - M8**
 - **Introduction**
 - **Build procedure to generate R-Pi image with Firmware upgrade**
 - **Repo Steps**
 - **Image Generation Steps**
 - **Overall Architecture**
 - **TR-181 Data Model Parameters**
 - **Pre-requisites - I**
 - **Flashing the image**
 - **Command to flash the image**
 - **Increasing the rootfs partition size in SD-CARD if Flashed for first time**
 - **Local Apache2 server set up**
 - **Server Set-up**
 - **HTTP Location folder**
 - **Pre-requisites - II**
 - **Execution Steps**
 - **Using dmcli commands**
 - **Using Webpa**
 - **prerequisite**
 - **Curl Commands**
 - **WebPA Set**
 - **WebPA Get**
 - **TroubleShooting**
 - **Limitation**
- **Self Heal Support in RPI - Design - 2019 M10**
 - **Introduction**
 - **Design Considerations**
 - **Resource Monitoring**
 - **Process Monitoring**
 - **Connectivity Test**
 - **SelfHeal Logs**
 - **Architecture**
 - **Self Heal DataModel Flow**
 - **Process Monitor Flow**
 - **Data Model**
 - **Lists of self heal supported DataModel commands**
 - **Limitations**
- **Backup and Restore Feature in RPI - Design**

- Introduction
 - Design Considerations
 - Backup Settings
 - Restore Settings
 - Architecture
 - Limitations
 - Future Enhancements
 - Backup Feature
 - Restore Feature
- RDKB CcspLogAgent Component in RPI - Design - 2019 M7
 - Introduction
 - How to add rdklogger to a new component
- Log Rotation Support in RPI User manual - Broadband - 2019 M8
 - LogRotate :
 - Limitation
- Band Steering User manual(With CcspWifiAgent) - RDKB
 - Introduction
 - Environment Setup
 - Executing System
 - Troubleshooting
 - Error Messages
 - Special Considerations
 - Support
- RDKB BandSteering(With CcspWifiAgent) - Design Approach
 - Introduction
 - Design Considerations
 - Decision Overview
 - WiFi HAL Changes
 - wifi_setBandSteeringEnable
 - wifi_getBandSteeringEnable
 - wifi_getBandSteeringCapability
 - Other Component Changes
 - Band Steering Use cases
 - Post-association Band Steering using RSSI Threshold
 - Description
 - Approach
 - Sequence diagram
 - Band Steering using Utilization Threshold (To be updated further)
 - Description
 - Approach
 - Sequence Diagram
 - implementation
 - Architecture
 - Implementation of Band Steering using RSSI Threshold:
- Data Model
 - Limitations
 - Future Enhancements
- Backup and Restore Feature in RPI - User Manual
 - Introduction
 - Environment Setup
 - Set-up Considerations
 - System Work Flow
 - Exiting the System
 - Executing System
 - Troubleshooting
 - Error Messages
 - Special Considerations
 - Support
- Self Heal Support in RPI - User manual - Broadband
 - Introduction
 - Environment Setup
 - Executing System
 - Resource Monitor - Monitors CPU and MEMORY
 - Process Monitor - Monitors the Process Periodically based on Process id's
 - Connectivity Test - Ping Functionality
 - Troubleshooting
- Telemetry(1.0) in RPI User manual Broadband - 2019 M9
 - Telemetry Introduction:
 - RDKB Telemetry Components:
 - RDKB Telemetry Architecture:
 - XConf Server Configuration procedure:
 - RPi device side configuration and testing process:
 - Future Enhancement:
- Telemetry(1.0) Support in RPI - Design - 2019 M9
- RDKB RPI Firmware Upgrade - Design - 2019 M7
 - Introduction
 - Limitations
 - Design approach
 - Sequence Diagram

- SNMP - RDKB RPI - User Manual - 2020 - M7
 - Introduction
 - Environment Setup
 - Systemd Status
 - Netstat Command
 - Execution Commands
 - Sample Snmpget Command
 - Sample SnmpSet Command
 - SnmpWalk Command
 - Troubleshooting
 - snmp Logs
 - Location of SNMP MIB and CONF files
- Integration of SNMP - RDKB RPI - Design - 2020 M7
 - Introduction
 - Design Considerations
 - Architecture
- Advanced config- Remote Management with HTTPS protocol
 - Introduction
 - TR-181 Data Model Parameter of Remote Management
 - Architecture Design
 - Set Parameter Work Flow
 - Sequence Diagram
- Telemetry 2.0 support in RDKB RPI
 - Introduction
 - Architecture Overview
 - Telemetry 2.0
 - Overview of Instrumenting RDKB components with T2 shared library (commonlib) APIs:
 - T2.0 Common Library
 - TR-181 DataModel
 - Types of Markers
 - T2 Report Profiles
 - Profiles Set Properties
 - profiles
 - Profile
 - name
 - versionHash
 - value
 - Description
 - Version
 - Protocol
 - EncodingType
 - ReportingInterval
 - ActivationTimeOut
 - TimeReference
 - GenerateNow
 - Parameter
 - HTTP
 - JSONEncoding
 - Examples
 - Example 1
 - Example 2
 - Example 3
 - Example 4
 - T2 ReportProfilesMsgPack
- CcspWiFiAgent support
 - Introduction
 - Branch
 - Build Instructions
 - RPI4 32bit
 - RPI4 64bit
 - Supported devices
 - Sanity Testing
 - 250References
 - EPIC/User Stories
- Port Triggering in RPI
 - Introduction
 - Branch
 - Build Instructions
 - RPI4 32bit/RPI4 64bit Dunfell Build.
 - RPI4 32bit/RPI4 64bit Kirkstone Build.
 - Procedure for Testing
 - Iperf
 - Supported devices
 - Sanity Testing 250250250250
 - References
 - EPIC/User Stories
- WebPA Client Support on RPI and Emulator
 - Parodus-WebPA :
 - WebPA :

- Parodus :
 - Parodus-WebPA Structure :
 - How Communication Happens :
 - Nanomsg Registration (provided by libparodus):
 - Service Keep Alive Message (provided by libparodus):
 - Upstream Communication:
 - Parodus2ccsp:
 - GET Parameter Sequence Flow of WebPA :
 - SET Parameter Sequence Flow of WebPA :
- BridgeMode Feature support in RPI
 - Scope of the work
 - Target Audience
 - High level Code Flow Design
 - After boot-up
 - During boot-up
 - Supported device
 - Build Instructions
 - TR-181 Data Model Parameter of Bridge Mode
 - Test Procedure
 - Enabling BridgeMode via dmcli
 - Enabling BridgeMode via WebUI
 - RPI Test Results
 - Router Bridge-Static
 - Bridge-Static Router
 - References
 - Ticket details
 - Code review links
- WebConfig Feature Validation in RPI
 - Scope
 - Target Audience
 - High level Code Flow Design
 - WebConfig Client Arch Design
 - RDK Telemetry Overview via webconfig
 - Supported device
 - Build Instructions
 - TR-181 Data Model Parameters
 - Webconfig Settings
 - Test Procedure and Results
 - OneWiFi validation
 - For privatessid subdoc
 - PAM Validation
 - For Lan subdoc
 - For DMZ subdoc
 - For PortMapping subdoc
 - Telemetry Multiprofiles Validation
 - References
- WebPA Client Support on RPI and Emulator - User Manual
 - Curl Commands :
 - GET Command Example: Get Request in Console to get RadioNumberofEntries
 - SET Command Example : Set Request in Console to set Wifi SSID for Privatewifi 2.4Ghz
 - Sample Success & Failure Parameters Response in WebPA Server
- RDK-B Turris
 - Turris Omnia RDK-B Gateway - Build Guide
 - Turris Omnia RDK-B Gateway - Yocto 3.1(Dunfell)
 - Introduction
 - Environment Setup
 - Host Environment
 - Build instructions for creating rdk-generic-broadband-image
 - Errors and Challenges
 - WebPA Support
 - Firmware Upgrade
 - OpenSync Verification, with Plume NOC and MeshAgent Support
 - Components Involved:
 - Version and branches of Components
 - Set-up Considerations
 - User Access Considerations
 - Limitations
 - Layers and its change-sets
 - Challenges/Issues observed
 - Features Validated
 - Known Issues
 - Challenges/Issues observed
 - Known Issues
 - Features Supported
 - Flashing procedure
 - Turris Omnia Reference Platform
 - Introduction
 - Environment Setup
 - Host Environment

- Build instructions for creating rdk-generic-broadband-image
 - Errors and Challenges
 - WebPA Support
 - Firmware Upgrade
 - OpenSync Verification, with Plume NOC and MeshAgent Support
 - Components Involved:
 - Version and branches of Components
 - Set-up Considerations
 - User Access Considerations
 - Limitations
 - Yocto 3.1 migration from topic dunfell for meta-turris and Opensync 2.0.5 build support
 - Build instruction
 - Layers and its change-sets
 - Migration and Validation from dunfell to rdk-next
 - Build Instruction
 - Challenges/Issues observed
 - Features Validated
 - Known Issues
 - Validation of RDKB features of dunfell Image with opensync
 - Build Instruction
 - Challenges/Issues observed
 - Set-up Considerations
 - Known Issues
 - Features Supported
 - Flashing procedure
- Turris Omnia RDK-B Gateway - Yocto 2.2(Morty)
 - Host Setup
 - Building
 - Flashing procedure
- Turris Omnia RDK-B Extender - Build Guide
 - RDKB Extender Yocto 2.2 - Morty
 - Host Setup
 - Building
 - Flashing procedure
 - RDKB Extender Yocto 3.1 - Dunfell
 - Host Setup
 - Building
 - Flashing procedure
- Turris Omnia Reference Platform: Flashing Instruction
 - Turris Omnia 2019 & 2020: Flashing Instruction
 - Hardware Information
 - Requirements
 - Flashing RDKB image (A Yocto Project based Distro)
 - Flashing with Medkit & Sysupgrade images
 - Creating additional partitions
 - RDK Firmware(Image) upgrade: (Yet to validate)
 - Approach 1:
 - Approach 2(Quick):
 - Turris Omnia RTROM01-2G & RTROM01-CE: Flashing Instruction
 - Hardware Information
 - Serial Port Access
 - Setting up Bootloader:
 - Setting default U-boot environment
 - Flashing
 - Flashing RDKB image (A Yocto Project based Distro)
 - RDK Firmware(Image) upgrade:
 - Approach 1:
 - Approach 2(Quick):
 - Approach 3:
 - Fallback to OpenWrt OS(Failsafe):
- Turris-Omnia Gateway Supported Features
 - Supported RDKB Features & Enhancements
 - Supported WiFi HAL APIs
- RDK Wi-Fi Extender with Turris Omnia
 - Status Reports
 - Meeting Notes
 - JIRA Ticket
 - Releases
 - IEEE 1905 Component
 - 1.Introduction
 - 1.1Abstraction Layer
 - 1.2High Layer Entity
 - The AL entity is executed like this:
 - The HLE entity is executed like this:
 - 2.Source code organization:
 - 3.Execution:
 - Wi-Fi Extender Features
 - WiFi Extender with Turris Omnia Reference Platform - Cloud Access Guide
 - Cloud Server Network Operation Center(NOC) access

- Account creation
 - Addition of devices in Global Inventory
 - Claim devices in your location
 - Network Management
 - OpenSync Backhaul credential
 - Plume's point of contact
 - Supported Features in Turris-Omnia Extender
 - List of Supported Features in Turris-Omnia Extender
 - TurrisOmnia Release builds
 - Building Turris GateWay image with musl C library
 - Errors and Challenges
- RDK-B Extenders
 - RDK-B OpenSync
 - OpenSync activities with RDKB
 - OpenSync supported platforms
 - OpenSync cloud Requirements
 - OpenSync connection establishment with cloud
 - Plume NOC
 - Current plans
 - RDKB list of committed and delivered tickets
 - RDK-REFPLT-2022-W35
 - RDK-REFPLT-2022-W37 (still drafting)
- Enabling valgrind tool for memory leak
 - Valgrind
 - Build steps
 - Installation
 - Execution
 - Defining the options:
- Enabling Gperf tool for memory leak
 - Gperf-tool
 - Build steps
 - Installation:
 - Execution
 - Usage:
 - Errors during execution
 - Limitations
- GDB for crash analysis
 - GDB
 - Build steps
 - GDB Installation
 - Execution
- OpenSync integration in RPI GW
 - Design considerations on VAP alignment for RPI WiFi HAL for OpenSync integration
 - Example Network Topology
 - Prerequisites
 - Design consideration for RPI target