

CAE專家教室 T1S1001 課程
10BASE-T1S Ethernet
SPE (Single Pair Ethernet)
T1S1001 v1.00



A Leading Provider of Smart, Connected and Secure Embedded Control Solutions



SMART | CONNECTED | SECURE

CAE Taiwan
Microchip Technology Inc.

May 21, 2026

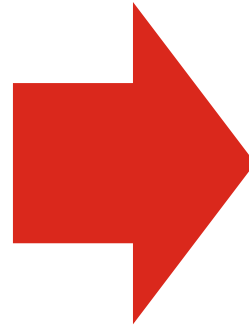
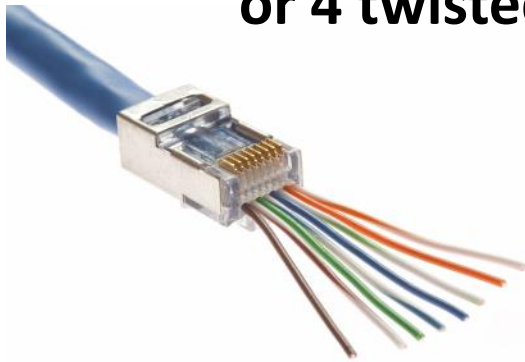
CAE Expert Classroom : T1S1001 Index

- Single Pair Ethernet (SPE) Introduction
 - Single Pair Technology
 - Introduction to T1S
 - Introduction to OPEN Alliance
 - Introduction to T1S Application
 - SPE Product Details
- PIC32CX-SG EVB APP055 v1.10 Introduction
- MCC Harmony Framework
- Start From New Project
- MPLAB® Code Configurator (MCC)
- TCP/IP Configurator
- Added TCP/IP Dependency, System Driver & Peripheral Library
- MCC-Configure the Added Modules
- MCC-Project Properties & Application
- EVB-LAN8670-USB Setup - 10BASE-T1S Ethernet Host

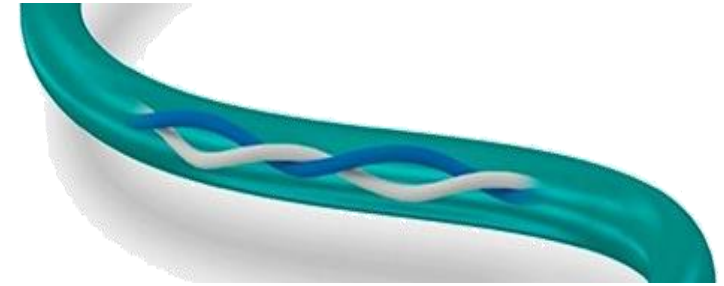
Single Pair Technology

What is Single Pair Ethernet (SPE)?

- Traditional ethernet uses 2 or 4 twisted pairs



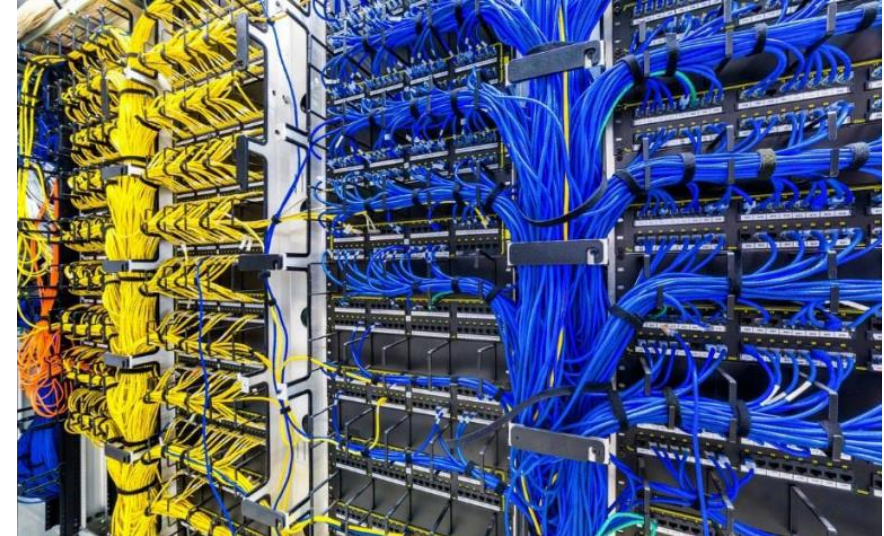
- SPE = Single Pair Ethernet
- T1 = 1-Twisted Pair



- Just a new ethernet transceiver (PHY)
- Network layers 2-7 remain the same
- Independent of speed; 10BASE-T1, 100BASE-T1, NGBASE-T1...

Why Single Pair Ethernet?

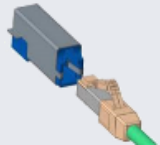
- Single Pair Ethernet reduces system cost, weight and wiring complexity
- **IEEE 802.3cg - 10BASE-T1S and 10BASE-T1L**
 - 10 Mbit/s connection over a single two-conductor cable
 - 10BASE-T1S: Short Reach (25m), multi-drop bus line or point-to-point connection
 - 10BASE-T1L: Long Reach (1 km), point-to-point connection
- **IEEE 802.3bw - 100BASE-T1**
 - 100 Mbit/s over single pair
- **IEEE 802.3bp - 1000BASE-T1**
 - 1000 Mbit/s over single pair cable
- **IEEE 802.3ch – NGBASE-T1**
 - 2.5Gbit/s, 5Gbit/s, 10Gbit/s over single pair cable



SPE Connector Standard – Current Status



- ISO/IEC 63171 CONNECTORS FOR ELECTRICAL AND ELECTRONIC EQUIPMENT**

	IEC 63171	IEC 63171-1	IEC 63171-2	IEC 63171-3	IEC 63171-4	IEC 63171-5	IEC 63171-6	IEC 63171-7
Initiator	N/A	Commscope	R&M	SIEMON	BKS	Phoenix Contact	Harting	TE
Picture								
Type		LC-Style	Rectangle	TERA 1P	Square-shaped	M8 / M12	Rectangle / M8 / M8 Hybrid/ M12	M12 Hybrid
Variants		1	1		1	2	4	7
IP-class		IP20	IP20	IP20	IP20	IP67	IP20 / IP67	IP67
Status Norm	published	ED1 published ED2 in progress	published	-	CDV	2. CDV publish in 2022	ED1 published ED2 in progress	CD
Target markets	N/A	Building	Building / Industry	-	Building / Data Center	Industry	Industry	Industry
Available products		-	Sockets, Patchcables	-	-	M8 sockets, M8 Patchcable	IP20 sockets, Patchcables	-

Longer reach application

- AN1829 - Understanding System Intrinsic Noise: Enabling Longer Reach and More Nodes on 10BASE-T1S Systems
- **Total 100M, 50 nodes , each node spacing for 2M**



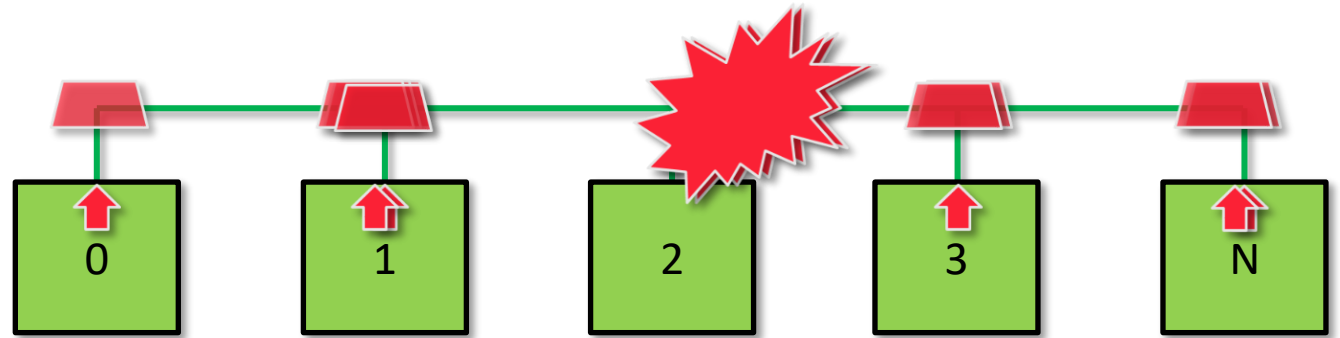
Introduction to T1S

What is 10BASE-T1S ?

- **Ethernet on a UTP bus line**

- 10 Mbps shared
- Half-duplex
- 2-50 nodes, depending on the environment
- Up to 25m bus length
- IEEE 802.3cg standard
- Low power
- No collisions (PLCA)

10 Mbps shared,
without PLCA, Collisions occur



What is 10BASE-T1S ?

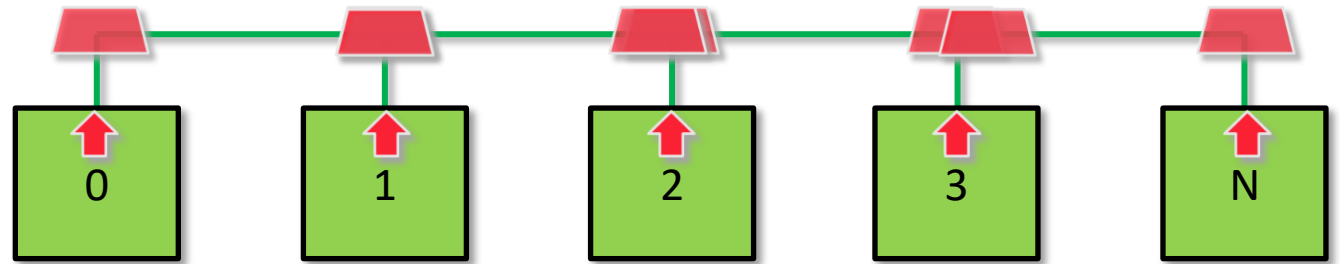
- **Ethernet on a UTP bus line**

- 10 Mbps shared
- Half-duplex
- 2-50 nodes, depending on the environment
- Up to 25m bus length
- IEEE 802.3cg standard
- Low power
- No collisions (PLCA)

- **Benefits**

- No switch required
- Inexpensive UTP cabling
- Inexpensive analog front end
- Less PHYs and cabling than with star topology
- Supports Functional Safety

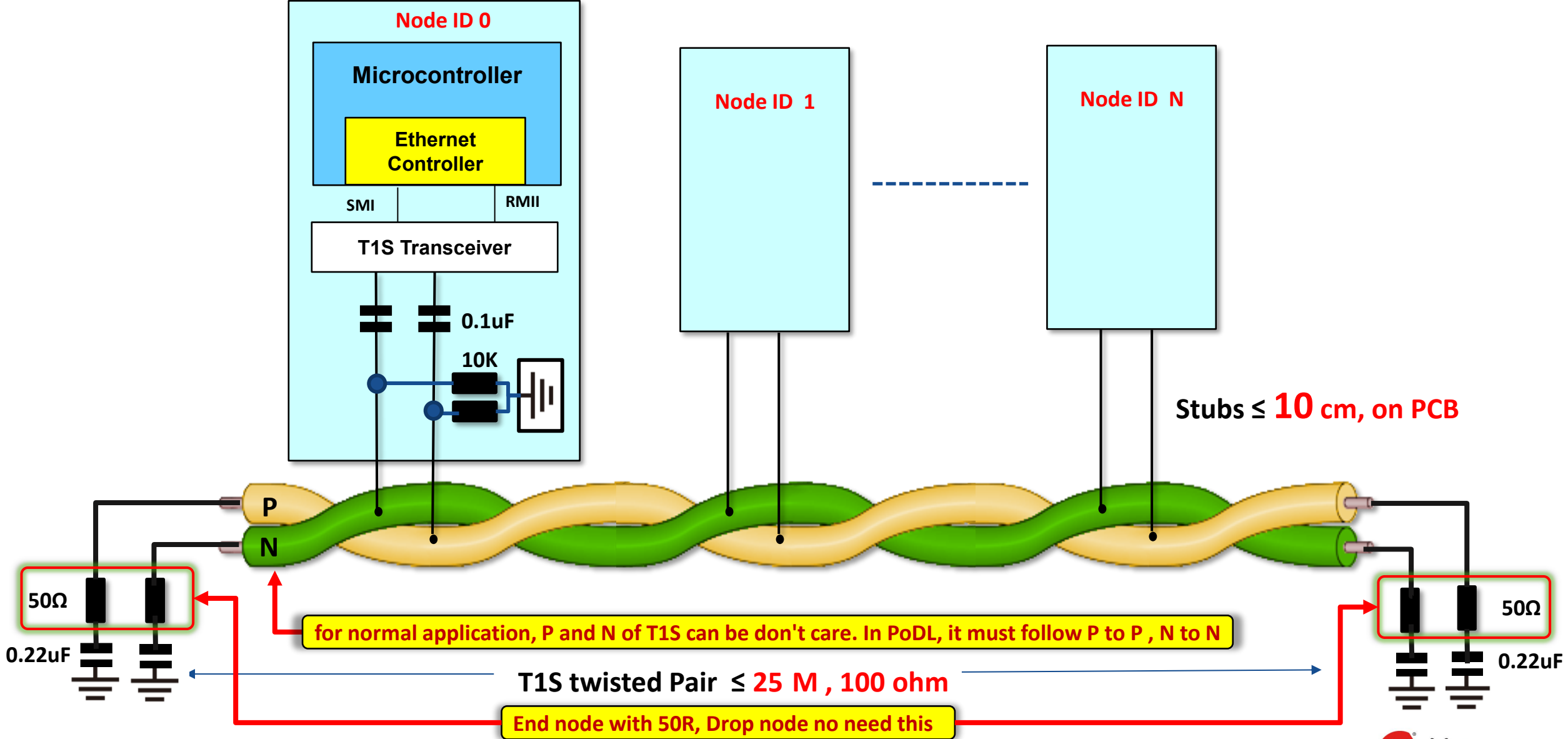
10 Mbps shared,
PLCA switched ON, avoids collisions



PLCA in IEEE 802.3 Ethernet Model

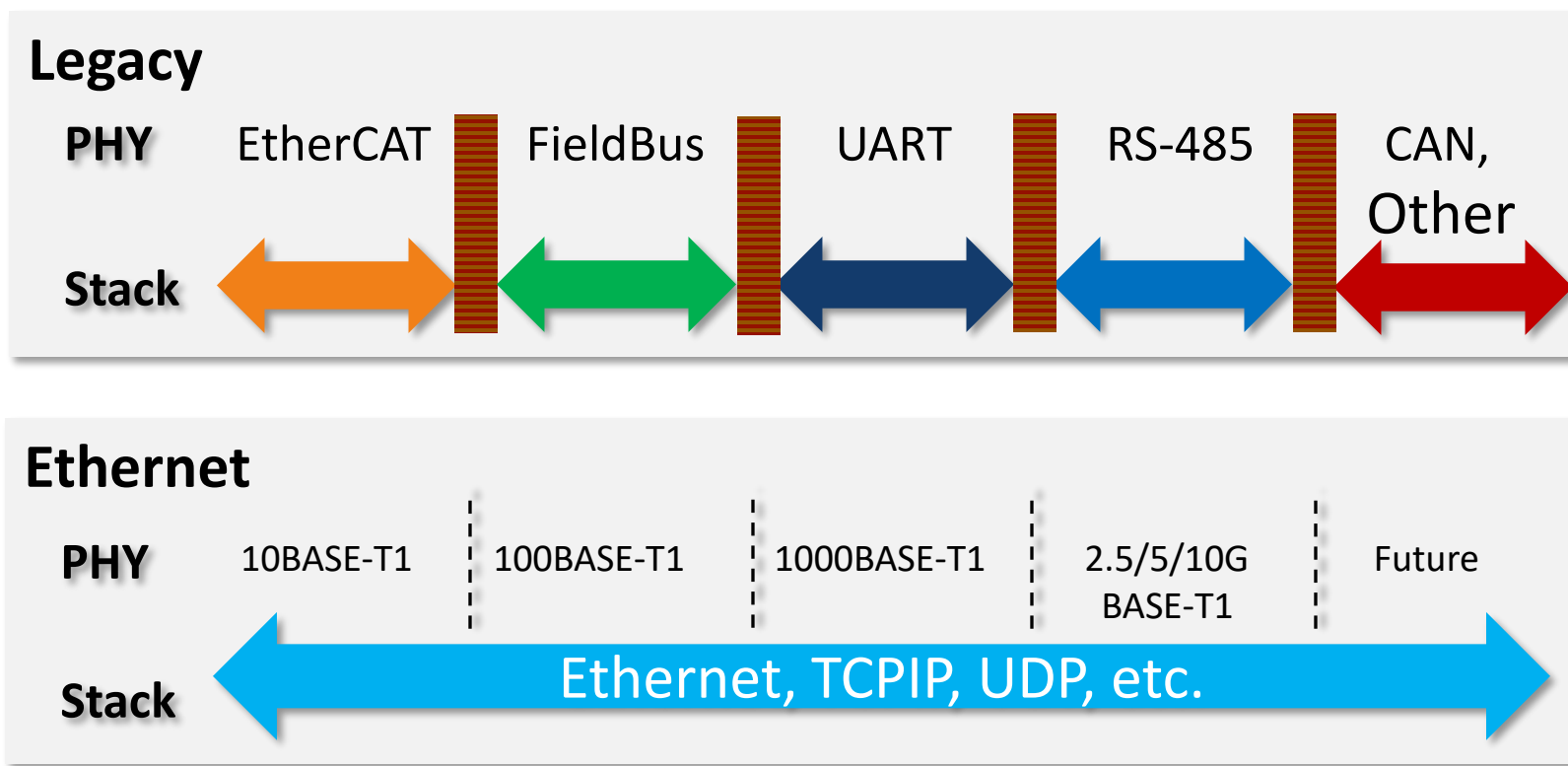
OSI Reference Model Layers	IEEE 802.3 Ethernet Model Layers	
Application	Higher Layers	
Presentation		
Session		
Transport		
Network		
Data Link	LLC = Logical Link Control (or other MAC client)	
	MAC Control	
	MAC = Media Access Control	
Physical	MII = Media Independent Interface	PHY
	PLCA Reconciliation	
	MII	
	PCS = Physical Coding Sublayer	
	PMA = Physical Medium Attachment	
	AN = Auto-Negotiation	
	MDI = Medium Dependent Interface	

T1S cabling example



Scalable from Low to High Bandwidths

- Common network software stacks
- Physical layer independent



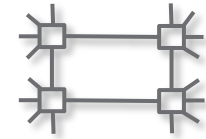
Motivation

Ethernet, a success story

- **Security**
 - Offering reliable standards (MACsec, IPsec)
- **Safety**
 - Preventing nodes from flooding network
 - Deterministic behavior
- **Synchronization**
 - Single time base for all nodes
- **Software defined system**
 - Sensors, actuators just offering a service
 - Independent applications operating on services

- **Wiring**

- Shorter cables through zone architecture
- Enabling machine-made wire-harness

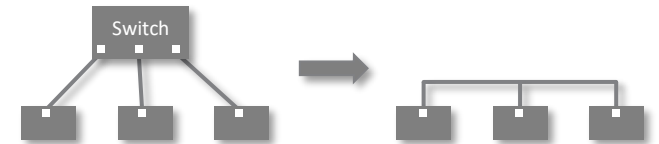


- **No Gateways**

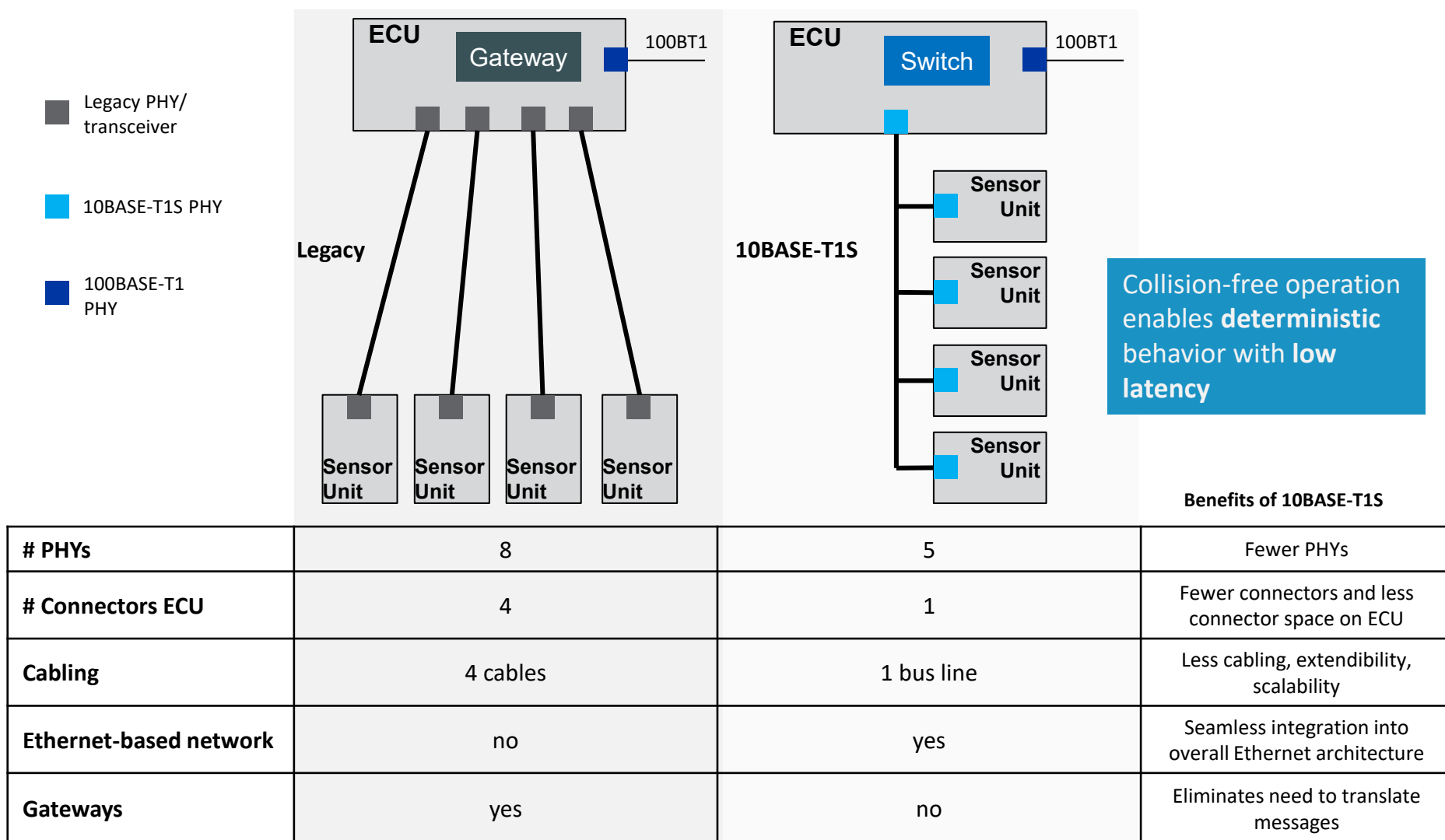
- Using switches instead of gateways

- **Multidrop**

- Fewer PHYs
- Less cables
- Fewer switch ports
- Simple UTP cabling



No Gateway, Less Phys and Cables, Reduce Costs



Introduction to OPEN Alliance

Standardization OPEN Alliance

- **TC6 (Technical Committee):**

- SPI Protocol for MAC-PHY
- Low pin count interface for Transceiver



- **TC10:**

- Sleep/Wake-up

- **TC14:**

- Compliance
- Interoperability
- EMC
- Diagnosis

- **TC17:**

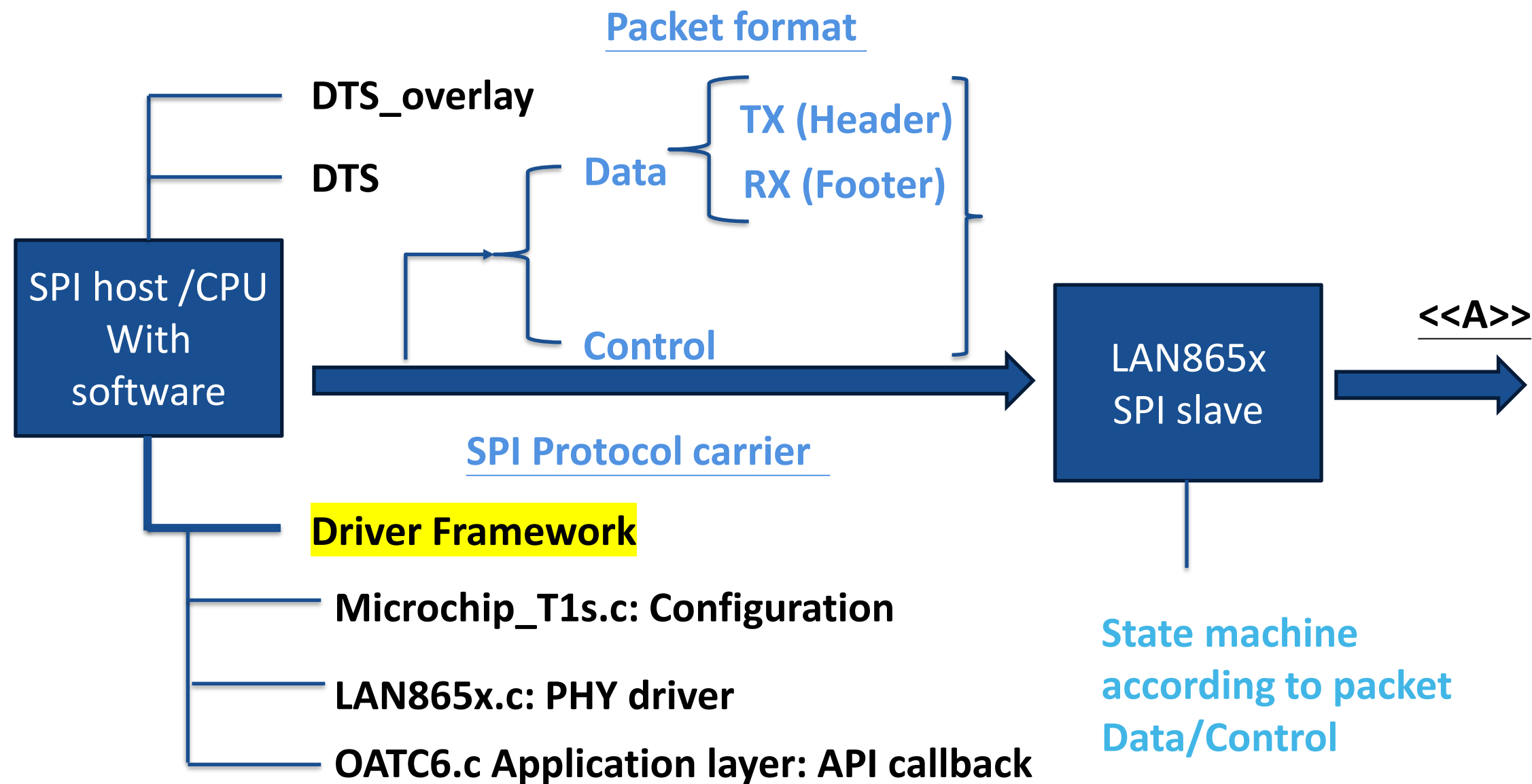
- MACsec

Founded in 2011, OPEN Alliance emerged with the idea of collaboration in the blossoming automotive Ethernet landscape. There was a gap in the industry that propelled its formation: a transparent space, governed by clear rules, for anyone to join and participate. Knowledge exchange, market development, and industrialization of the technology were its pillars. But the true cornerstone was the desire to break free from single-company specifications. OPEN Alliance envisioned a world where access to these crucial specifications wouldn't depend on favoritism or opaque promises, but on an open, fair, and universally accessible framework. By fostering transparent peer review and non-discriminatory access, OPEN Alliance aimed to pave the way for a collaborative future where innovation in Ethernet could truly thrive.

OPEN: One Pair Ethernet

小组	名称	Status
TC 1	Interoperability & Compliance Tests for 100BASE-T1 PHYs 100BASE-T1 PHY的物理层互操作性和一致性测试	on hold
TC 2	100BASE-T1 Ethernet Channel & Components 100BASE-T1 PHY的以太网通道和组件测试	on hold
TC 3	1000BASE-T1 CMC Requirements 1000BASE-T1 PHY CMC要求	rescoped
TC 4	Tools 工具	completed
TC 5	Gap Identification 差距识别	on hold
TC 6	Common Media Independent Interface Definitions for Automotive Purposes 车用通用媒体独立接口定义	in progress
TC 7	Gigabit Ethernet over Optical Fiber 千兆光纤以太网	in progress
TC 8	Automotive Ethernet ECU Test Specification 车载以太网ECU测试规范	on hold
TC 9	Automotive Ethernet Channel & Components 车载以太网通道和组件	in progress
TC 10	Automotive Ethernet Sleep/Wake-Up 车载以太网睡眠/唤醒	in progress
TC 11	Ethernet switch requirements and qualification 以太网交换机要求和资格	in progress
TC 12	Test specifications for the compliance testing of IEEE 1000BASE-T1 (IEEE802.3bp, Clause 97) Physical Interface (PHY) devices 1000BASE-T1 PHY设备一致性测试规范	in progress
TC 13	New Test House Qualification Requirements 测试机构资格新要求	in progress
TC 14	Interoperability & Compliance Tests for 10BASE-T1S PHYs 10BASE-T1 PHY互操作性和一致性测试	in progress
TC 15	Multi Gig Interoperability and Compliance Tests 多千兆互操作性和一致性测试	in progress
TC 16	EEE Interoperability and Compliance EEE互操作性和一致性	in progress
TC 17	MACsec Automotive Profile MACsec汽车行业配置	in progress

TC6 - Overview



TC10 - Key goal

- new service primitives provided by the ISO/OSI layer 1 (PHY) and supporting a controlled link shutdown and a **fast global wake-up** within an Ethernet network, The sleep and wake up process in a PHY shall fulfill SPEC'S timing requirement (next page
- Power consumption The following guidelines on the device power consumption in sleep mode target typical Ethernet products such as single and multiport PHYs and switches. A single-port PHY product should have a quiescence current of **35μA**. A multi-port PHY or switch product should have a quiescence current of 25μA plus 10μA for each port4 .

Pins and command about TC10

- Pins

Pin Name	Description
VBAT	This pin powers the sleep or wake management block.
INH	This is an open-drain output pin that goes low to disable power. During normal operation, an external resistor pulls it high to enable power.
WAKE_IN	This is an input pin that can be used to wake up the PHY from Sleep state.
WAKE_OUT	This is an output pin that is asserted when the PHY receives a wake-up request.

- Command

Command Name	Description
Wake-Up Request (WUR)	This is the MDI wake-up request command sent on an active link.
Wake-Up Pulse (WUP)	This is the MDI wake-up request command sent on an inactive link.
Low-Power Sleep (LPS)	This is the MDI sleep request and acknowledge command.

TC10 Specification / Terminology

- **Low Power Sleep (LPS)**
 - Command to indicate a sleep request to the link partner. Request transition to SLEEP state.
 - 100BASE-T1: Encoded in the IDLE scrambler stream. Sent for a minimum of 64 bits.
 - 1000BASE-T1: Sent in OAM (Operations, Administration, Maintenance) Frame.
- **Wake-Up Request (WUR), for PHY is link up case**
 - Command indicating a wake-up request to the link partner.
 - Can be sent by a node PHY or switch PHY to distribute the wake-up request over a link, which is already active.
 - 100BASE-T1: Encoded in the IDLE scrambler stream. Sent for a minimum of 64 bits.
 - 1000BASE-T1: Sent in OAM (Operations, Administration, Maintenance) Frame.
- **Wake-Up Pulse (WUP): for PHY is link down case**
 - Command indicating wake-up request to link partner. Sent when link is not active.
 - Transmitted using link training (idle) codes over the network, for a duration of 1 ms (+/- 0.3ms)

Result (both 100 and 1000 mode are PASS)

Testhouse

C&S group GmbH

Schweigerstr. 13A

D-38302 Wolfenbuettel

Phone: +49 5331/ 90 555 - 0

Web: www.cs-group.de

C&S

OPEN

ALLIANCE

Preliminary

P24_0286_LAN8870_1000B_TC10_Test_Report_00D_00

Date of Release: 2024-Nov-29

Test Report

Device Under Test

Device Name

LAN8870

Manufacturer

Microchip

Type / Version

-

Sample marking

LAN8870v VB000e3
241146Q

Customer

Order No.

P24_0286

Name

Microchip Technology Inc.

Address

80 Arkay Dr, Ste 100
11788 Hauppauge NY
USA

Number of Pages

38

Test Period

from 2024-Nov-19 until 2024-Nov-28

Test Method / Test Requirement

1000BASE-T1 Interoperability Test Suite

Performed Tests and References

1000BASE-T1 Interoperability Test Suite v1.9.1
(2024-Feb-13) - TC10 Sleep/Wake-up functionality

Interoperability Test Results

1

1000BASE-T1 Interoperability
Test Suite¹

PHY Sleep/Wale-up Functionality: PASS

	Link Partners		
Chapter	RTK RTL9010AA	BCM 89884	MIC LAN8870 ²
Group 5: Wake-up reception and signaling	PASS	PASS	PASS
Group 6: Wake-up transmission	PASS	PASS	PASS
Group 7: Wake-up forwarding	NOT APPLICABLE ³	NOT APPLICABLE ³	NOT APPLICABLE ³
Group 8: Sleep	PASS	PASS	PASS

Testhouse

C&S group GmbH
Schweigerstr. 13A
D-38302 Wolfenbuettel
Phone: +49 5331/ 90 555 - 0
Web: www.cs-group.de

C&S

OPEN

ALLIANCE

Preliminary

P24_0286_LAN8870_100B_TC10_Test_Report_00D_02

Date of Release: 2024-Nov-29

Test Report

Device Under Test

Device Name LAN8870
Manufacturer Microchip
Type / Version -
Sample marking LAN8870v VB000e3
 241146Q

Customer

Order No. P24_0286
Name Microchip Technology Inc.
Address 80 Arkay Dr, Ste 100
 11788 Hauppauge NY
 USA

Number of Pages

40

Test Period

from 2024-Nov-6 until 2024-Nov-15

Test Method / Test Requirement

100BASE-T1 Interoperability Test Suite

Performed Tests and References

100BASE-T1 Interoperability Test Suite v1.2
(2024-Feb-13) - TC10 Sleep/Wake-up functionality

Interoperability Test Results

The test results refer to the delivered device and the applicable test cases.

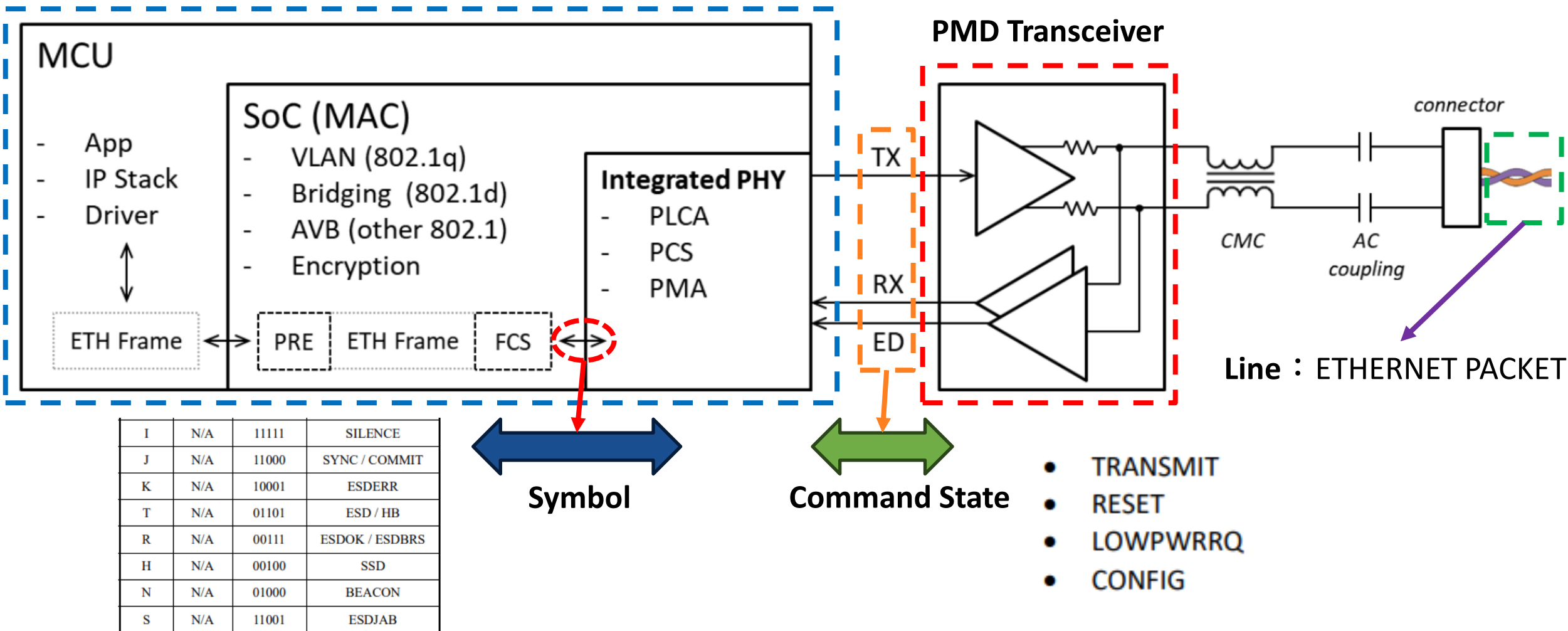
1 100BASE-T1 Interoperability Test Suite¹

PHY Sleep/Wale-up Functionality: PASS

	Link Partners			
Chapter	RTK RTL9000AA	MVL 88Q1110	NXP TJA1101B	MIC LAN8870 ²
Group 5: Wake-up reception and signaling	PASS	PASS	PASS	PASS
Group 6: Wake-up transmission	PASS	PASS	PASS	PASS
Group 7: Wake-up forwarding	NOT APPLICABLE ³	NOT APPLICABLE ³	NOT APPLICABLE ³	NOT APPLICABLE ³
Group 8: Sleep	PASS	PASS	PASS	PASS

TC14 - OA3P (open alliance 3 pins)

MCHP SoC / Endpoint



MCHP solution (SoC / Endpoint)

- LAN8660 , LAN8661 : IO to OA3p

- **Control**

- Sensors
- Actuators

- **Audio**

- Microphones
- Speakers

- **Light**

- Interior lights
- Exterior lights
- Small displays

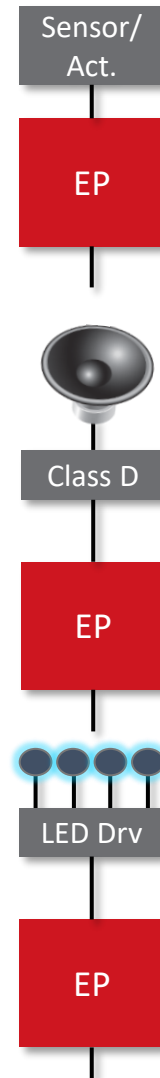
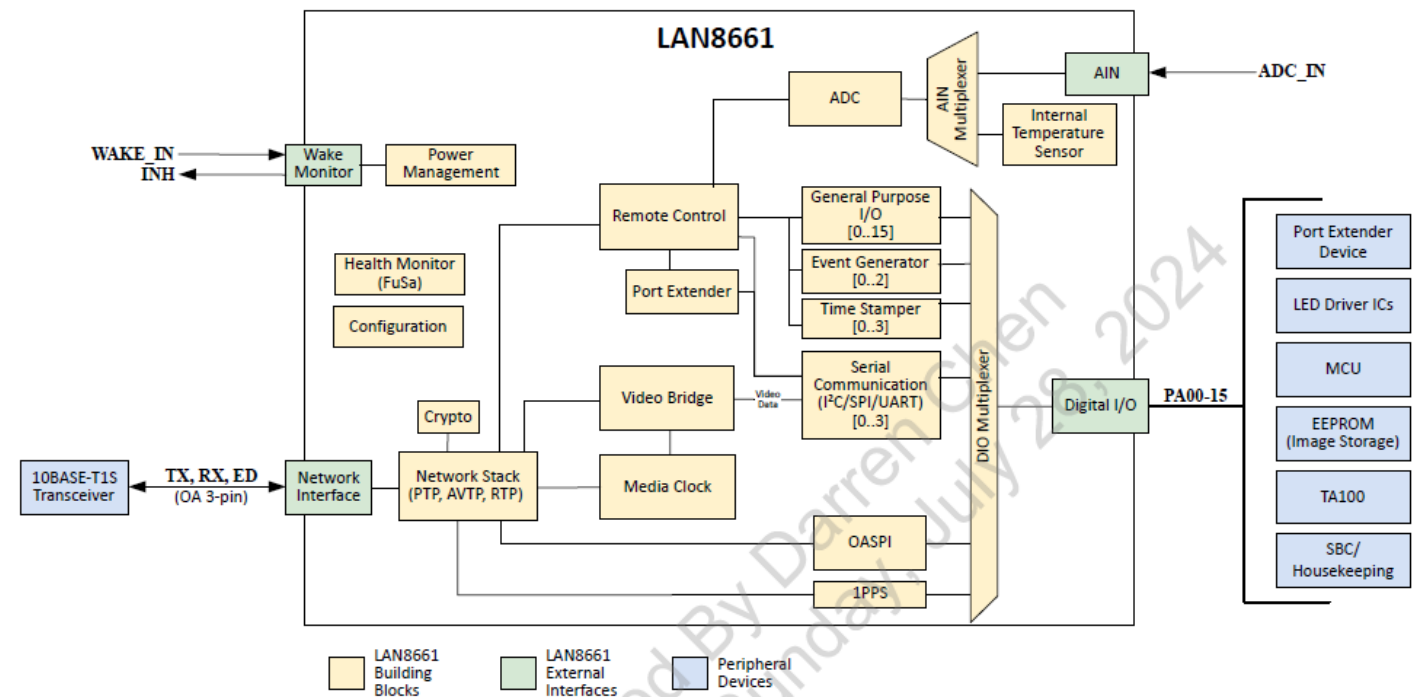
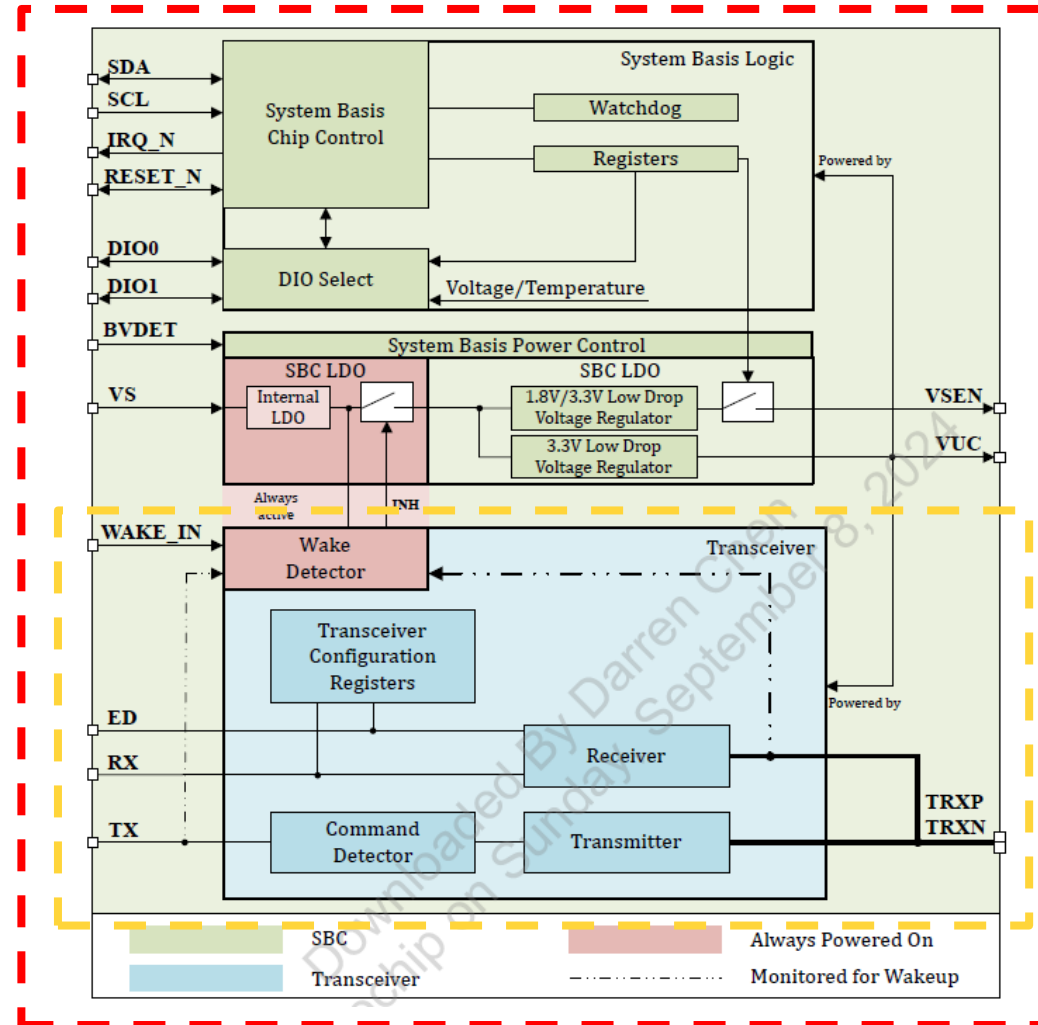


Figure 1-1. LAN8661 Block Diagram



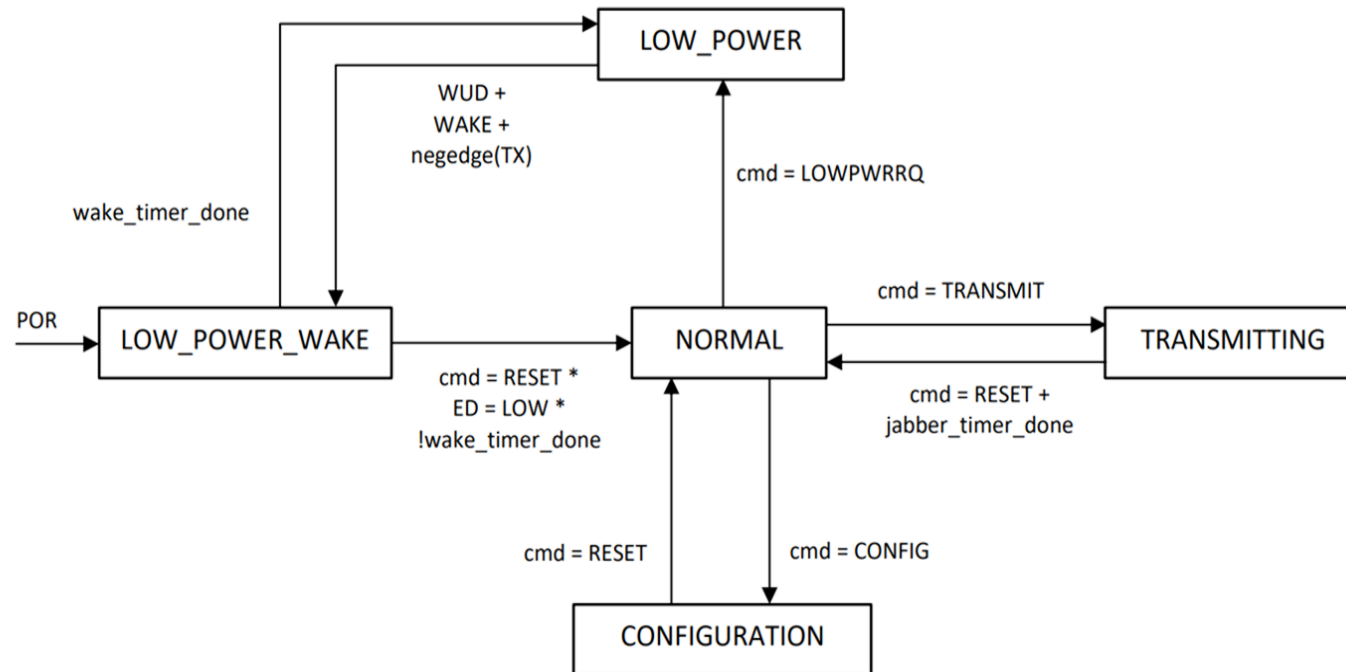
MCHP solution (PMD Transceiver)

- **LAN8679** : OA3p to T1S
- **LAN8680** : OA3p to T1S >> TC14 and I2C interface (SBC)



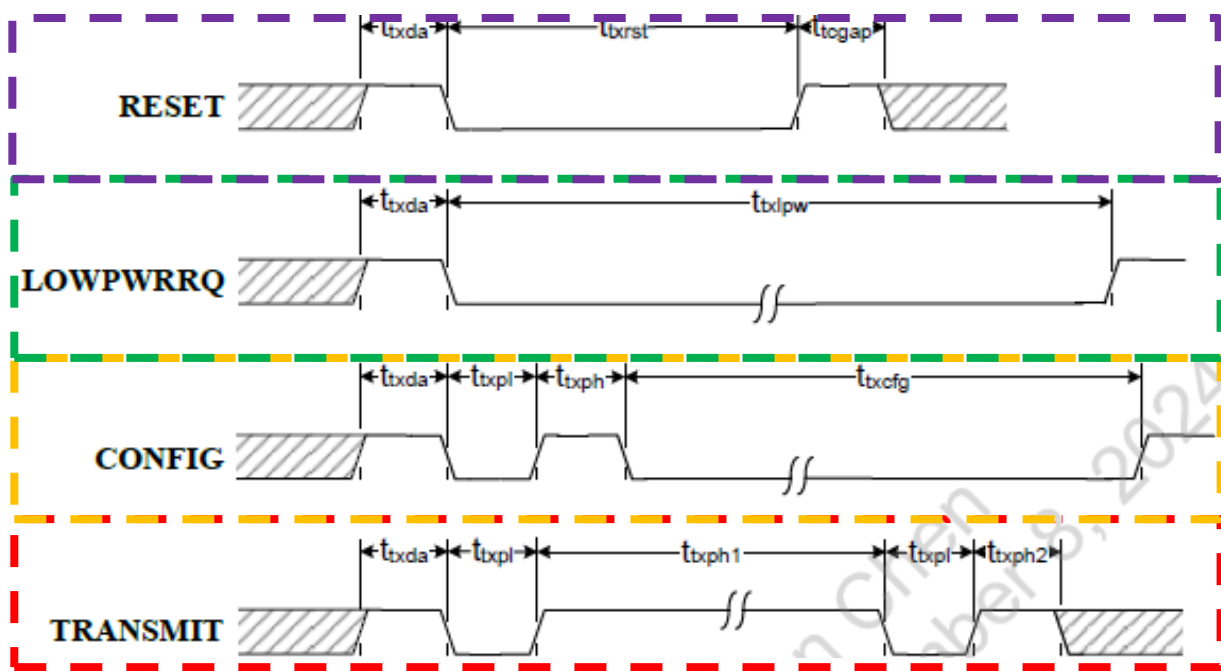
Functional States available in Transceiver

- The host controller uses the TX pin to send commands and activate the different function states of the transceiver. These states include the **TRANSMITTING** state, **NORMAL** state, **LOW-POWER** state, and **CONFIGURATION** state.
- Instead of using a data line with a clock to transmit data and activate the functions in the transceiver, the TX pin instead sends pulses of defined length. This is done to reduce electromagnetic emissions.



Command timing on TX pin

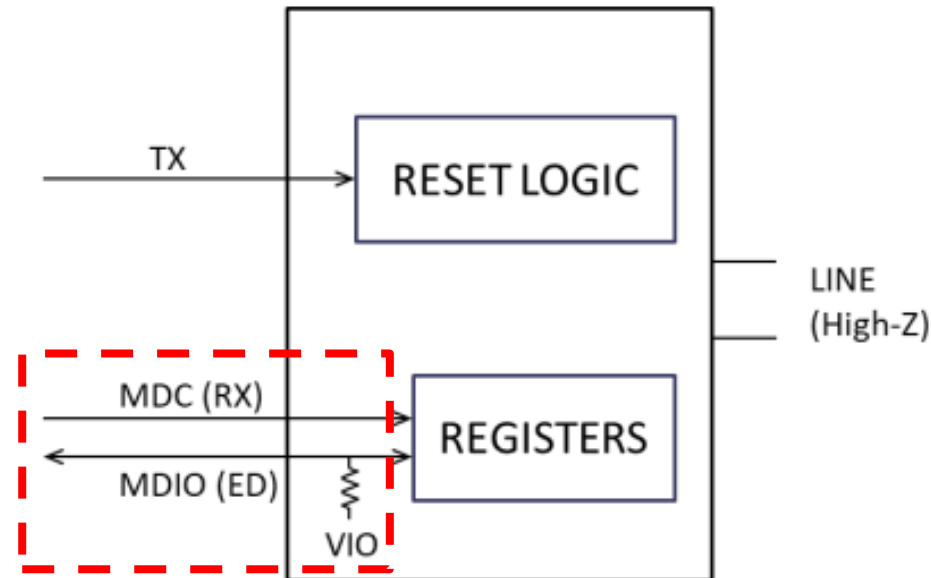
- The TX pin is a high-speed low-voltage 3.3V output to an external PMD transceiver. It encodes the transmit data and controls the PMD reset, configuration, and entering/exiting low-power states.



Parameter	Symbol	Min	Typ	Max	Units
Multiple Commands					
Deassertion time before a command	t_{txda}	TBD	20	-	ns
Low pulse duration	t_{txpl}	TBD	20	TBD	ns
RESET Command					
RESET command assertion time	t_{txrst}	TBD	80	TBD	ns
Gap between commands	t_{tcgap}	TBD	20		ns
LOWPWRRQ Command					
LOWPWRRQ command assertion time	t_{txlpw}	TBD	16	-	μ s
CONFIG Command					
High pulse duration	t_{txph}	TBD	20	TBD	ns
CONFIG command assertion time	t_{txcfg}	TBD	16	-	μ s
TRANSMIT Command					
TRANSMIT command high pulse duration	t_{txph1}	TBD	180	TBD	ns
TRANSMIT end high pulse duration	t_{txph2}	TBD	20	TBD	ns

Functional States available in Configuration

- When the PMD transceiver is set to configuration mode, the RX and ED pins are used as MII management interface (MIIM). pins are used as the MDC and MDIO pins to access configuration registers.
- During configuration mode, the transceiver does not drive any differential voltage at the line interface with low impedance, regardless of any activity on the TX pin.

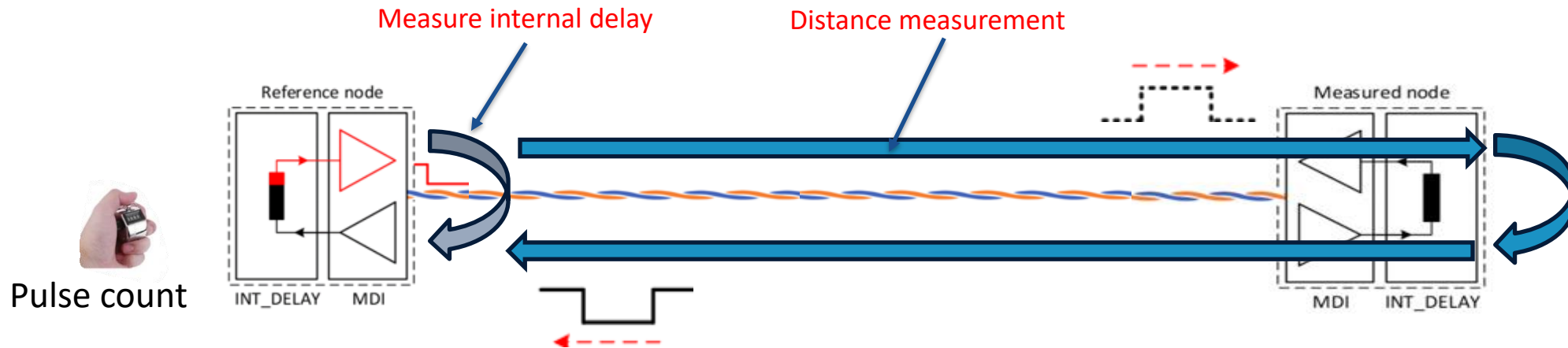


Topology Discovery

- **TC14's topology discovery function :**

- Automatically identify network topology:
 - In automotive Ethernet, 10BASE-T1S supports bus topology with multi-point connections. Topology discovery can help identify the physical connection relationship between nodes.
- Diagnosis and troubleshooting:
 - Through topology discovery, the system can monitor the network structure in real time and detect and locate fault points early.
- Improve system reliability:
 - Accurate topology information helps optimize data transmission paths, reduce latency and packet loss, and improve overall network performance.

Topology Discovery



● Procedure

- Only one Reference /Measure pair operating per time (topology configuration phase) , the rest nodes keep “listening/receive “ and PLCA suggestion to turn off during this phase
- Enter CONFIGURATION mode by set the TPEN bit and configure the TPREFN bit in register CTRL0 and leave CONFIGURATION mode using a RESET command (for both reference (host) and measured node)
- TX will keep generating Generate TP_pulse , this pulse will keep cycling within the loop. The reference node need keep counting how many pulse has been counted
- “**unfiltered**” ED signal (Filtered signal will introduce delay) to calculate the pulse traveling time (T_p) between reference node and measure node at ED pin's MEAS FSM period
- Counting the # of “local count” as the reference (reference node loop back) : **Blue**
- Counting the # of “remote count” as the measured node : **Orange**
- Longer distance will get more count
- Program convert the counting # to distance
- Normally , cable propagation delay about 5ns/m

Enabling TD in the LAN8680

Bit 1 – TPREFN Topology Discovery Reference Node select

When the PMD is configured for topology discovery operation, this bit will select which topology discovery mode the PMD operates in. When set, the PMD will function as a topology discovery measured node. When clear, the PMD will function as a topology discovery reference node. This configures the PMD topology discovery internal scrambler for the correct operation as a measured or reference topology discovery node.

Note: This bit has no effect when the Topology Discovery Enable (TPEN) bit is clear.

Value	Description
0	PMD operates as a topology discovery measured node
1	PMD operates as a topology discovery reference node

Bit 0 – TPEN Topology Discovery Enable

When set, the PMD operates in topology discovery mode.

Value	Description
0	Normal operation
1	Topology discovery mode

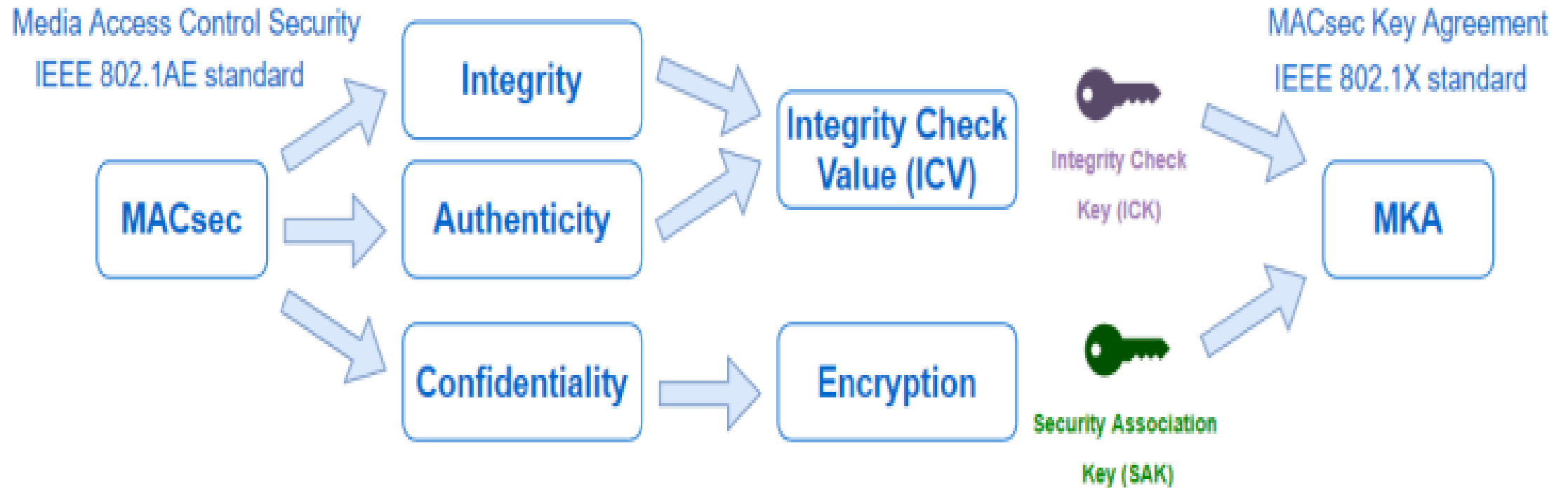
This does not include the setup and operation of the host.

Open Alliance TC17 MACsec Auto Profile

- **MACSec = Media Access Security IEEE 802.1AE**
 - Point to point security protocol with Confidentiality, Integrity and Authentication
 - Transmit side Adds Security Tag (SecTag) and Integrity Check Value (ICV) while encrypting the payload
 - Receive side identifies and decrypts packets
 - Identifies packets
 - Decrypts
 - Prevents replay
 - Removes SecTag and ICV discarding invalid packets
 - All unencrypted frames dropped other than diagnostic and key agreement
- **MKA can be cumbersome in automotive applications**
- **Some OEM's mandating keys are stored in secure elements / HSM's with tamper protection**

Goal

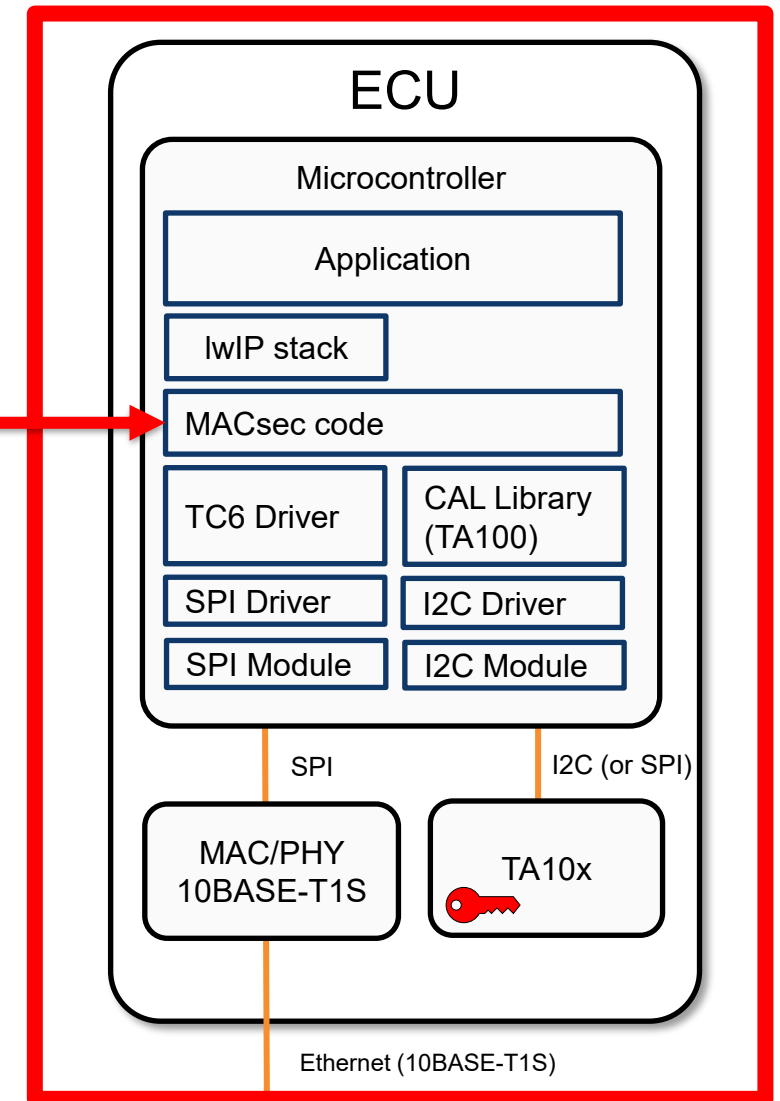
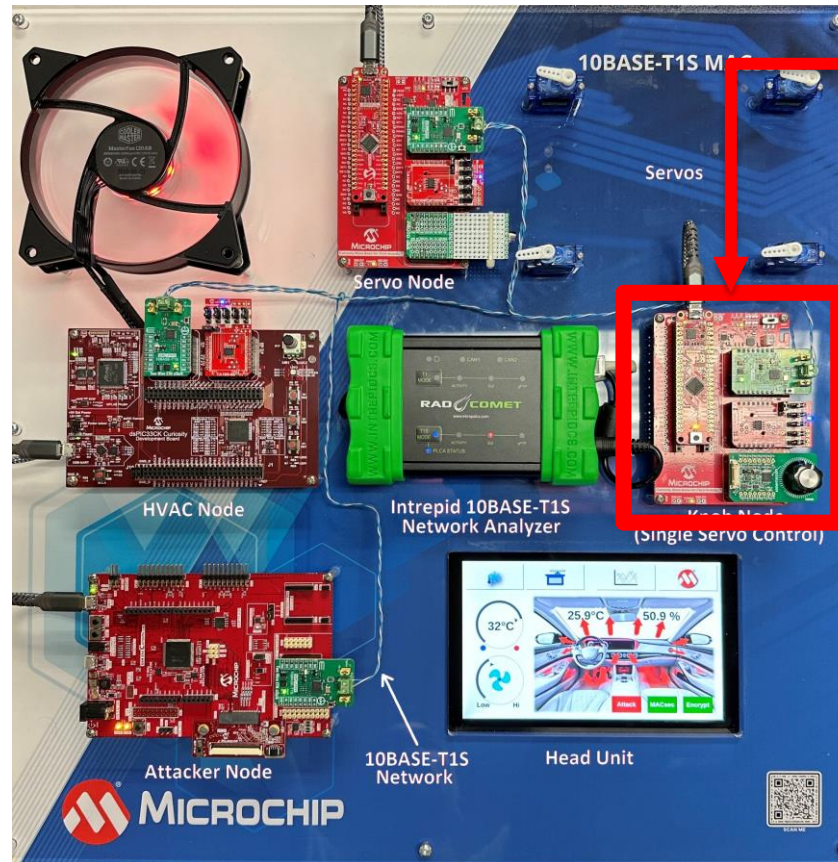
- It ensures the security of Ethernet data frames through functions such as identity authentication, data encryption, integrity verification, and replay protection, preventing devices from processing messages that pose security threats.



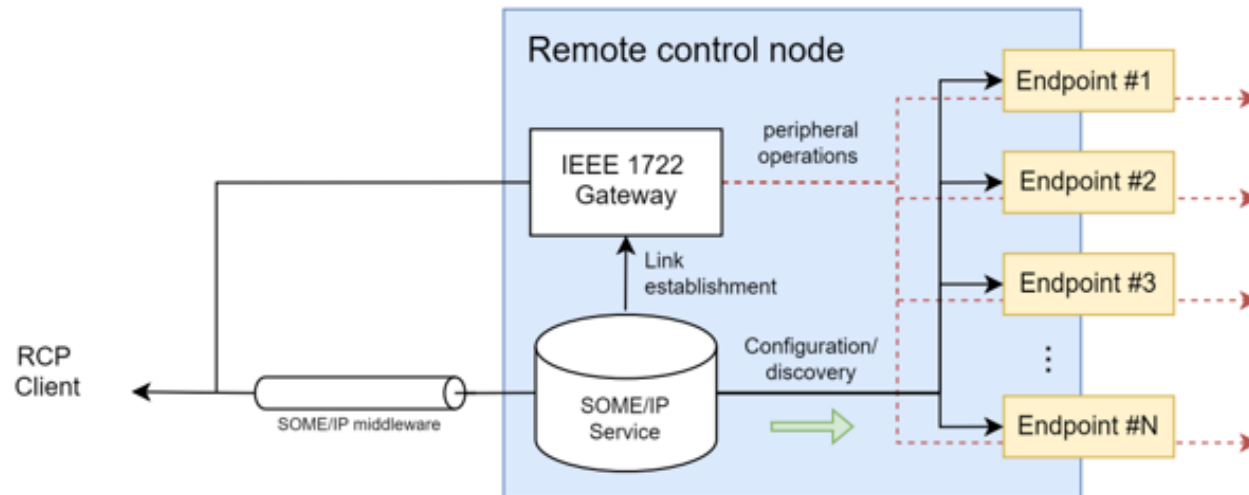
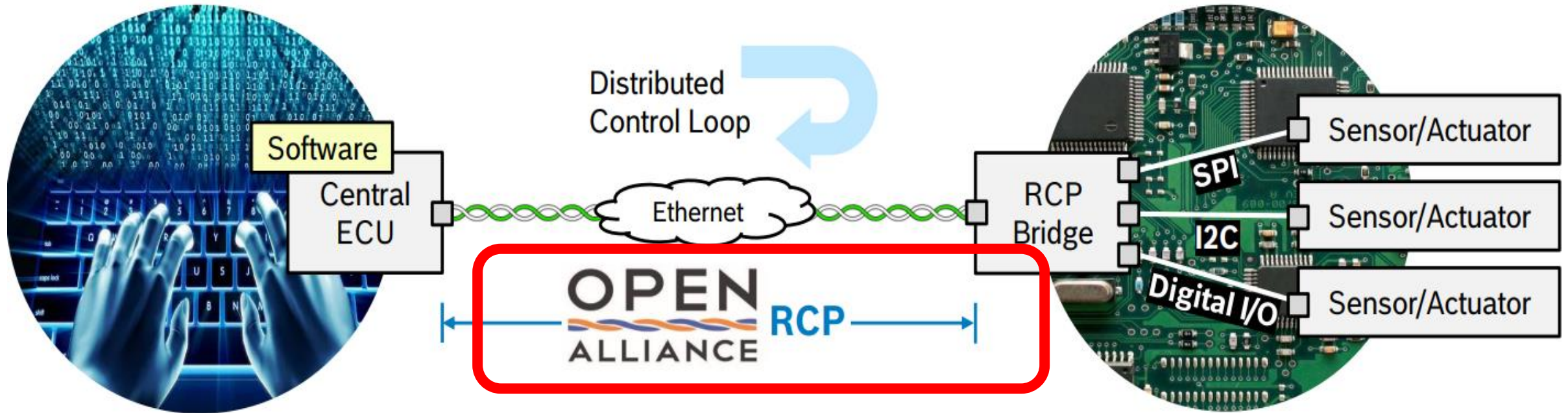
TC17 MACSEC for automotive

- **10M SPE :**

- The MACSEC is implement by software suite
- Key was loaded in the TA100
- JIL FISP certificated



Remote Controlled Peripherals/Protocol



10BASE-T1S Control/Lighting/Audio Endpoints

Optimized for MCU-less edge devices

- **Features**

- Control protocol
 - SOME/IP
 - TC18 RCP
- Streaming protocol
 - AVTP
 - UDP/RTP
- Security / Safety
 - MACsec, E2E Profile 4, SecOC
 - External HSM TA100 (opt.)
 - Secure Configuration / Firmware Update
- Content delivery modes (LAN8661X only)
 - Video streaming
 - Edge Node Rendering (graphic commands)
 - Play pre-stored graphics, videos
- gPTP Slave
- PDM to PCM conversion (LAN8662X only)
- External MCU via I2C or SPI (optional)

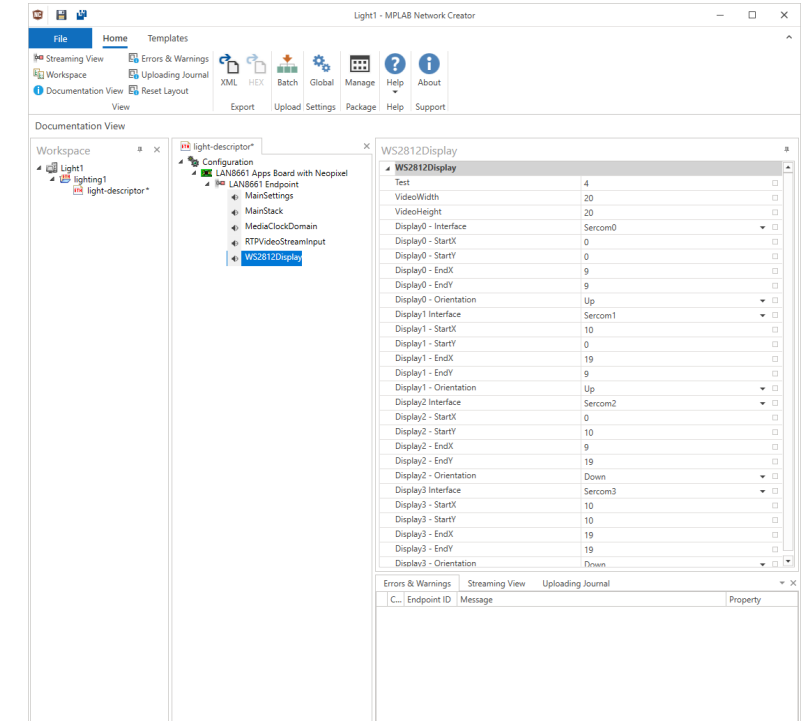
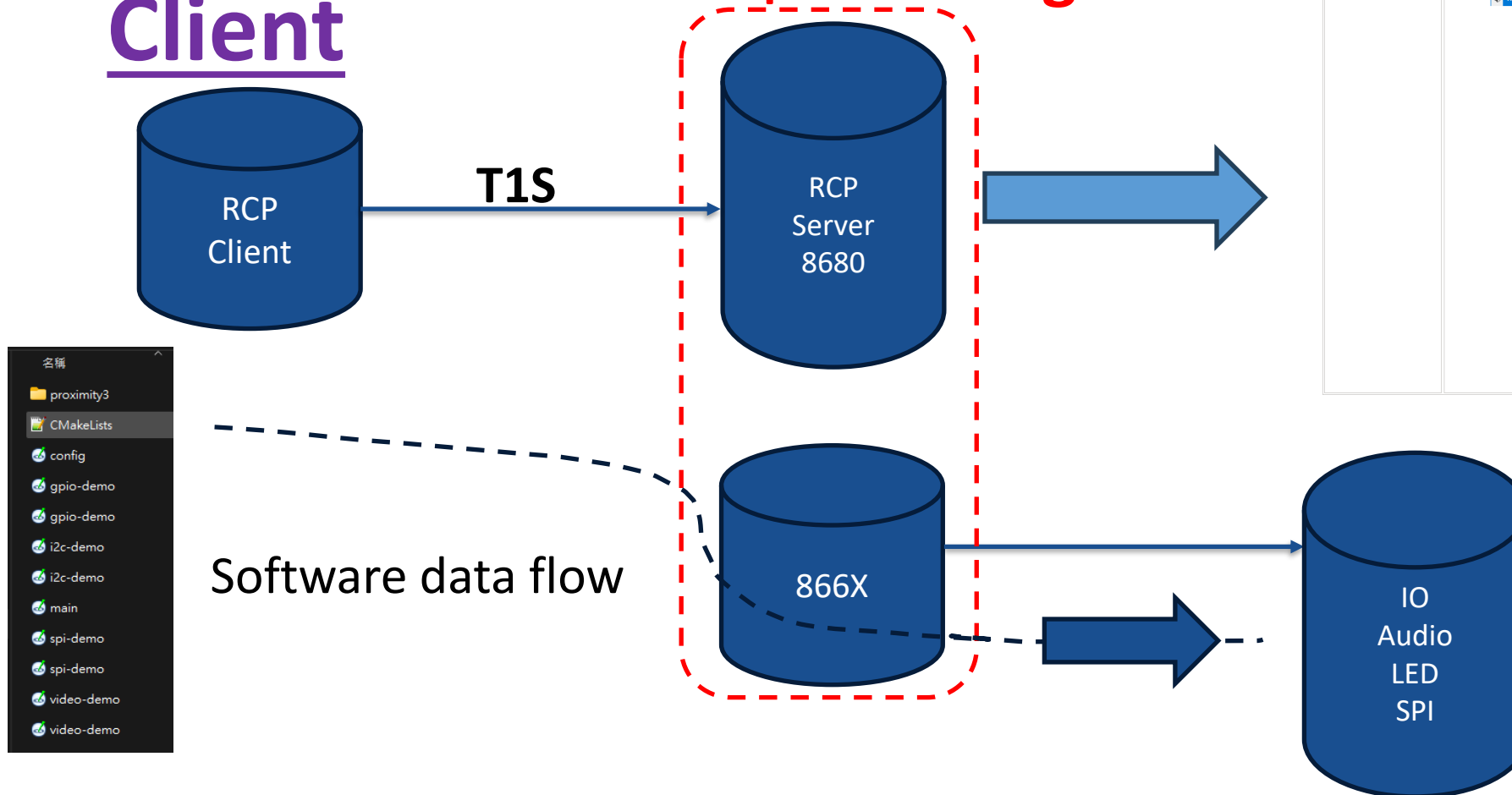
Control/Lighting/Audio

Available Features	LAN8660X	LAN8661X	LAN8662X
SOME/IP	✓	✓	✓
RCP	✓	✓	✓
OPEN Alliance TC10/14 Wake/Sleep	✓	✓	✓
gPTP	✓	✓	✓
PWM Interface	✓	✓	—
AVTP	—	✓	✓
RTP	—	✓	✓
Video Streaming and LED Driver IC Control	—	✓	—
Audio Interfaces	—	—	✓
CAN FD Controller	✓	✓	—
SERCOMs	Up to 4	Up to 4	Up to 3
GPIOs	Up to 16	Up to 16	Up to 12
ADC Channels	1	1	1

Remote- Control- Client

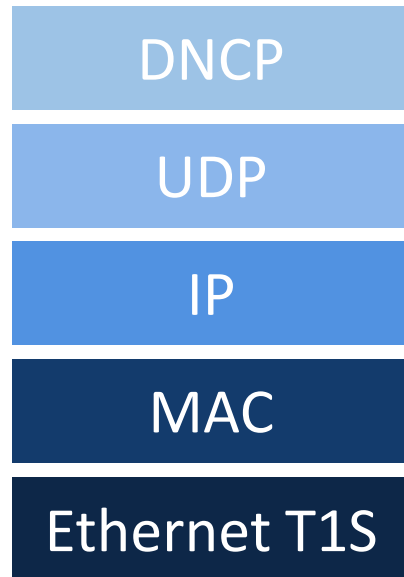
MCU less

End point configurator

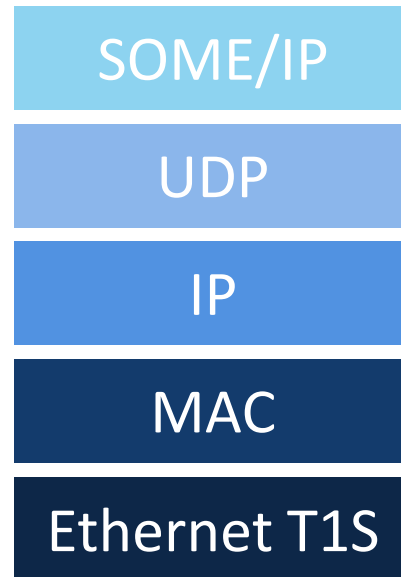


LAN866x Protocols

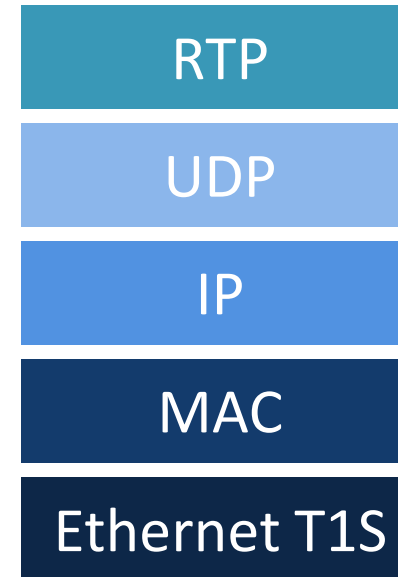
Configuration:



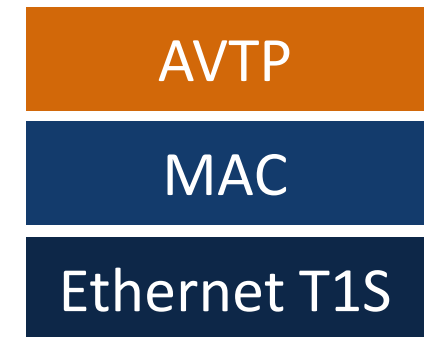
Control:



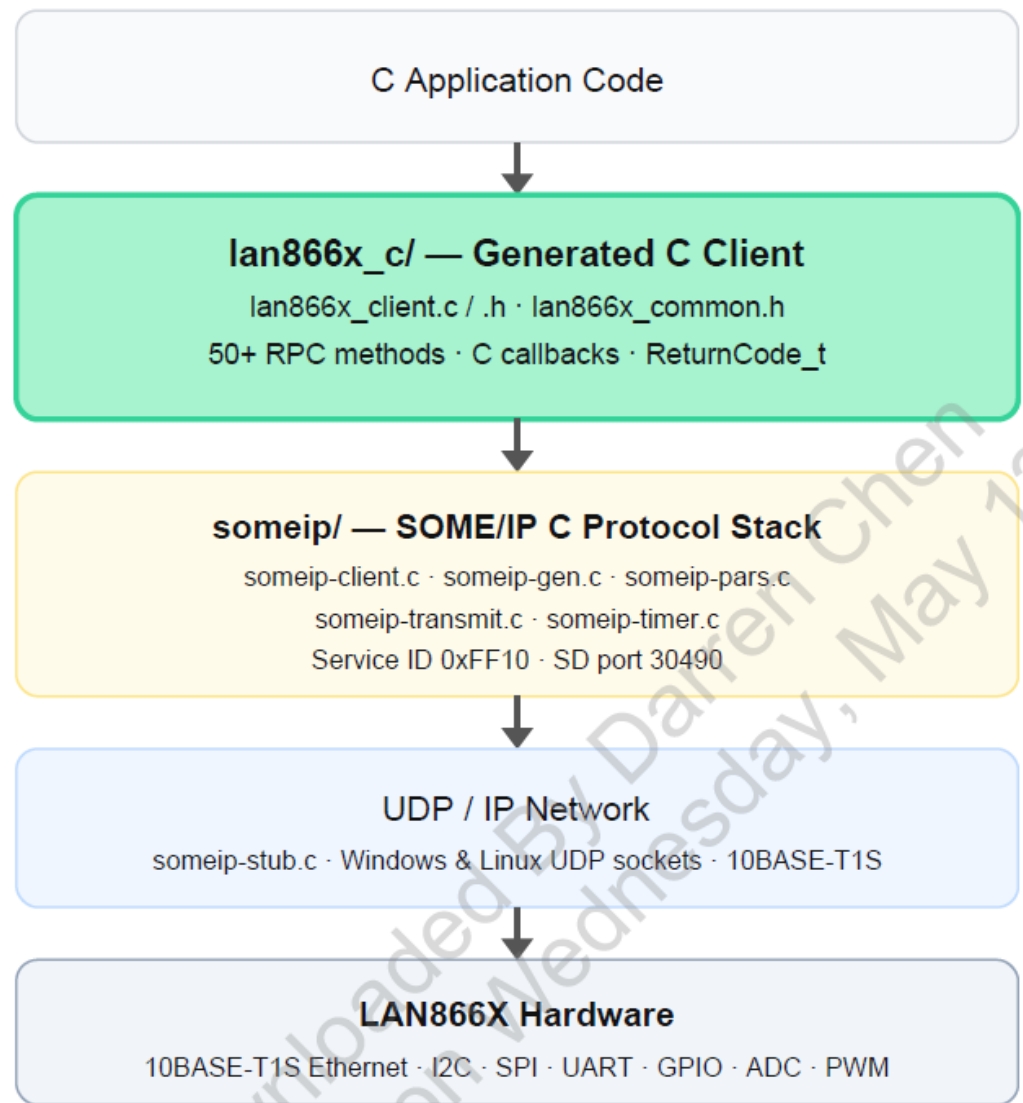
Video:



Audio:



Client Tools

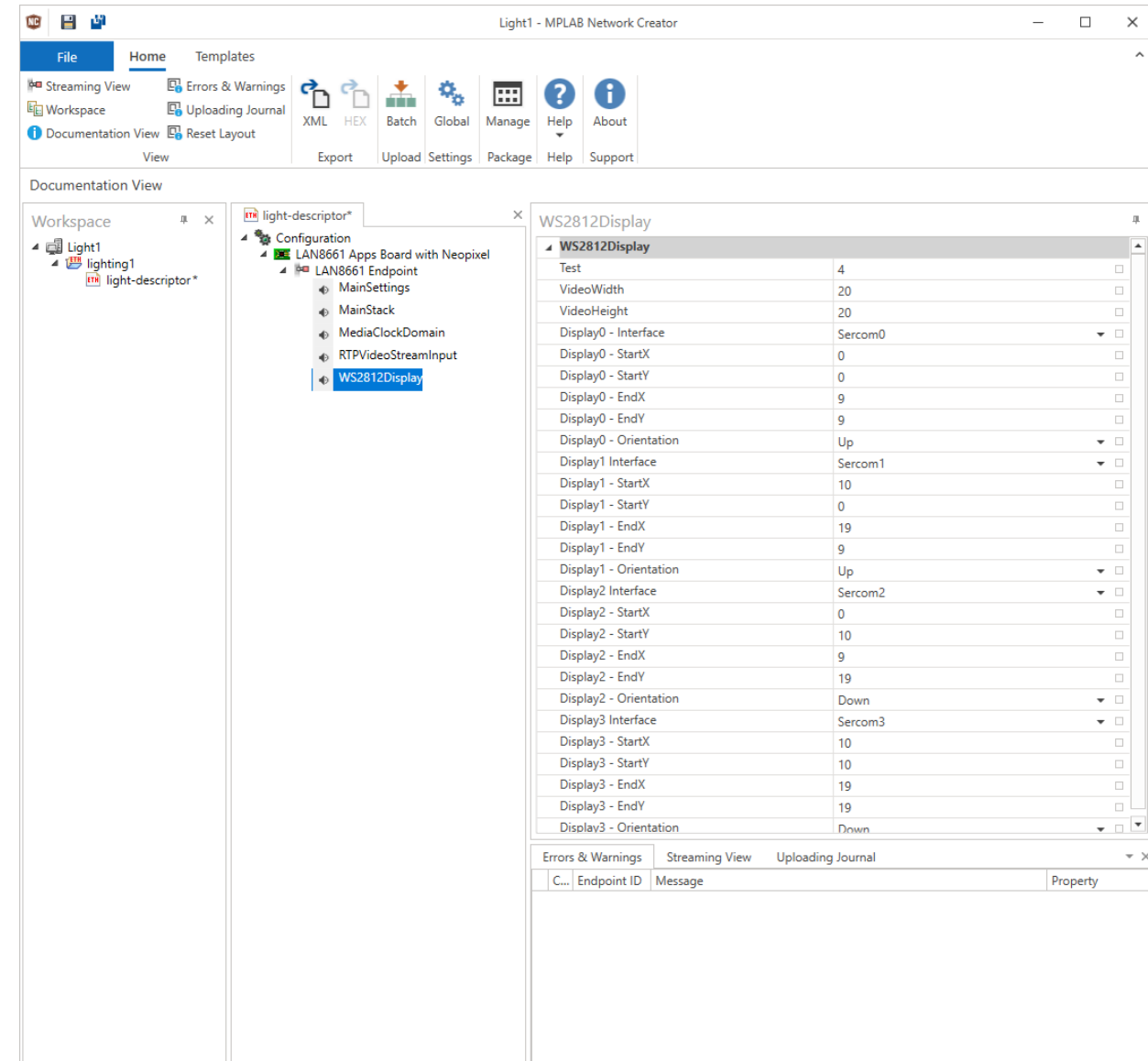


We support multiple tools

Folder	Content
generator/	Code generator: lan866x.proto, Jinja2 templates, generate-lan866x.{bat,sh}, Doxyfile, version-sync helper.
lan866x_c/	Generated C client (lan866x_client.c/.h + lan866x_common.h) with C examples.
lan866x_cpp/	Generated C++ client with C++20 examples.
lan866x_python/	Python client (pybind11 bindings + wrappers), prebuilt Windows wheels, examples.
lan866x_csharp/	.NET 8 C# client, NuGet package, examples.
lan866x_java/	Java 11 client, JNI shim, prebuilt Windows JAR, examples.
lan866x_rust/	Rust client, FFI bindings, prebuilt Windows DLL, examples.
lan866x_blockly/	Drag-and-drop Blockly frontend for the Python client.
lan866x_arxml/	Generated AUTOSAR XML (lan866x.arxml).
lan866x_wireshark/	Wireshark SOME/IP dissector config files.
lan866x_doxygen-cpp/	Generated Doxygen HTML for the C++ API.
someip/	Git submodule: the SOME/IP protocol stack.
tools/	Developer utilities (ARXML validator, CA bundle maker).
docs/	Architecture SVGs, design documents, specifications.

Endpoint Configuration

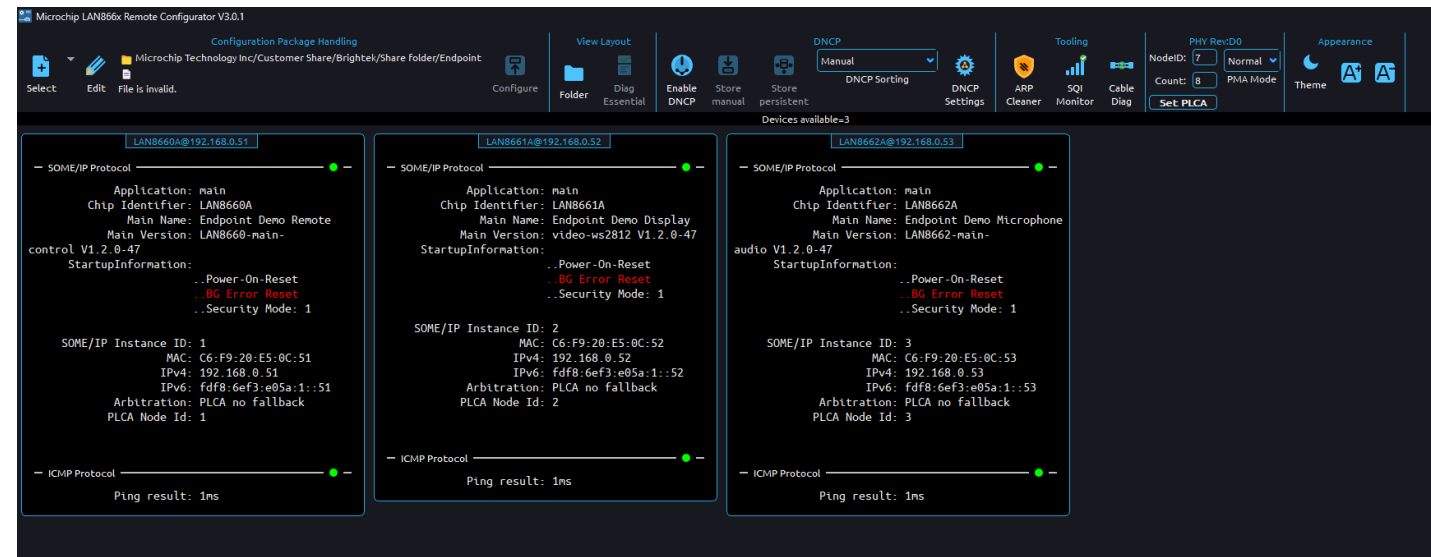
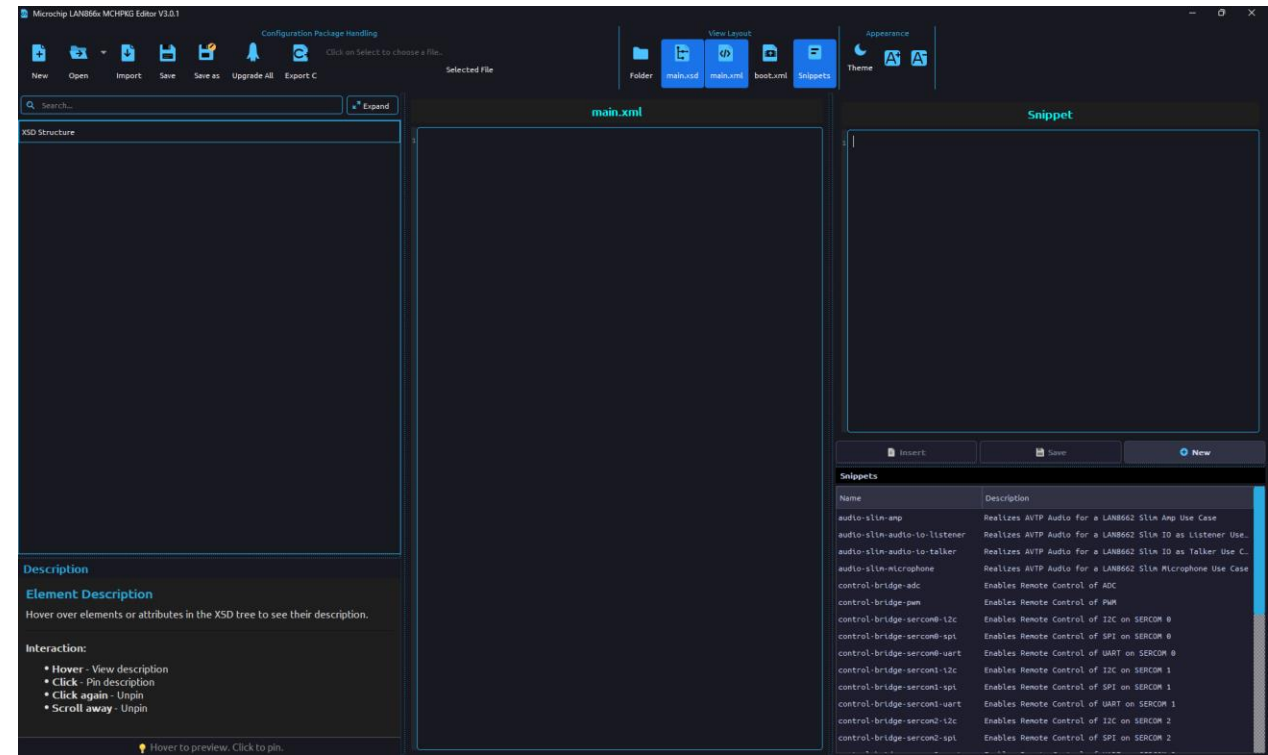
- **MPLAB Network Creator**
 - SAME GUI for config PHYs, Switches, Endpoints
 - Access to all available parameters
 - Selection of Peripheral Modules
 - Output of configuration
 - Config file for End-Of-Line programming
 - MDI/network (e.g. during OTA or in repair shop)



Remote Configurator

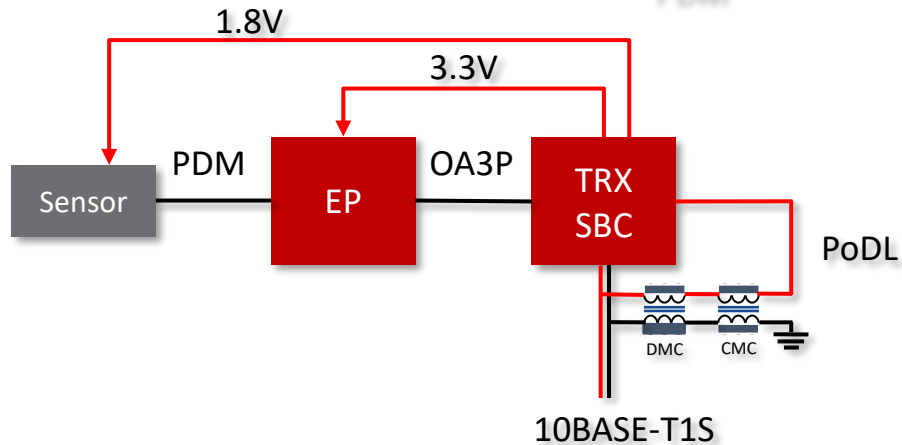
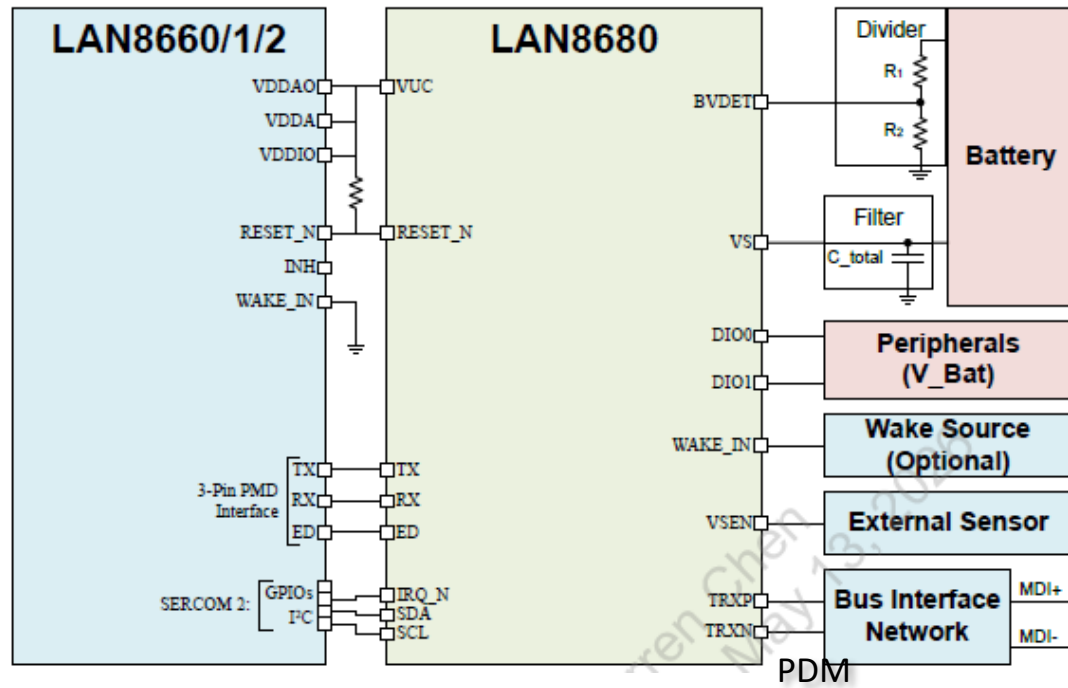
GUI:

- Configure over Network
- Uses config file from MPLAB Network Creator
- Display found EPs at once
- List all parameters
- Send to sleep / wakeup
- Windows / Linux support

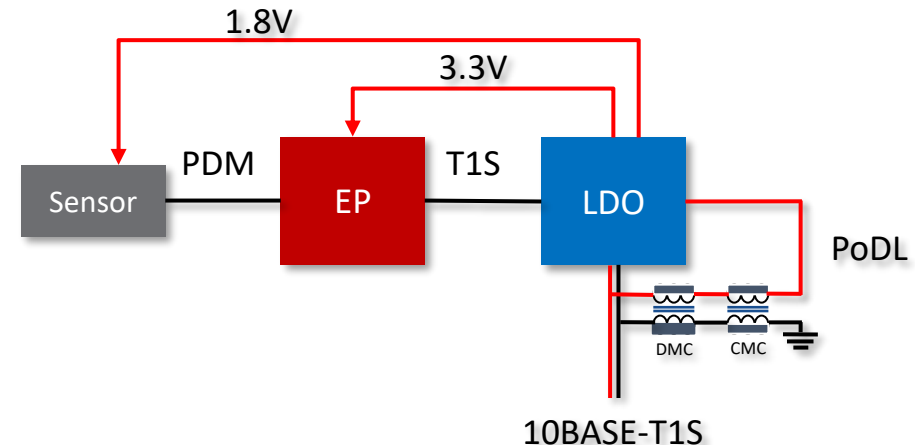
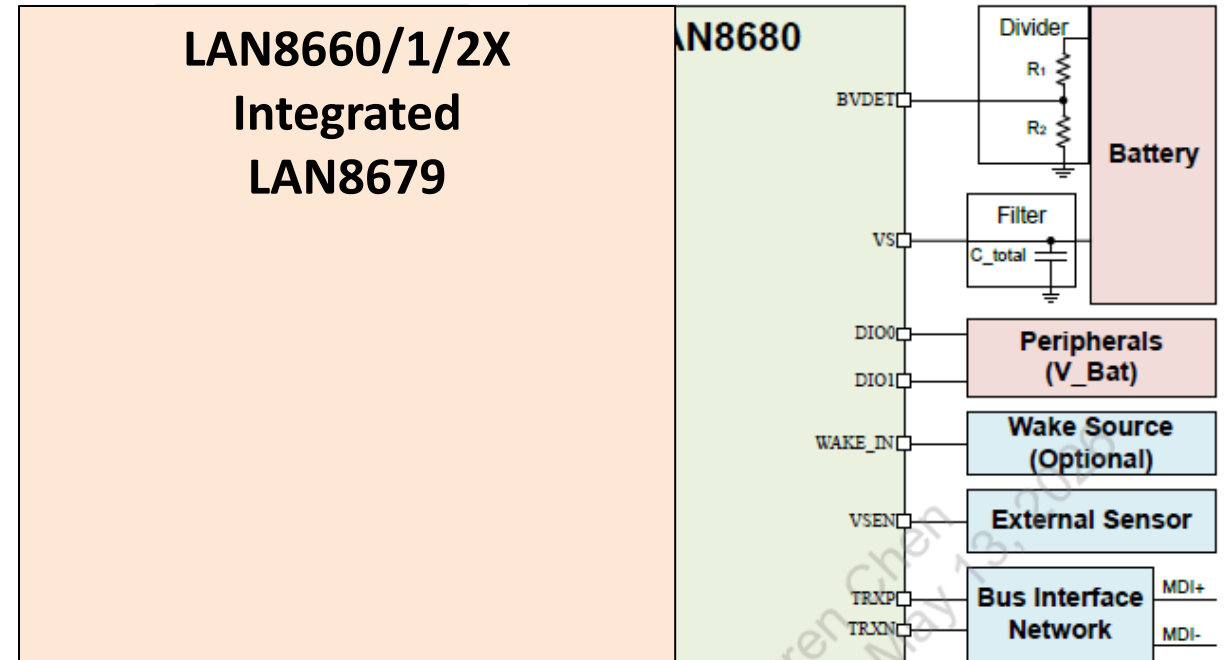


Microchip solutions

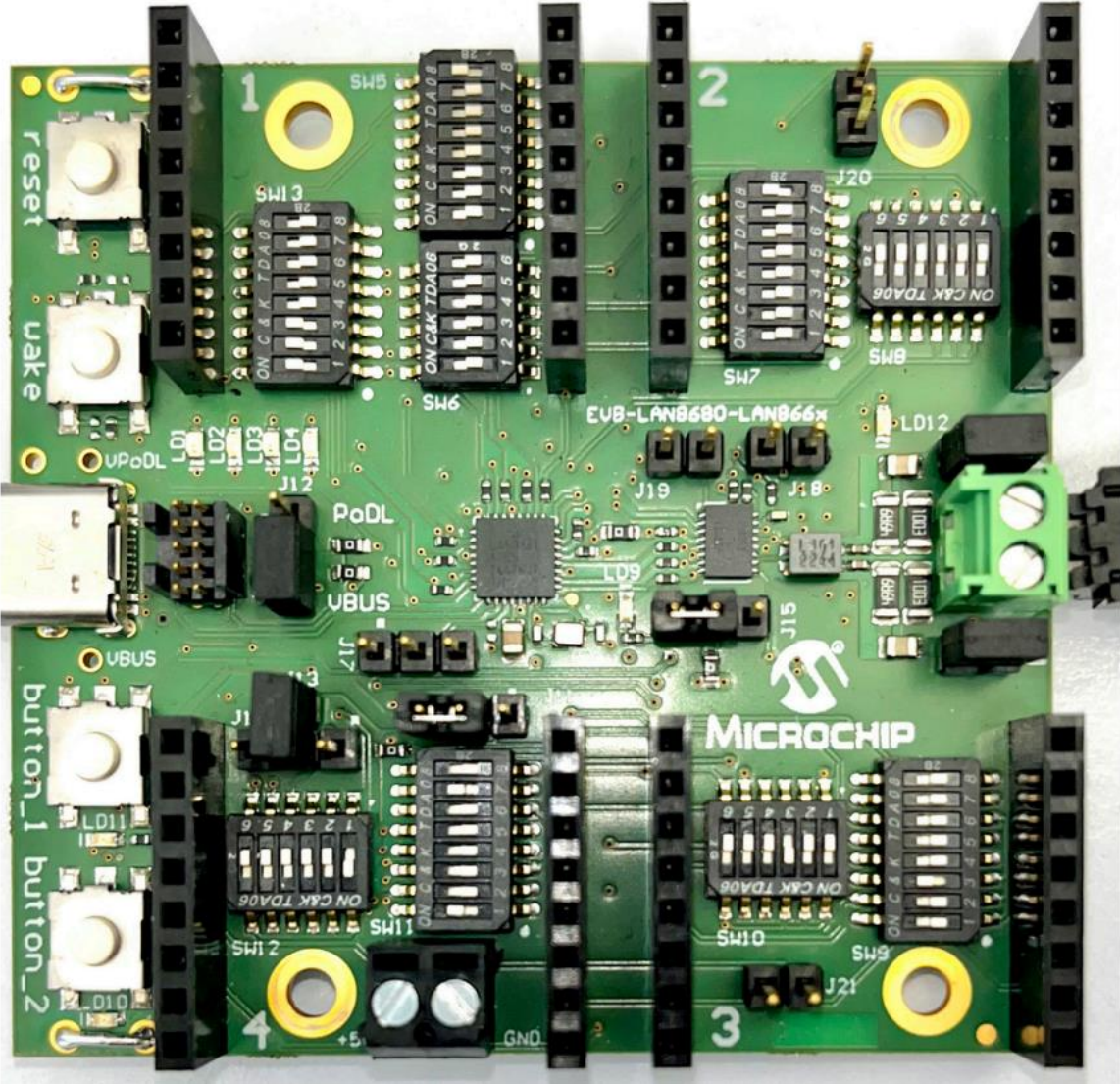
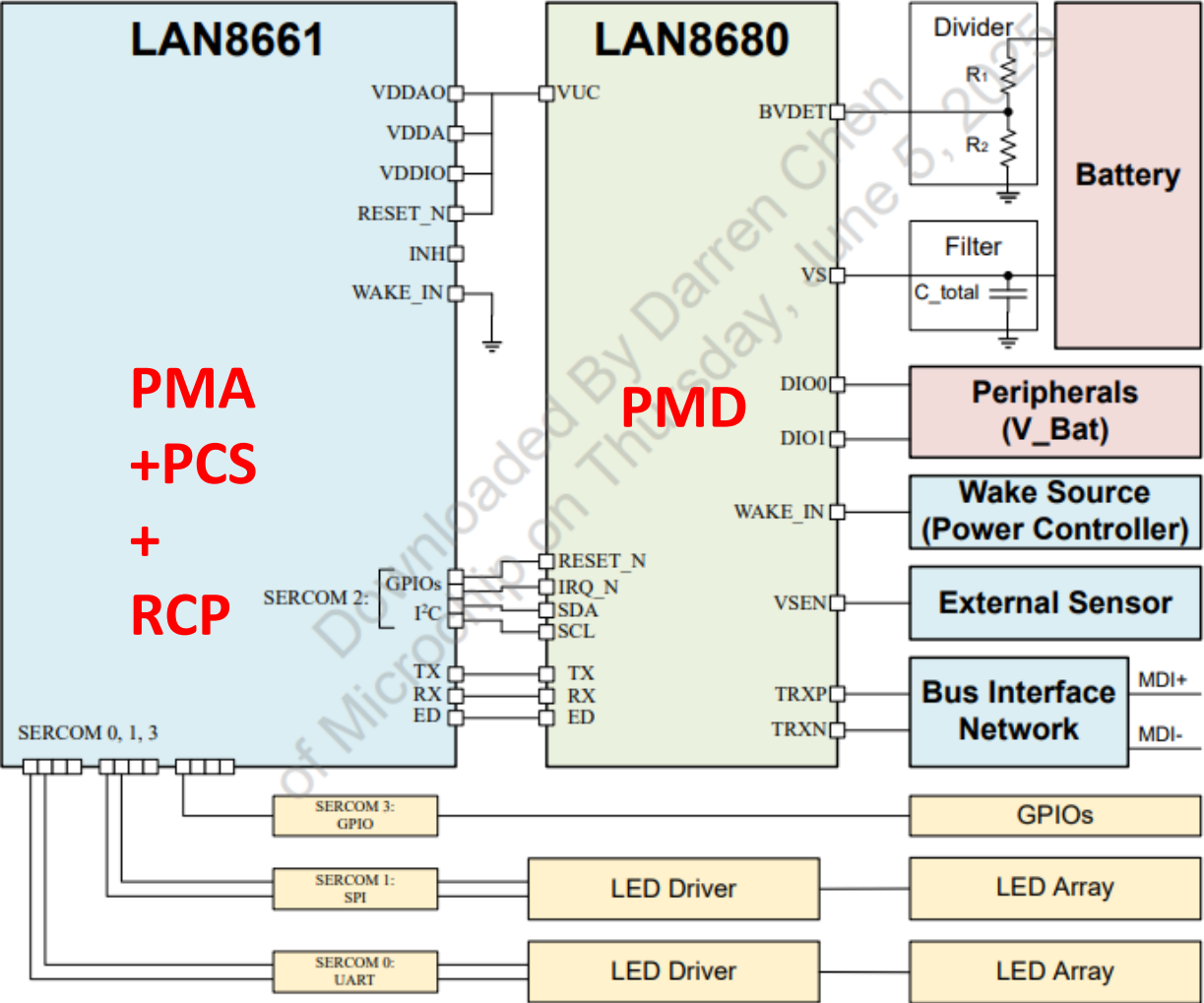
- Endpoint + PMD Transceiver



- Endpoint Integrated PMD Transceiver



MCHP RCP solution



Introduction to T1S Application

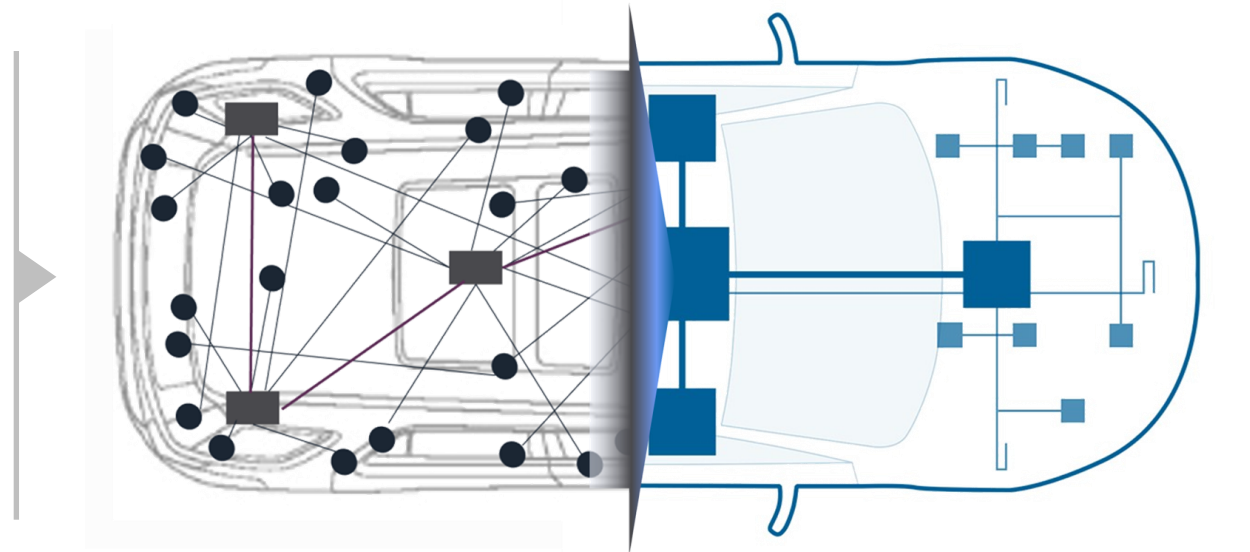
Evolving In Vehicle networking

Microchip's Long-Term Vision for In-Vehicle Networking (IVN)

*Today's Automotive
Networking Landscape...*



*Future Automotive
Networking Landscape...*



Low Speed & Backbone

OPEN
ALLIANCE
Ethernet

HPC & Storage

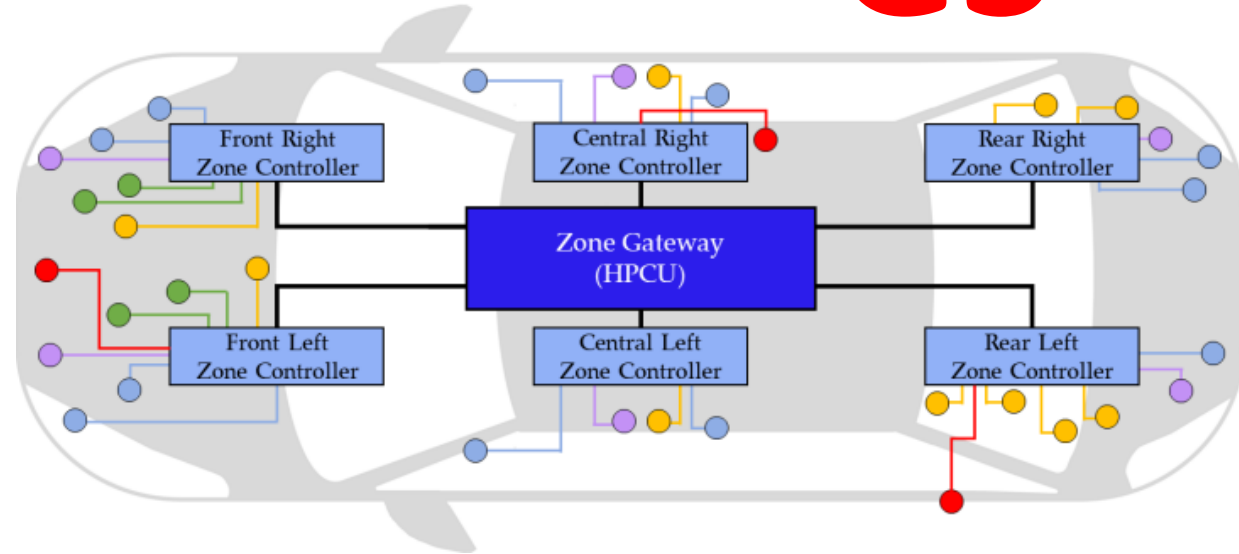
PCI
EXPRESS

Camera & Display

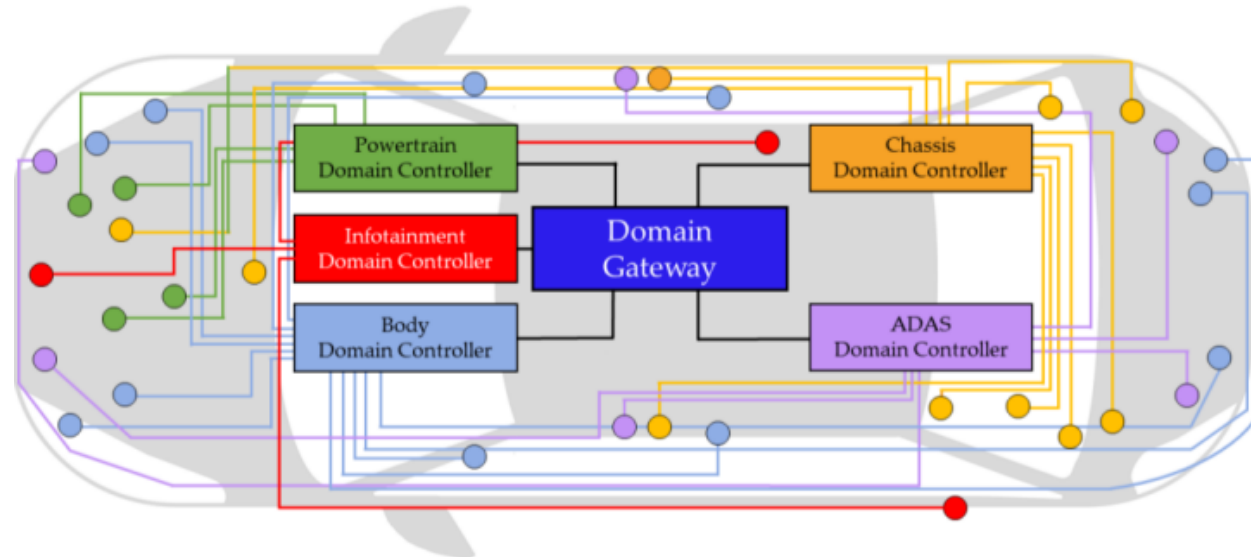
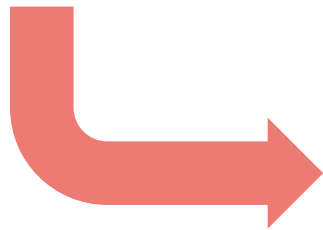


Generations of ECU connections

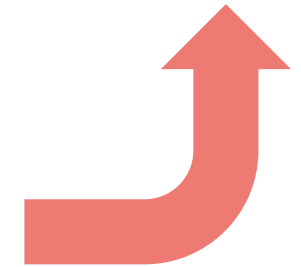
G3



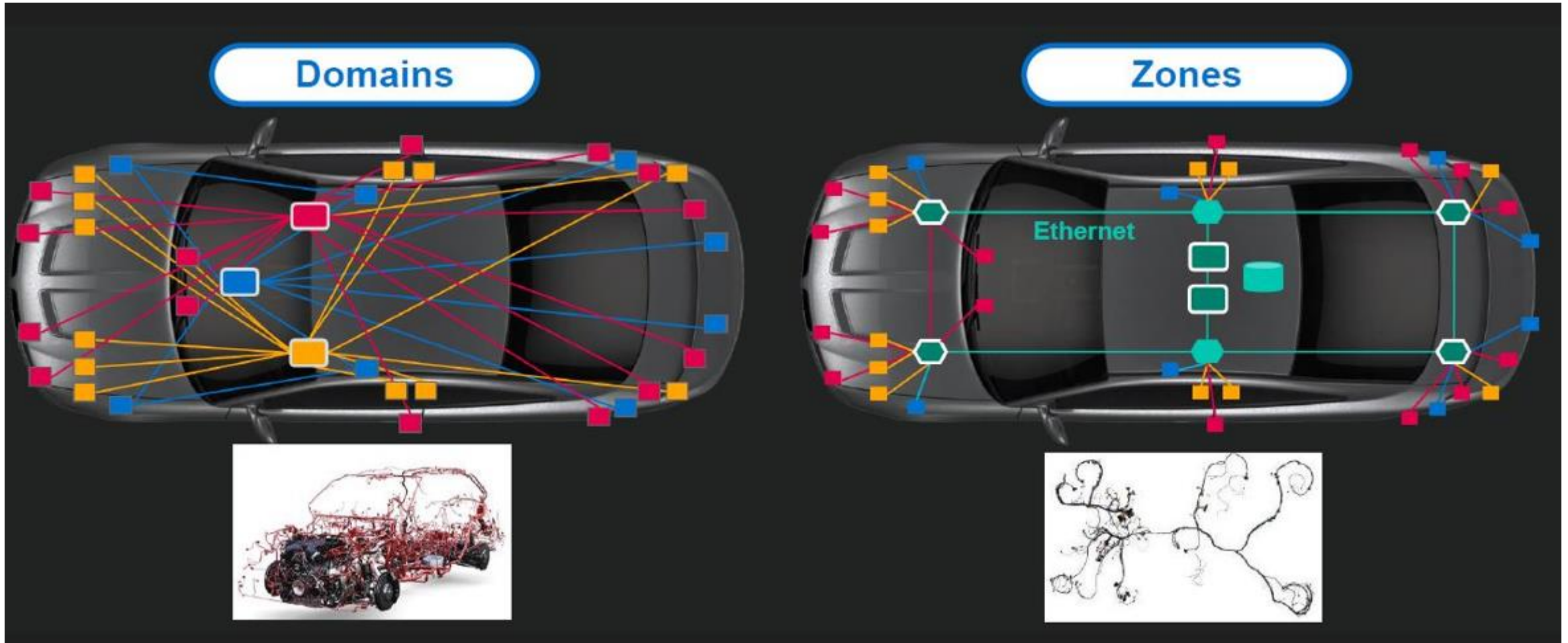
G1



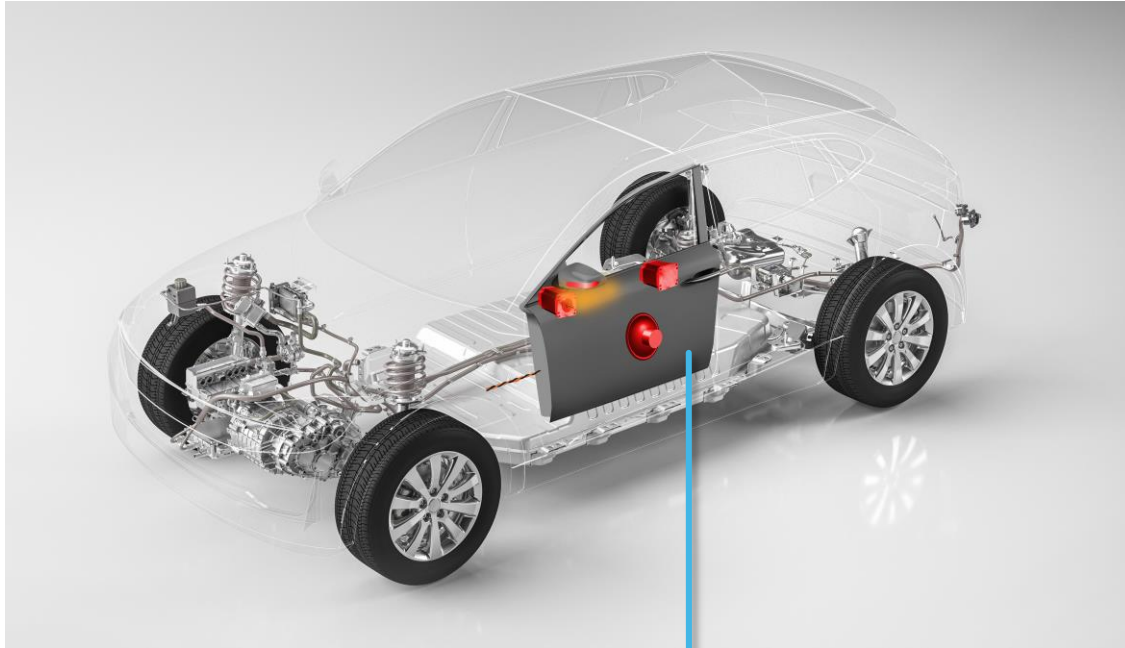
G2



Saving in cable and harness



Automotive Zone Architecture



Example: Door Zone

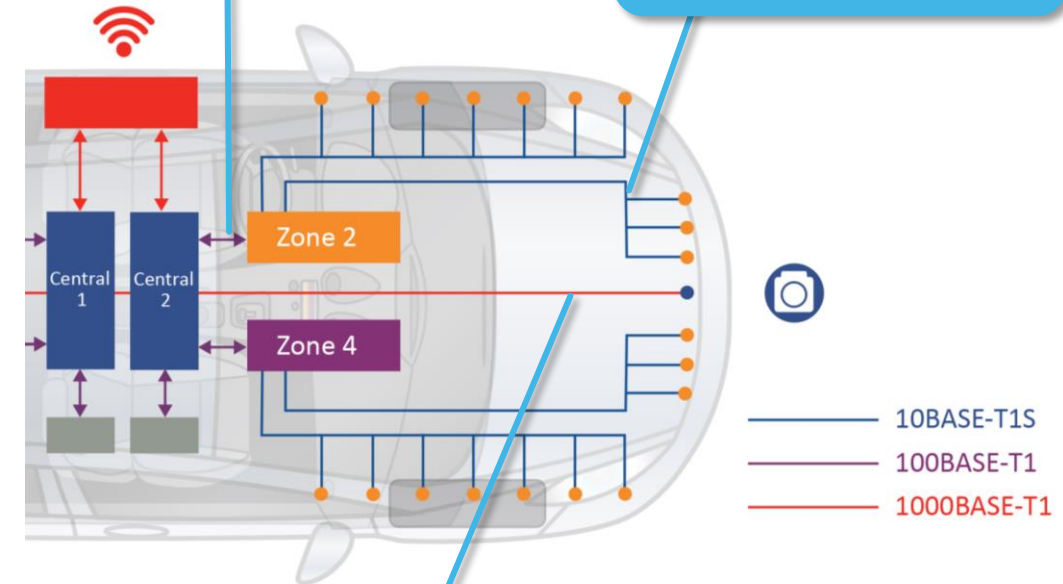
- Window lifter
- Mirror control
- Speakers
- Lock
- Ambient/interior light

10BASE-T1S bus:
Multiple applications
on a single cable!

Typical Zone Architecture using All-Ethernet

100/1000BASE-T1
connects zone and
central computer

10BASE-T1S
connects sensors and
actuators of a zone

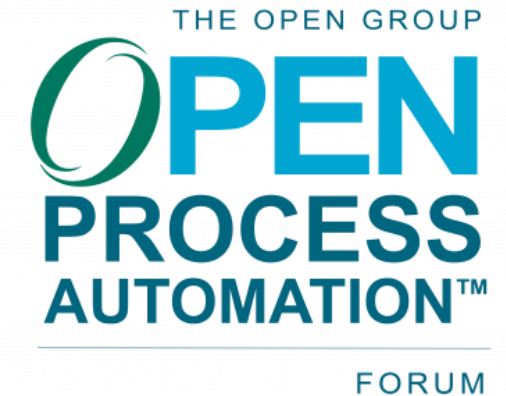


NGAUTO-T1 backbone
connects several central
computers and maybe cameras

DCS (Distributed control system)

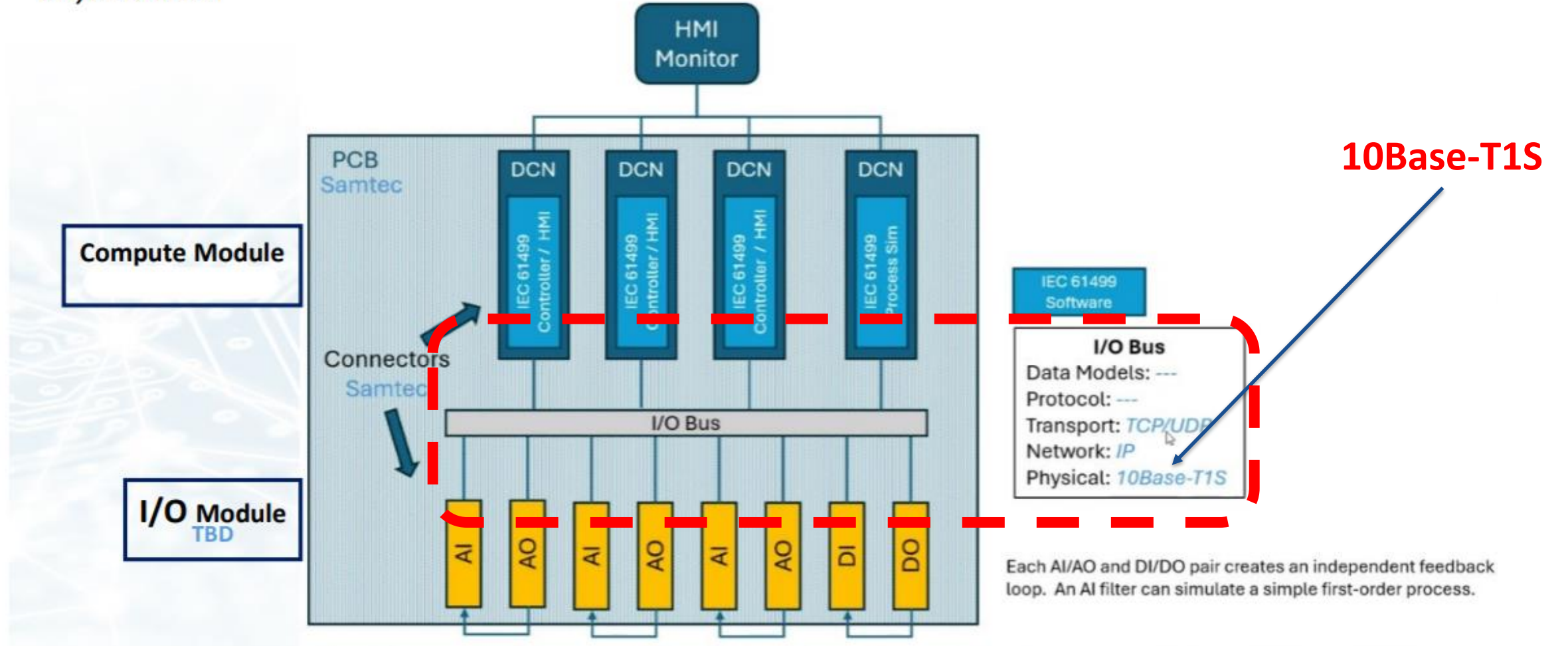


**Schneider Electric, Intel, and
Red Hat explore modern DCSs**

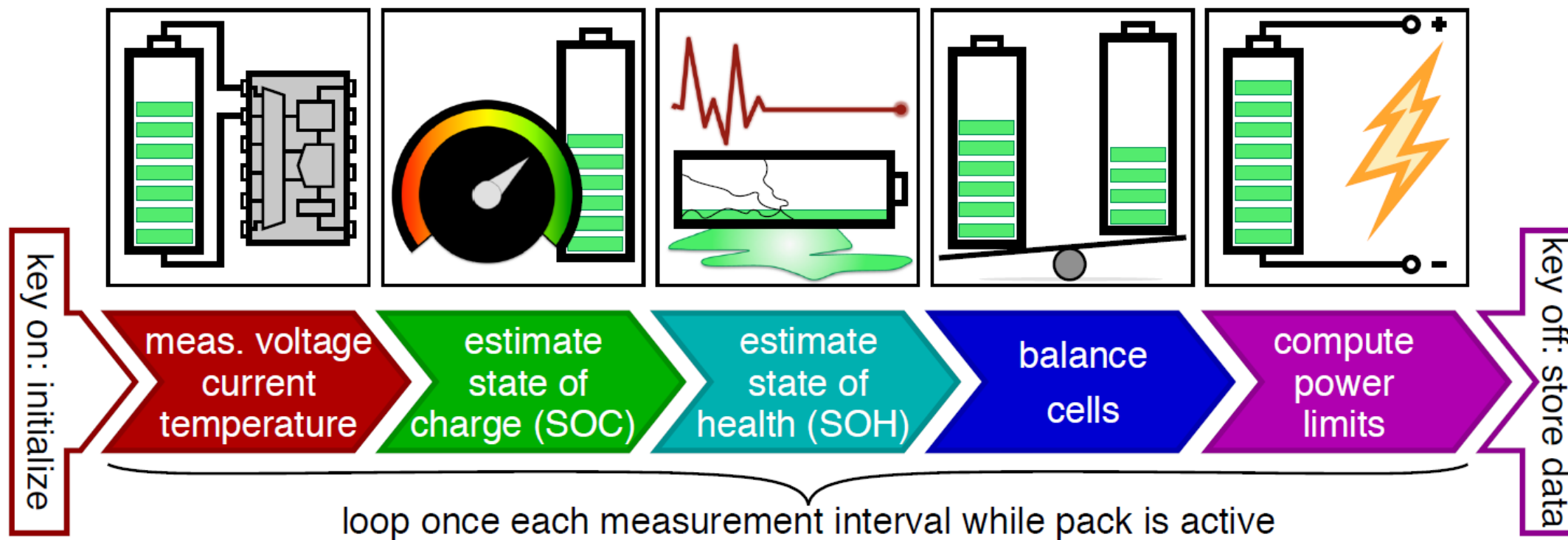


DCN : field terminal & controller connection

Physical View

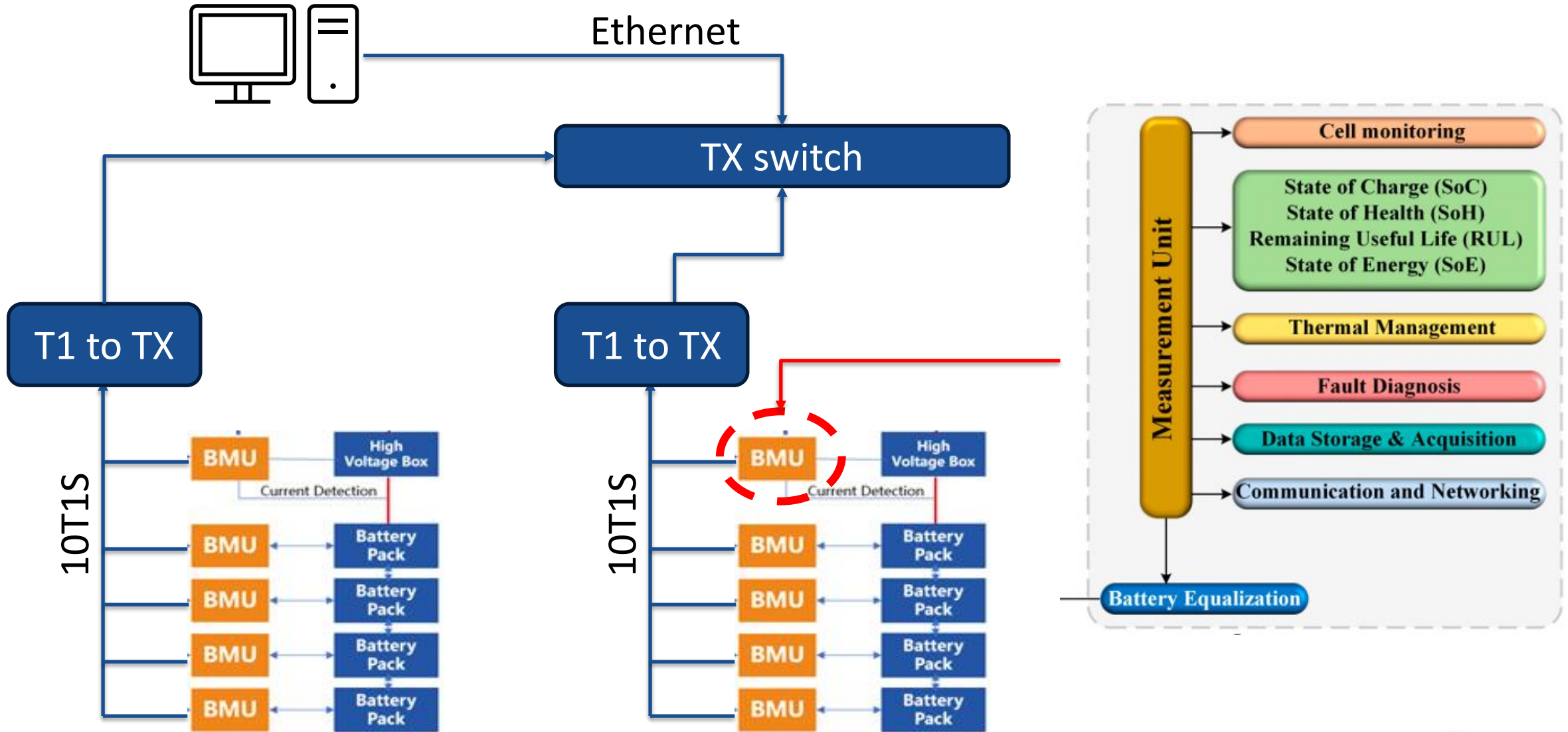


Software flow of a BMS



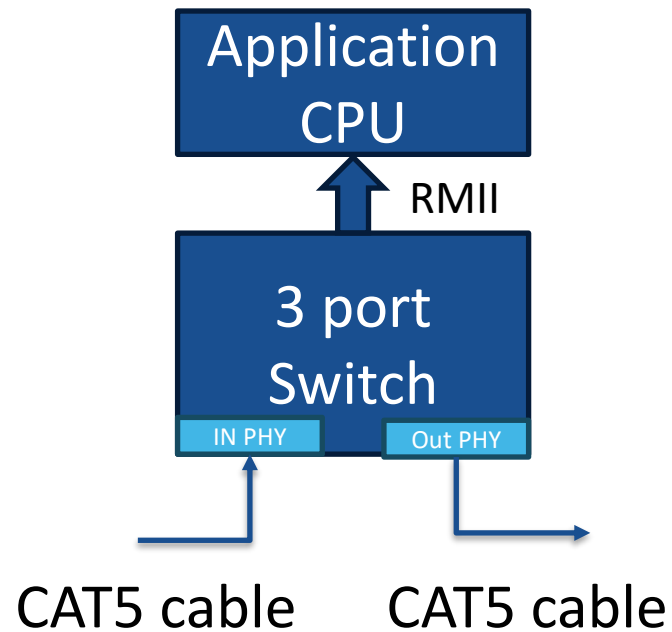
BMU based EMS (communication)

Management PC

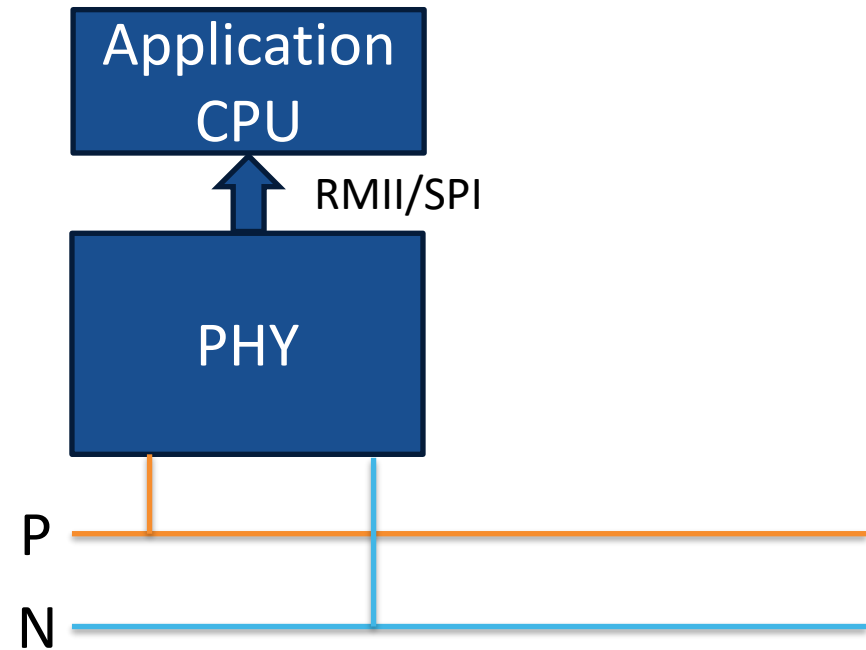


Daisy chain replacement

- 3 port switch implement

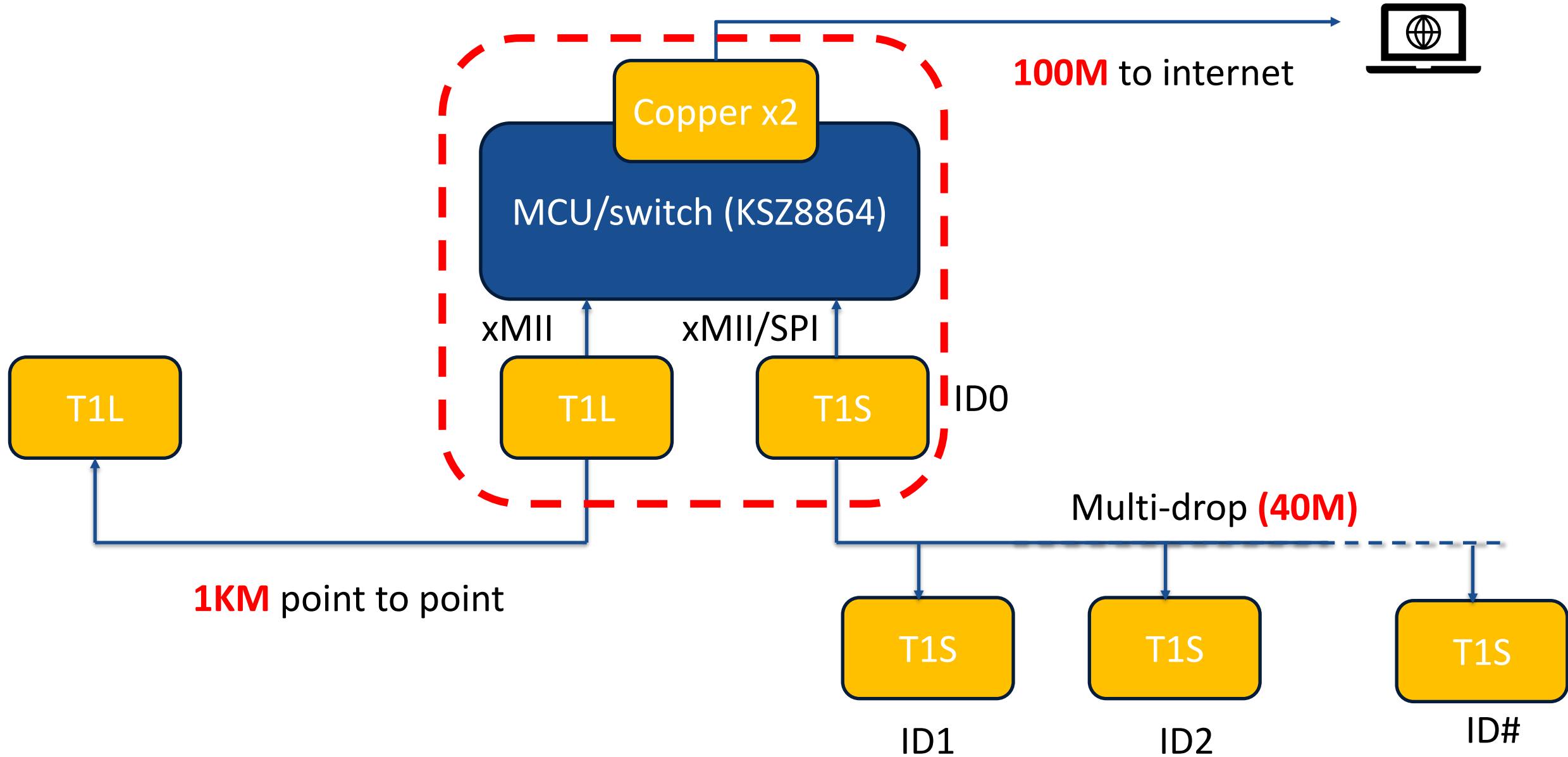


- 10Base-T1S PHY implement



Less cost

Full range process automation



SPE Product Details

Microchip SPE Solutions

- Flexible and scalable system solutions
- Extended cable reach
- Smallest packages
- Lowest power, wake/sleep
- Security support
- Functional safety ready
- Ambient temperatures up to 125 °C
- Comprehensive software & tool support

10BASE-T1S
PHY

10BASE-T1S
Controller

100BASE-T1
PHY

GigE
AVB/TSN
Switch

1000BASE-T1
PHY

10/100T1
AVB/TSN
Endpoint

10BASE-T1S Roadmap

Development

Sampling

Production

PHY (MII, RMII)

LAN8670
32 QFN
MII/RMII

LAN8671
24 QFN
RMII

Updated to
latest OA Spec
Q3 2025

MAC-PHY (SPI)

LAN8650
32 QFN
Ext. LDO

LAN8651
32 QFN
Int. LDO

LAN865x
Latest OA Spec
48V

Samples Q4 2026

Transceiver SBC (OA3p)

LAN8680
16 DFN
12V LDO

RTP Q3 2026

Transceiver (OA3p)

LAN8679
8 DFN

RTP Q2 2026

LAN8681
14 DFN
Wake/Sleep, 48V

Samples Q4 2026

Endpoint (OA3p)

LAN8660
32 QFN
Control EP, RCP

LAN8661
32 QFN
Lighting EP, RCP

LAN8662
32 QFN
Audio EP, RCP

LAN8660X/1X/2X
32 QFN
integrated TRX

RTP Q3 2026

MCU, Switch (OA3p)

PIC32CM
32 QFN
MCU

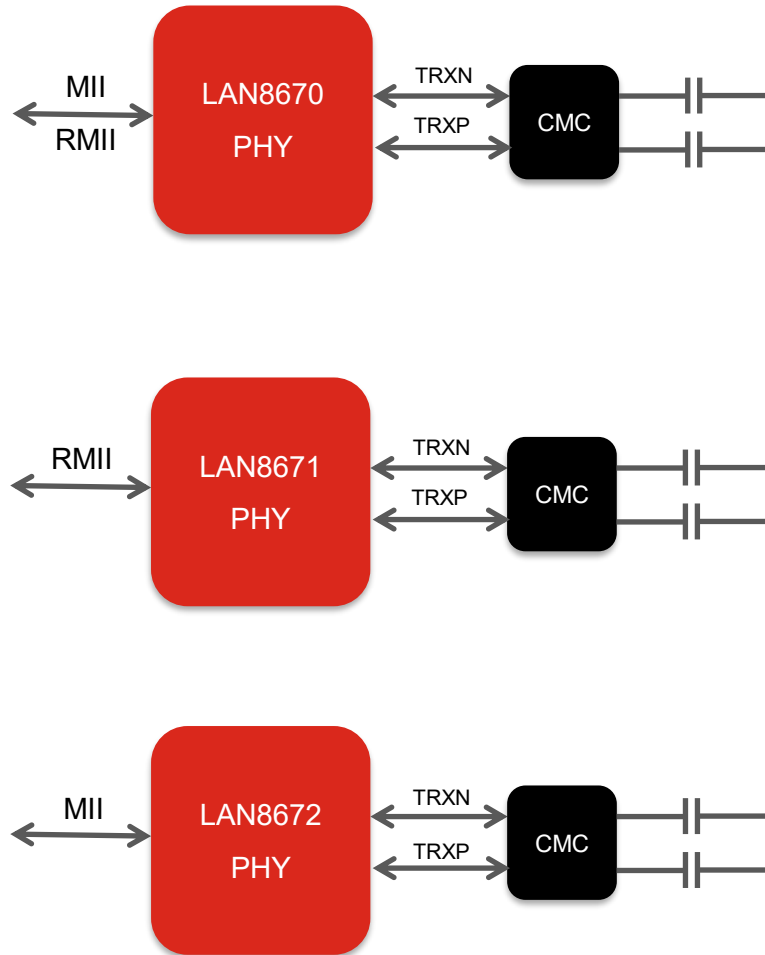
dsPIC33A
48 QFN
MCU

Hallberg
30G, 100/1000T1
Switch

Buzz
3G, 100T1/SGMII
Switch

LAN867x 10BASE-T1S PHYs

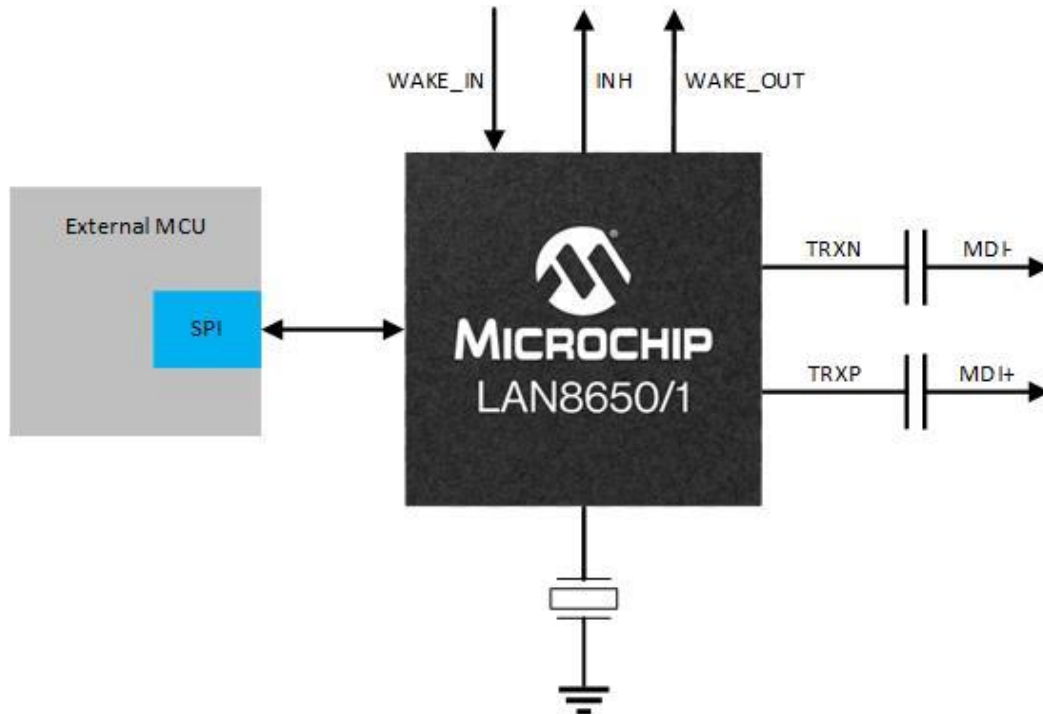
10BASE-T1S Ethernet PHY



- **Designed according to IEEE Std 802.3cg-2019™**
 - 10 Mbps over single balanced pair of conductors
 - Half-duplex point-to-point segments up to at least 15 m
 - Half-duplex multidrop segments up to at least 25 m with up to at least 8 nodes
- **High bandwidth utilization and latency optimization**
 - Physical Layer Collision Avoidance (PLCA)
 - Multiple PLCA IDs per node
 - Burst mode
- **Time Sensitive Networking (TSN) support**
 - Enables high-precision clock recovery with ultra-low jitter (IEEE Std 802.1AS™ / IEEE 1588™)
- **Standard interface**
 - MII / RMII
 - SMI (MDIO, MDC) for rapid register access
- **Ultra-low power sleep mode**
 - Wake on network activity or input pin
- **Functional Safety Ready**
 - SGS TUV Certified ISO 26262 ASIL B ready
 - FMEDA, Safety Manual,
- **Small footprint**
 - 24 & 32 & 36-pin
 - VQFN package with wettable flanks
- **AEC-Q100 Grade 1 (-40 to +125C)**

LAN865x 10BASE-T1S PHYs

10BASE-T1S MAC-PHY Ethernet Controller with SPI

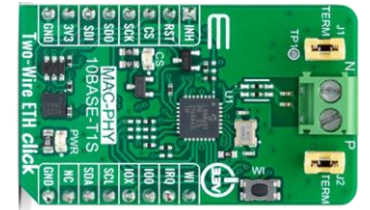
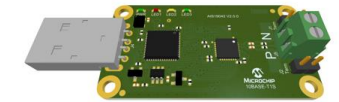
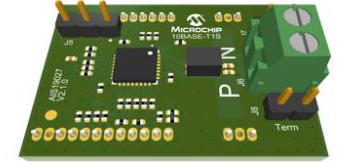


- **Designed according to IEEE Std 802.3cg-2019™**
 - 10 Mbps over single balanced pair of conductors
 - Half-duplex point-to-point segments up to at least 15 m
 - Half-duplex multidrop segments up to at least 25 m with up to at least 8 nodes
- **High bandwidth utilization and latency optimization**
 - Physical Layer Collision Avoidance (PLCA)
 - Multiple PCLA IDs per node
 - Burst mode
- **Standard interface**
 - Serial Peripheral Interface (SPI)
 - Designed to the OPEN Alliance 10BASE-T1x MAC-PHY Serial Interface specification, V1.1
- **Ultra-low power sleep mode**
 - Wake on network activity or input pin
- **Functional Safety Ready**
 - SGS TUV Certified ISO 26262 ASIL B ready
 - FMEDA, Safety Manual
- **Small footprint**
 - 32-pin (5x5 mm)
 - VQFN package with wettable flanks
- **AEC-Q100 Grade 1 (-40 to +125C)**

10BASE-T1S Evaluation Boards

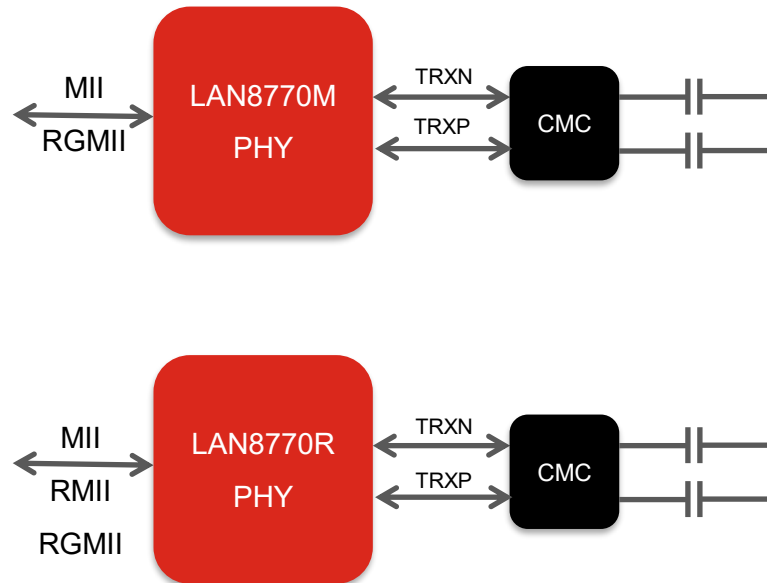
Evaluation Boards and MPLAB® Harmony

- **10BASE-T1S PHY RMII Evaluation Board EVB-LAN8670-RMII**
 - Fits to many Microchip MCU boards
 - Small form factor
 - Available from MicrochipDirect (EV06P90A)
- **10BASE-T1S PHY USB Evaluation Board EVB-LAN8670-USB**
 - Makes every USB host a potential 10BASE-T1S node
 - Drivers for Linux and Windows
 - Available from MicrochipDirect (EV08L38A)
- **10BASE-T1S MAC-PHY SPI Evaluation Board**
 - Enables Ethernet communication for various Microchip evaluation kits that support click boards™ and the SPI interface
 - Available from MikroE (www.mikroe.com), Two-Wire Eth Click, MIKROE-5543)
- **MPLAB® Code Configurator Development Framework**
 - Integration into development framework for Microchip microcontrollers and microprocessors



LAN877x 100BASE-T1 PHY

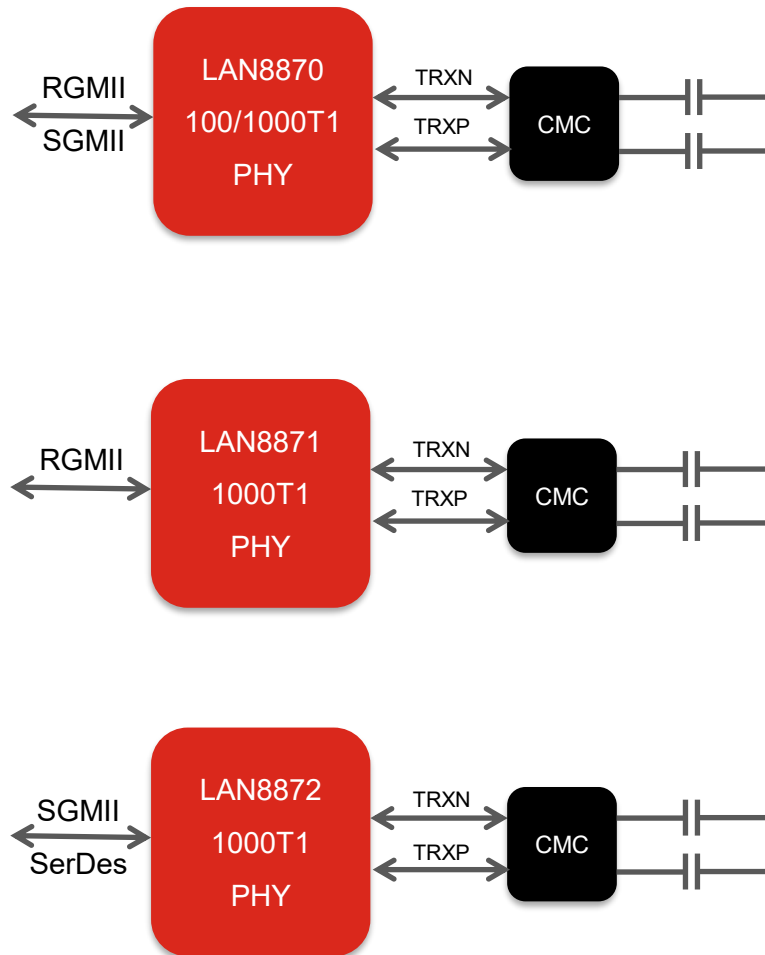
100BASE-T1 Ethernet PHY



- **Fully compliant IEEE802.3bw Single-Port PHY**
- **Leading Energy Efficiency**
 - Low Power ~130mW operation
 - TC10 Wake up / Sleep (< 15uA)
- **Reduced BOM**
 - Fully integrated line termination and filtering
 - RGMII, RMII, MII w/ FlexPWR®
 - 125MHz RGMII output
- **Enhanced LinkMD™ diagnostics**
 - OPEN Alliance (TC1) Advanced diagnostics compliant
 - SQI counter w/ MSE, peak, threshold interrupt
- **Functional Safety Ready**
 - SGS TUV Certified ISO 26262 ASIL B ready
 - FMEDA, Safety Manual,
- **Smallest T1 PHY**
 - 32-Pin & 36-Pin Wettable QFN (5x5mm, 6x6mm)
- **AEC-Q100 Grade 1 (-40 to +125C)**

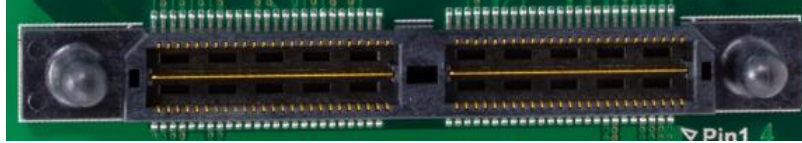
LAN887x 1000BASE-T1 PHY

1000BASE-T1 Ethernet PHY

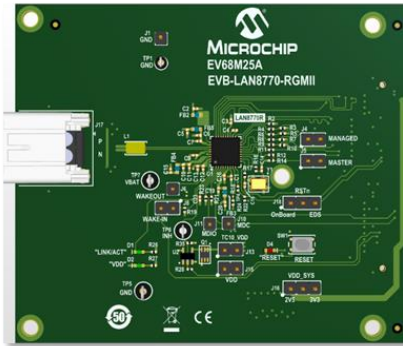


- **Fully compliant IEEE802.3bw/bp Single-Port PHY**
 - Force or auto-negotiation mode
- **IEEE 1588 / .1AS Time synchronization support**
 - 8ns jitter accuracy
- **Cable reach**
 - Type A (15m) and Type B (40m)
 - Extended reach (>40m*)
- **TC10 Sleep and Wakeup support**
 - <20uA standby power consumption
- **Enhanced Diagnostics**
 - Cable diagnostics, Signal Quality Indicator
 - BIST, link, error, temperature, voltage and clock monitoring
- **Functional Safety Ready**
- **48-Pin (7mm x 7mm) package**
- **AEC-Q100 Grade 2 (-40degC to +105degC)**

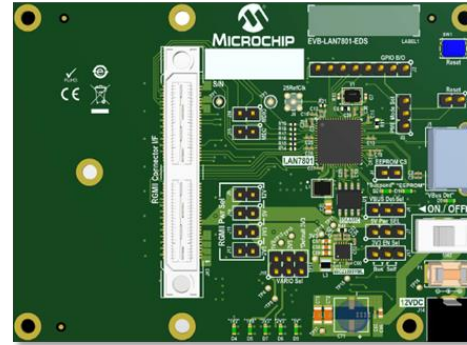
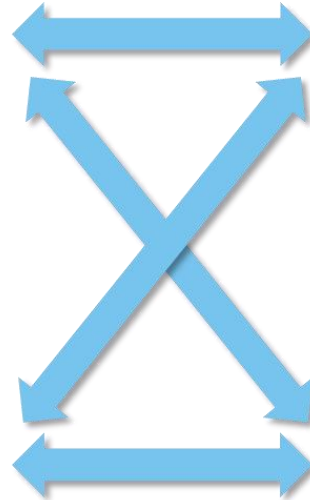
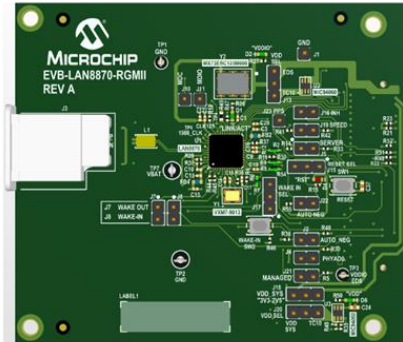
100/1000 BASE-T1 Plug-in Bridge Solutions



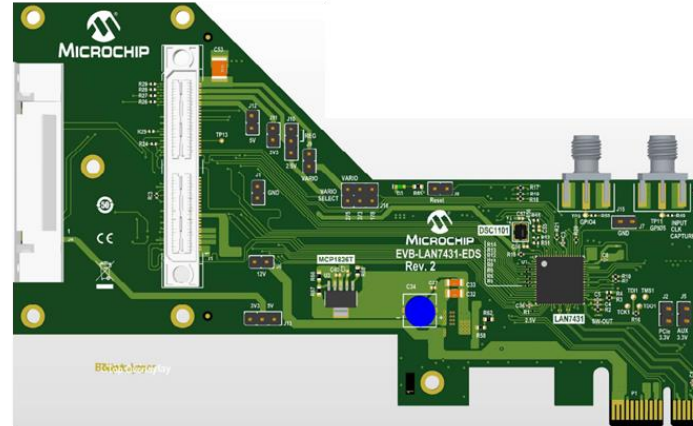
100BASE-T1
LAN8770



1000BASE-T1
LAN8870



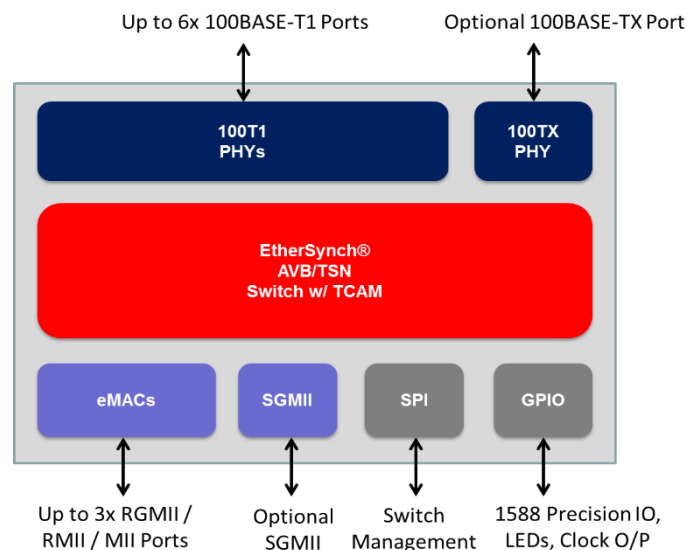
LAN7801



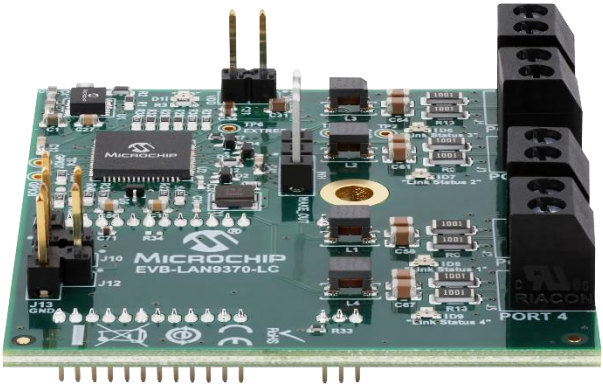
LAN7431



LAN937x 100T1 GigE Switch



Part Number	T1 PHY Ports	TX PHY Ports	RGMII/ RMII/MII Ports	SGMII Ports	Temperature	Package
LAN9370	4	0	1	0	-40°C +105°C	64-QFN
LAN9371	3	1	2	0	-40°C +105°C	128-TQFP
LAN9372	5	1	2	0	-40°C +105°C	128-TQFP
LAN9373	5	0	2	1	-40°C +105°C	128-TQFP
LAN9374	6	0	2	0	-40°C +105°C	128-TQFP



- 5 to 8-Port AVB/TSN Switch
- Cascade for up to 14-Ports
- Software and pin compatible
- TC11 compliant
- TC10 wake up and sleep
- AEC-Q100 Grade 2

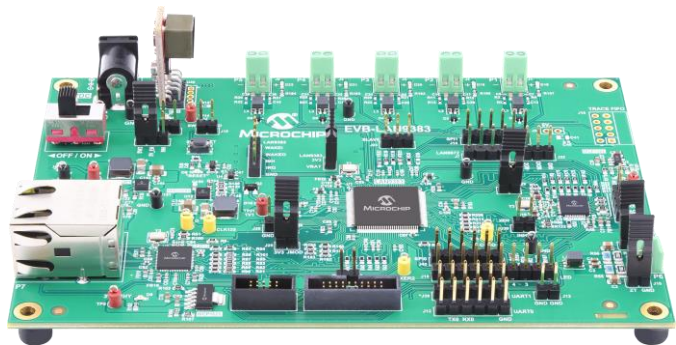
Advanced AVB/TSN support

- 802.1AS Time Sync
- 802.1Qat Stream Reservation
- 802.1Qav Shaper
- 802.1Qbv Scheduler
- 802.1Qci Ingress Policing

EVB-LAN9370

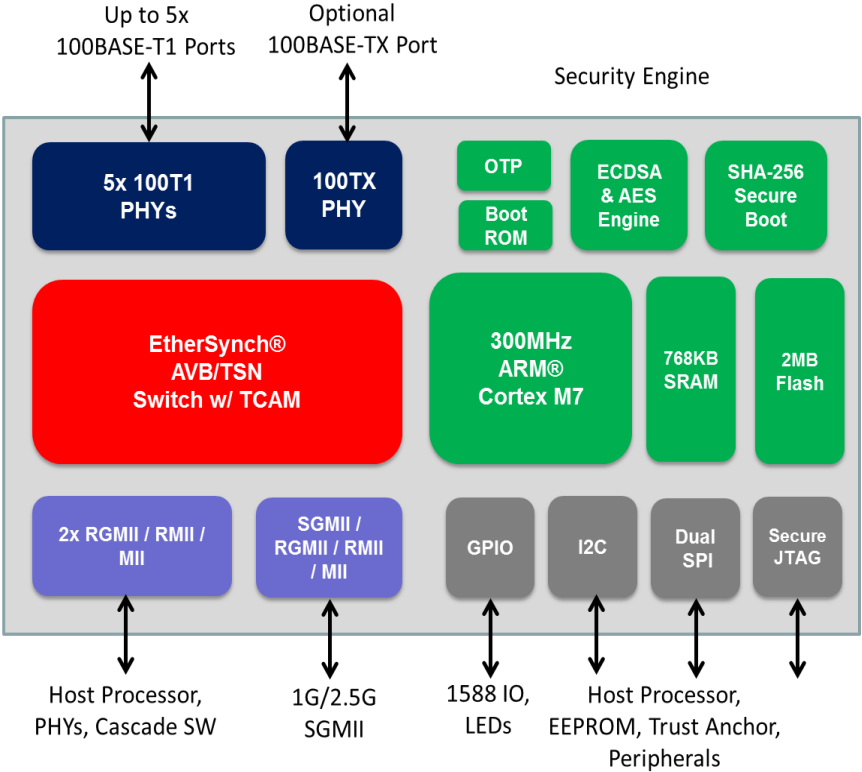
- Partner: SAME70 XULT, SAMA5D3-EDS, Audio End Point
- Software: Autosar, FreeRTOS, Linux

LAN938x 100T1 GigE Switch w/ CPU



Part Number	T1 PHY Ports	TX PHY Ports	RGMII / RMII / MII Ports	SGMII OR RGMII MUX Port	Temperature	Package
LAN9384	5	1	0	1	-40 +105	128-TQFP
LAN9383	5	0	1	1	-40 +105	128-TQFP
LAN9382	4	1	1	1	-40 +105	128-TQFP
LAN9381	4	0	2	1	-40 +105	128-TQFP

- 7-Port AVB/TSN Switch
- 300MHz Cortex M7
- Integrated Flash & ECC RAM
- 1G/2.5G SGMII Uplink
- Advanced TSN support
 - 802.1AS-2020 Time Sync
 - 802.1Qbv Scheduler
 - 802.1Qci Ingress Policing
 - 802.1CB Redundancy
- Advanced Security Engine
- TC10 wake up and sleep
- Functional Safety ready
- AEC-Q100 Grade 2

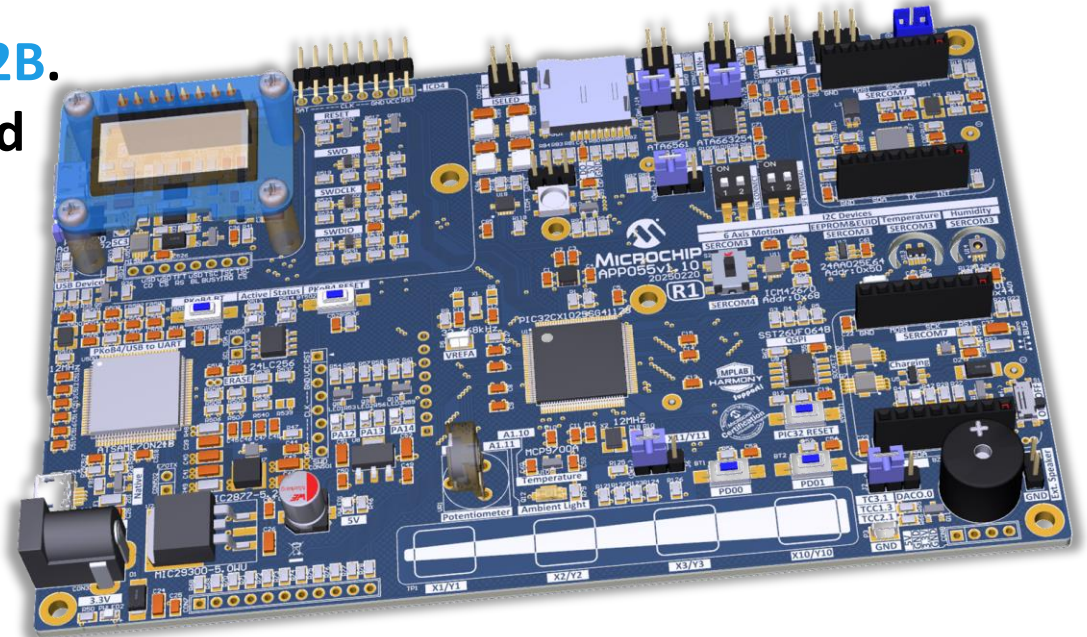


PIC32CX-SG EVB

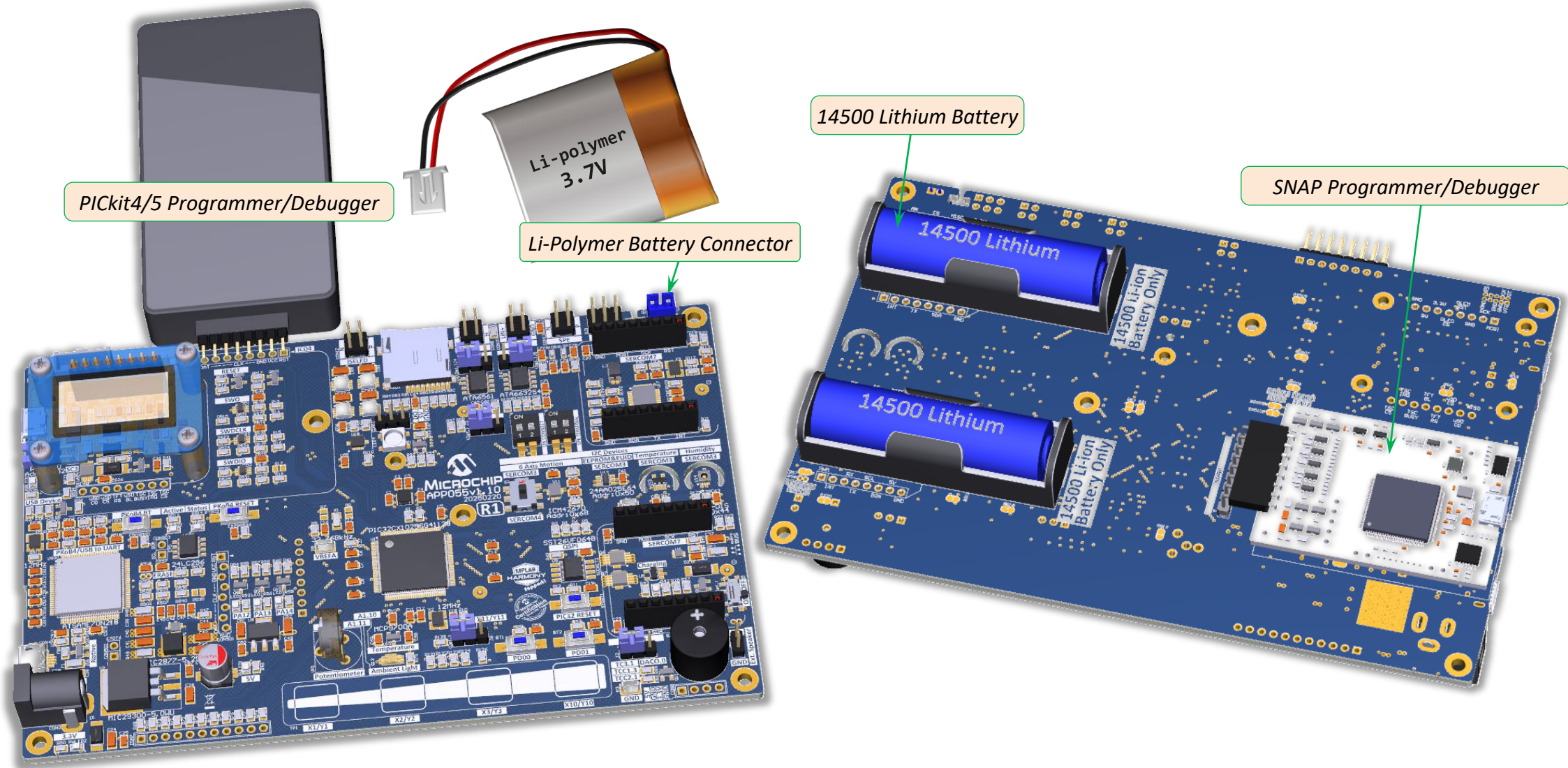
APP055 v1.10R1 Introduction

APP055 v1.10_{R1} Features

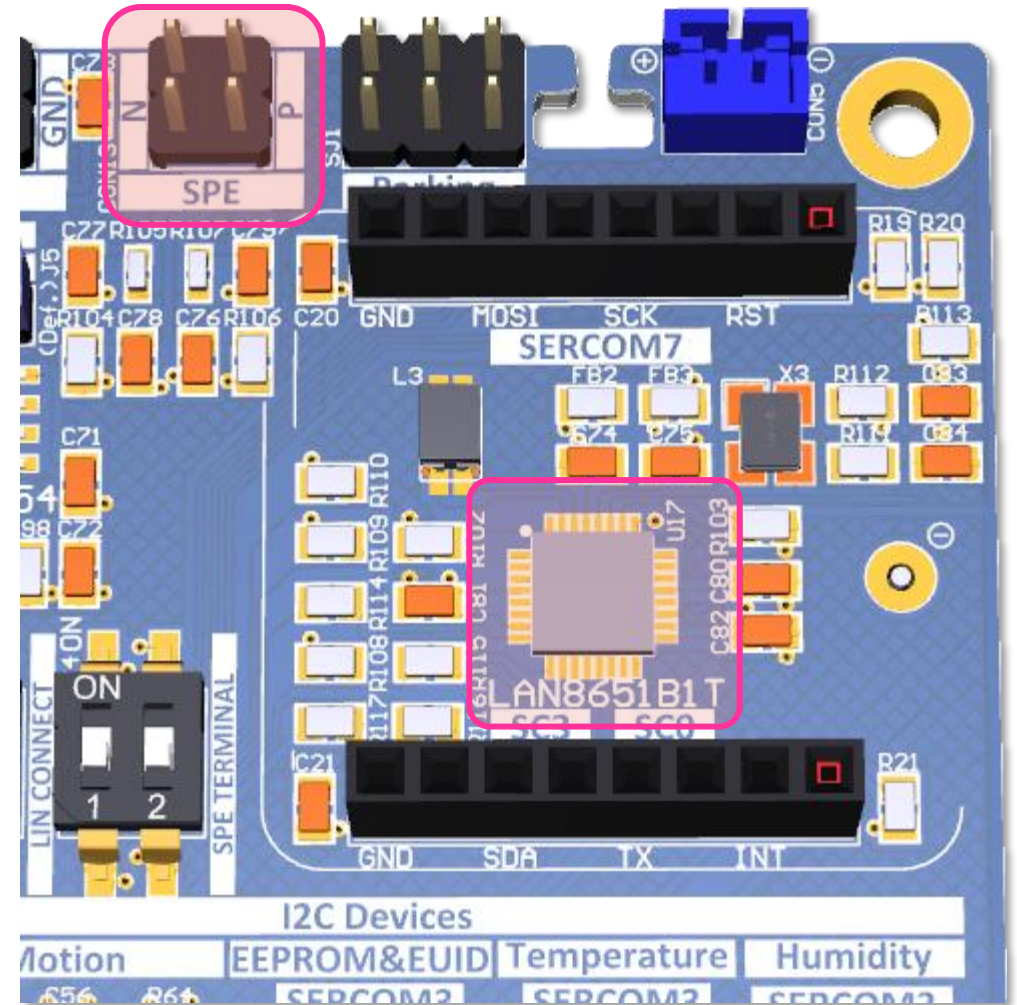
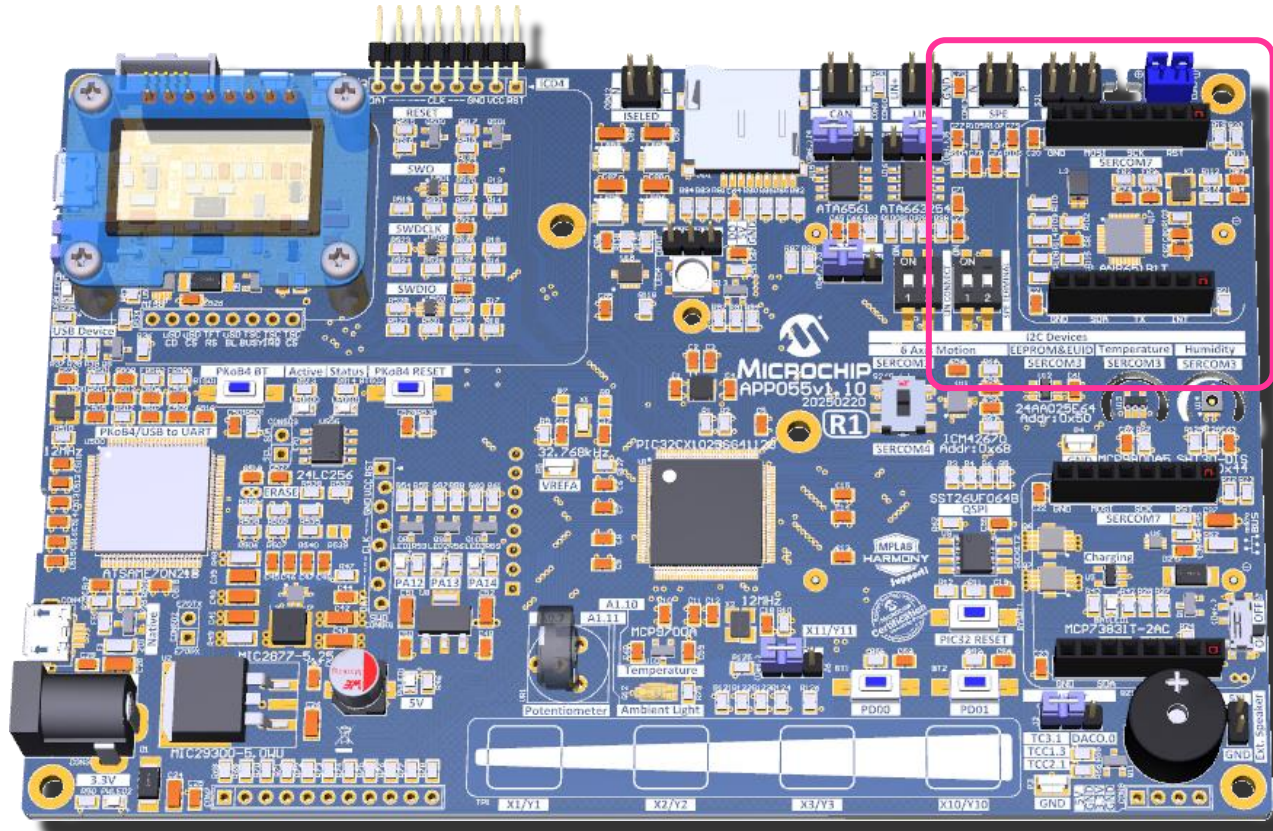
- **PIC32CX1025SG41128** 32Bit Arm Cortex M4F Microcontroller with **FPU**.
- **Built-in PKoB4 Programmer/Debugger with USB CDC Virtual COM.**
- **Multiple Power Source** and support Li-on battery with battery charger.
- **ISELED, Single Pair Ethernet, CAN-FD, LIN** support ready.
- **Capacitive QTouch®** Slider & Buttons with Driven Shield.
- **mikroBUS™** Click Board Socket ready.
- **Digital IO : 2 x Buttons** and **3 x LEDs** and **WS2812B**.
- **Analog : Potentiometer, Temperature Sensor** and **Ambient Light Sensor**.
- **I2C : Temperature, EEPROM, 6 Axis Motion** and **Humidity Sensor**.
- **SPI : OLED (128 x 64, SSD1306 Compliant)**
- **QSPI : Flash Memory.**
- **SDHC : MicroSD Socket.**
- **PWM, DAC : Buzzer.**



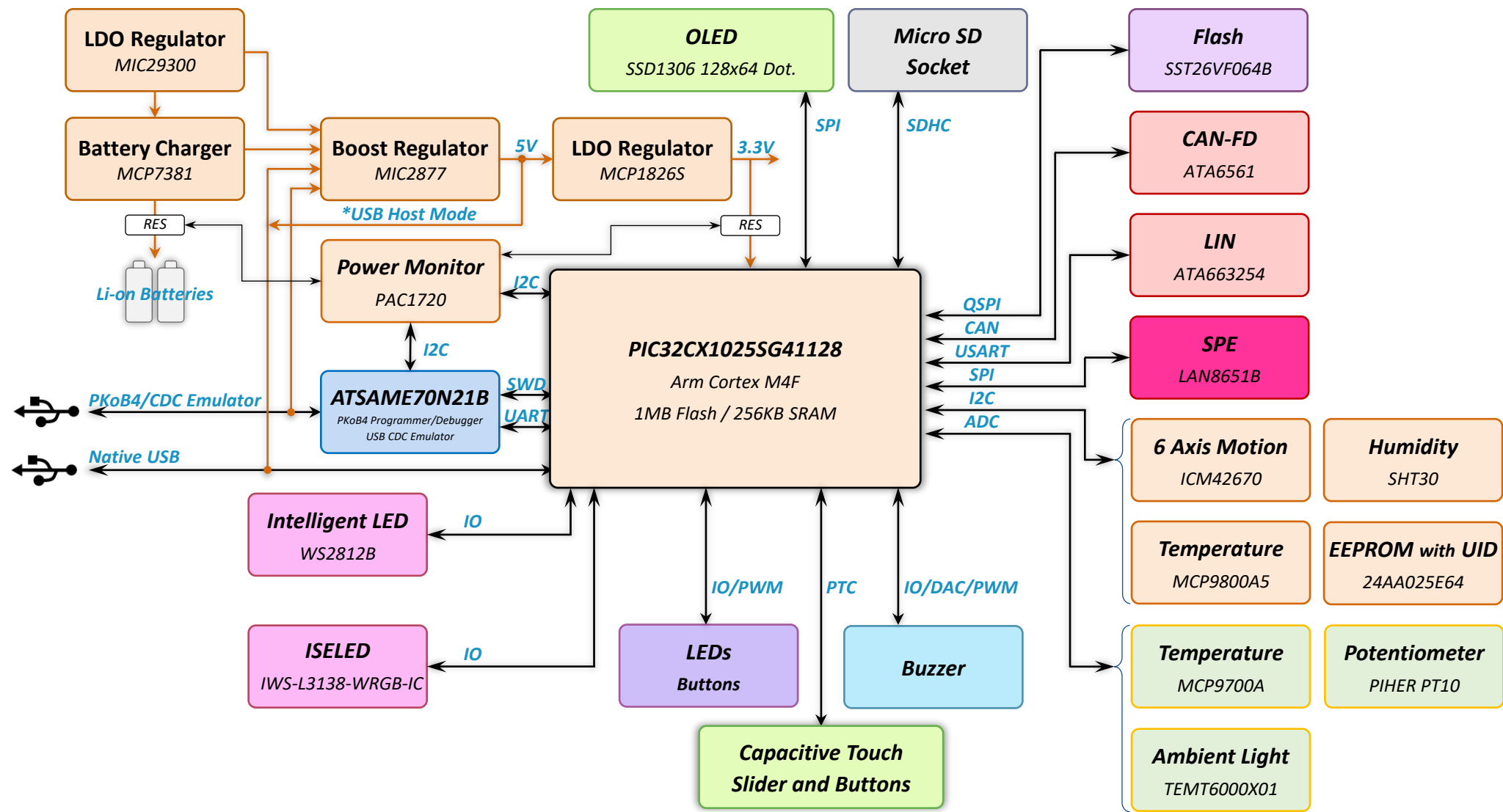
APP055 v1.10_{R1} Accessories



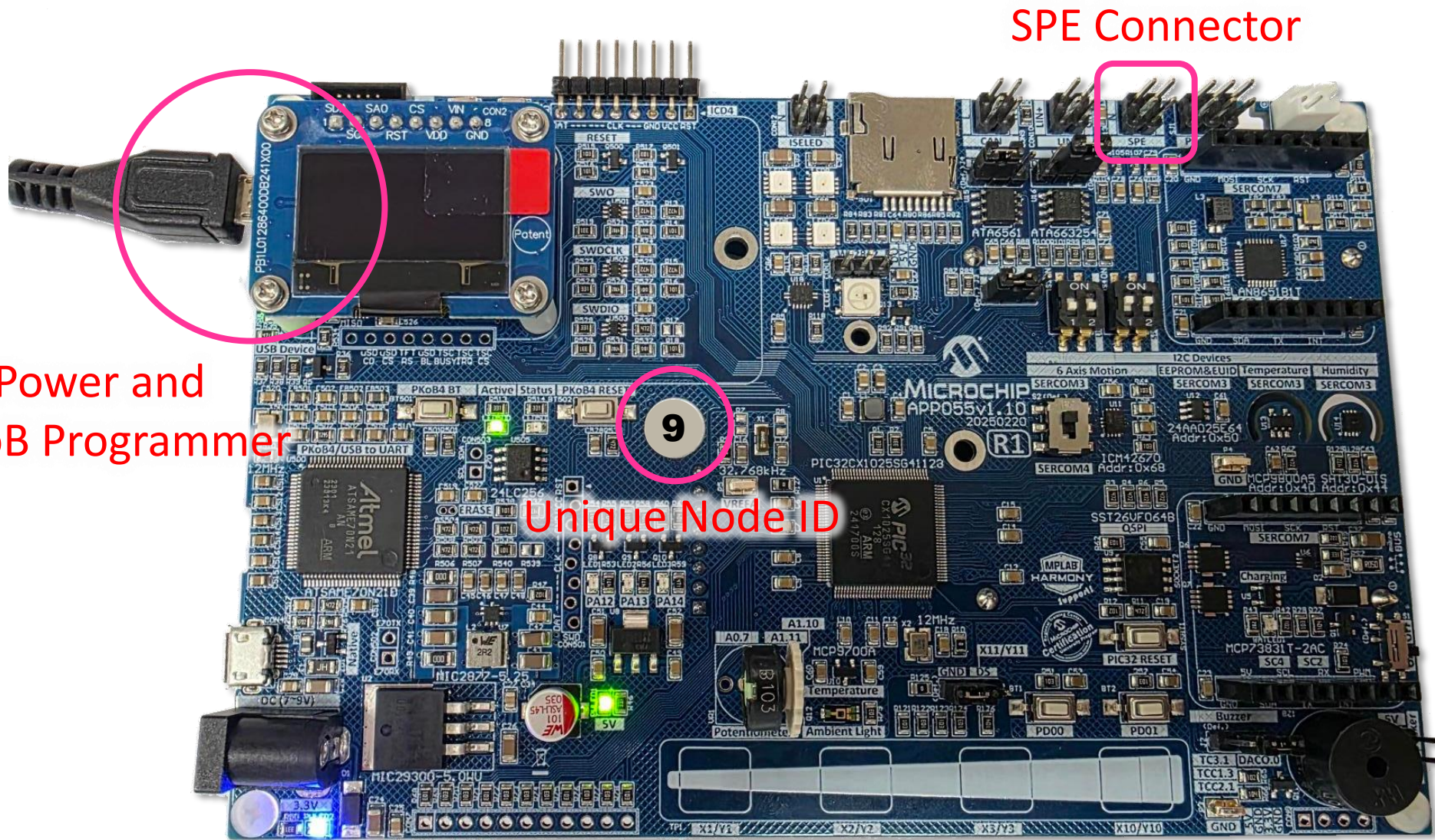
Single Pair Ethernet (SPE) LAN8651- 10BASE-T1S Ethernet



APP055 v1.10_{R1} Block Diagram



APP055 v1.10_{R1} Connection



MCC Harmony Framework

The Software Framework for MCC

Software Request In this Course

When the versions of the software and the framework differ from the created project, issues may occur. Therefore, it is recommended to use the following setting in this course.

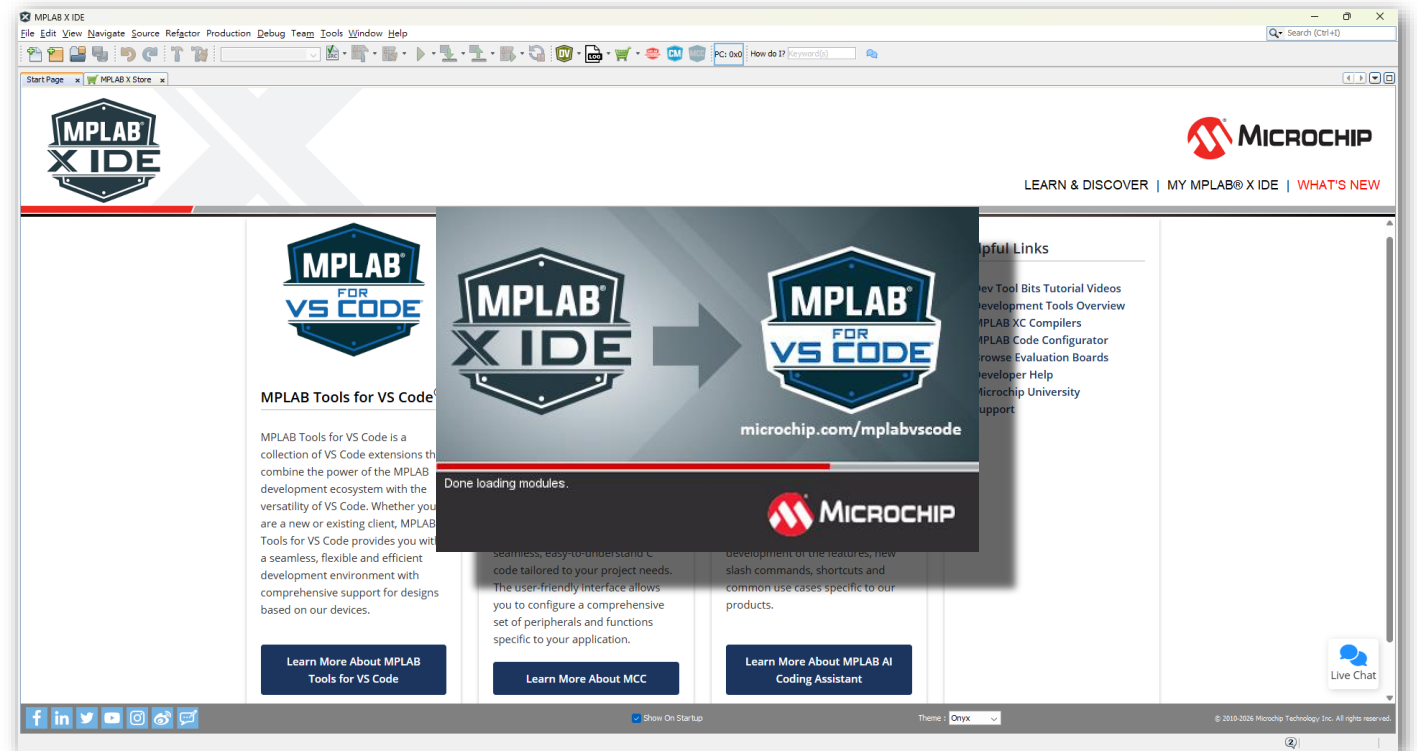
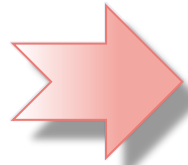
- **MPLAB® X IDE v6.30** <https://www.microchip.com/mplab/mplab-x-ide>
- **MPLAB® XC32 Compiler v5.00** <https://www.microchip.com/mplab/compilers>
- **MPLAB® Code Configurator (MCC) v5.6.4.2**
- **MCC Content Manager v2.0.4.2**
- **MCC-Service-Provider-Plugin v1.2.0**
- **MCC Core v5.8.4**
- **MCC Harmony Framework (From X IDE MCC Content Manager)**
 - MCC Harmony Core (com.microchip.mcc.harmony.Harmony3Library) v1.5.5
 - Harmony 3 - Chip Support Package (**csp**) v3.25.1
 - Harmony 3 - Core (**core**) v3.16.2
 - Harmony 3 - Core (**bsp**) v3.25.0
 - Harmony 3 - Networking Stack and Solutions (**net**) v3.14.5
 - Harmony 3 - Networking Stack and Solutions (**net_10base_t1s**) v1.4.3
 - Harmony 3 - WolfSSL solutions (**wolfssl**) v5.4.0
 - Harmony 3 - Harmony Services (**harmony-services**) v1.5.0
 - arm CMSIS (**CMSIS_5**) v5.9.0
- **MPLAB® X IDE Packs**
 - PIC32CX-SG41_DFP v1.6.172 (Device Family Pack)
 - CMSIS v6.2.0 or Newest (Device Family Pack)
- **MCC Core and Framework Update from X IDE > MCC > Content Manager**
- **PIC32CX-SG41_DFP Update from X IDE > Tools > Packs**
- **Tera Term v5.5** <https://github.com/TeraTermProject/teraterm/releases>

(Lasted Update : 2026/04/22)

MCC Harmony Framework

Step 1

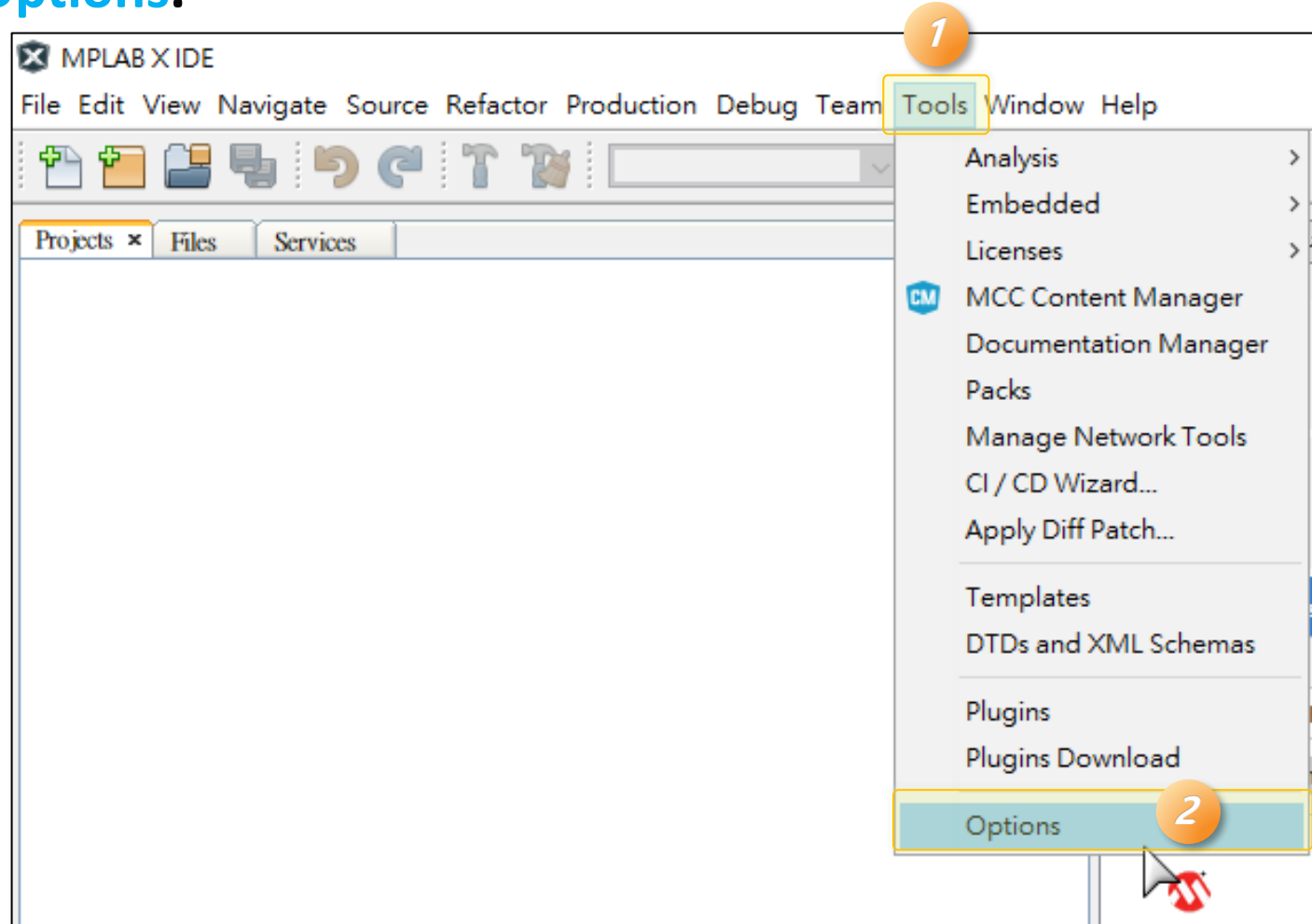
- Open **MPLAB X IDE v6.30**.



MCC Harmony Framework

Step 2

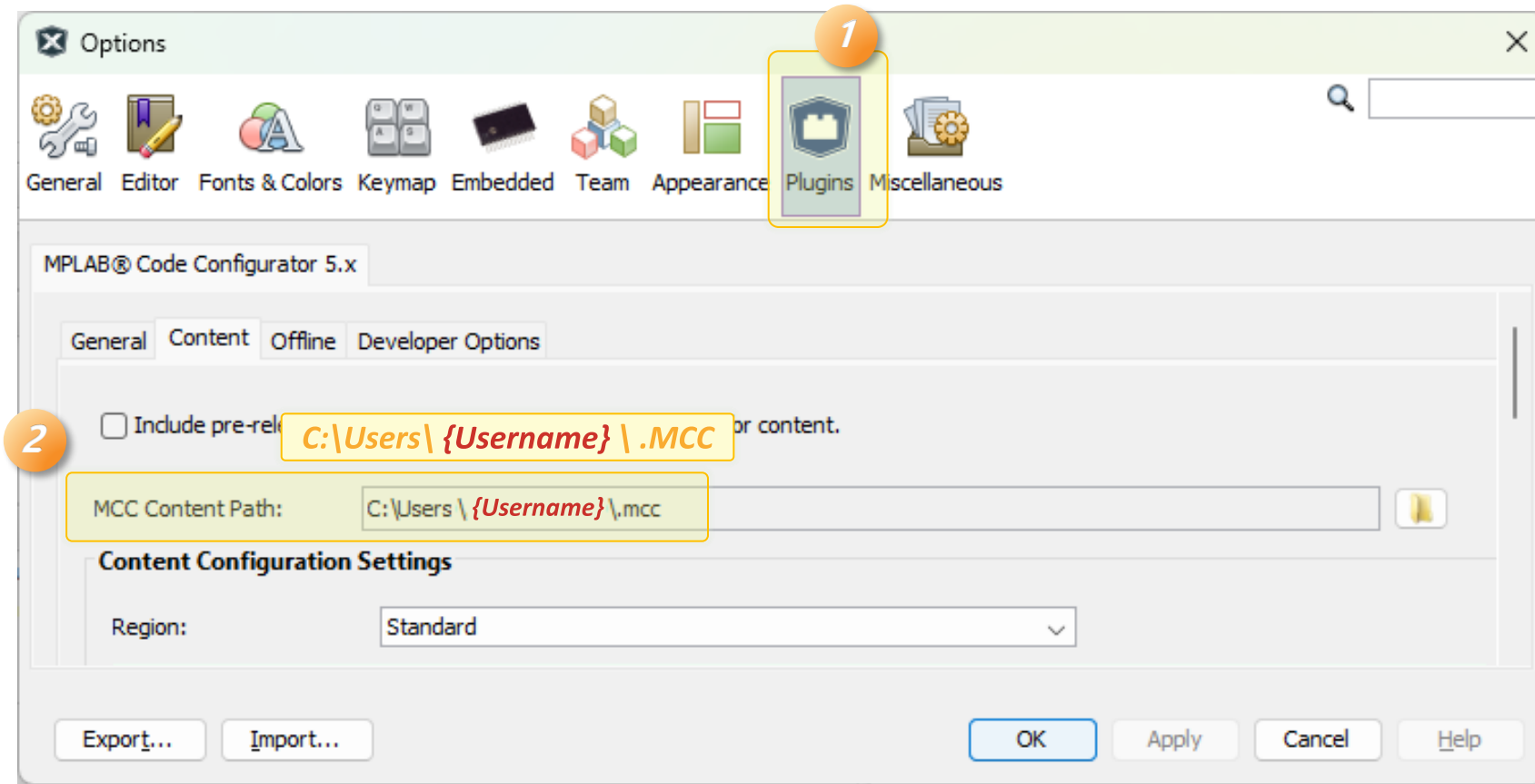
- Click **Tools** > **Options**.



MCC Harmony Framework

Step 3

- Click **Plugins** and check the **Harmony Content Path**

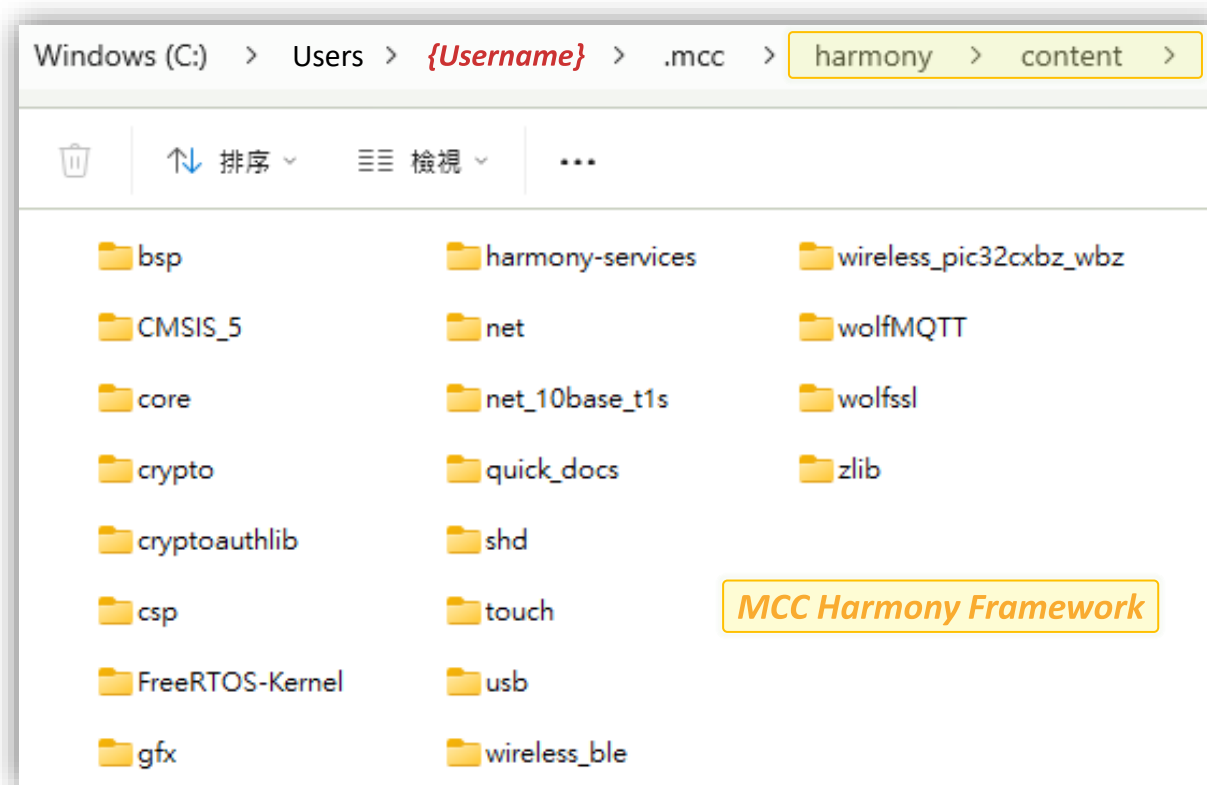


MCC Harmony Framework

Step 4

- Locate the MCC Harmony Framework folder in your computer.

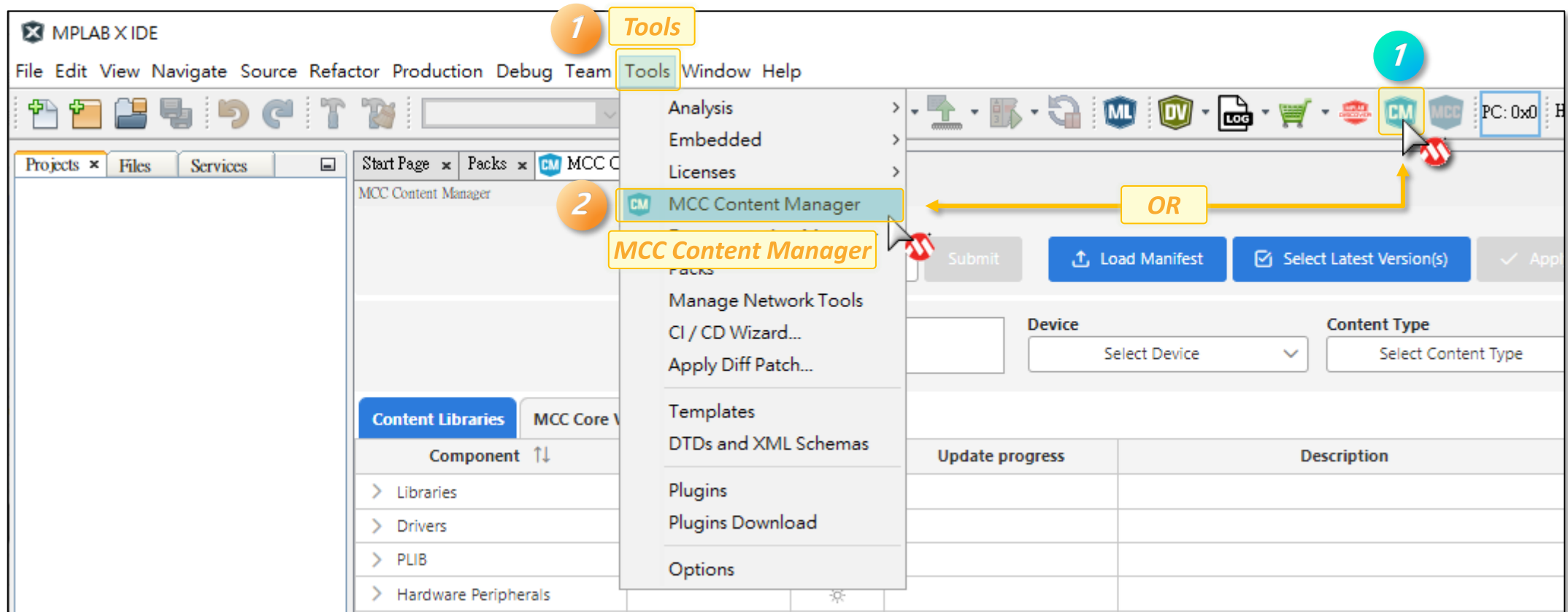
C:\Users\{Username}\.mcc\harmony\content



MCC Harmony Framework

Step 5

- Click **Tools** > **MCC Content Manager** or Click icon 



MCC Harmony Framework

Step 6

- Type "t1s" in quick search, select to latest **net_10base_t1s** module then next page.

The screenshot shows the MPLAB X IDE interface with the MCC Content Manager window open. The search bar contains "t1s" (labeled with a yellow box and "1"). The "Content Libraries" tab is active, showing a table of components. The "Harmony 3 - Networking Stack and Solutions" component is expanded, and the "net_10base_t1s" module is selected (labeled with a yellow box and "2"). The version dropdown for "net_10base_t1s" shows "v1.4.3" (labeled with a yellow box and "V1.4.3"). A pink box with the text "Wait a few minutes for the selected modules to download" is overlaid on the right side of the table.

Harmony 3 - Networking Stack and Solutions

net_10base_t1s

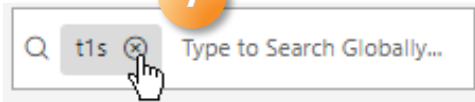
V1.4.3

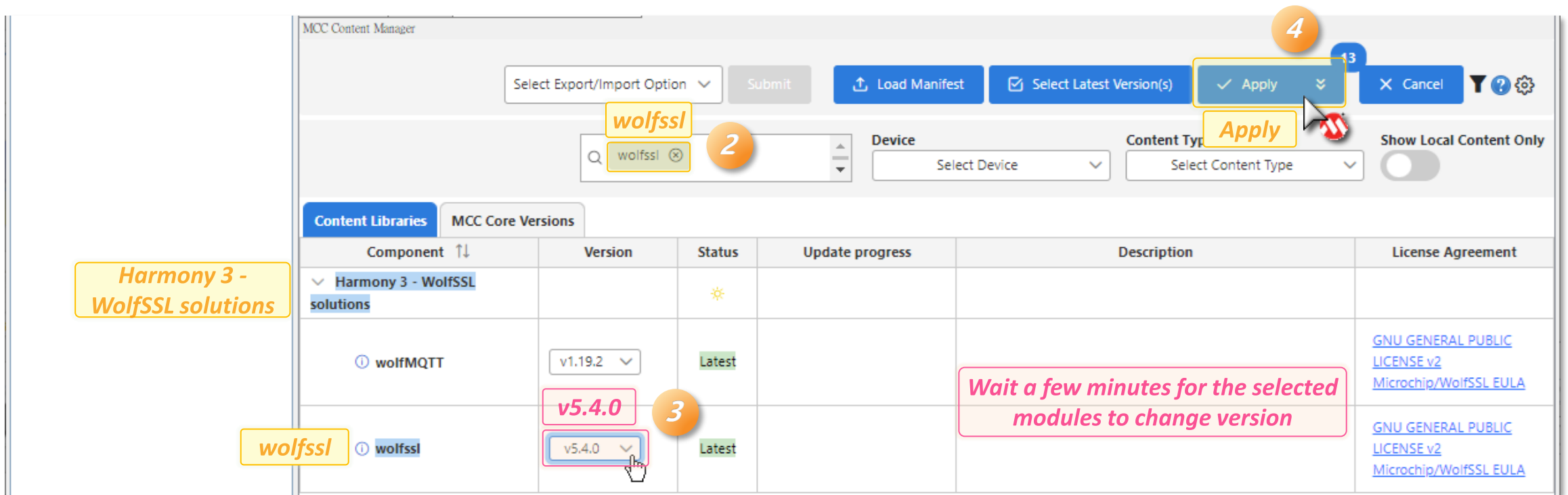
Wait a few minutes for the selected modules to download

Component	Version	Status	Update progress	Description	License Agreement
Harmony 3 - Networking Stack and Solutions					
net_10base_t1s	v1.4.3				
	v1.4.3 - [remote]				
	v1.3.3 - [remote]				

MCC Harmony Framework

Step 7

- Clean "**t1s**" by  in quick search, Input "**wolfssl**" to find module and change the version of **wolfssl** to **v5.4.0** and click on **Apply** to make changes then remember close the **Content Manager** before execute **MCC**.



Harmony 3 - WolfSSL solutions

wolfssl

wolfssl

v5.4.0

v5.4.0

Wait a few minutes for the selected modules to change version

Apply

Component	Version	Status	Update progress	Description	License Agreement
Harmony 3 - WolfSSL solutions					
wolfMQTT	v1.19.2	Latest			GNU GENERAL PUBLIC LICENSE v2 Microchip/WolfSSL EULA
wolfssl	v5.4.0	Latest			GNU GENERAL PUBLIC LICENSE v2 Microchip/WolfSSL EULA

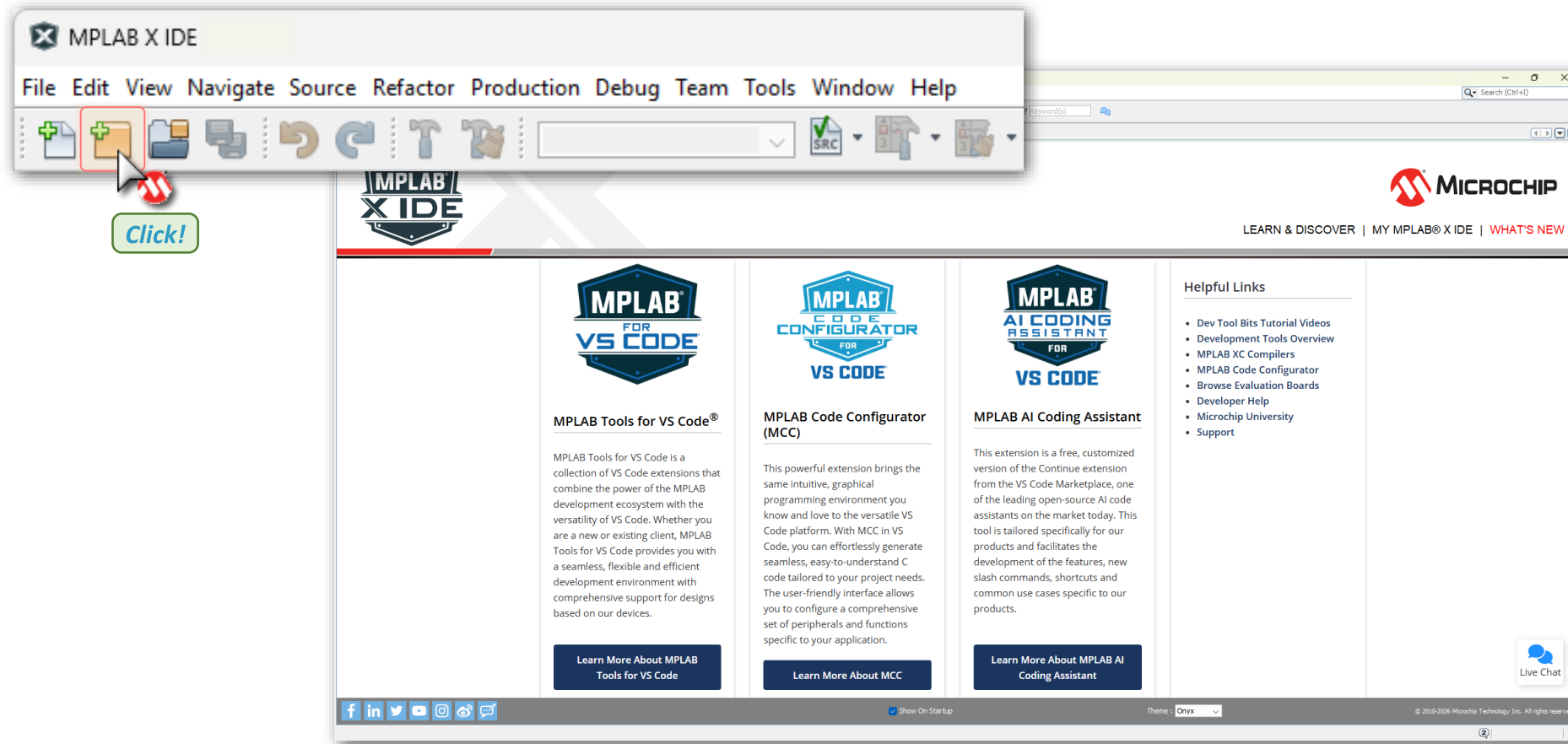
Start From New Project

Create a New Project in MPLAB X IDE

Start From New Project

Step 1

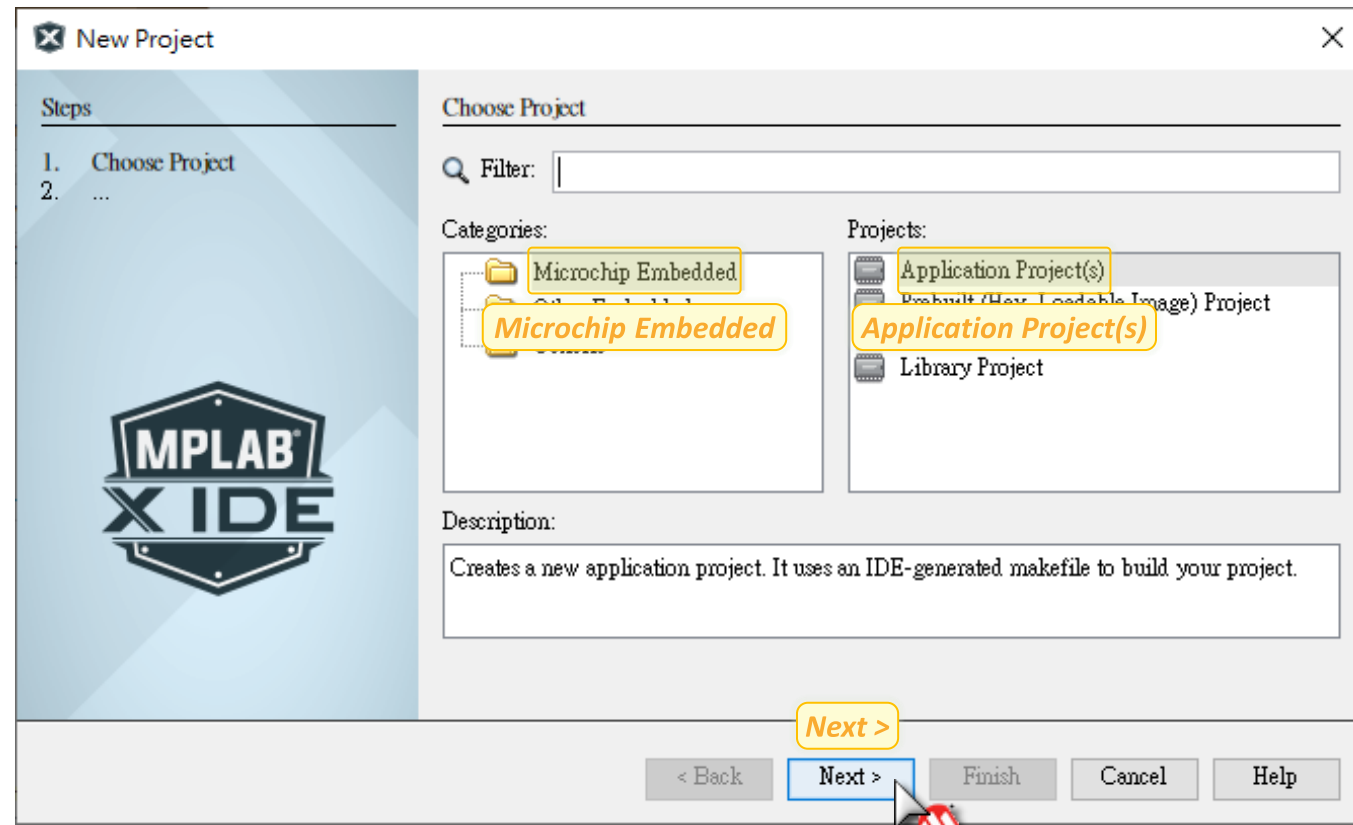
- Select to **File > New Project** or **Click icon** 



Start From New Project

Step 2

- Categories : **Microchip Embedded**
- Projects : **Application Project(s)**



Start From New Project

Step 3

- Family : All Families
- Device : PIC32CX1025SG41128 **Don't miss any character**
- Tool : RTC-APP055-PKoB4 SN: CAExxxxxxxxxxxxxx

New Project

Steps

1. Choose Project
2. Select Device
3. Select Header
4. Select Plugin Board
5. Select Compiler
6. Select Project Name and Folder
7. (Optional) Add Project

MPLAB X IDE

Select Device

Family: All Families

Device: PIC32CX1025SG41128

Tool: RTC-APP055-PKoB4 SN: CAExxxxxx

Input : PIC32CX1025SG41128

RTC-APP055-PKoB4 SN: CAExxxxxx

Next >

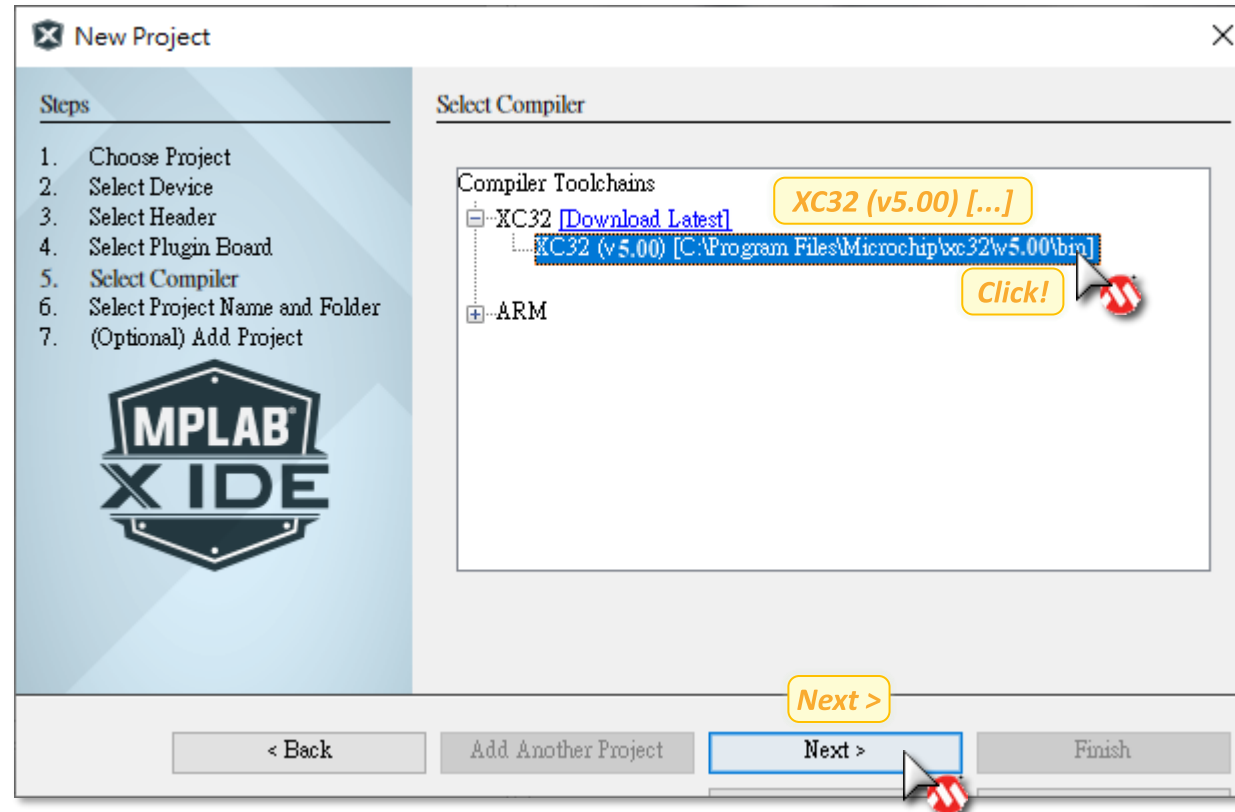
< Back Add Another Project Next > Finish

Start From New Project

Step 4

Assign Compiler Toolchains

- XC32 : **XC32 (v5.00)[...]**



Start From New Project

Step 5

Configure the Project Location and Project Folder Name.

- Project Name : **10BASE-T1S** *No white space of project name or path*
- Project Location : **C:\Exercises\10BASE-T1S**

New Project

Steps

1. Choose Project
2. Select Device
3. Select Header
4. Select Plugin Board
5. Select Compiler
6. **Select Project Name and Folder**
7. (Optional) Add Project

Select Project Name and Folder

Project Name: **10BASE-T1S**

Project Location: **C:\Exercises\10BASE-T1S**

Project Folder: **C:\Exercises\10BASE-T1S\10BASE-T1S.X**

☐ Overwrite existing project.

☐ Also delete sources.

☒ Set as main project

☒ Open MCC on Finish

☐ Use project location as the project folder

Encoding: **UTF-8**

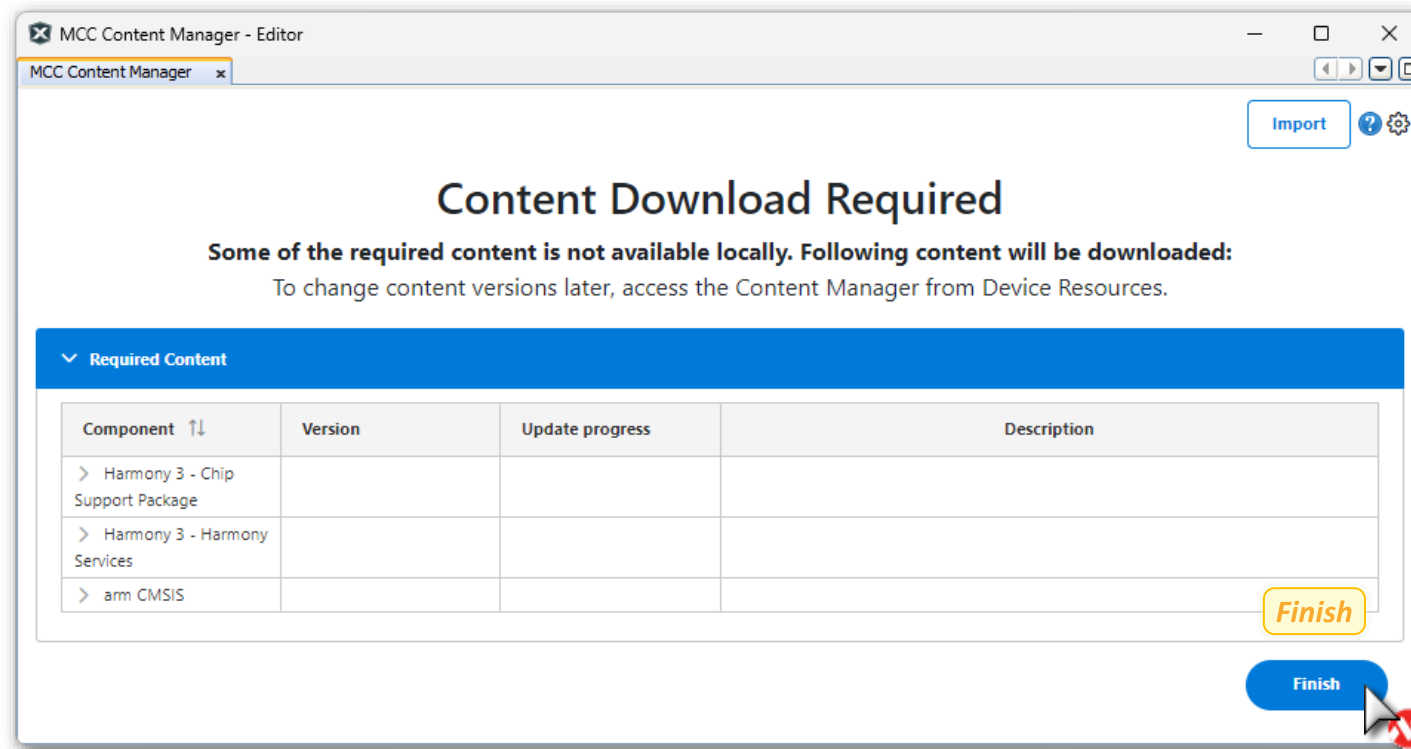
Finish

< Back Add Another Project Next > **Finish**

Start From New Project

Step 6 (optional)

- Click **Finish** and **Waiting for automatic download MCC Framework**.



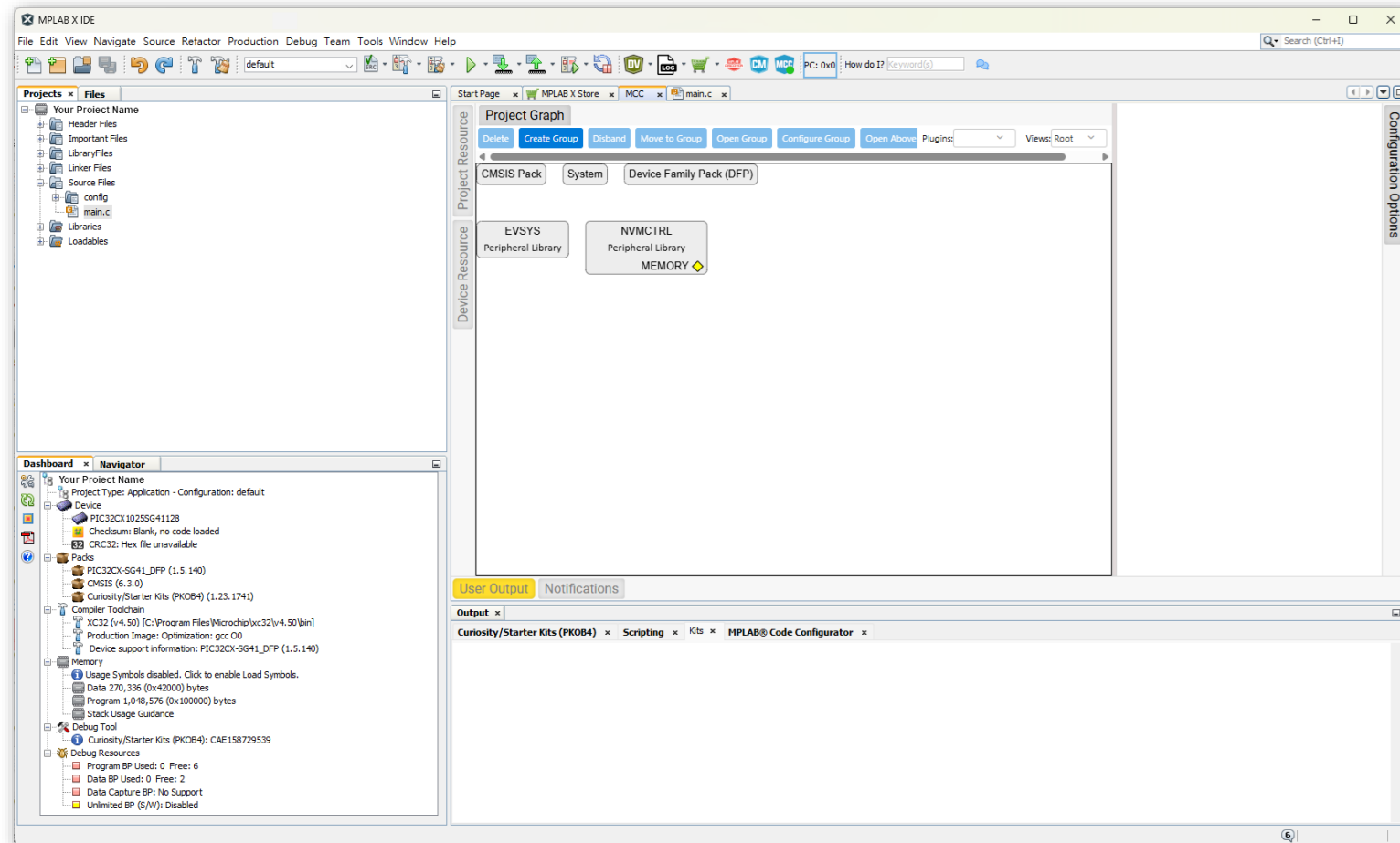
** MCC will download the latest Framework, which only needs to be performed once.*

** This step will not occur in the course because the MCC Framework is already installed on the computer.*

Start From New Project

Step 7

- The MCC Harmony Configurator interface appears.

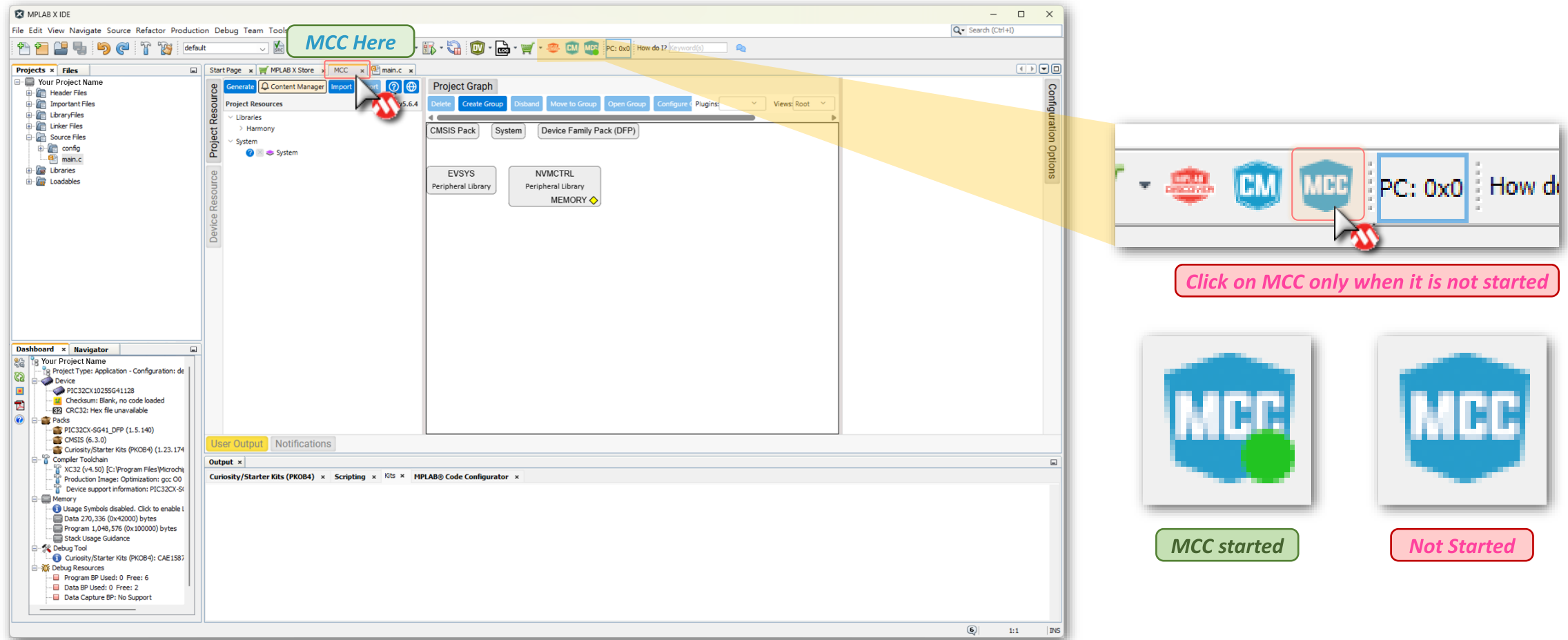


MPLAB® Code Configurator (MCC)

Harmony Interface Introduction

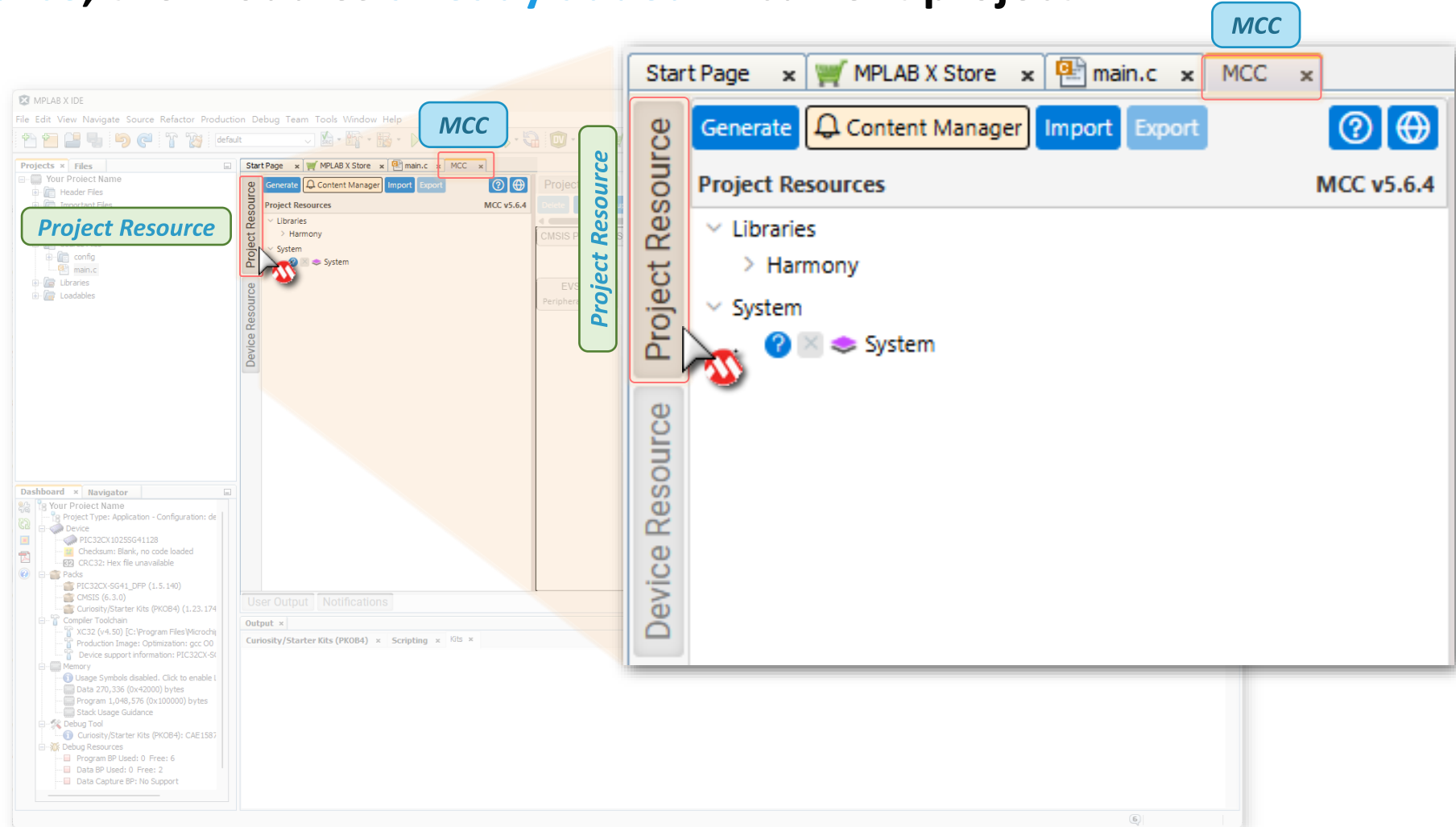
MCC Harmony Interface

- Let's take a brief look at the MCC Harmony Configurator interface.
- Notice !** If you close MCC already , please click the  icon and reopen MCC.



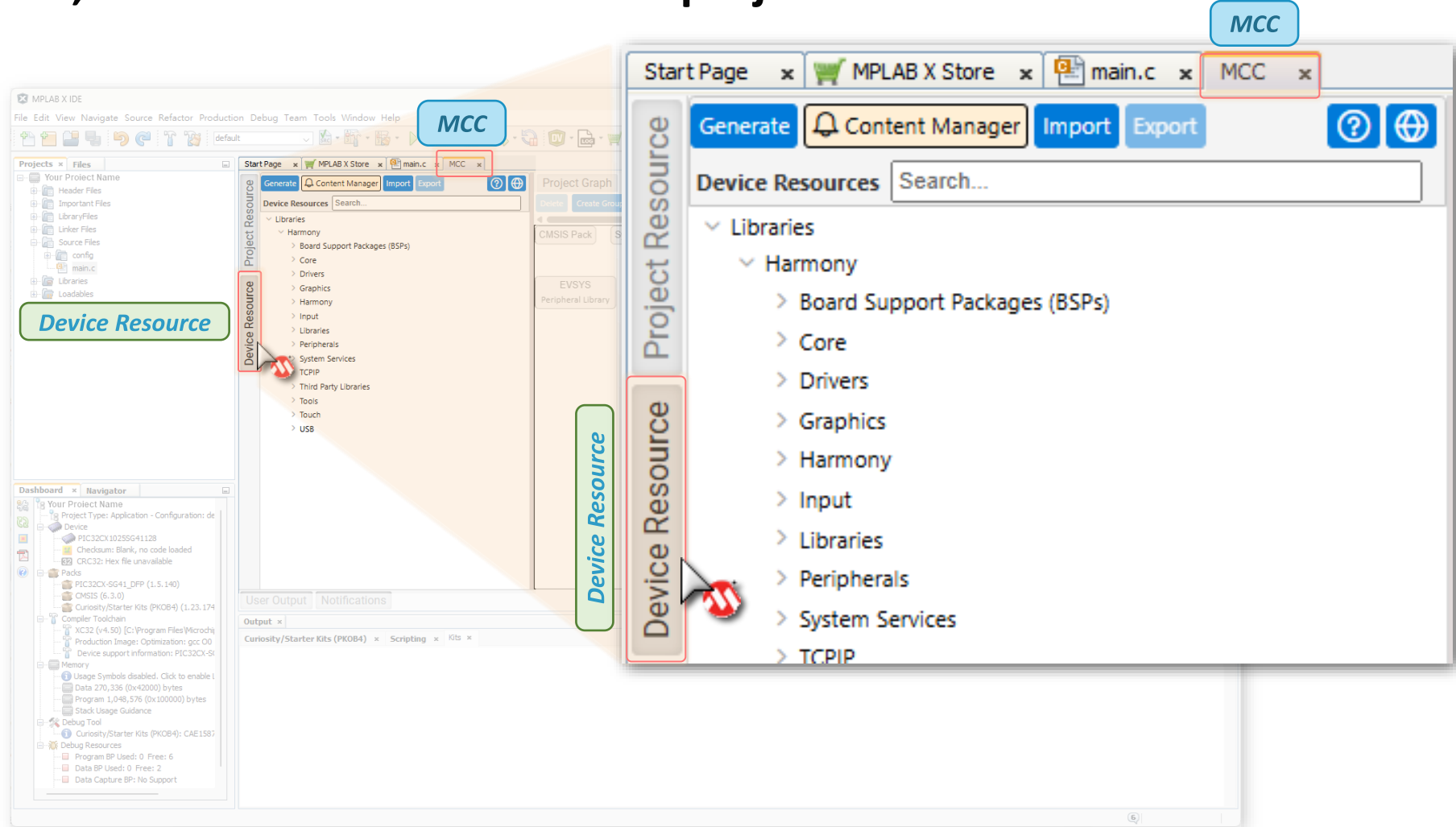
MCC Harmony Interface

- **Project Resource**, the modules already added in current project.



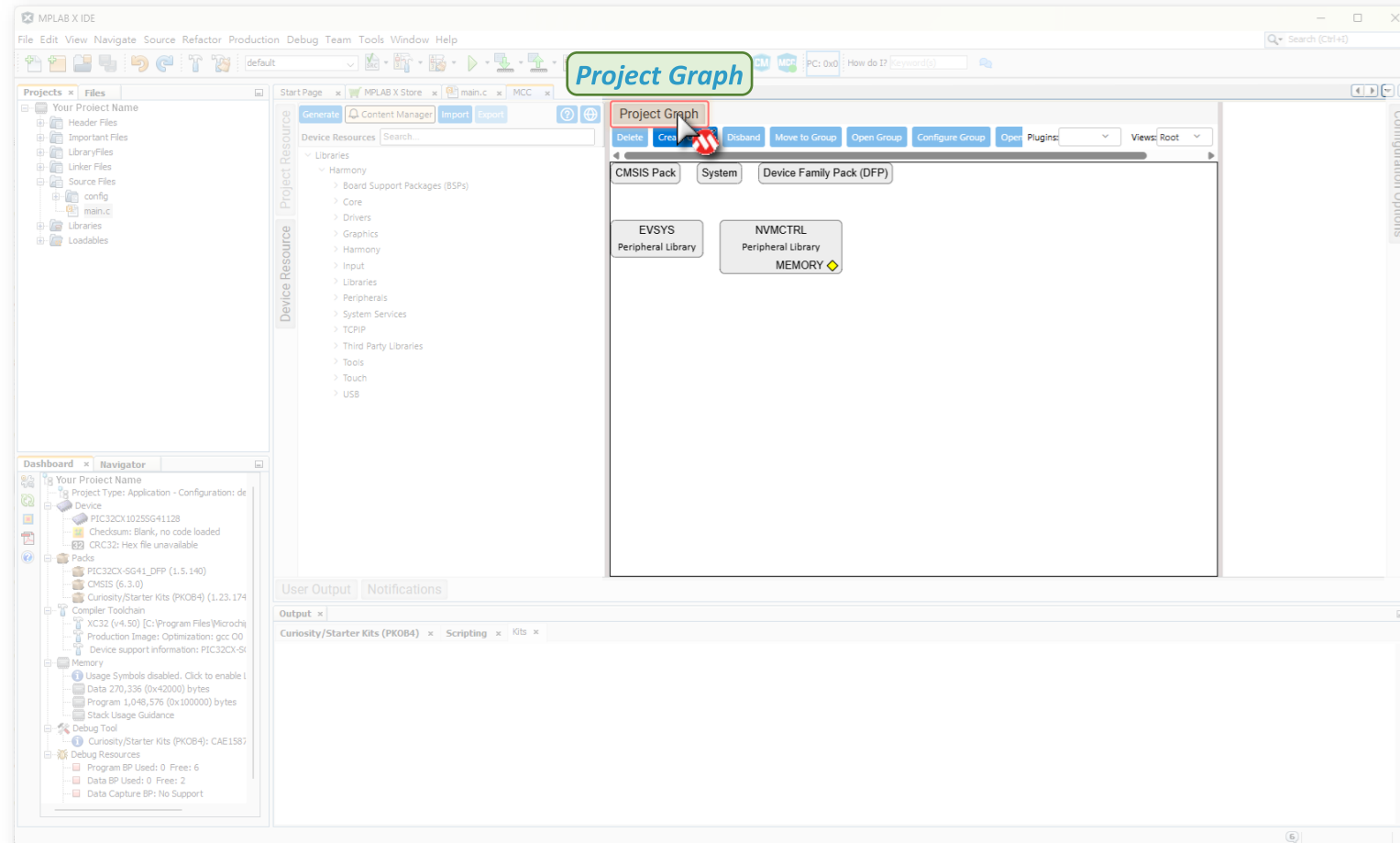
MCC Harmony Interface

- **Device Resource**, the modules not in current project but **available to be added**.



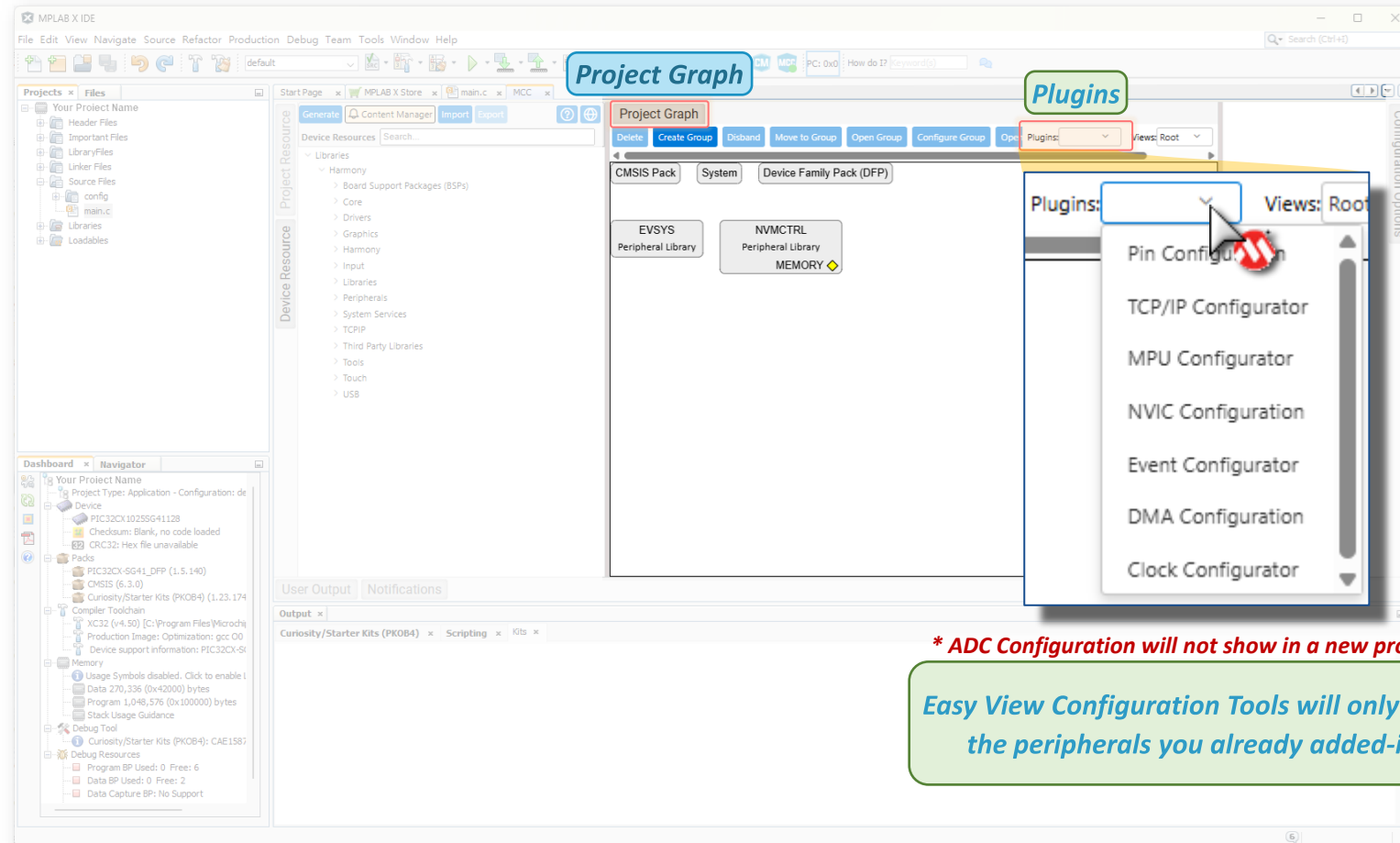
MCC Harmony Interface

- **Project Graph**, the connection diagram between modules.



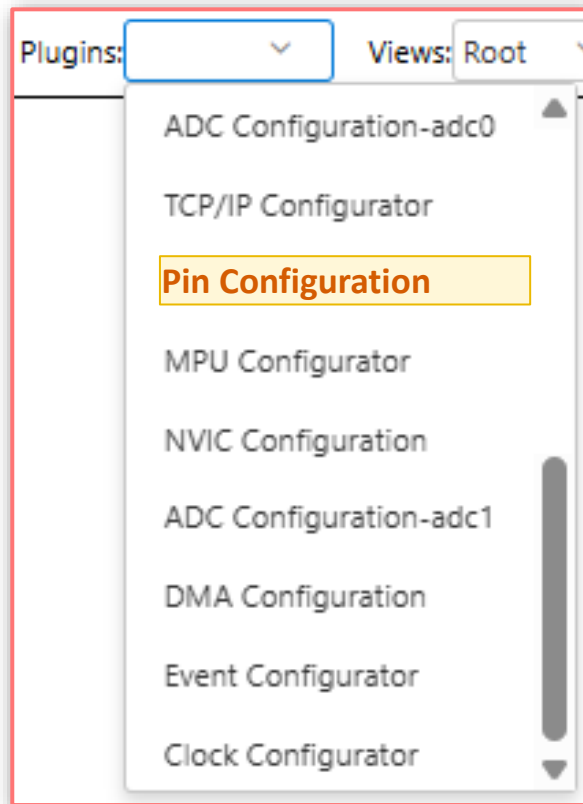
MCC Harmony Interface

- **Project Graph** → **Plugins**, the Easy View Configuration of modules.



MCC Harmony Interface

- Project Graph → Plugins : Pin Configuration

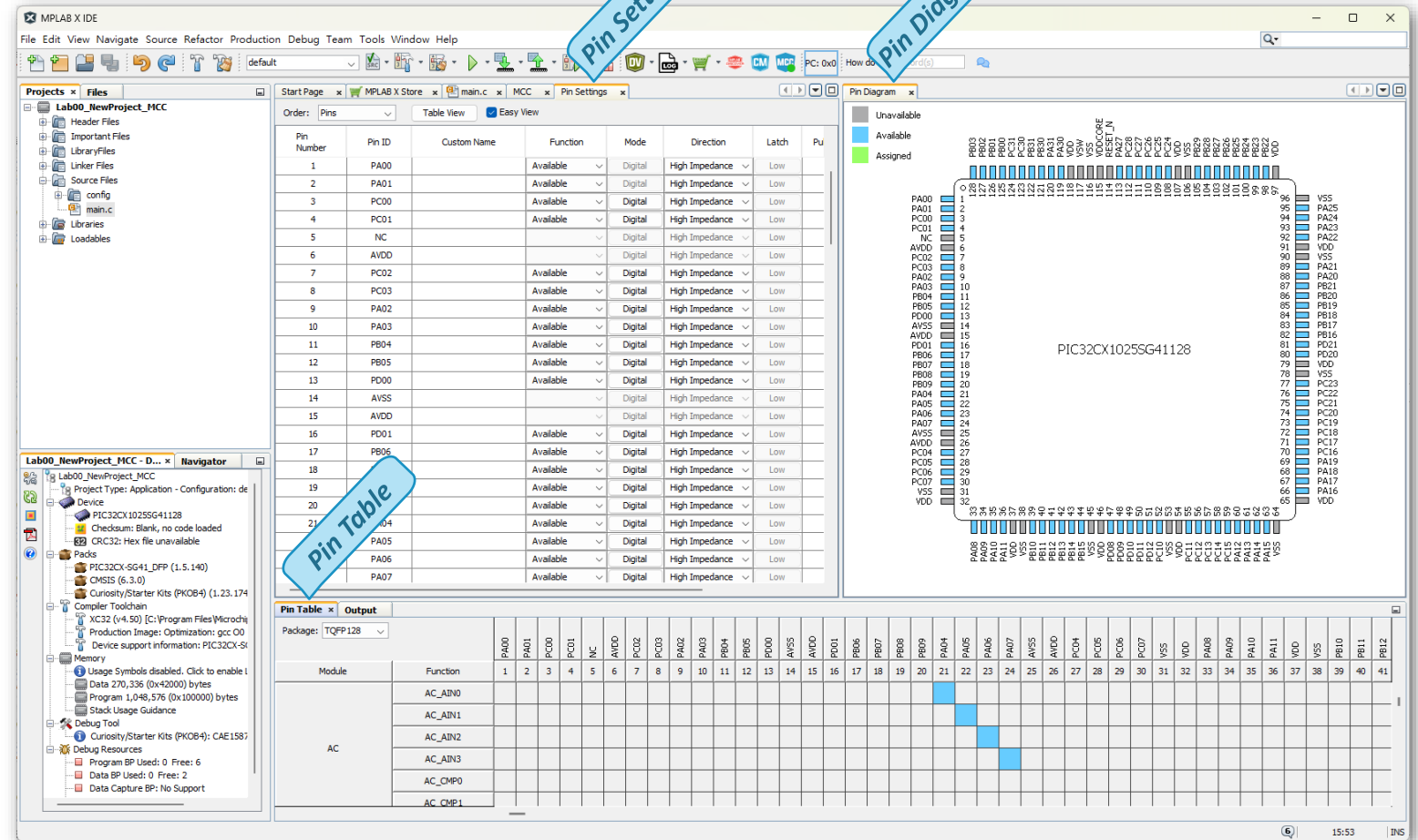
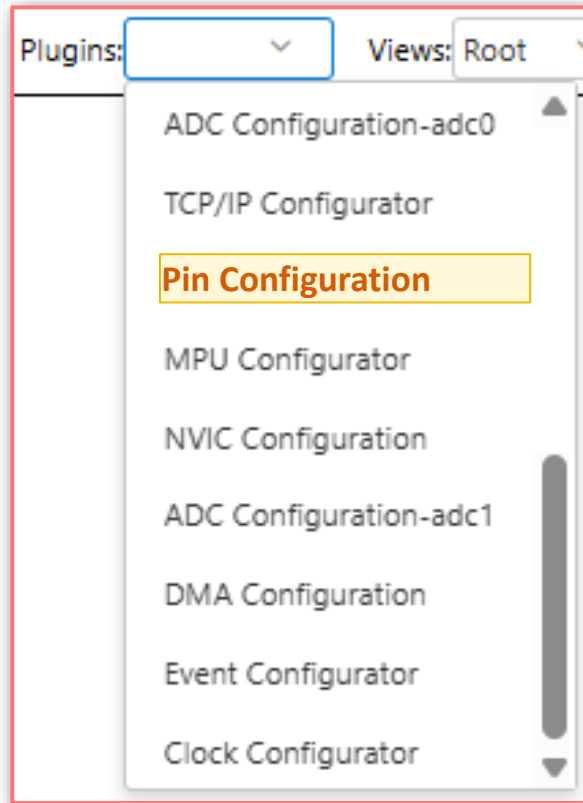


A screenshot of the MPLAB X IDE Pin Configuration window. The window displays a table of pin configurations for a PIC32CX1025G41128 device. The table has columns for Pin Number, Pin ID, Custom Name, Function, Mode, Direction, Latch, Pull Up, Pull Down, and Drive Strength. The 'Pin Table' tab is selected, and the table shows 37 pins. Callouts point to the 'Pin Diagram', 'Pin Table', and 'Pin Settings' tabs.

Pin Number	Pin ID	Custom Name	Function	Mode	Direction	Latch	Pull Up	Pull Down	Drive Strength
1	PA00		Available	Digital	High Impedance	Low	☐	☐	NORMAL
2	PA01		Available	Digital	High Impedance	Low	☐	☐	NORMAL
3	PC00		Available	Digital	High Impedance	Low	☐	☐	NORMAL
4	PC01		Available	Digital	High Impedance	Low	☐	☐	NORMAL
5	NC			Digital	High Impedance	Low	☐	☐	NORMAL
6	AVDD			Digital	High Impedance	Low	☐	☐	NORMAL
7	PC02		Available	Digital	High Impedance	Low	☐	☐	NORMAL
8	PC03		Available	Digital	High Impedance	Low	☐	☐	NORMAL
9	PA02		Available	Digital	High Impedance	Low	☐	☐	NORMAL
10	PA03		Available	Digital	High Impedance	Low	☐	☐	NORMAL
11	PB04		Available	Digital	High Impedance	Low	☐	☐	NORMAL
12	PB05		Available	Digital	High Impedance	Low	☐	☐	NORMAL
13	PD00		Available	Digital	High Impedance	Low	☐	☐	NORMAL
14	AVSS			Digital	High Impedance	Low	☐	☐	NORMAL
15	AVDD			Digital	High Impedance	Low	☐	☐	NORMAL
16	PD01		Available	Digital	High Impedance	Low	☐	☐	NORMAL
17	PB06		Available	Digital	High Impedance	Low	☐	☐	NORMAL
18	PB07		Available	Digital	High Impedance	Low	☐	☐	NORMAL
19	PB08		Available	Digital	High Impedance	Low	☐	☐	NORMAL
20	PB09		Available	Digital	High Impedance	Low	☐	☐	NORMAL
21	PA04		Available	Digital	High Impedance	Low	☐	☐	NORMAL
22	PA05		Available	Digital	High Impedance	Low	☐	☐	NORMAL
23	PA06		Available	Digital	High Impedance	Low	☐	☐	NORMAL
24	PA07		Available	Digital	High Impedance	Low	☐	☐	NORMAL
25	AVSS			Digital	High Impedance	Low	☐	☐	NORMAL
26	AVDD			Digital	High Impedance	Low	☐	☐	NORMAL
27	PC04		Available	Digital	High Impedance	Low	☐	☐	NORMAL
28	PC05		Available	Digital	High Impedance	Low	☐	☐	NORMAL
29	PC06		Available	Digital	High Impedance	Low	☐	☐	NORMAL
30	PC07		Available	Digital	High Impedance	Low	☐	☐	NORMAL
31	VSS			Digital	High Impedance	Low	☐	☐	NORMAL
32	VDD			Digital	High Impedance	Low	☐	☐	NORMAL
33	PA08		Available	Digital	High Impedance	Low	☐	☐	NORMAL
34	PA09		Available	Digital	High Impedance	Low	☐	☐	NORMAL
35	PA10		Available	Digital	High Impedance	Low	☐	☐	NORMAL
36	PA11		Available	Digital	High Impedance	Low	☐	☐	NORMAL
37	VDD			Digital	High Impedance	Low	☐	☐	NORMAL

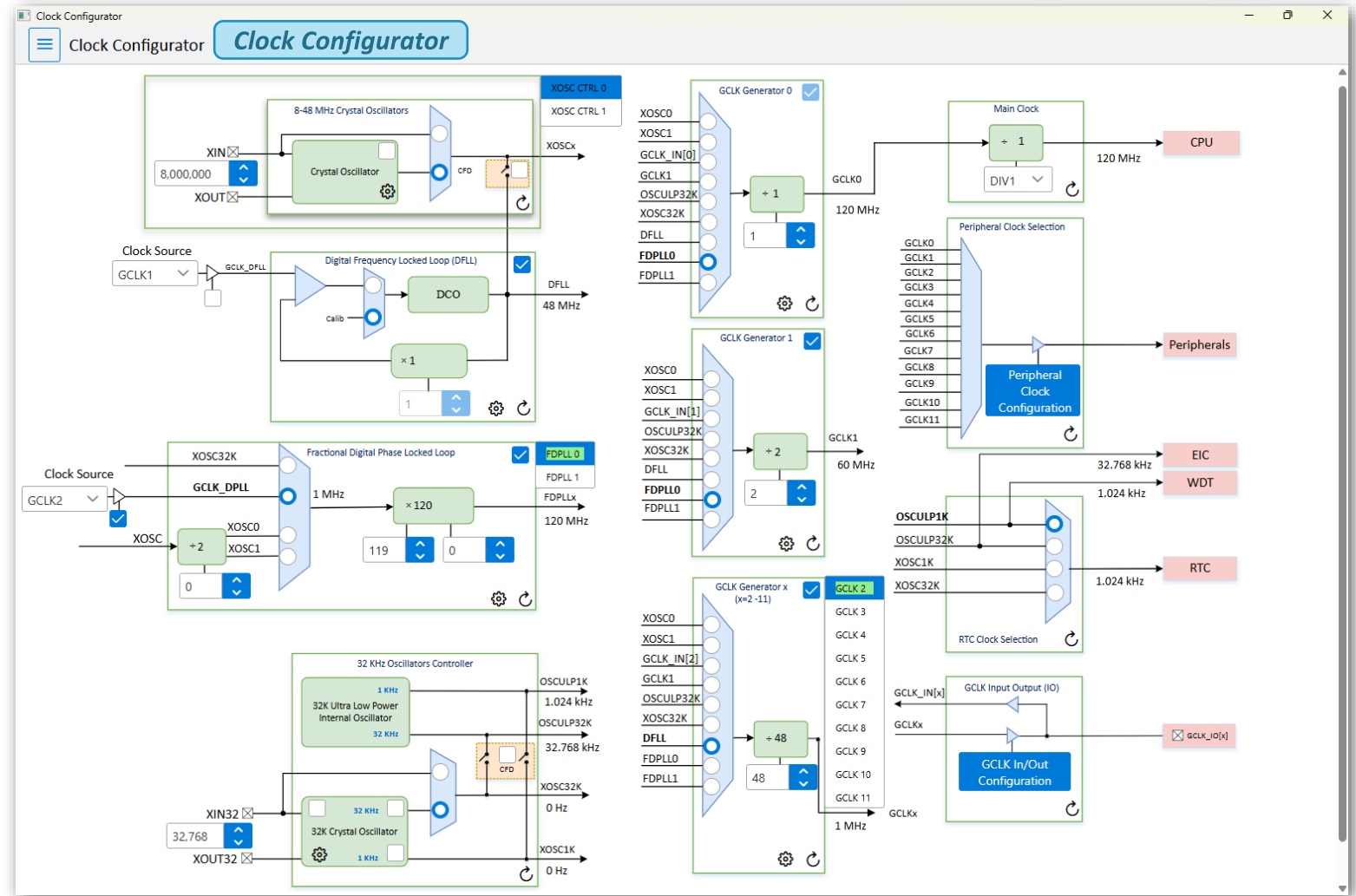
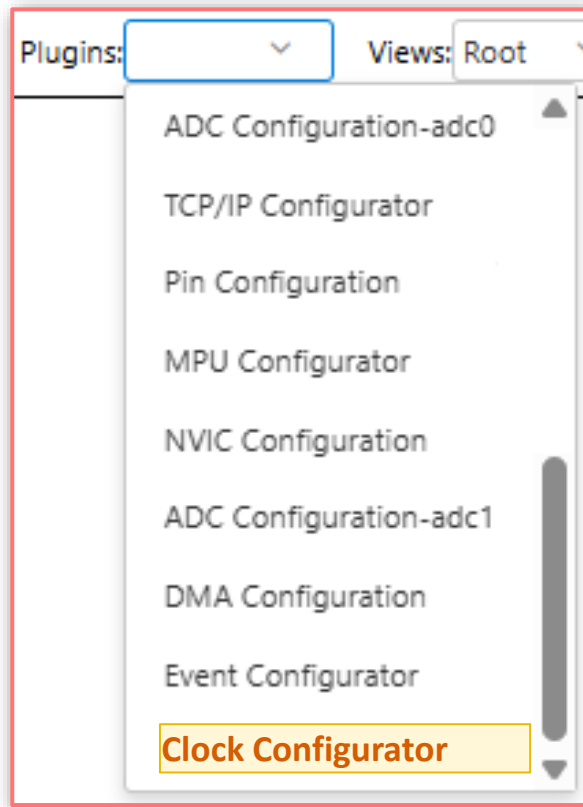
MCC Harmony Interface

- **Project Graph** → **Plugins** : **Pin Configuration** (Dock three windows to different locations)



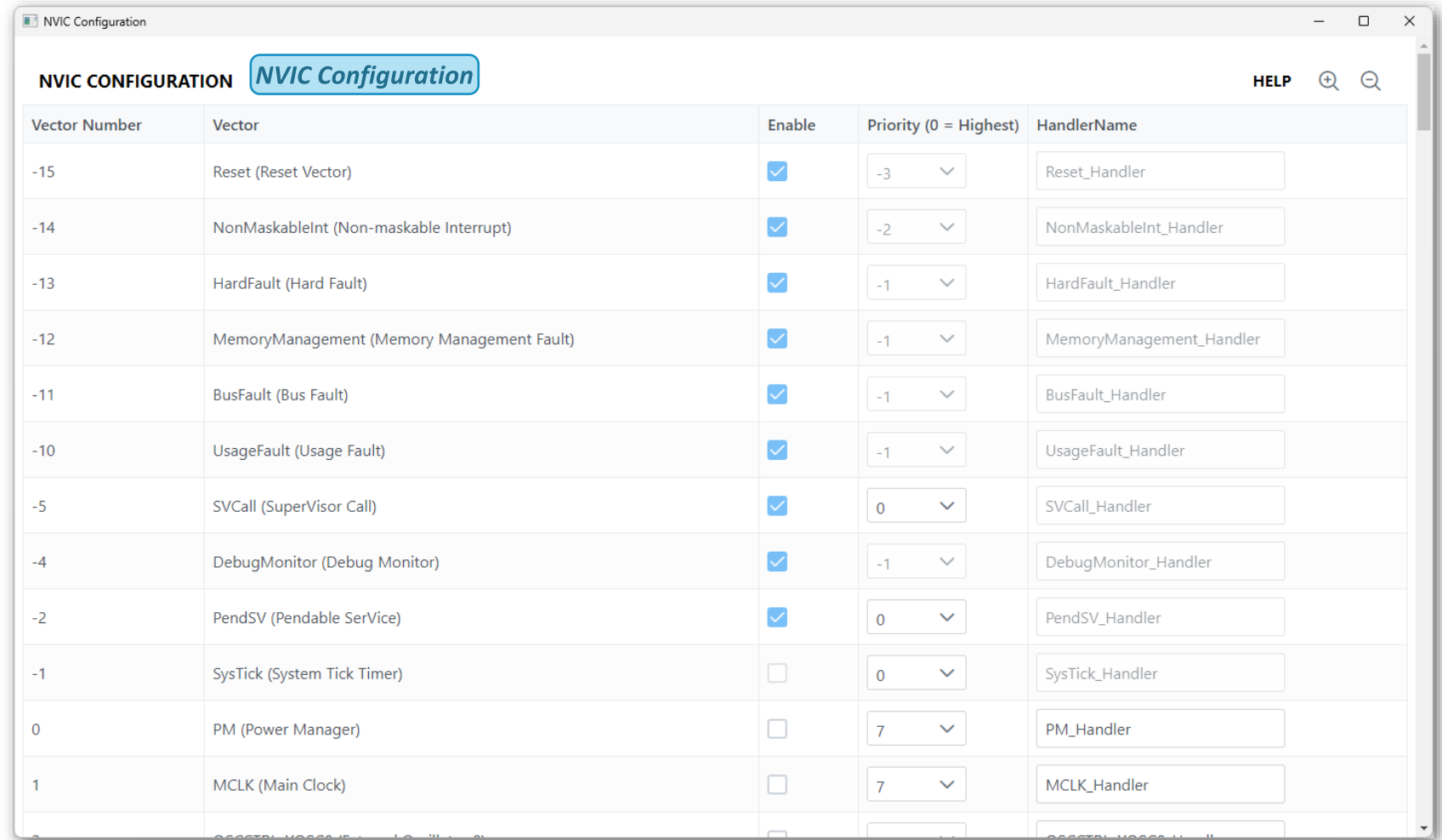
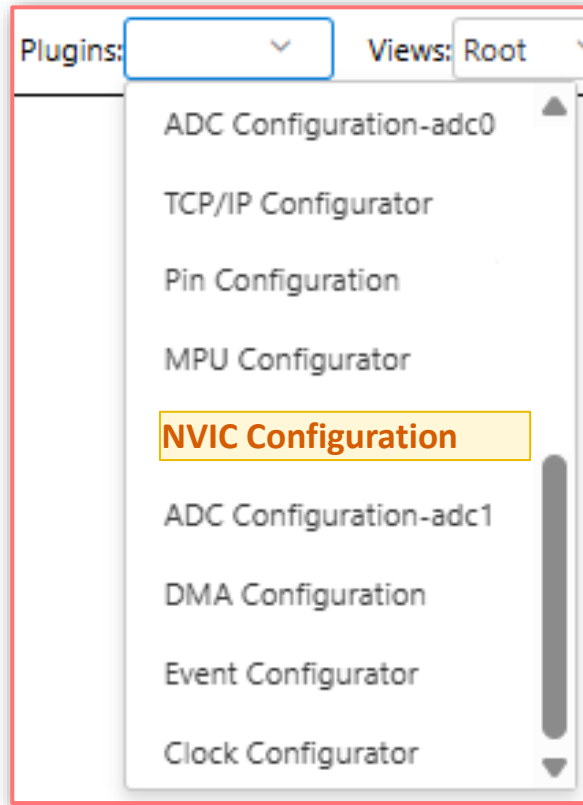
MCC Harmony Interface

- Project Graph → Plugins : Clock Configuration



MCC Harmony Interface

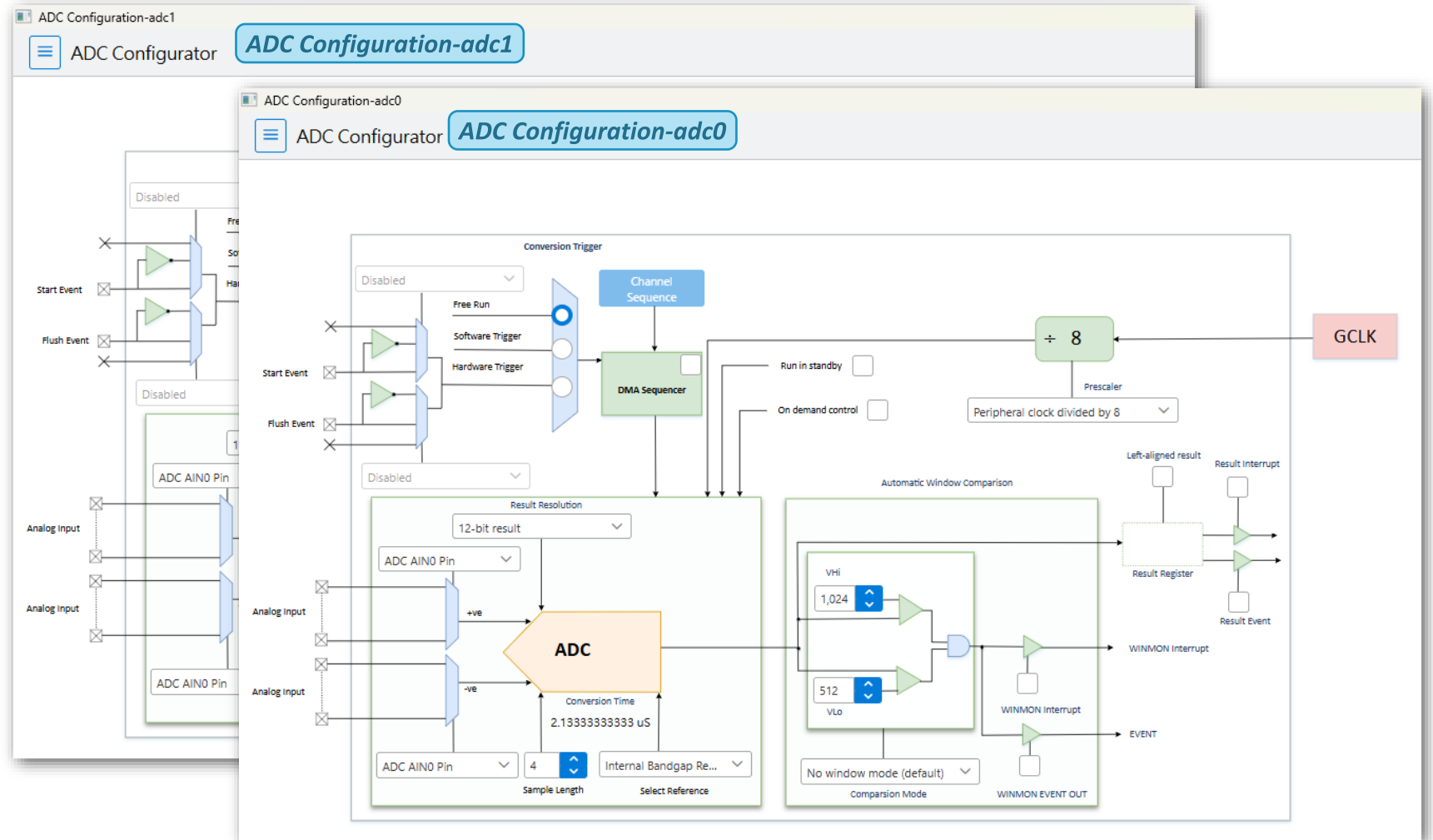
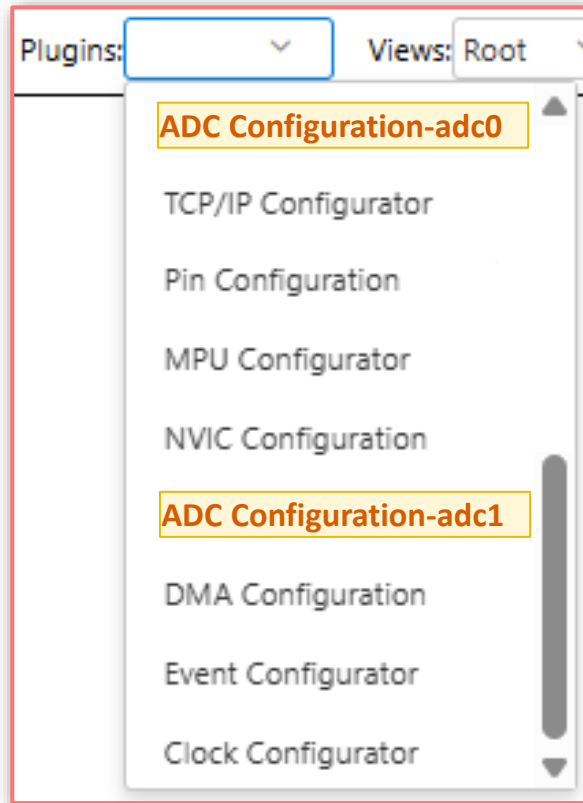
- **Project Graph** → **Plugins** : **NVIC Configuration** (Open a new window after clicked)



MCC Harmony Interface

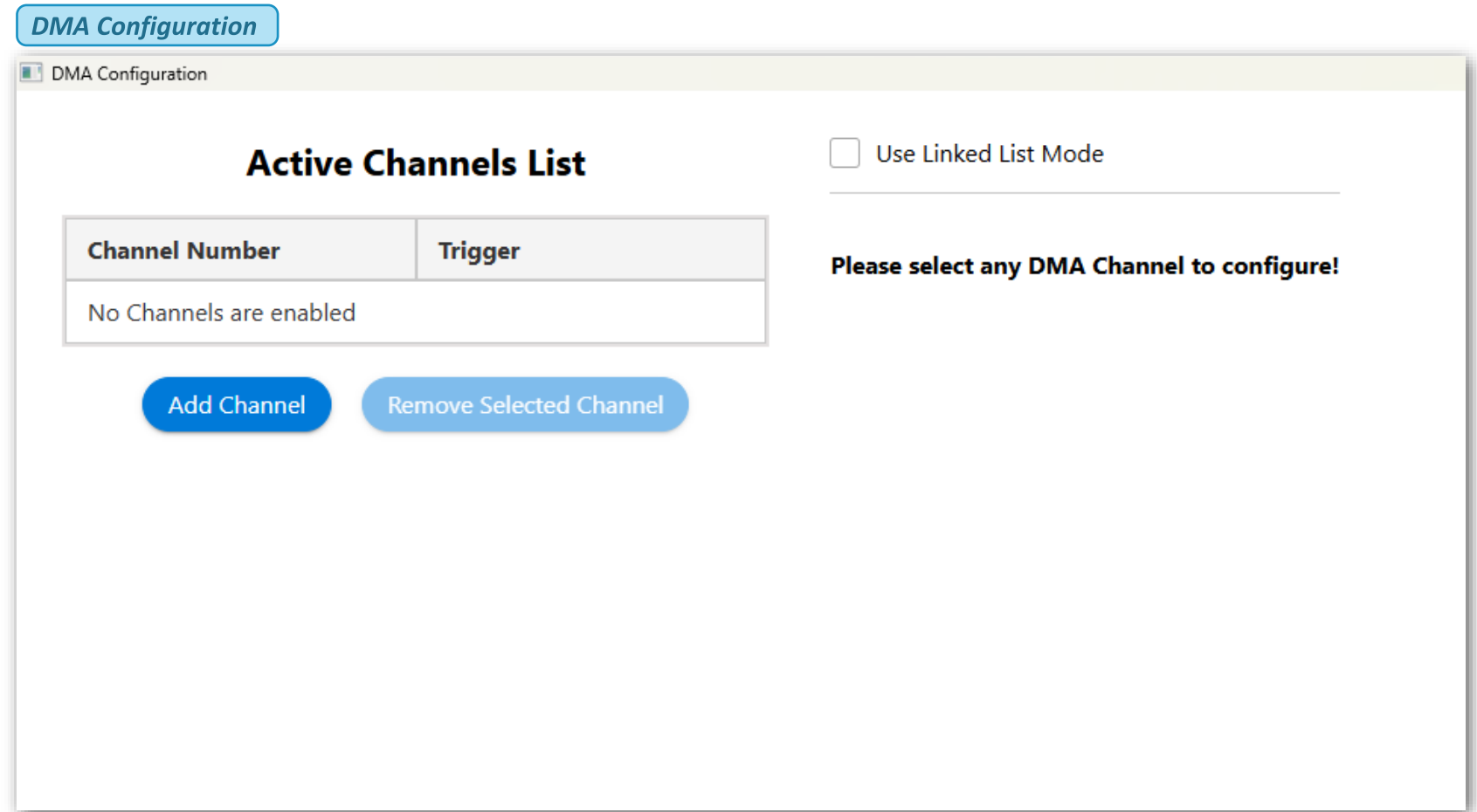
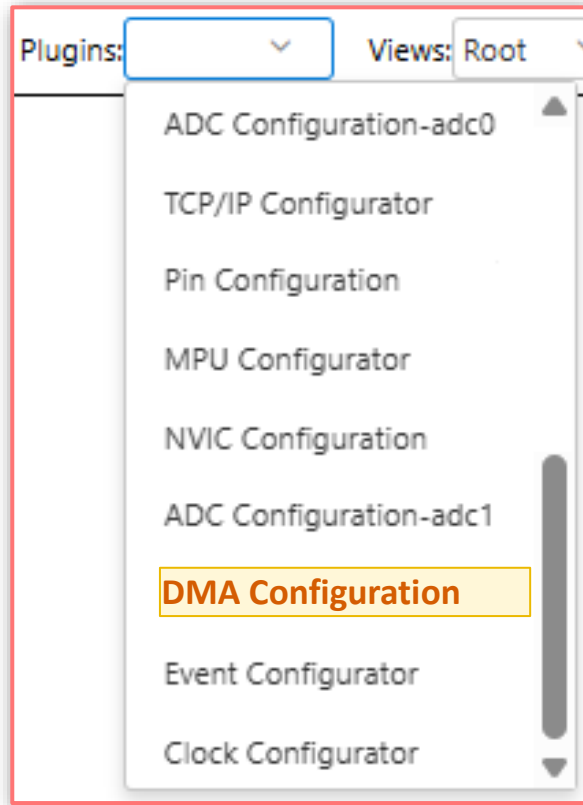
- **Project Graph** → **Plugins : ADC Configuration** (Open a new window after clicked)

** ADC Configuration will not show in a new project*



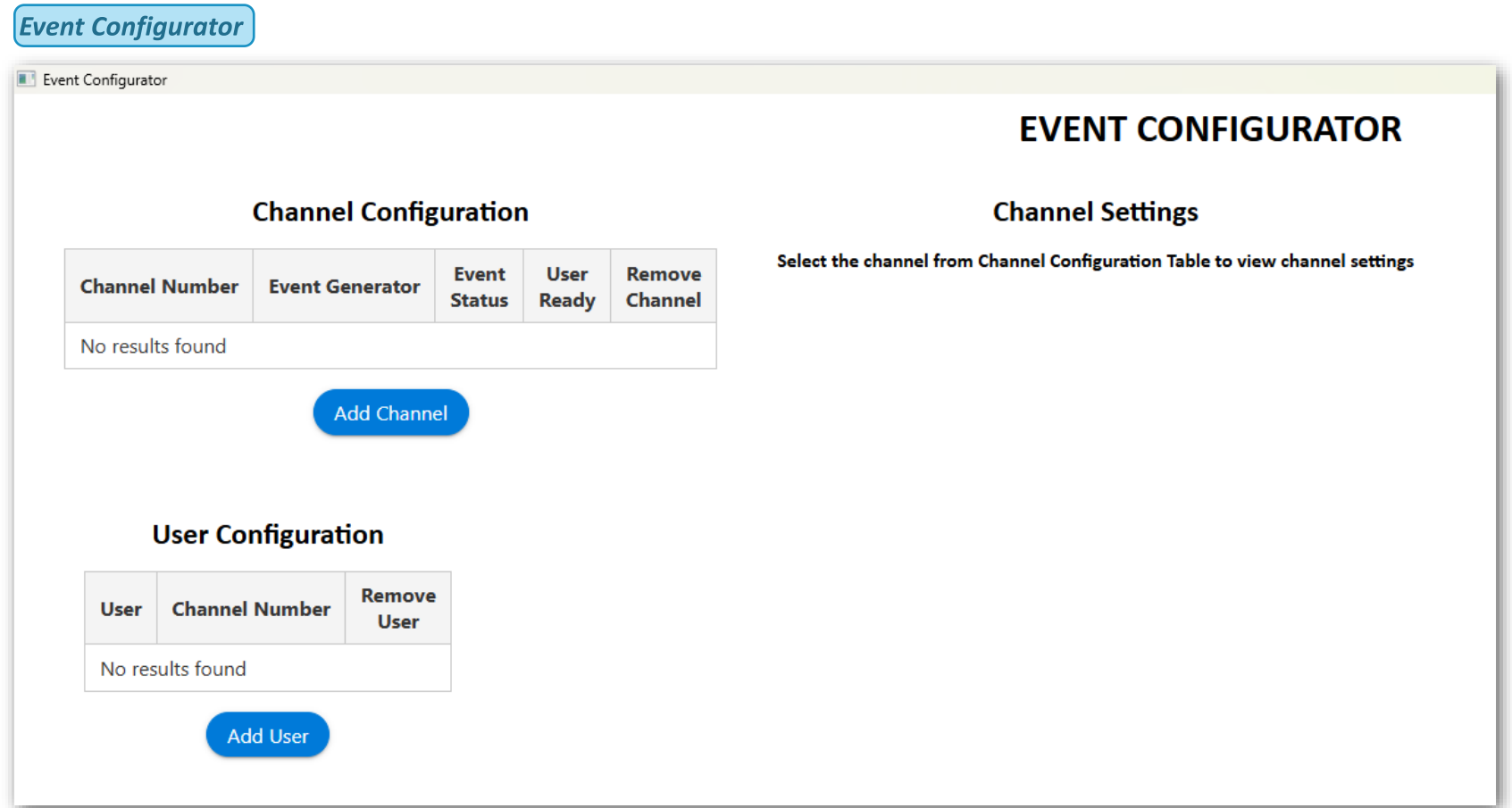
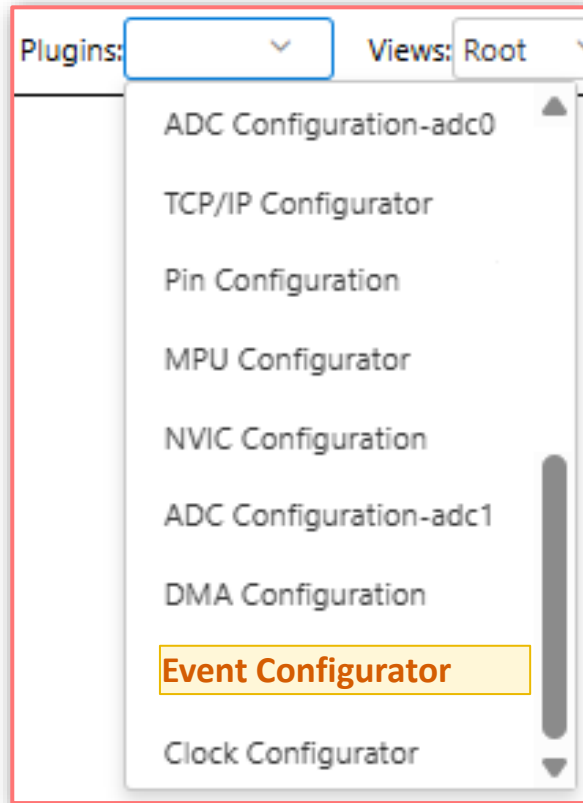
MCC Harmony Interface

- **Project Graph** → **Plugins** : **DMA Configuration** (Open a new window after clicked)



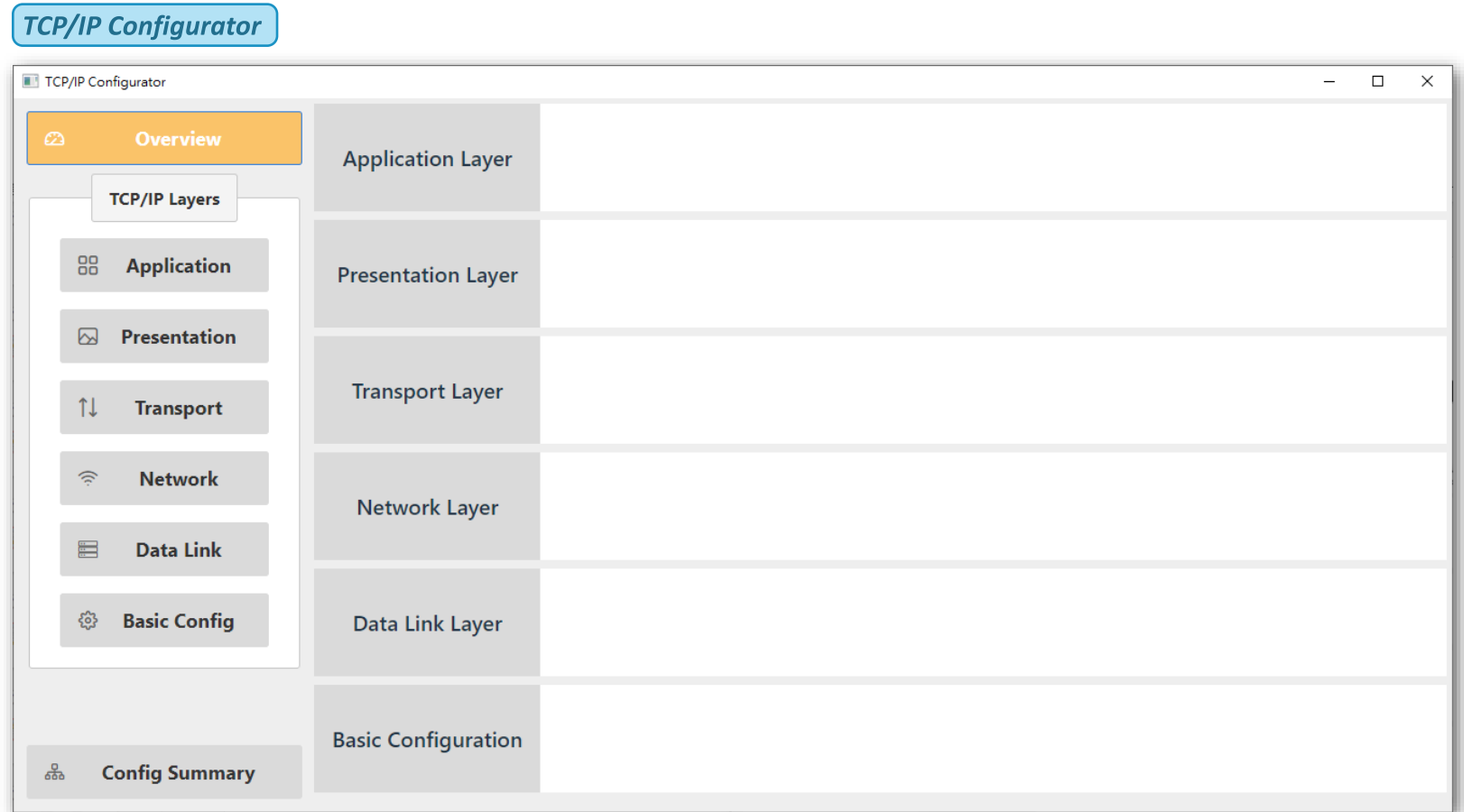
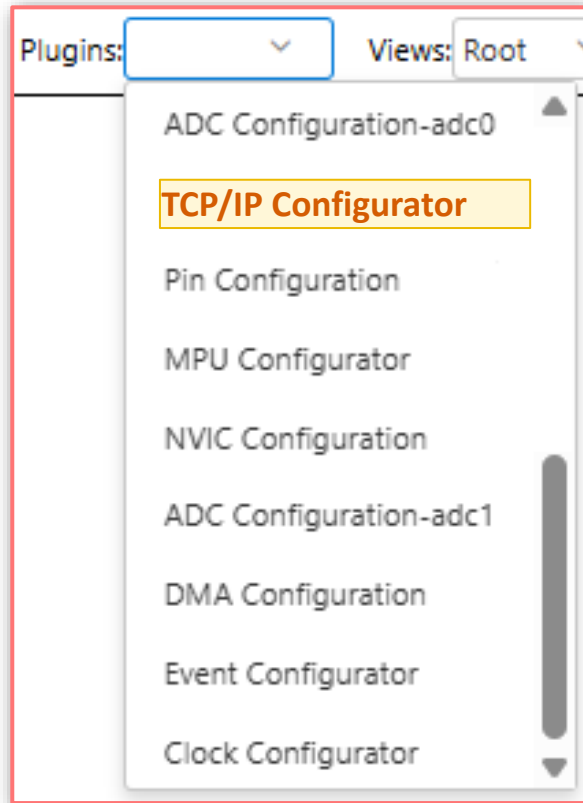
MCC Harmony Interface

- **Project Graph** → **Plugins** : **Event Configurator** (Open a new window after clicked)



MCC Harmony Interface

- **Project Graph** → **Plugins** : **TCP/IP Configurator** (Open a new window after clicked)



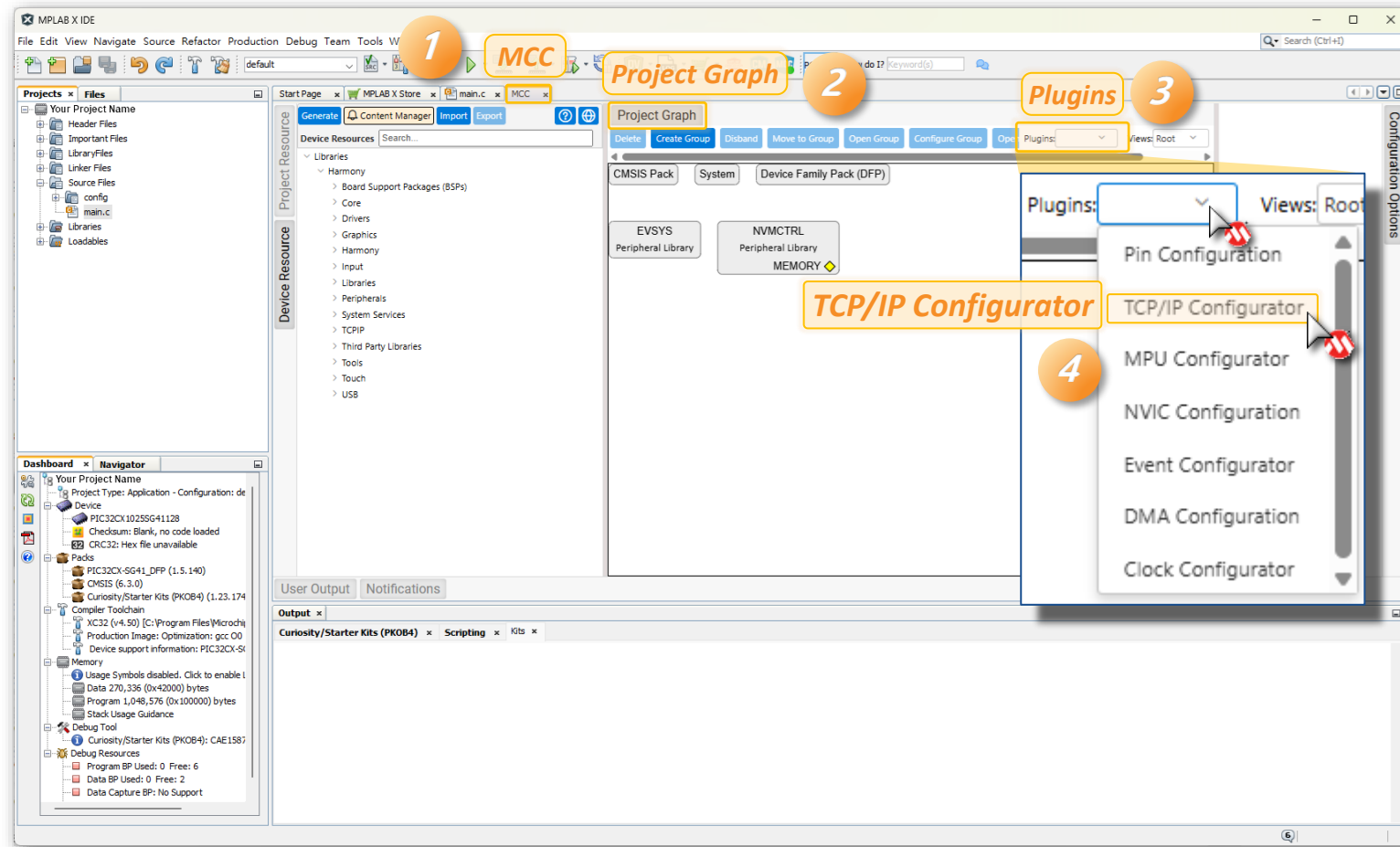
MPLAB[®] Code Configurator (MCC) TCP/IP Configurator

An Easy Configuration Tool for TCP/IP

MCC - TCP/IP Configurator

Step 1

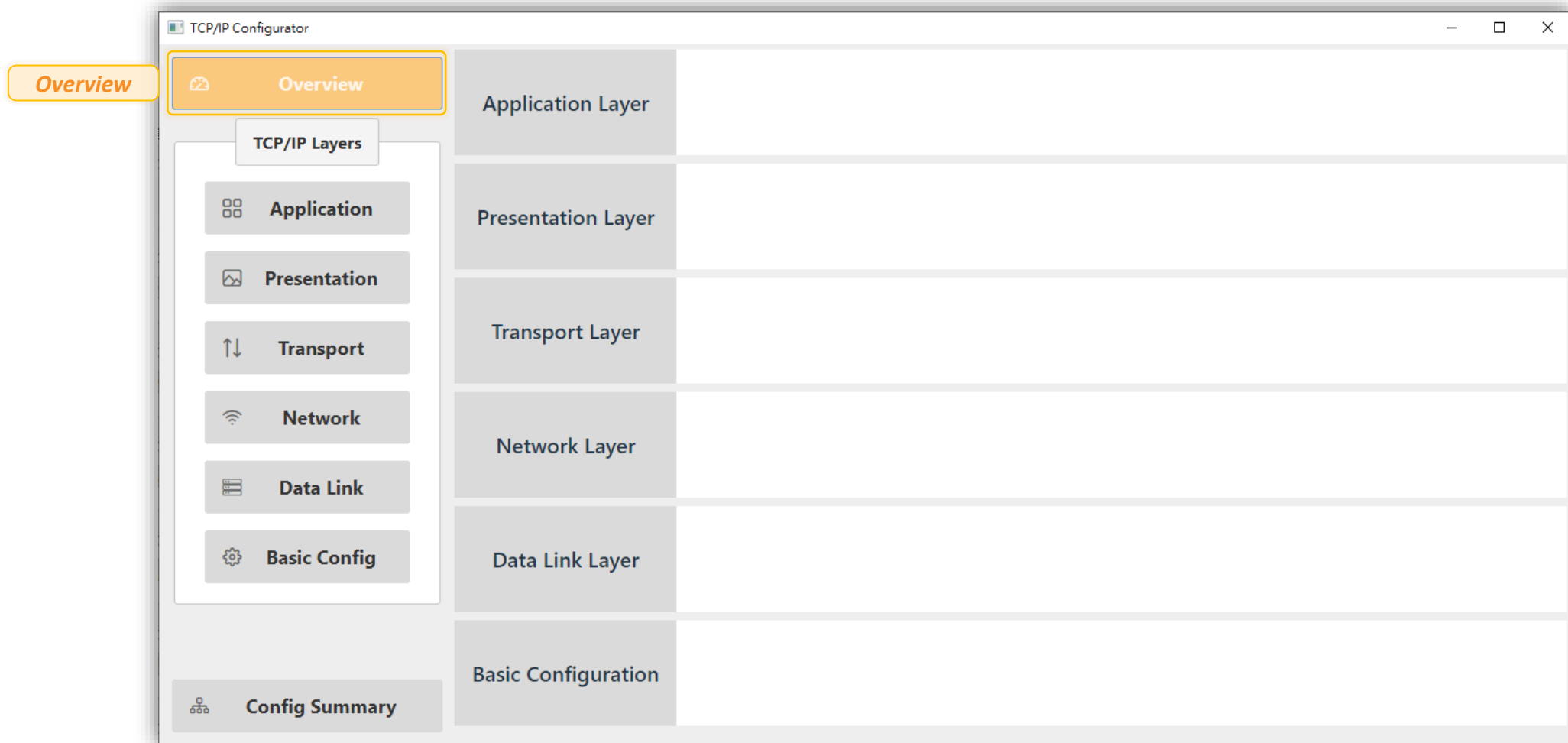
- Under **MCC**, click on **Project Graph** > **Plugins** > **TCP/IP Configurator**.



MCC - TCP/IP Configurator

Step 2

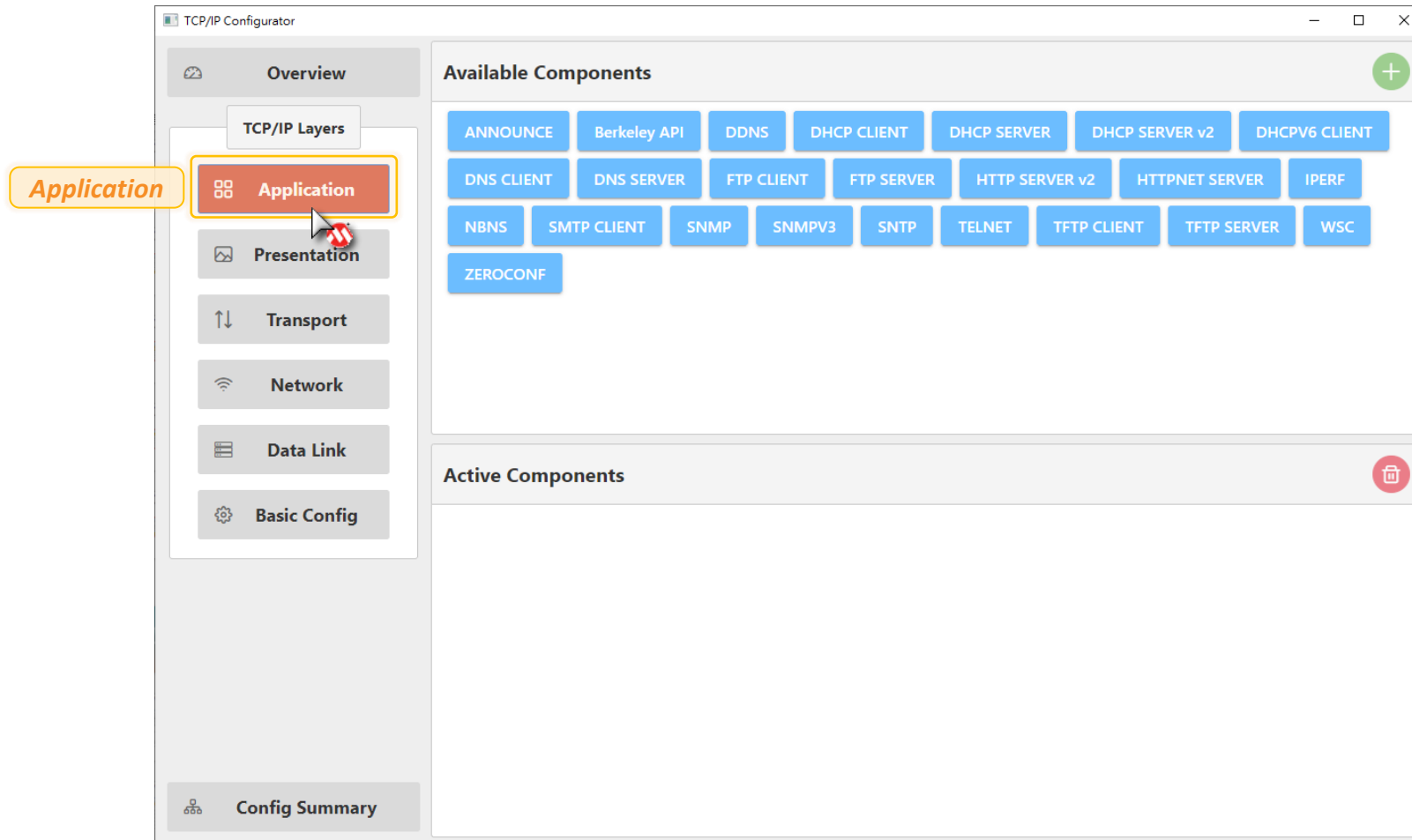
- The **TCP/IP Configurator** interface appears.



MCC - TCP/IP Configurator

Step 3

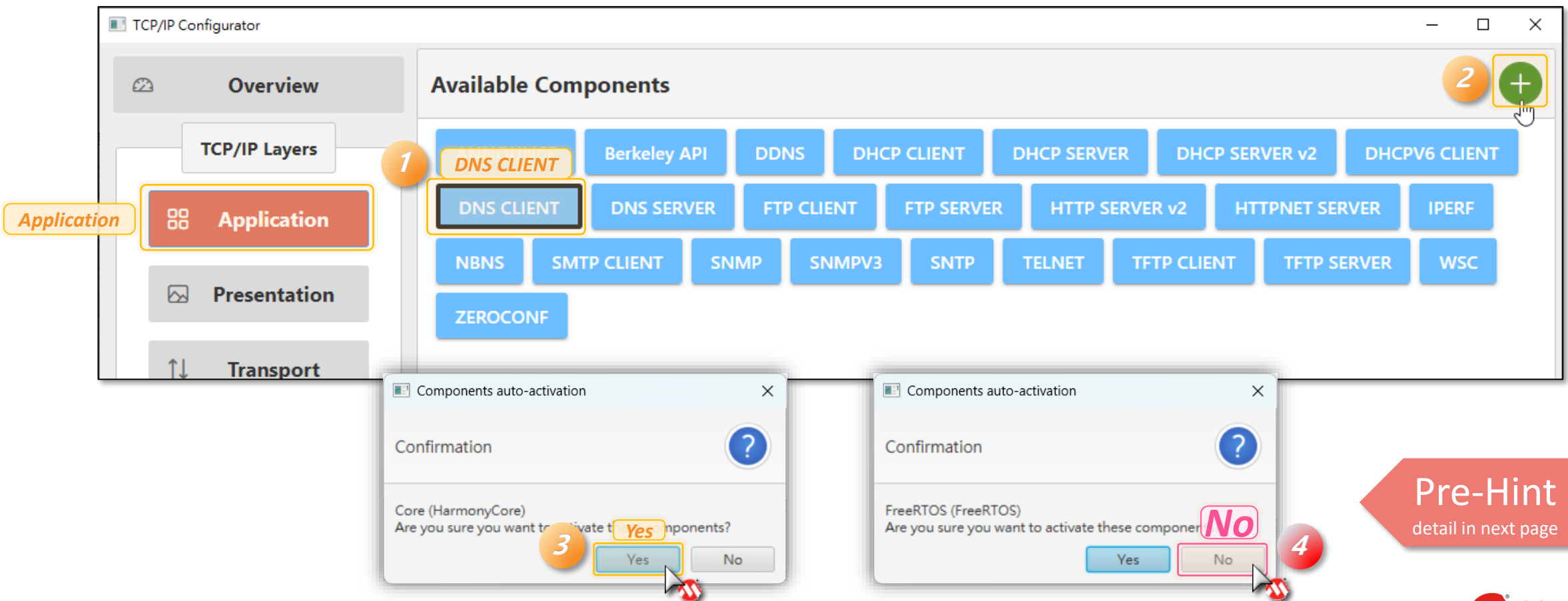
- Click the **Application** button to select the **Application Layer**.



MCC - TCP/IP Configurator

Step 4

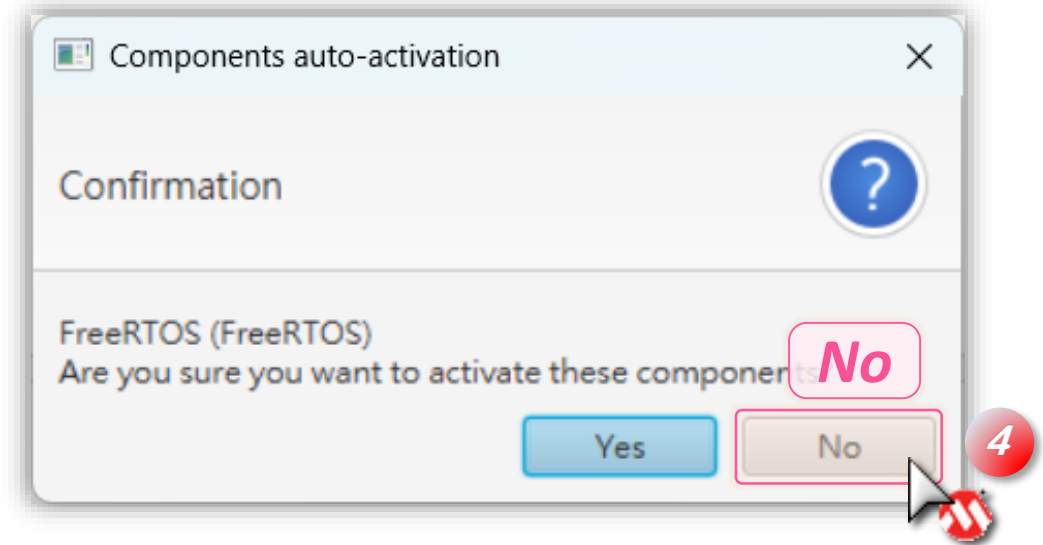
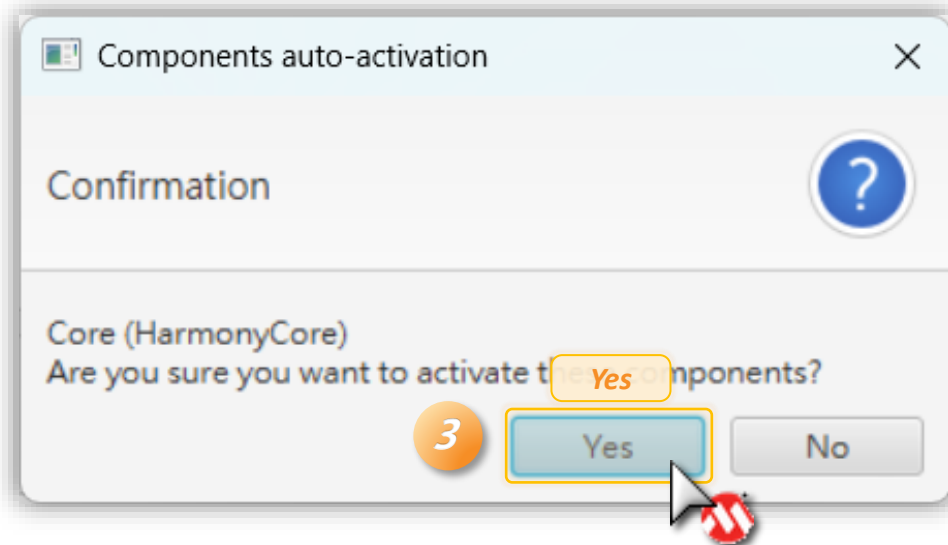
- Select "**DNS Client**" from the **Available Components** and click  to add it to the **Active Components**.



MCC - TCP/IP Configurator

Step 5

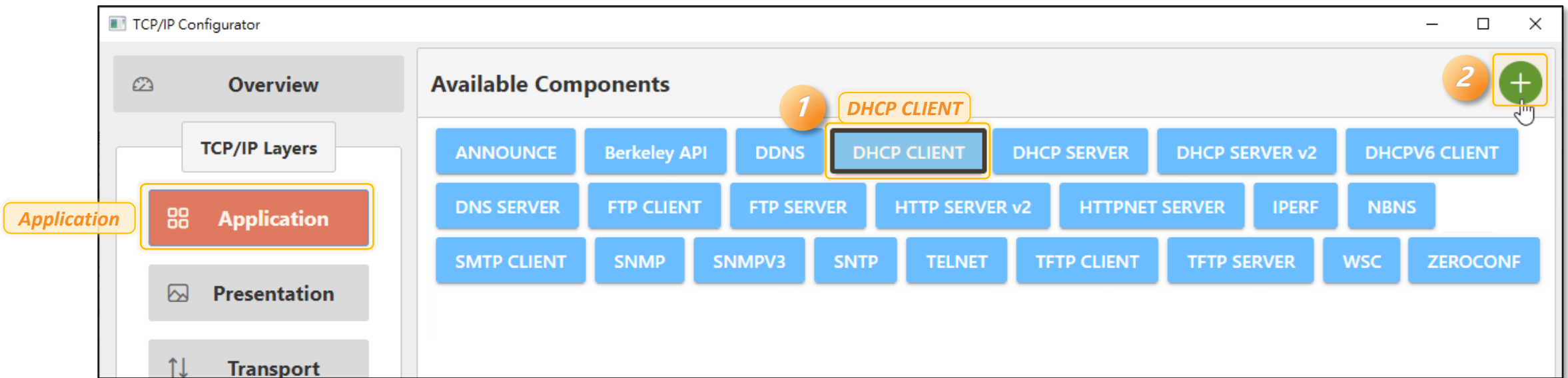
- Once you click  , two dialog boxes will pop up.
- Click **Yes** to add the "Harmony Core" component.
- Click **No** to **not use** the "FreeRTOS". (We use bare metal to complete the project)



MCC - TCP/IP Configurator

Step 6

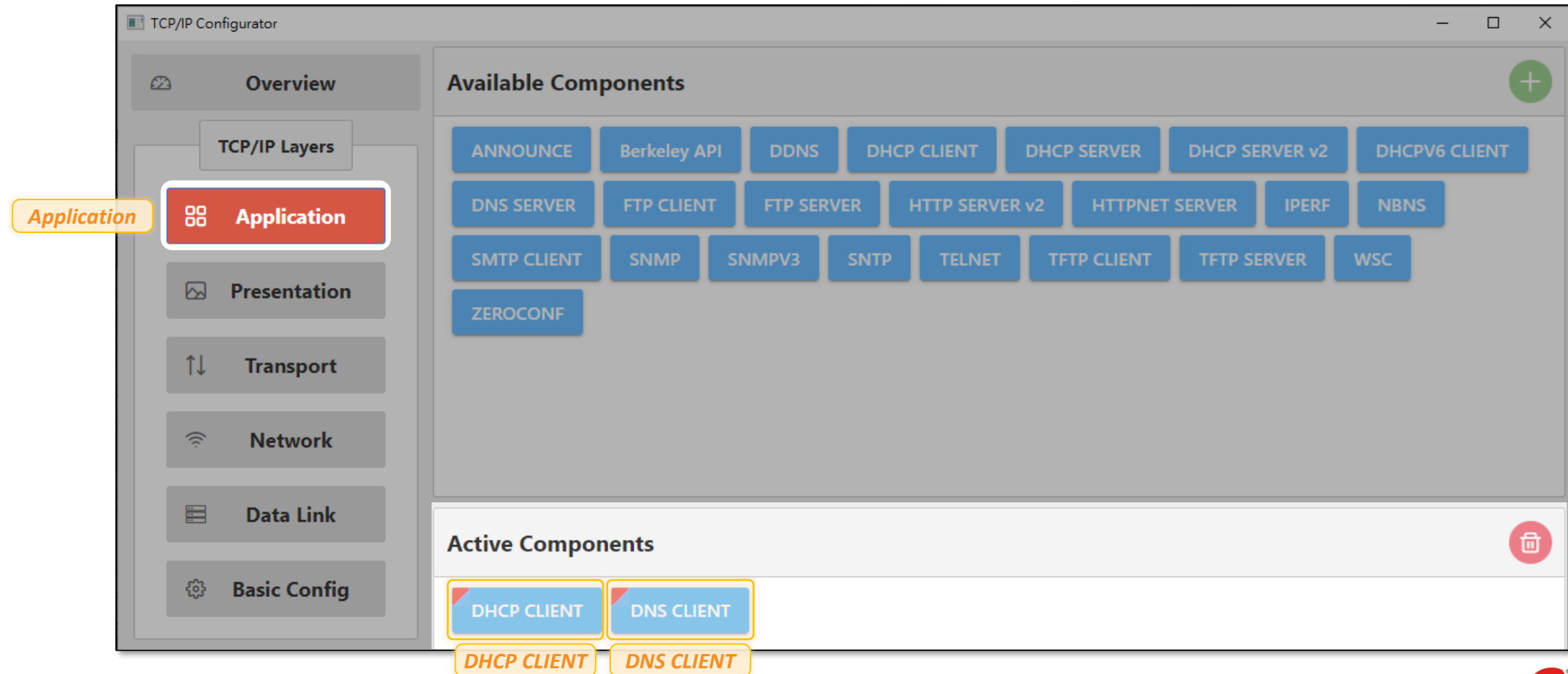
- Select another module "**DHCP Client**" from the **Available Components** and click  to add it to the **Active Components**.



MCC - TCP/IP Configurator

Step 7 – Result Check

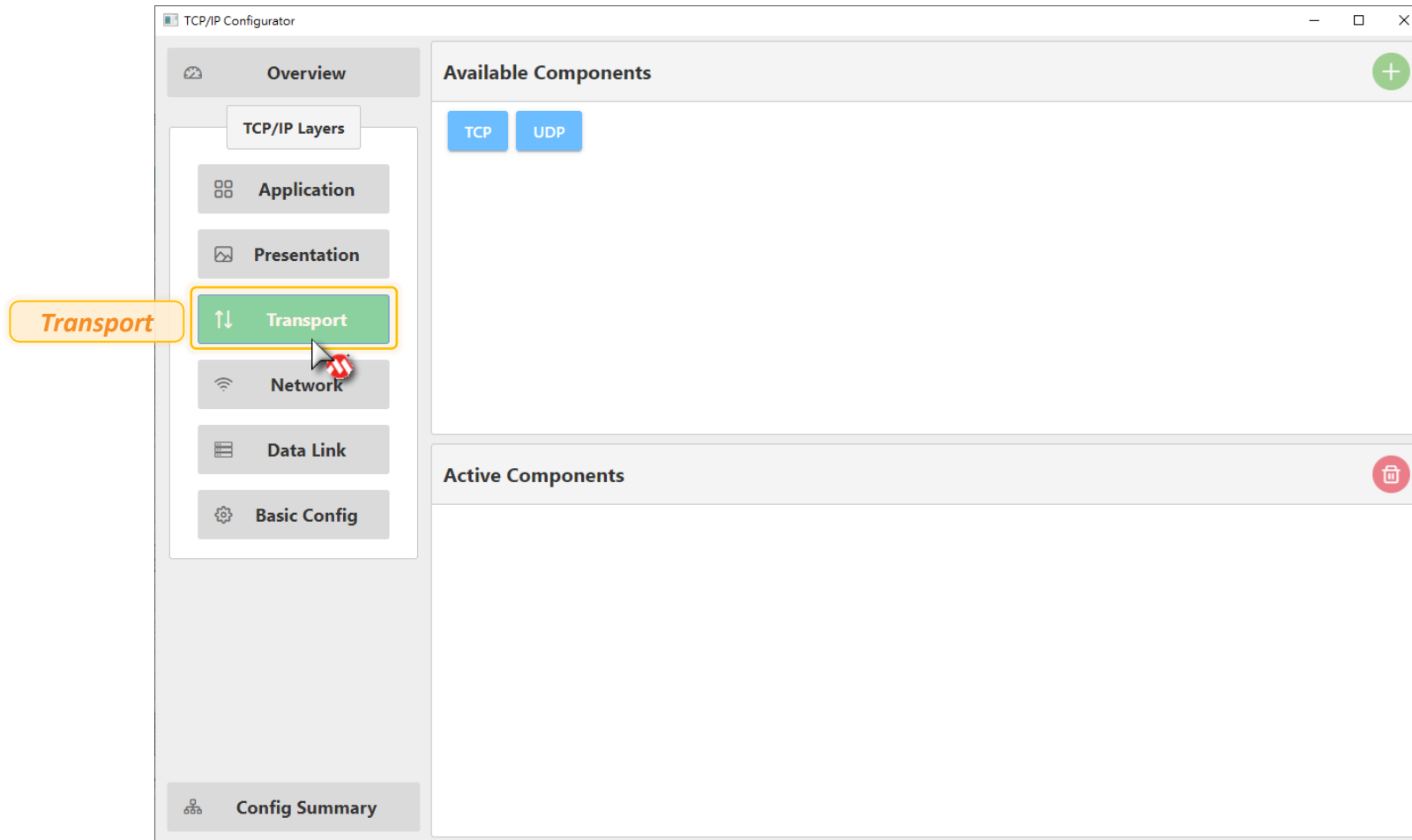
- The "**DHCP Client**" and "**DNS Client**" will be added to the **Active Components**.



MCC - TCP/IP Configurator

Step 8

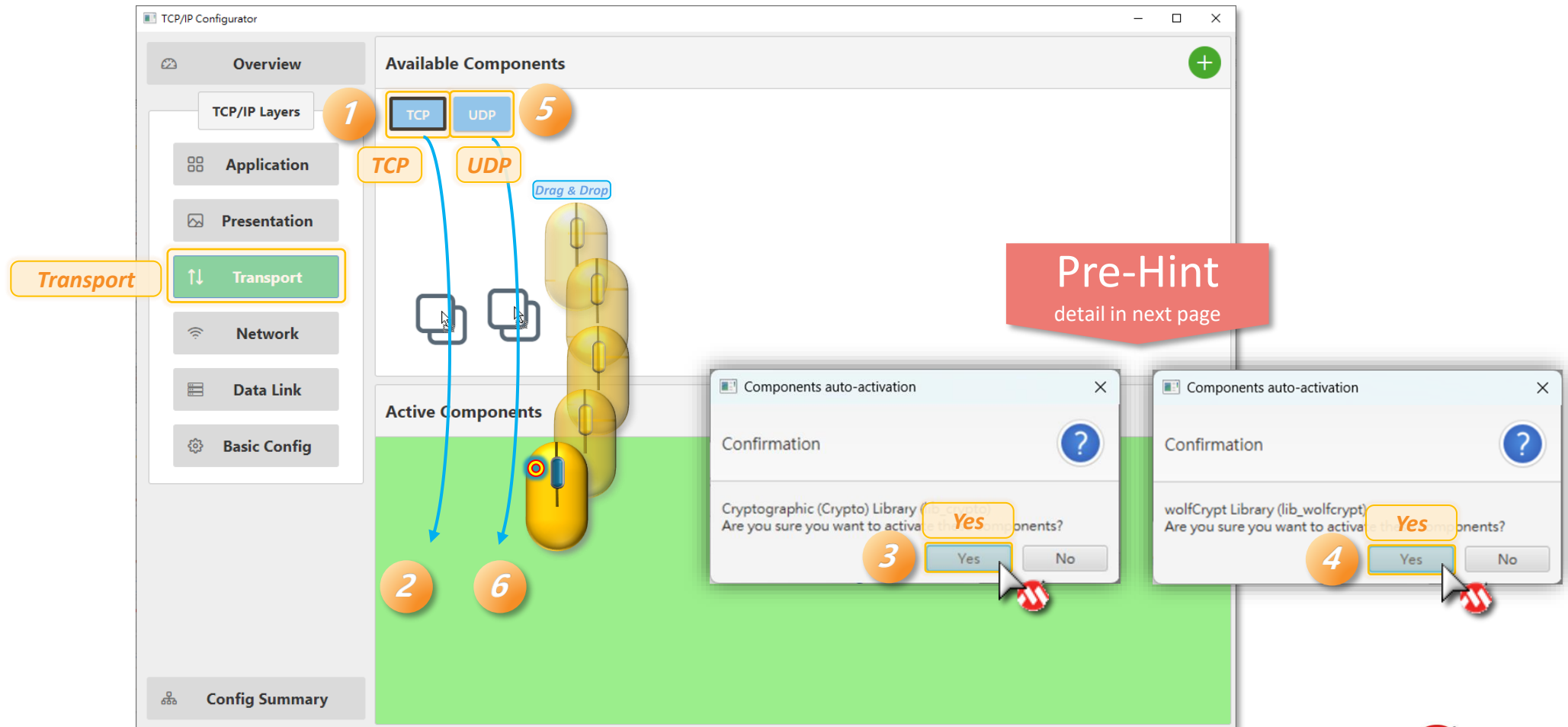
- Click the **Transport** button to select the **Transport Layer**.



MCC - TCP/IP Configurator

Step 9

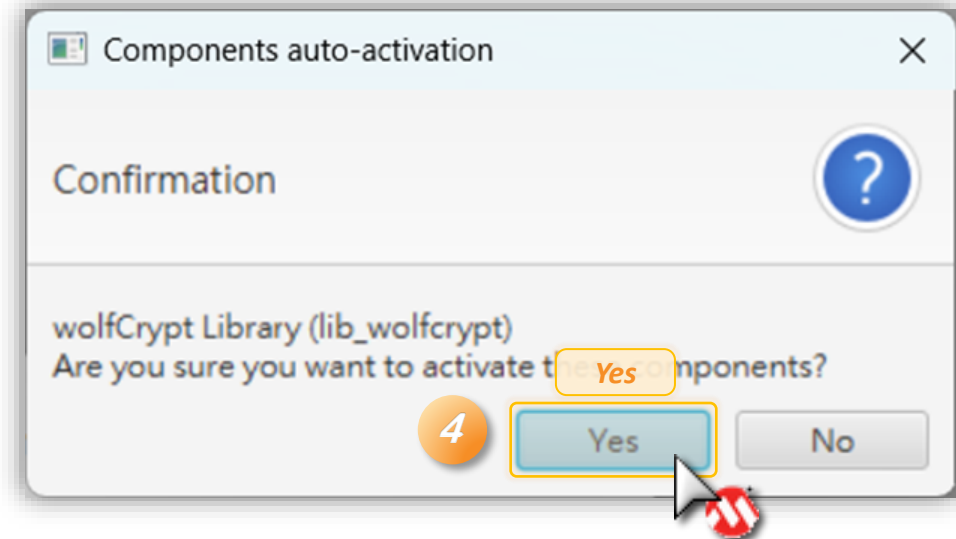
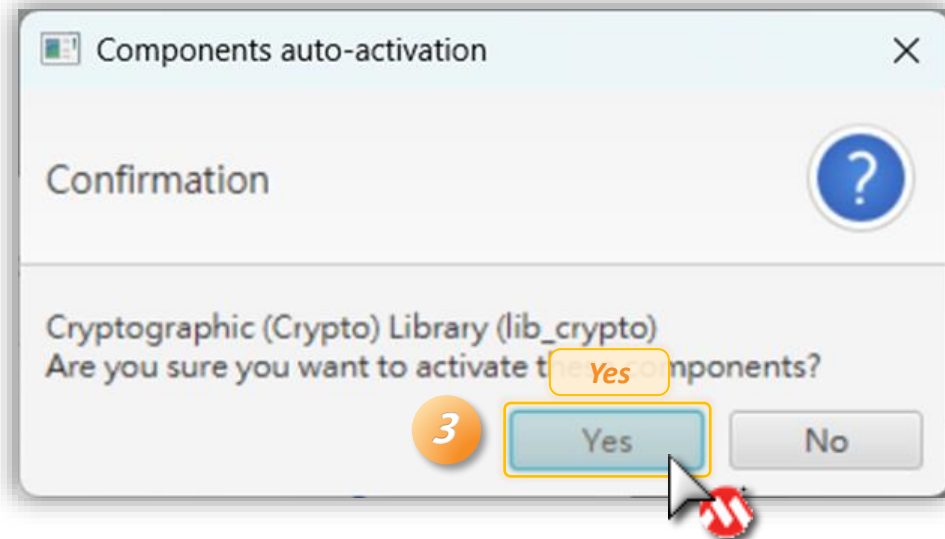
- You can also add "TCP" and "UDP" to the **Active Component** by dragging and dropping.



MCC - TCP/IP Configurator

Step 10

- Once you drop the component to **Active Components**, three dialog boxes will pop up.
- Click **Yes** to add the "**lib_crypto**" component.
- Click **Yes** to add the "**lib_wolfcrypt**" component.



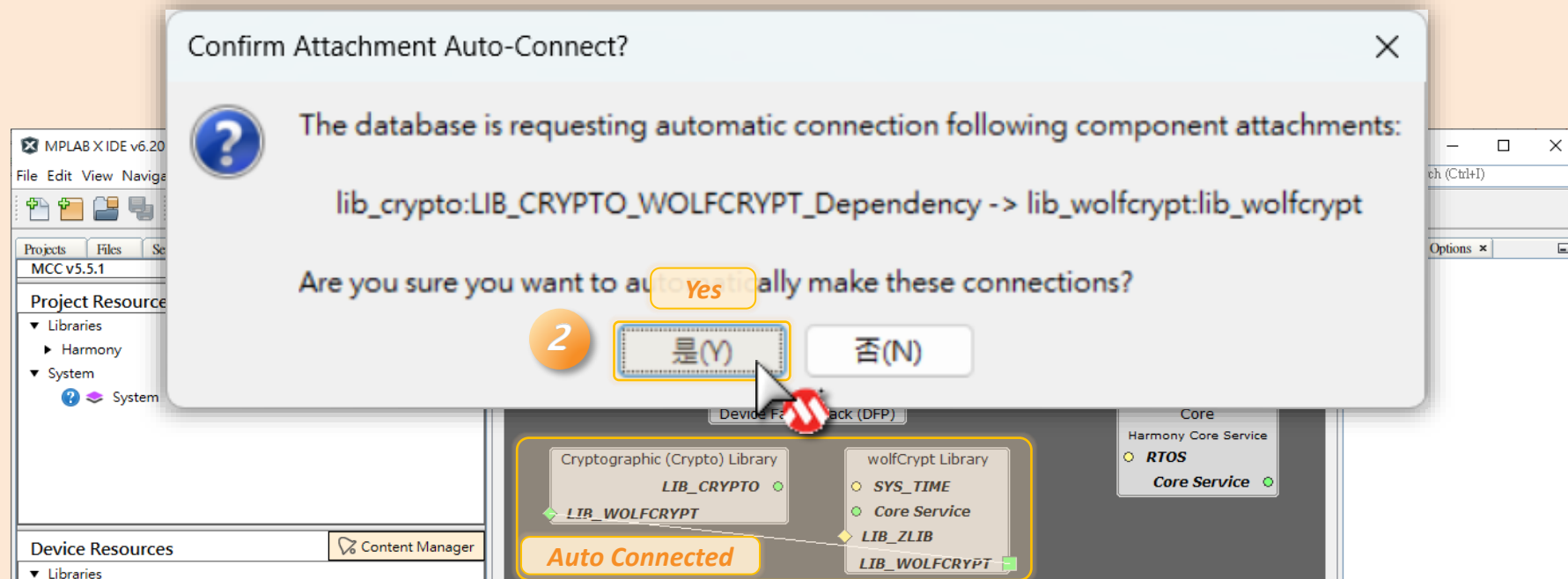
MCC - TCP/IP Configurator

Step in Older Version MCC

- The third dialog box popup appears behind the **TCP/IP Configurator** window. Therefore, minimize the **TCP/IP Configurator** window to view this popup.



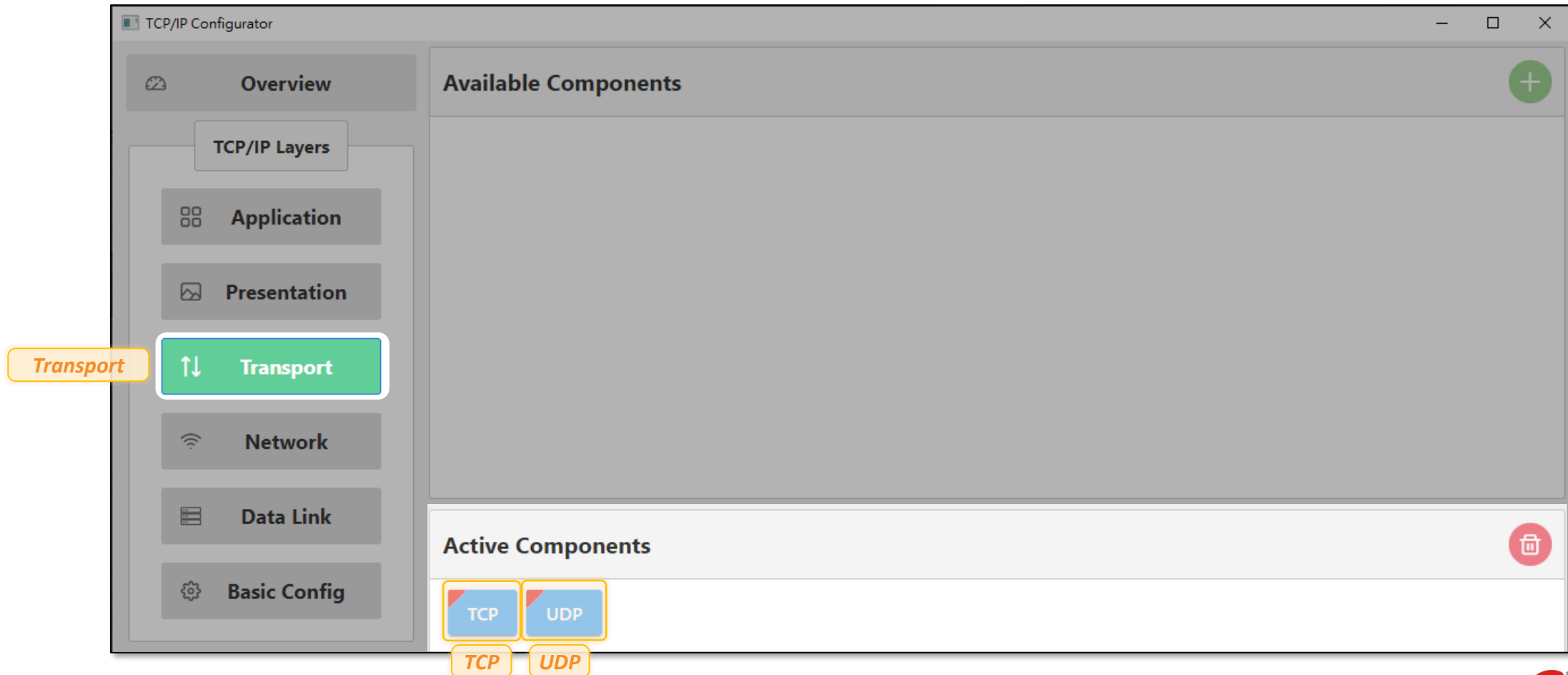
- Click **Yes** to **Auto-Connect** the "**lib_crypto**" and "**lib_wolfcrypt**".



MCC - TCP/IP Configurator

Step 11 – Result Check

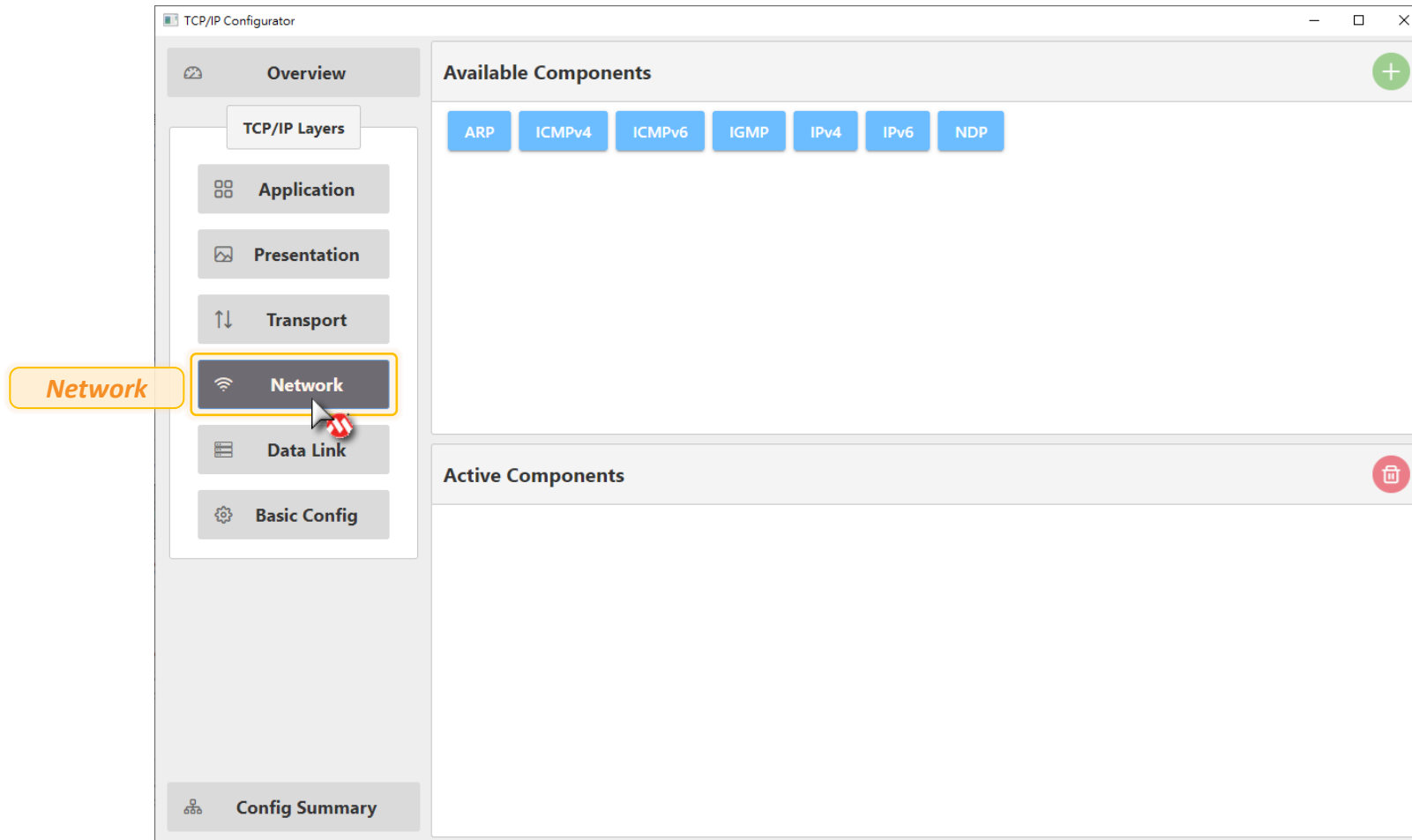
- The "TCP" and "UDP" success to be added to the **Active Components**.



MCC - TCP/IP Configurator

Step 12

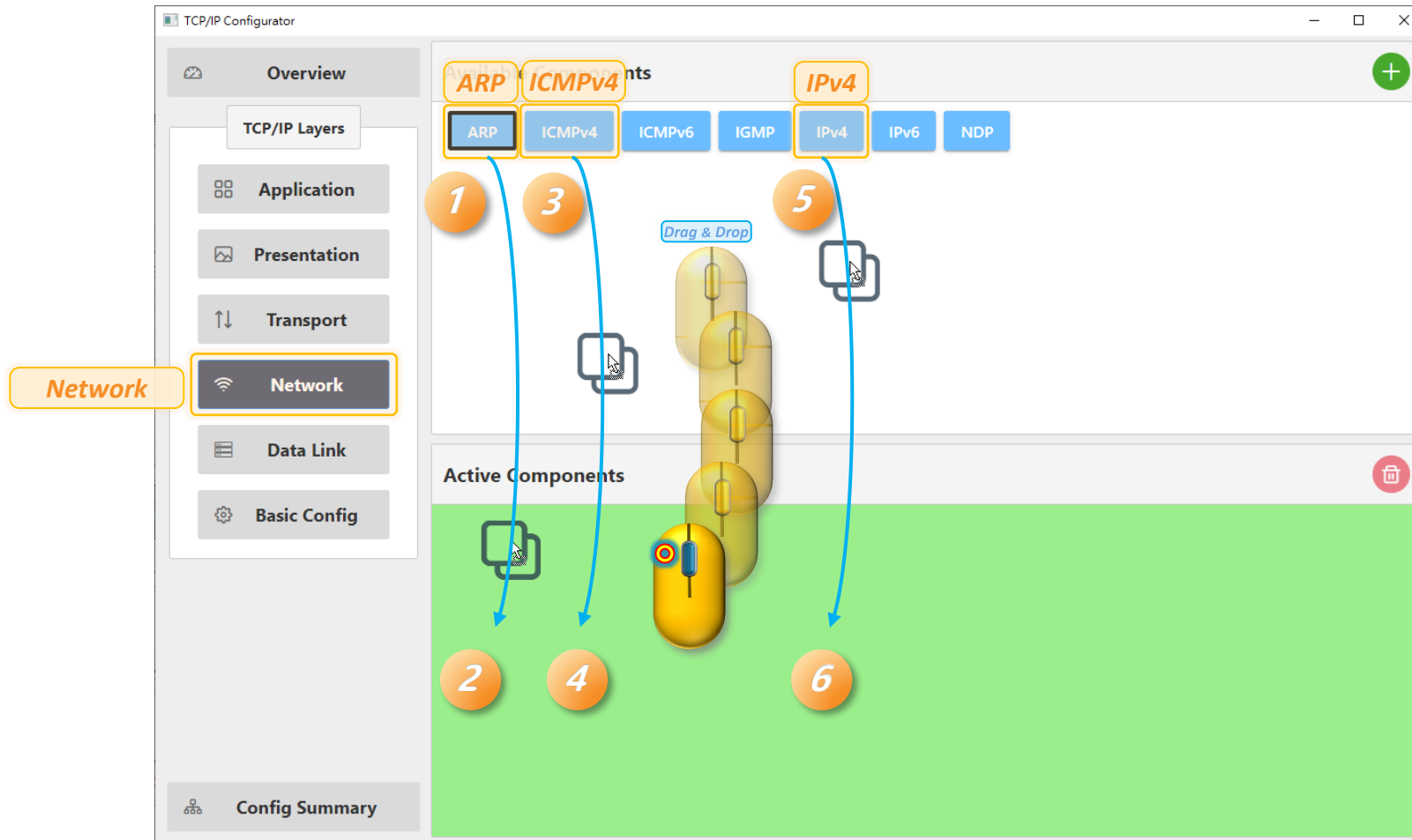
- Click the **Network** button to select the **Network Layer**.



MCC - TCP/IP Configurator

Step 13

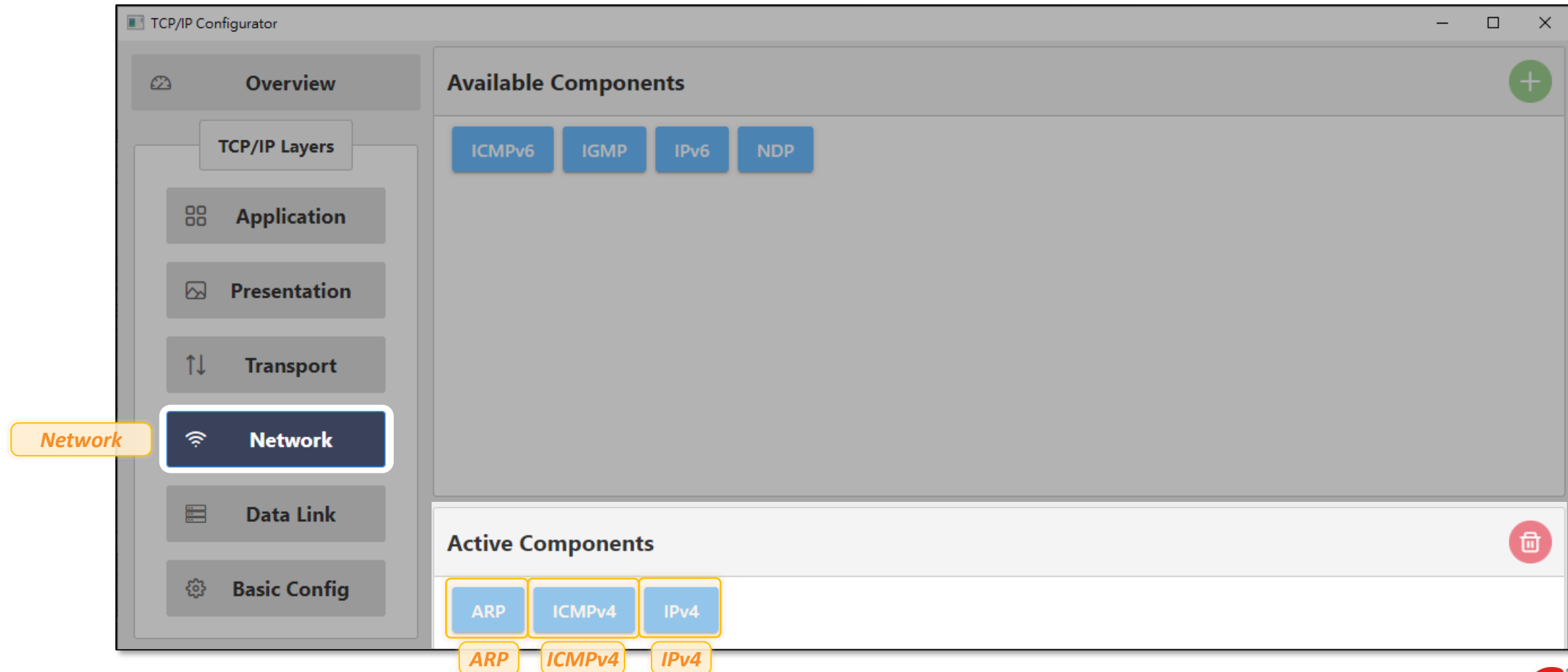
- Add "ARP", "IPv4" and "ICMPv4" to the **Active Component** by dragging and dropping.



MCC - TCP/IP Configurator

Step 14 – Result Check

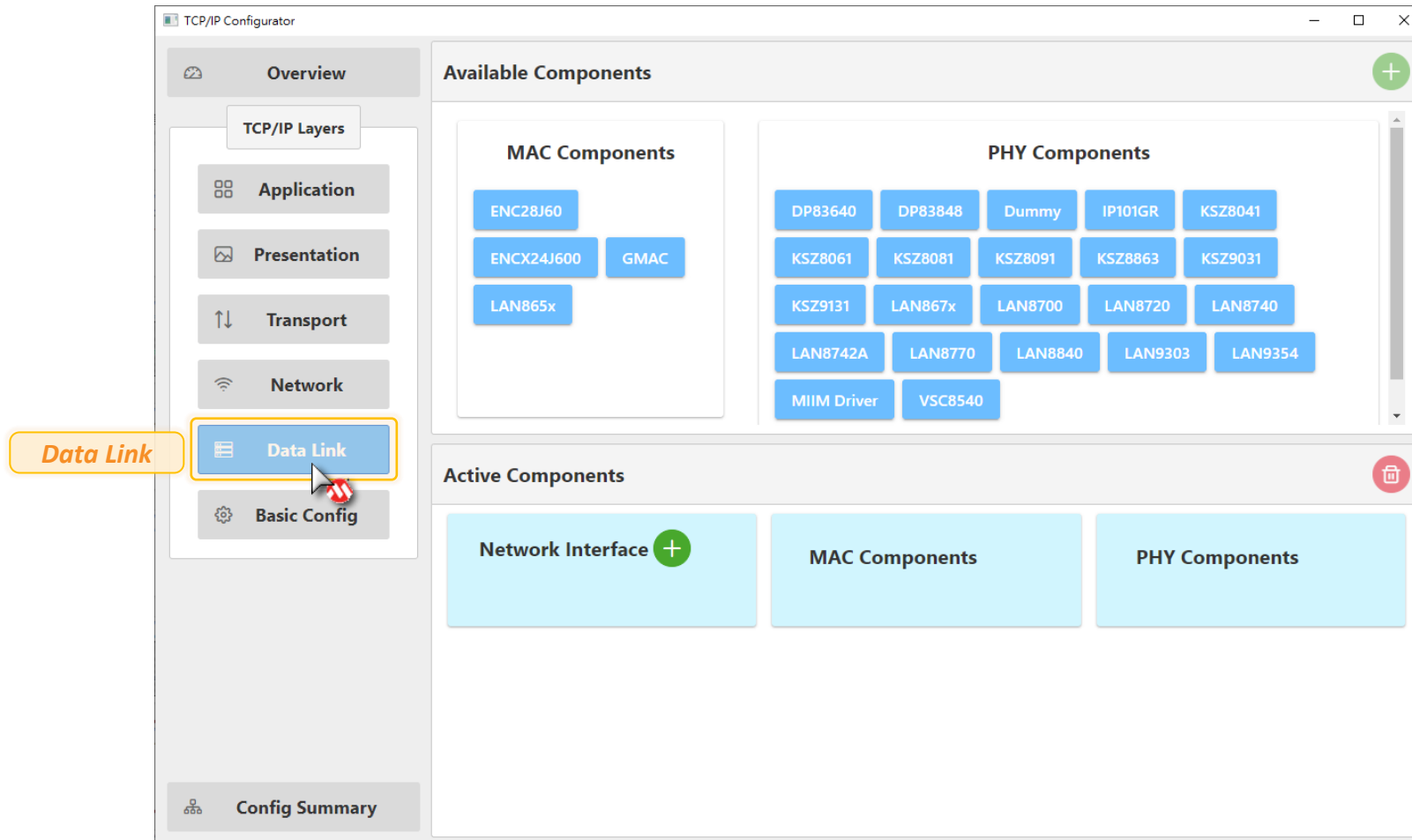
- The "ARP", "IPv4" and "ICMPv4" success to be added to the **Active Components**.



MCC - TCP/IP Configurator

Step 15

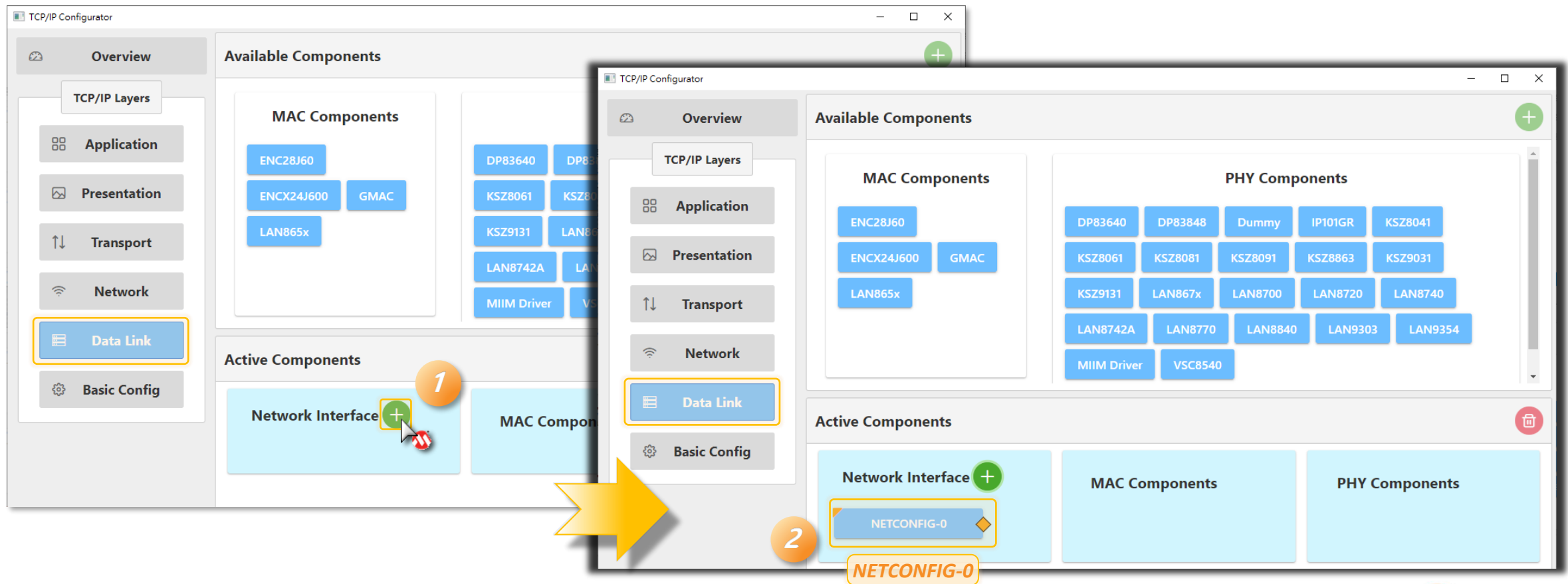
- Click the **Data Link** button to select the **Data Link Layer**.



MCC - TCP/IP Configurator

Step 16

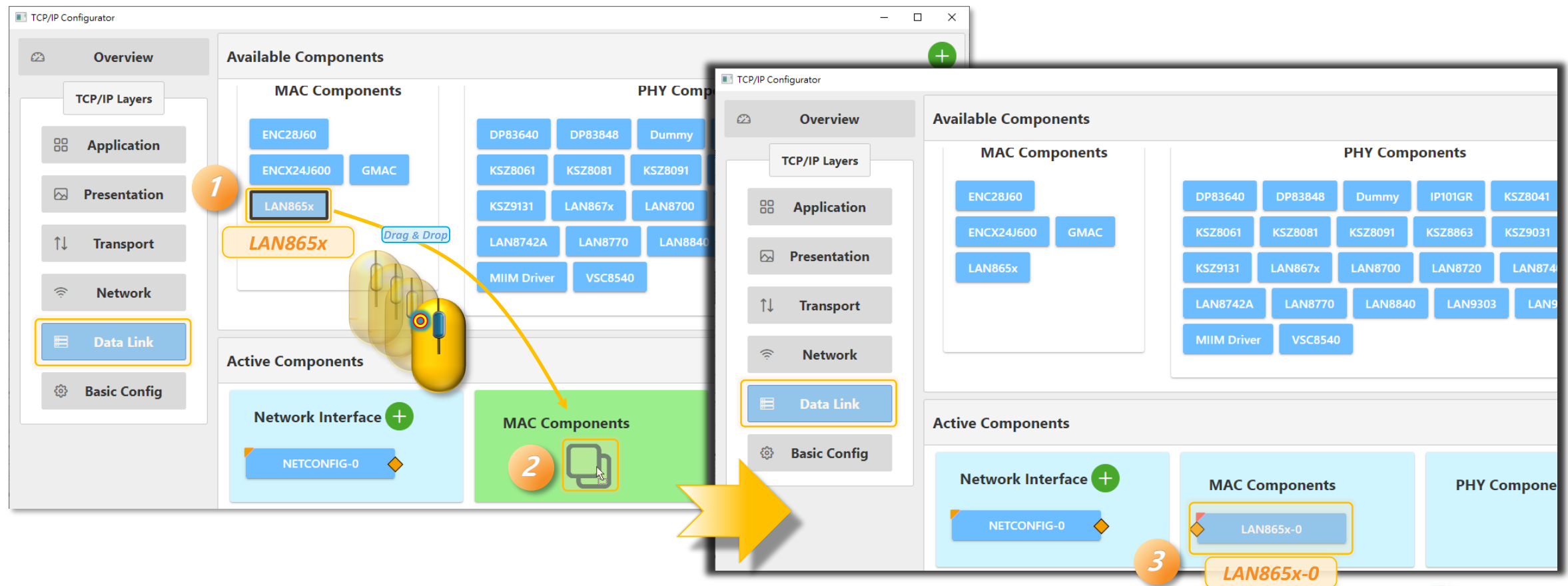
- Click the **+** on the "Network Interface" in the Active Components area.
- A new instance **NETCONFIG-0** will be added in Network Interface.



MCC - TCP/IP Configurator

Step 17

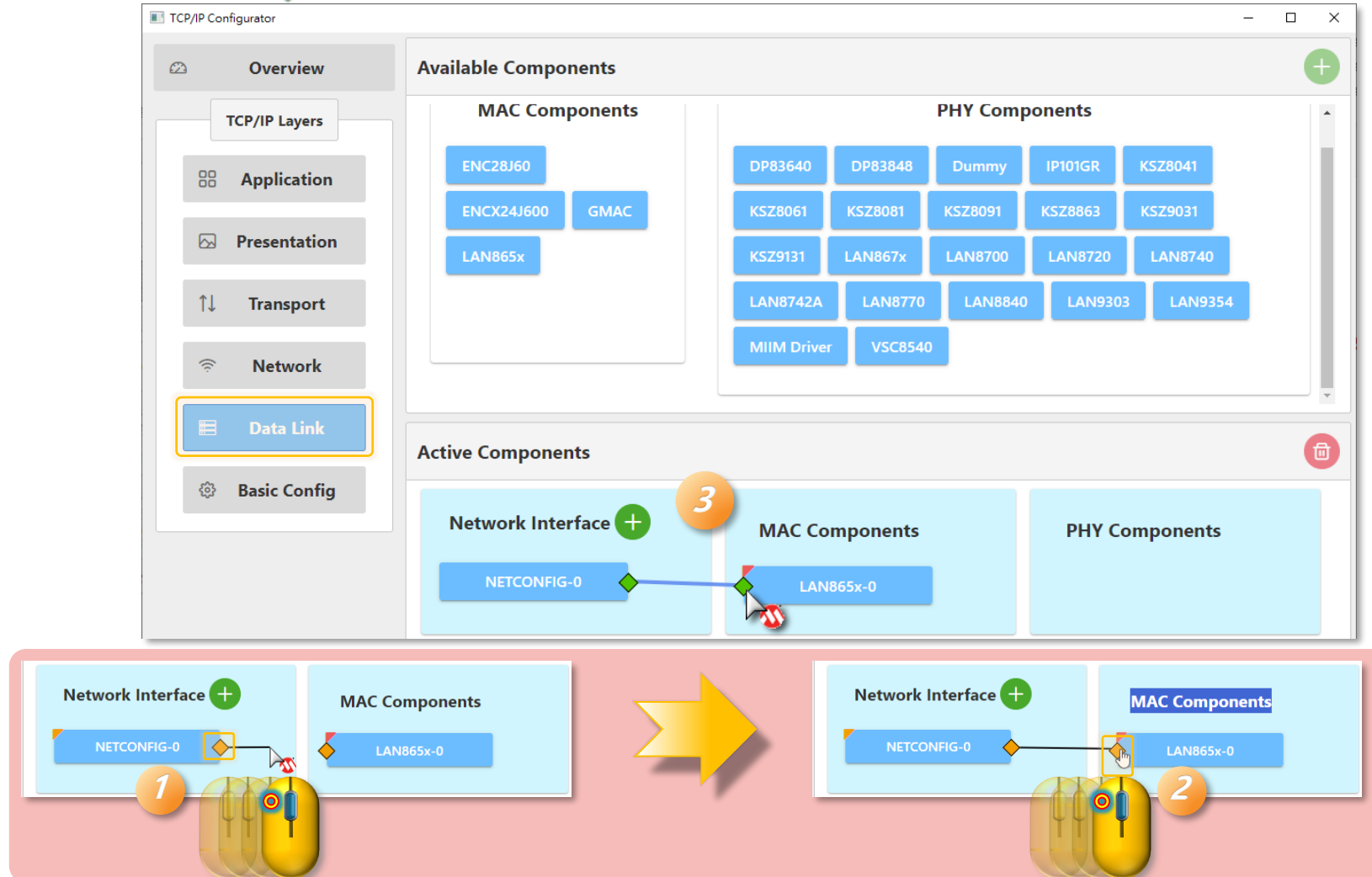
- Drag and drop "LAN865x" from **MAC Components** of **Available Components** to the **Active Components** area, a new instance **LAN865x-0** will add to **MAC Components**.



MCC - TCP/IP Configurator

Step 18

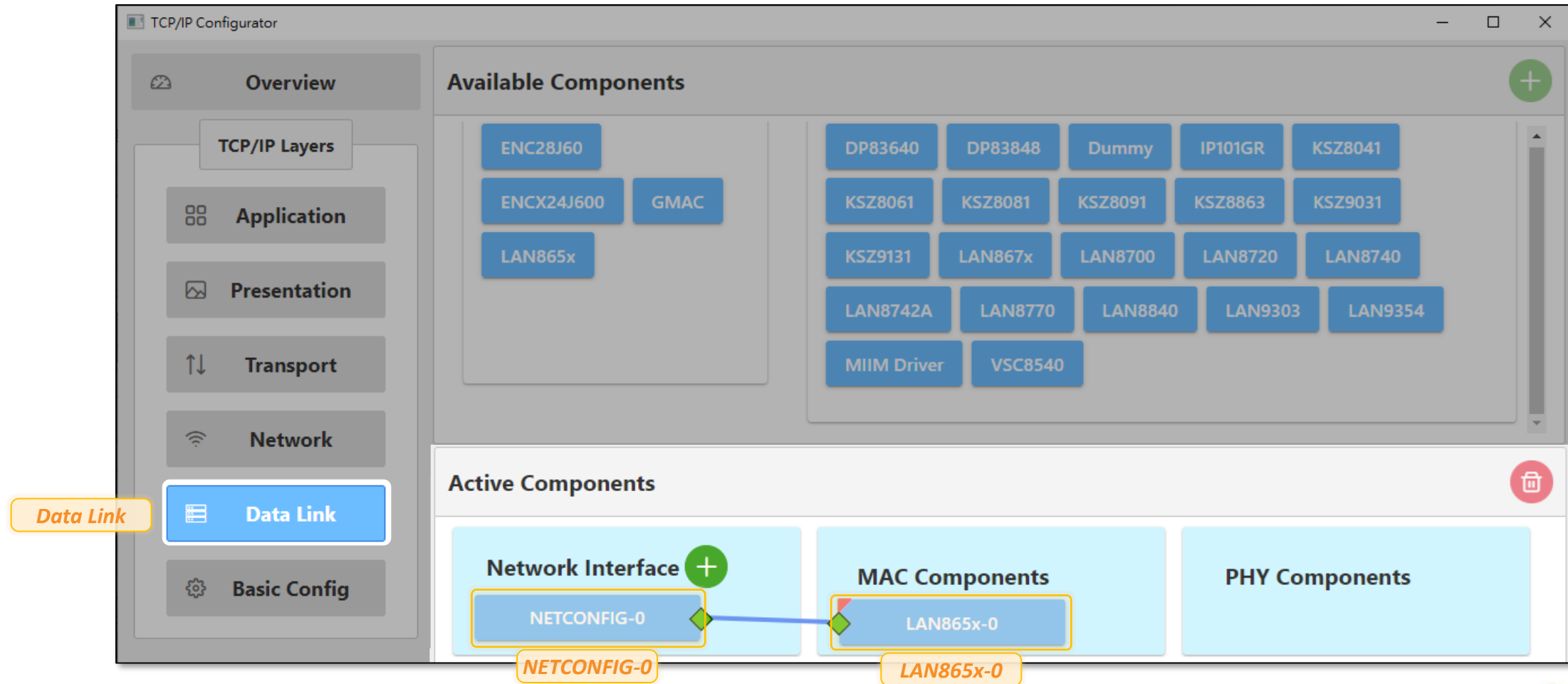
- Drag a line from the  diamond to connect "NETCONFIG-0" and "LAN865x-0".



MCC - TCP/IP Configurator

Step 19 – Result Check

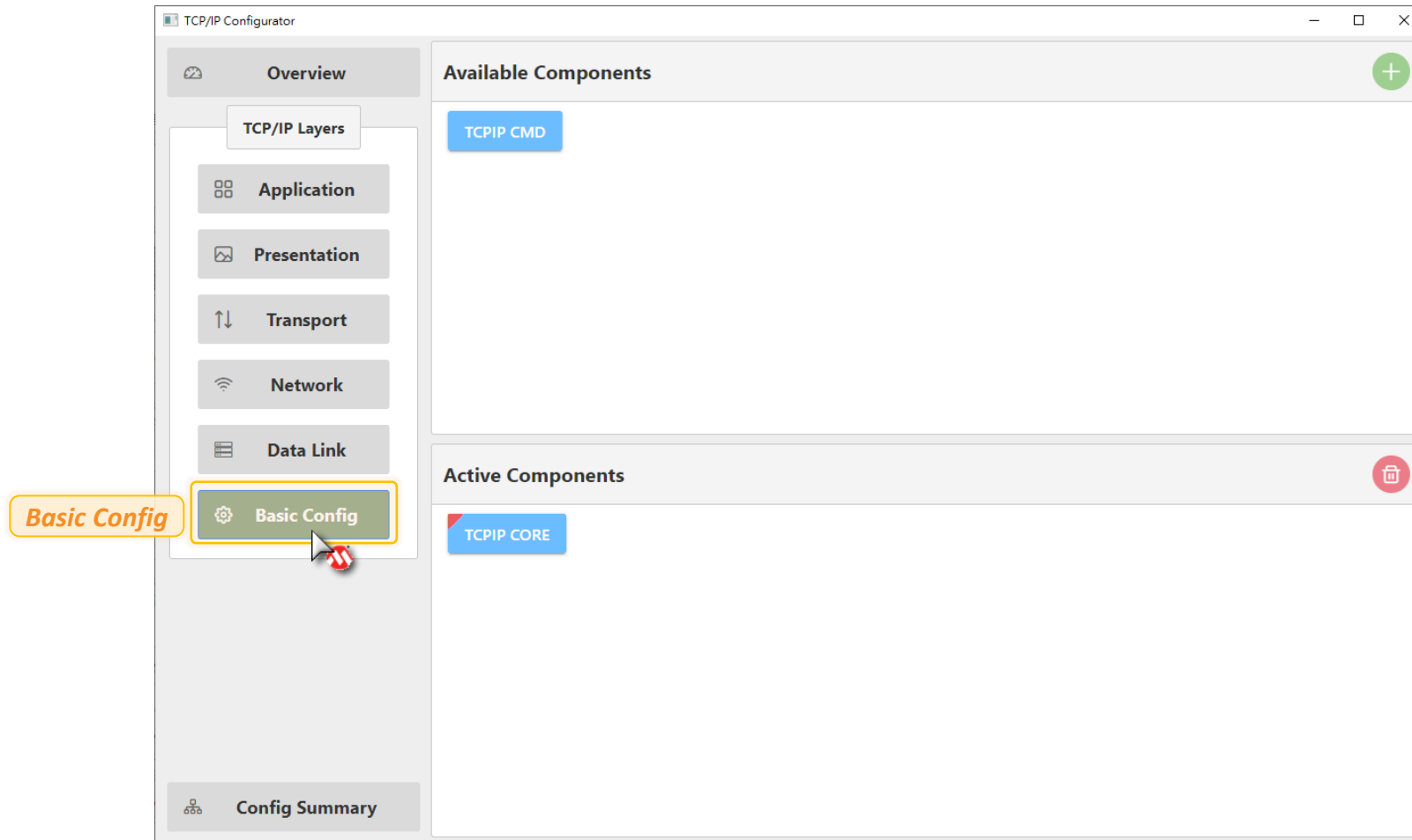
- The "NETCONFIG-0" and "LAN865x-0" success connected in Active Components.



MCC - TCP/IP Configurator

Step 20

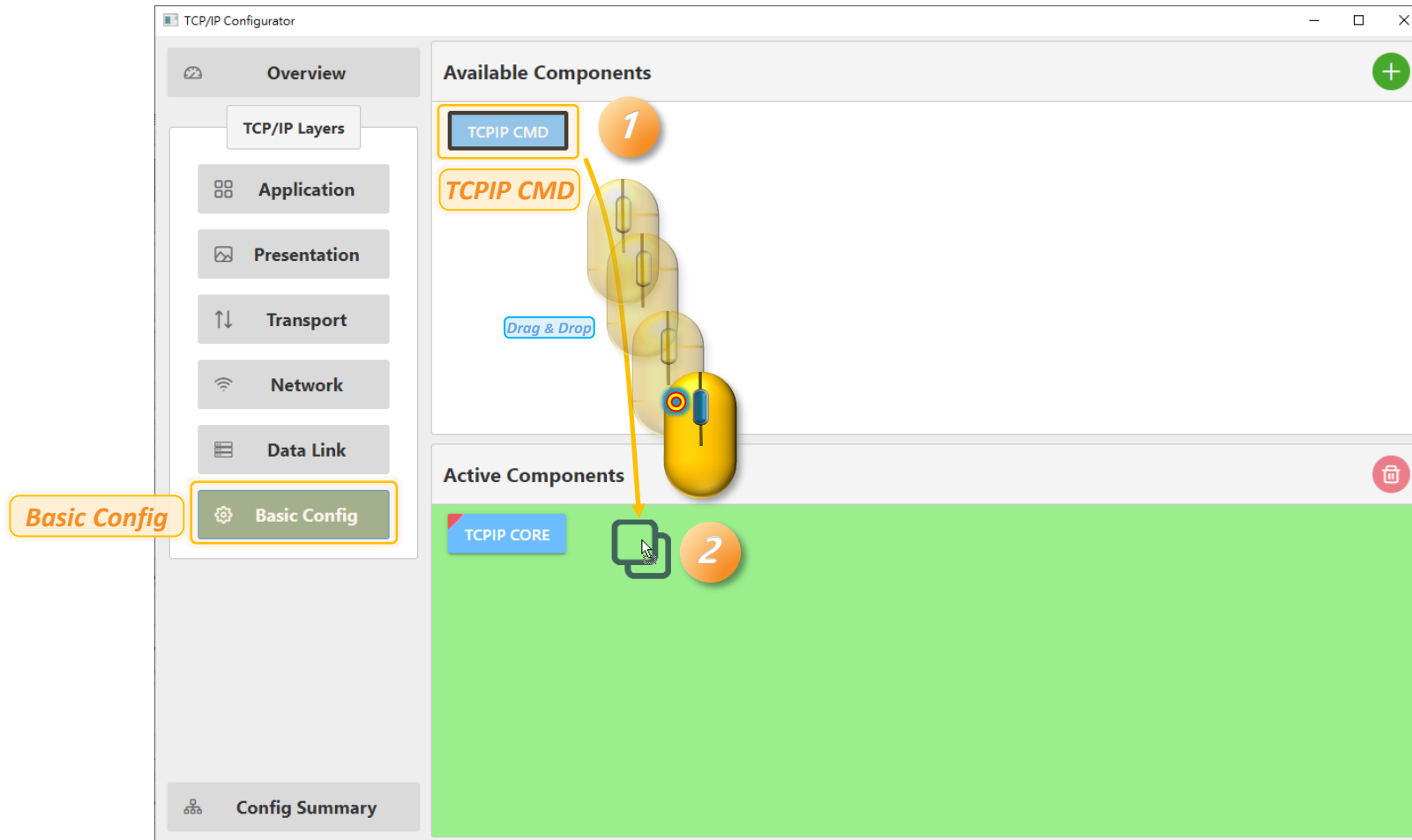
- Click the **Basic Config** button to select the **Basic Config Layer**.



MCC - TCP/IP Configurator

Step 21

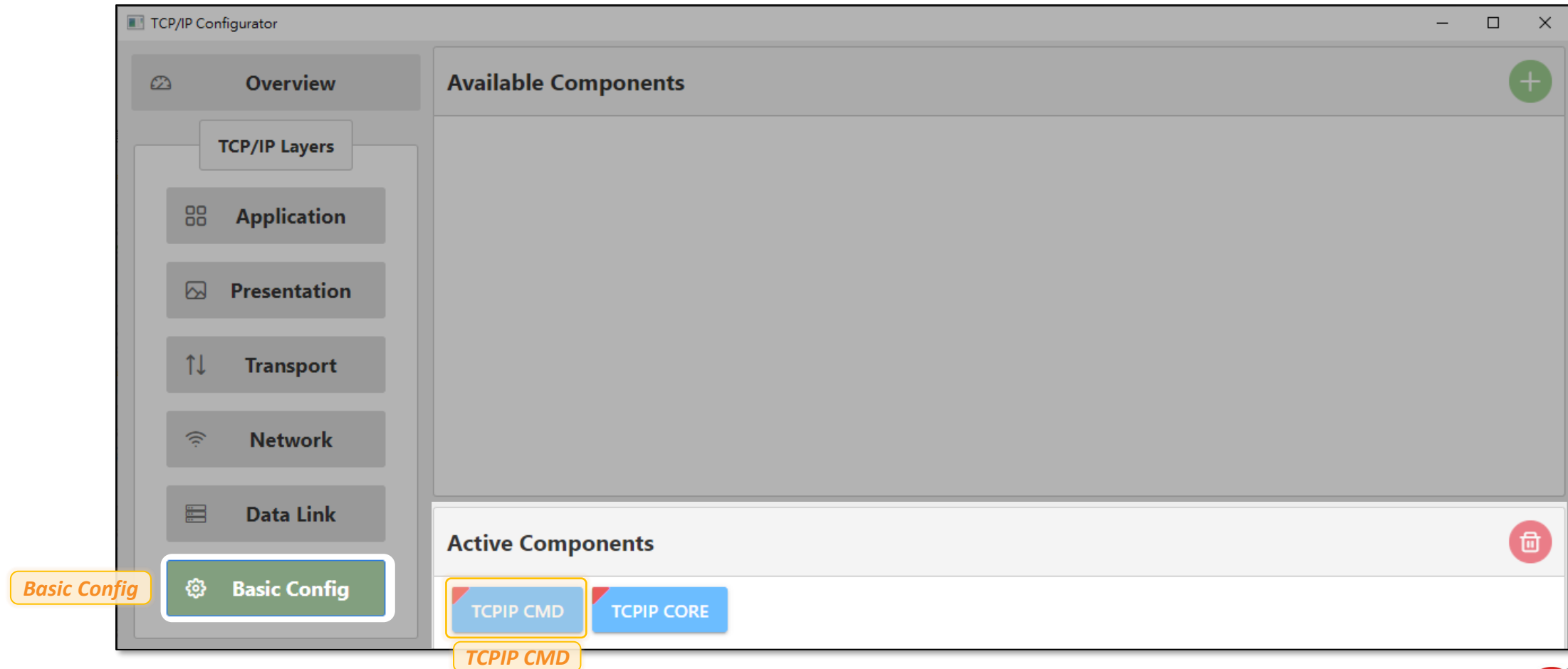
- Drag and Drop "**TCPIP CMD**" into **Active Components**.



MCC - TCP/IP Configurator

Step 22 – Result Check

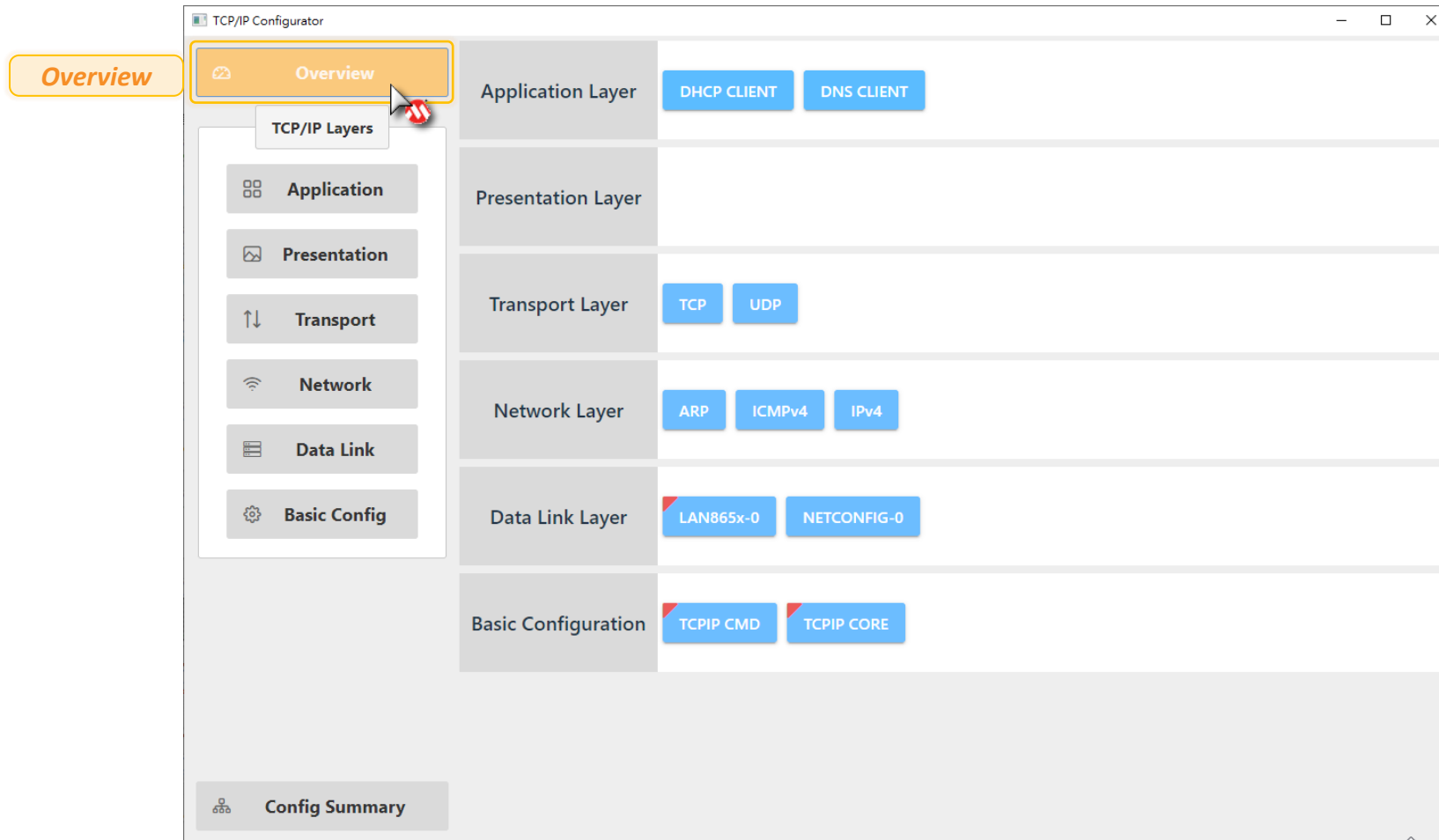
- The "**TCPIP CMD**" success to be added to the **Active Components**.



MCC - TCP/IP Configurator

Step 23

- Click the **Overview** button to see all added components.



MCC - TCP/IP Configurator

Step 24

- Click the **Config Summary** button to view the dependency status.

Configuration Summary

TCP/IP Modules | Interface | System

TCP/IP Stack Configuration Status

TCP/IP Layers	Enabled TCP/IP Module
Application	DHCP CLIENT DNS CLIENT
Presentation	None
Transport	TCP UDP
Network	ARP ICMPv4 IPv4
Data Link & Physical	LAN865x-0 NETCONFIG-0
Basic Configuration	TCPIP CMD TCPIP CORE

Unsatisfied Dependency Status

Required Dependency Status

LAN865x-0	: DRV_SPI (Driver)
TCPIP CMD	: SYS_COMMAND (System Service)
TCPIP CORE	: SYS_TIME (System Service)

Optional Dependency Status

TCPIP CORE	: SYS_CONSOLE (System Service)
------------	--------------------------------

Some Required Dependency Not Ready yet !

Config Summary

MPLAB® Code Configurator (MCC)

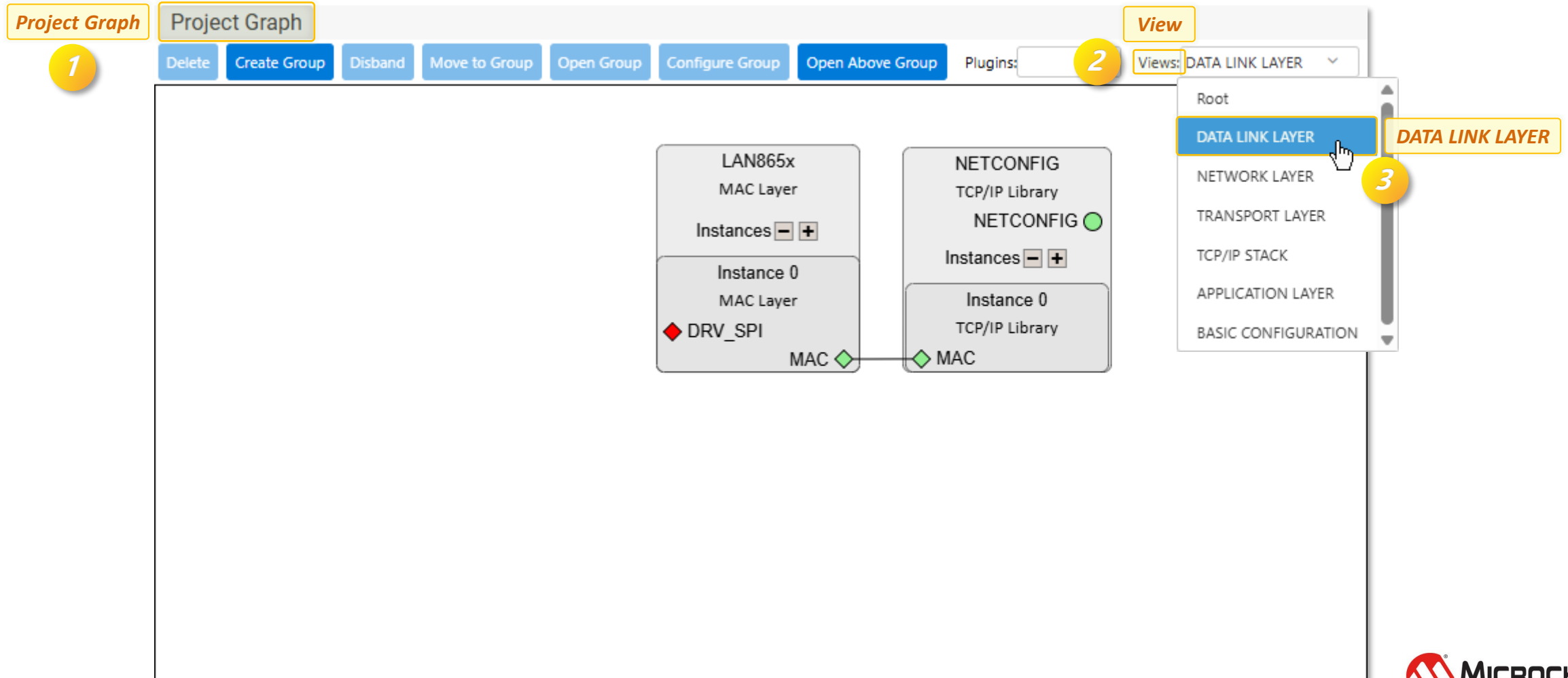
Added TCP/IP Dependency, System Driver & Peripheral Library

MCC Harmony Configuration

Added TCP/IP Dependency, System Driver & Peripheral Library

Step 1

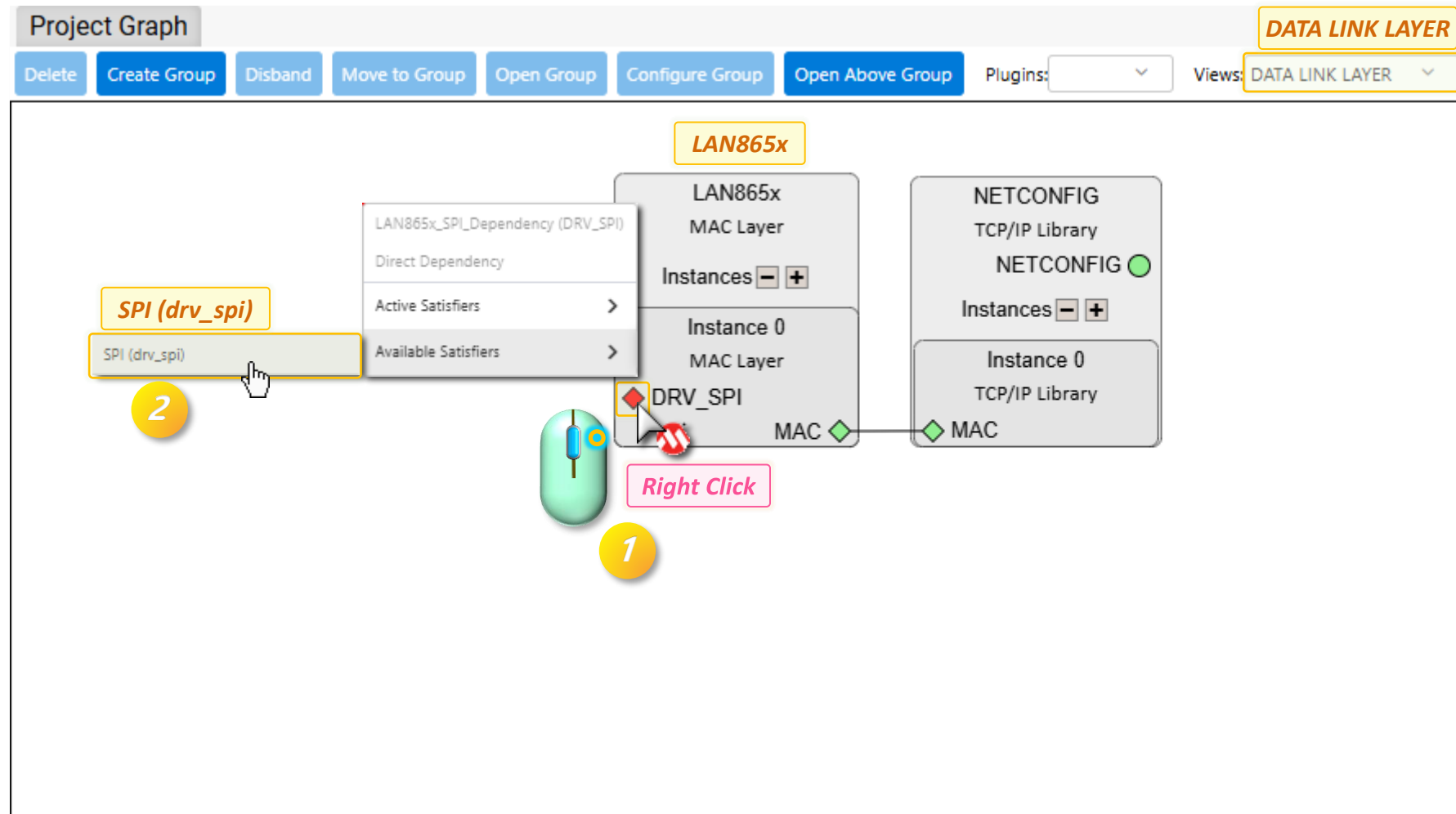
- Return "Project Graph" of MCC Harmony, Select "DATA LINK LAYER" from the "View".



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 2

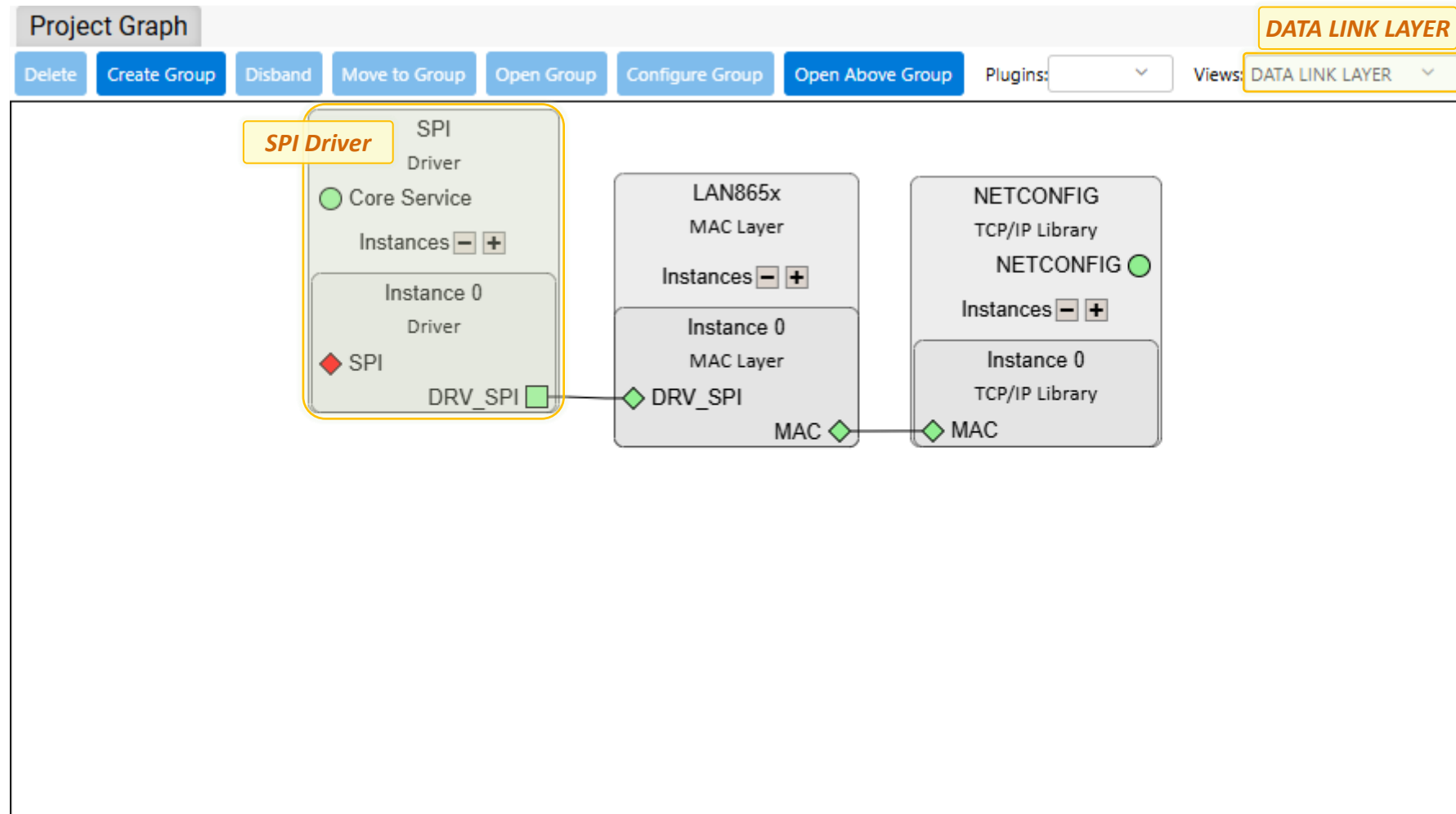
- Right click on  "DRV_SPI" of LAN865x , select the "SPI (drv_spi)" as Satisfiers of it.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 3

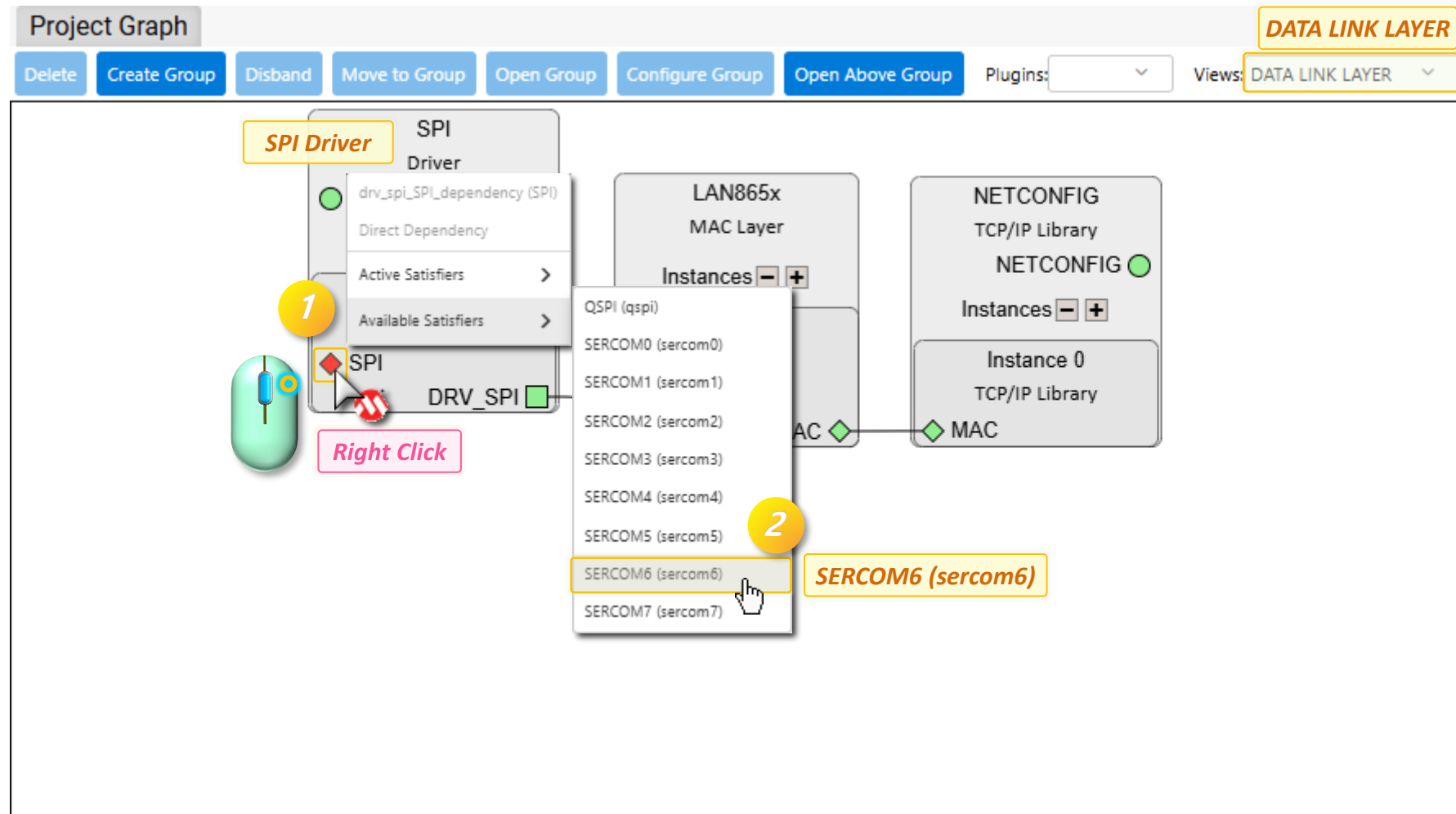
- Successfully added "SPI Driver" to satisfy the instance 0 of "LAN865x" dependency.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 4

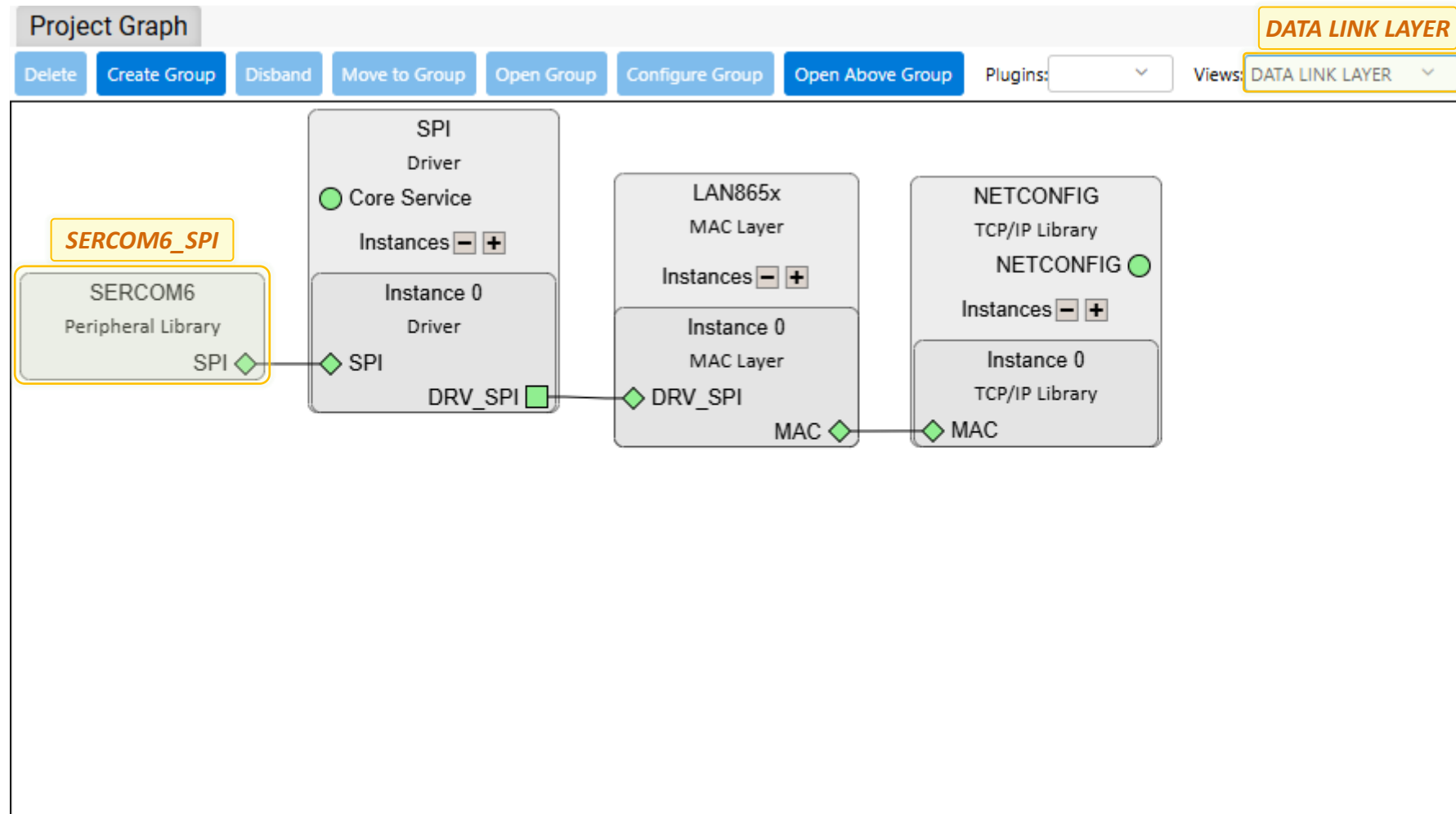
- Right click on  "SPI" of SPI Driver, select the "SERCOM6" as Satisfiers of it.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 5

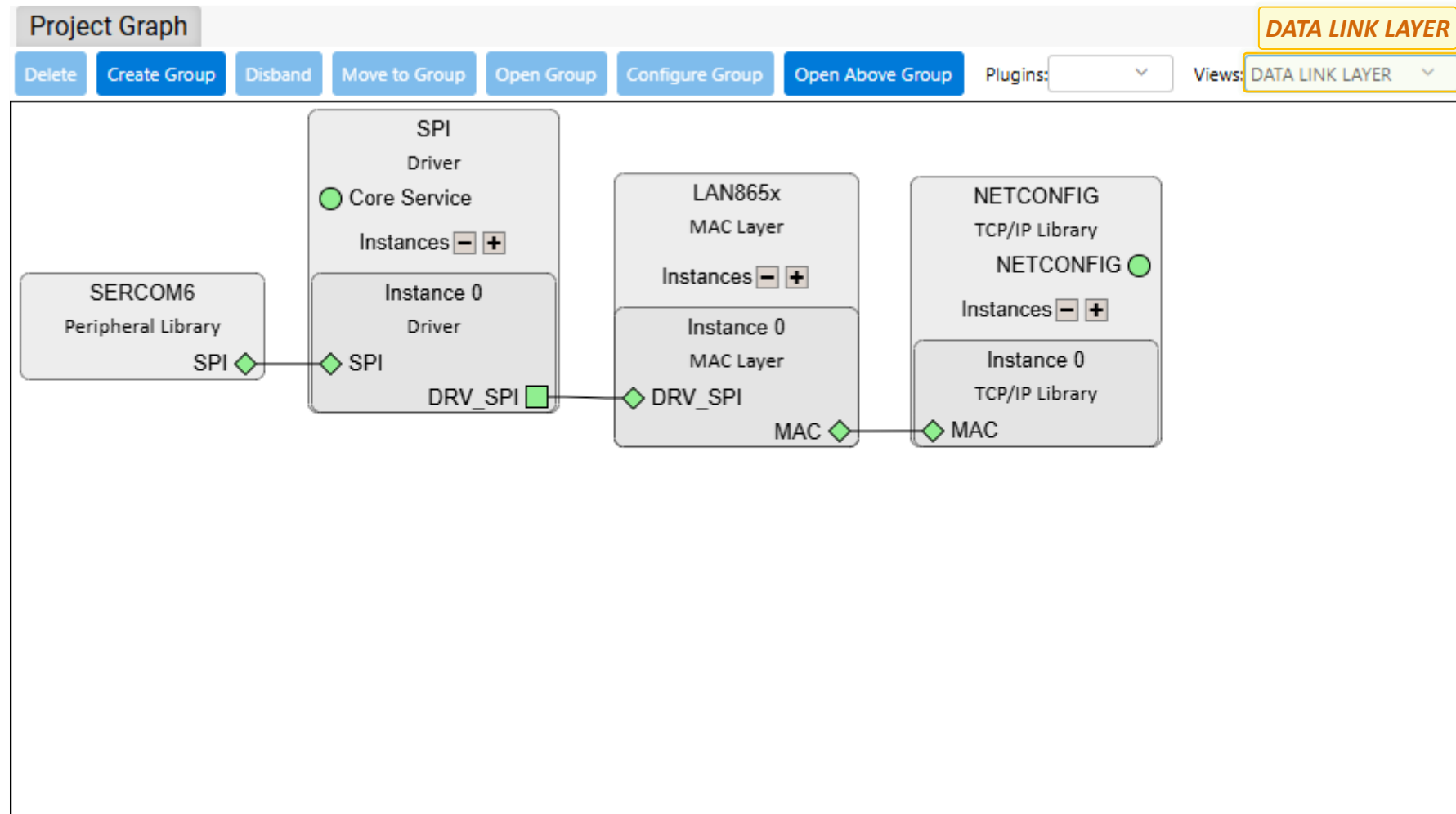
- Successfully added "**SERCOM6_SPI**" to satisfy the "**SPI Driver**" dependency.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 6

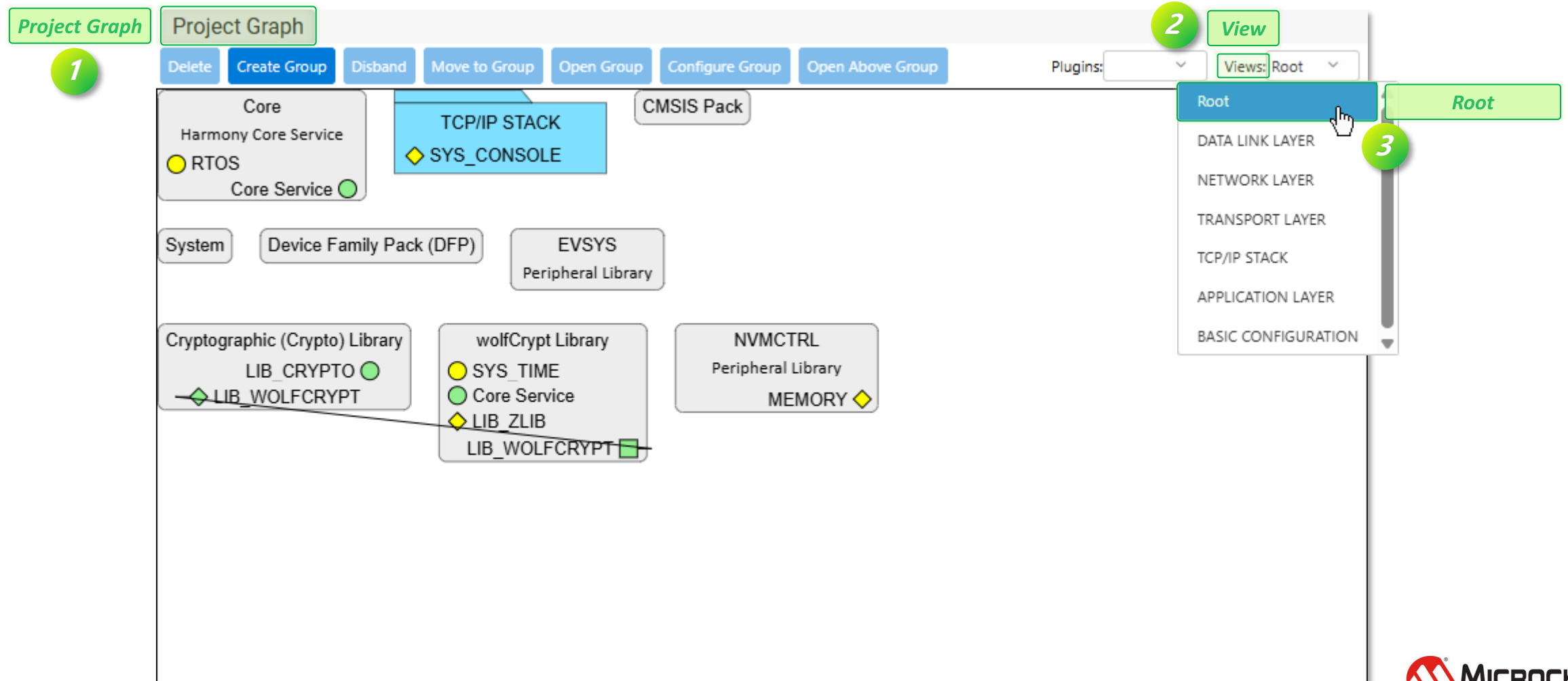
- Finally, the "DATA LINK LAYER" should look like this.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 7

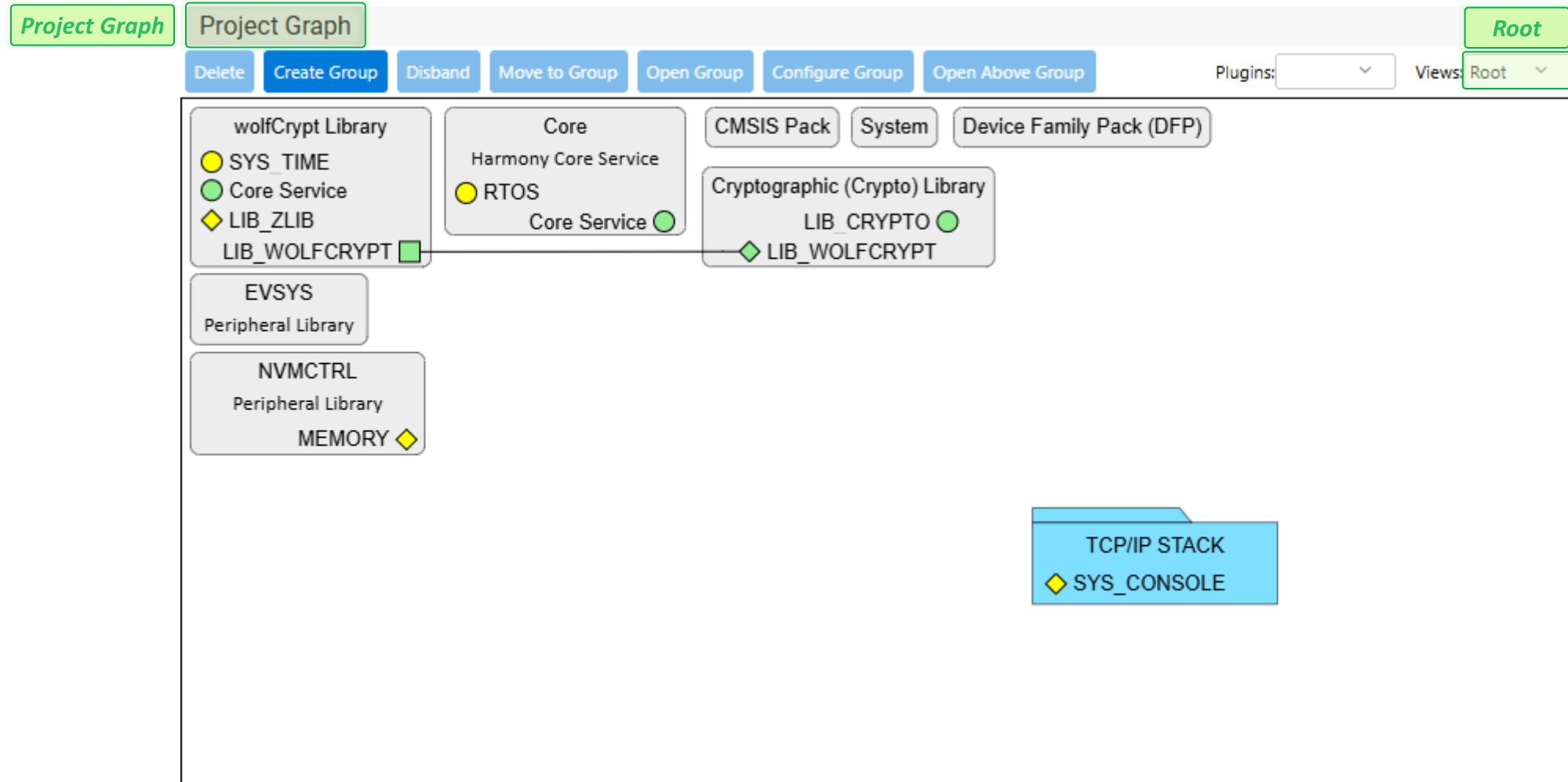
- Select "Root" from the "View".



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 8

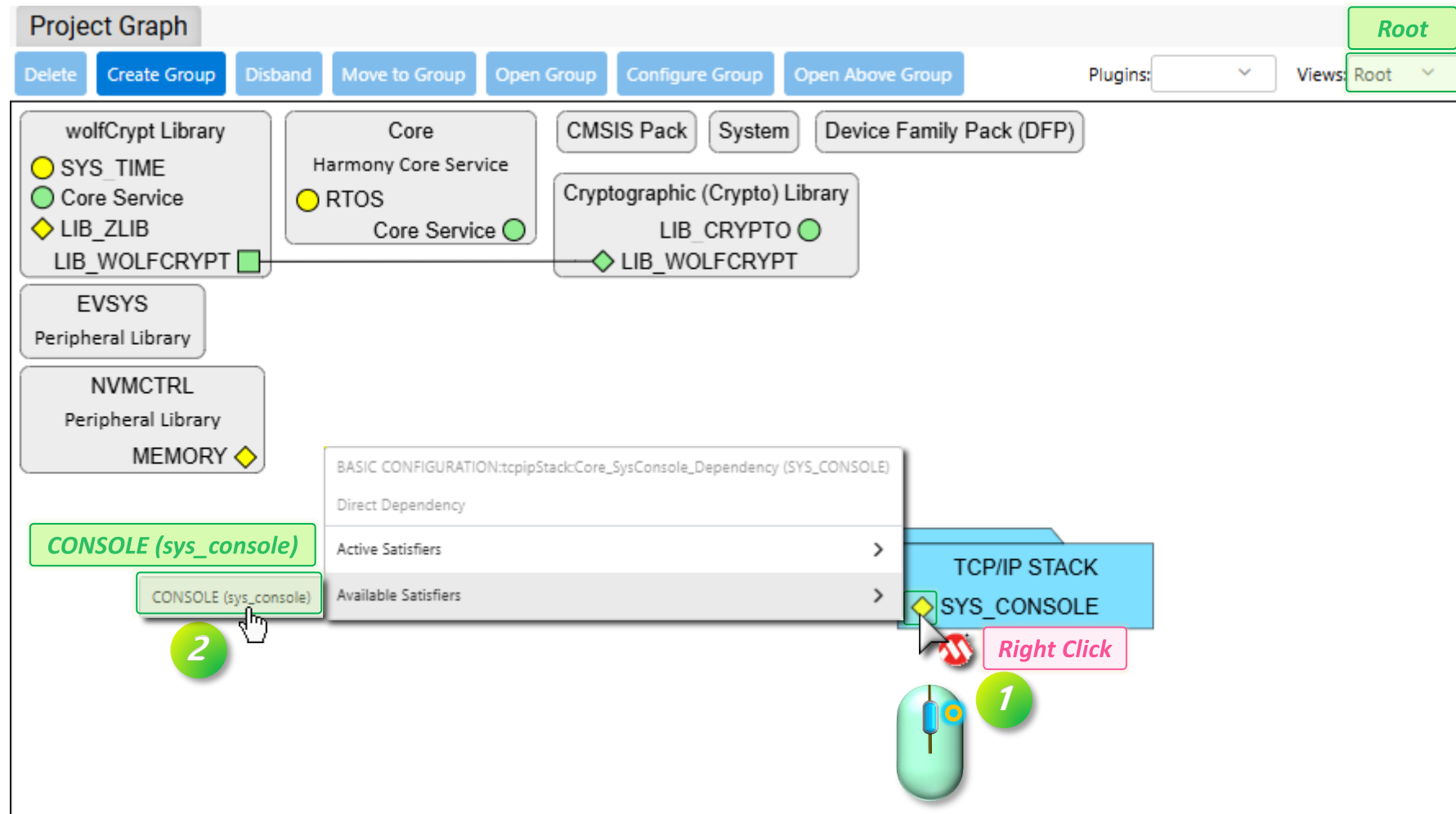
- Trying to rearrange it a bit so that the hierarchy of modules is easier to understand.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 9

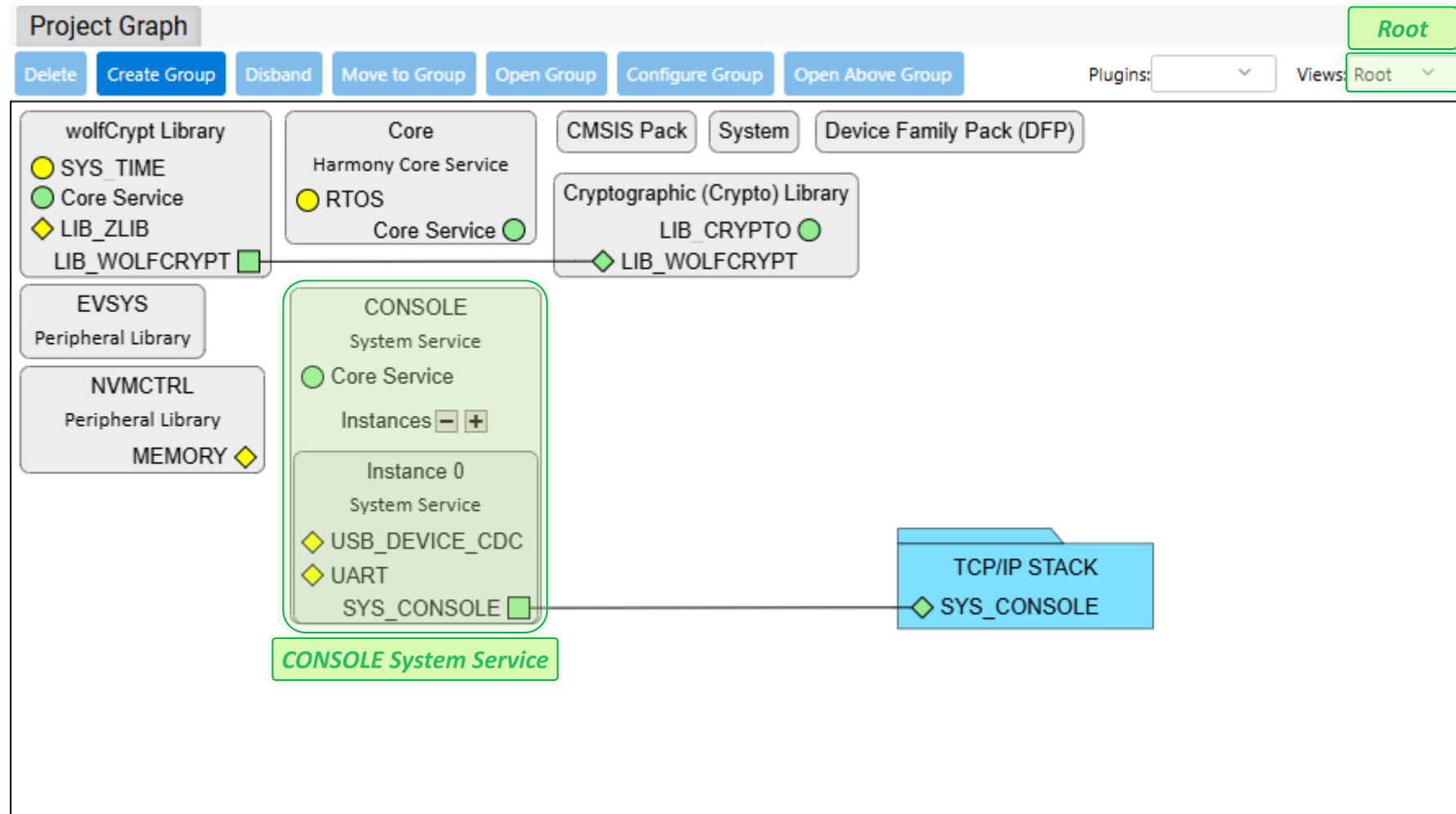
- Right click on  "SYS_CONSOLE" of TCP/IP STACK, select "CONSOLE (sys_console)" as its Satisfiers.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 10

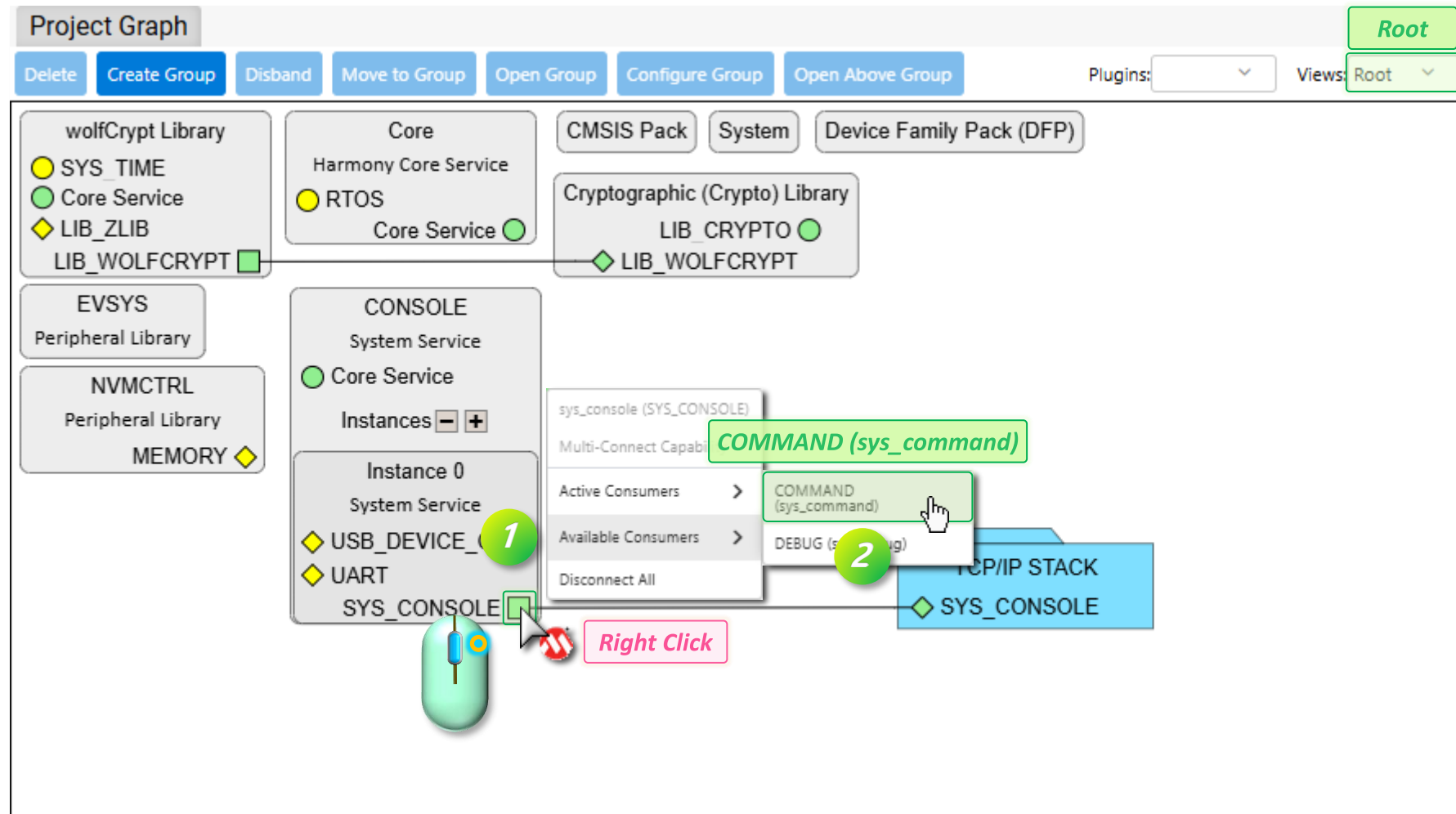
- Successfully added "**CONSOLE System Service**" to satisfy the "**TCP/IP STACK**" console requirement.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 11

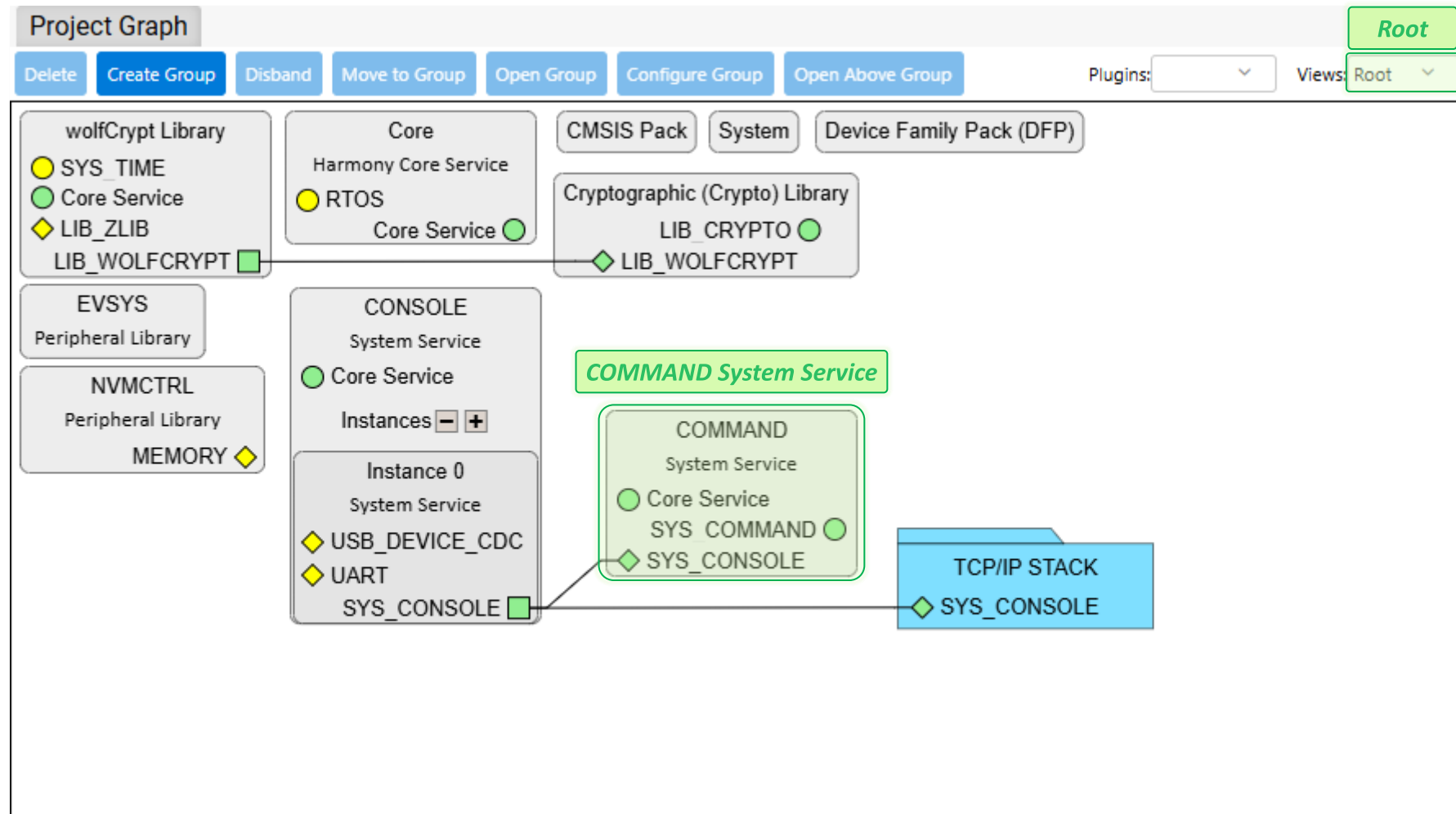
- Right click on **"SYS_CONSOLE"** of **CONSOLE System Service**, let **"COMMAND"** as its Consumers.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 12

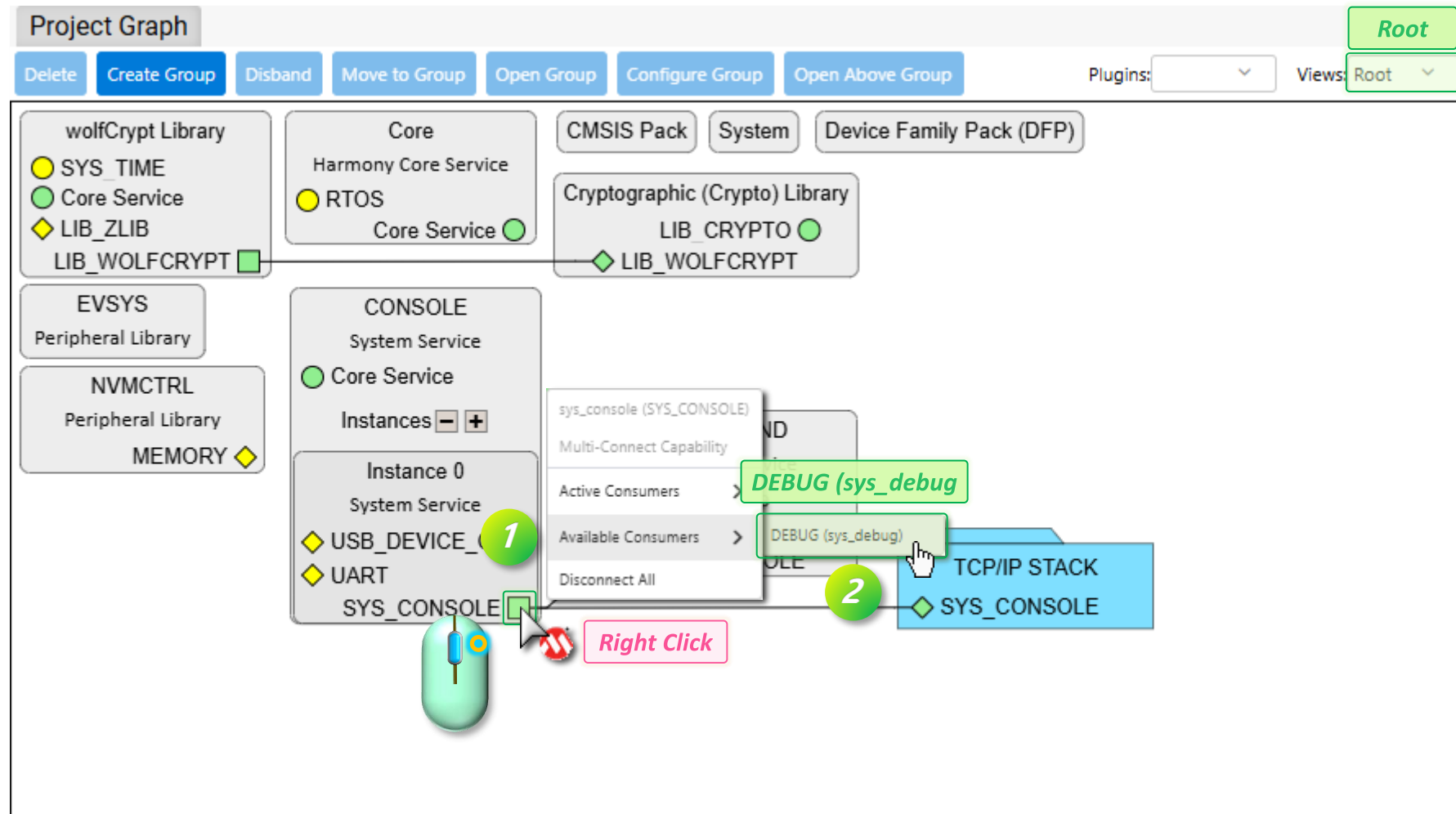
- Successfully added "**COMMAND System Service**" as Consumer of "**CONSOLE System Service**".



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 13

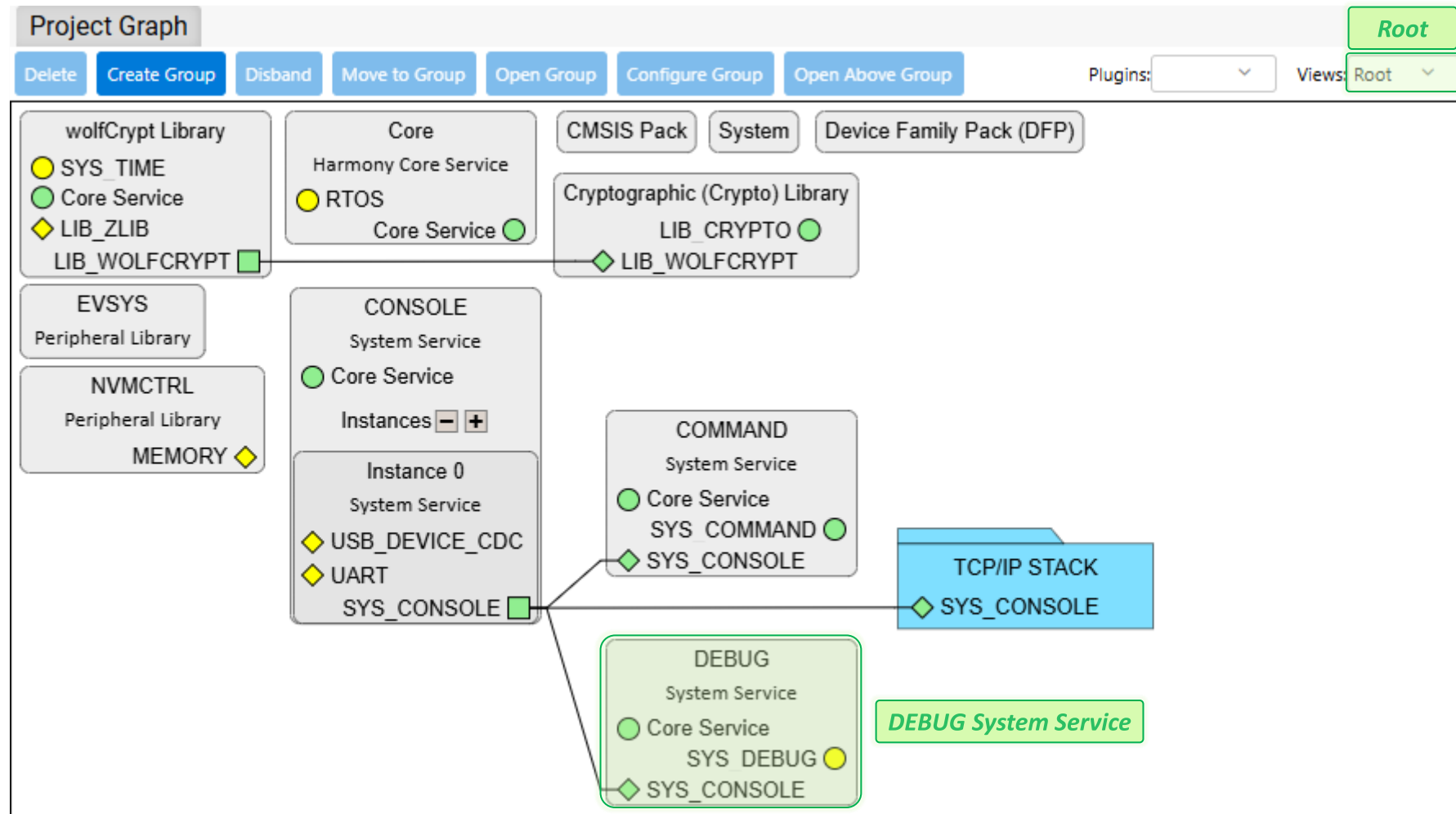
- Right click on **"SYS_CONSOLE"** of **CONSOLE System Service** and add **"DEBUG"** as Consumers.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 14

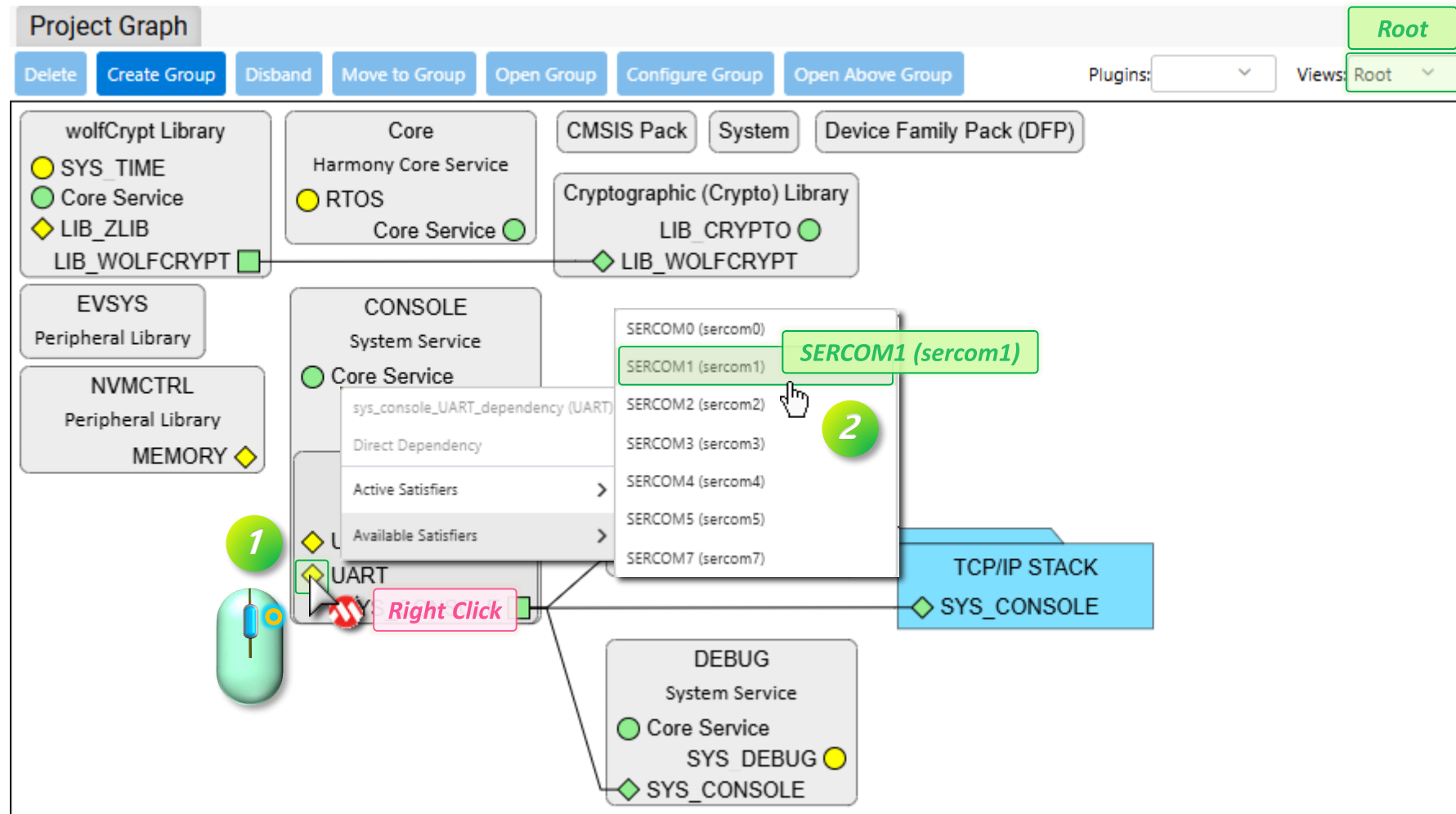
- Successfully added "**DEBUG System Service**" as Consumer of "**CONSOLE System Service**".



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 15

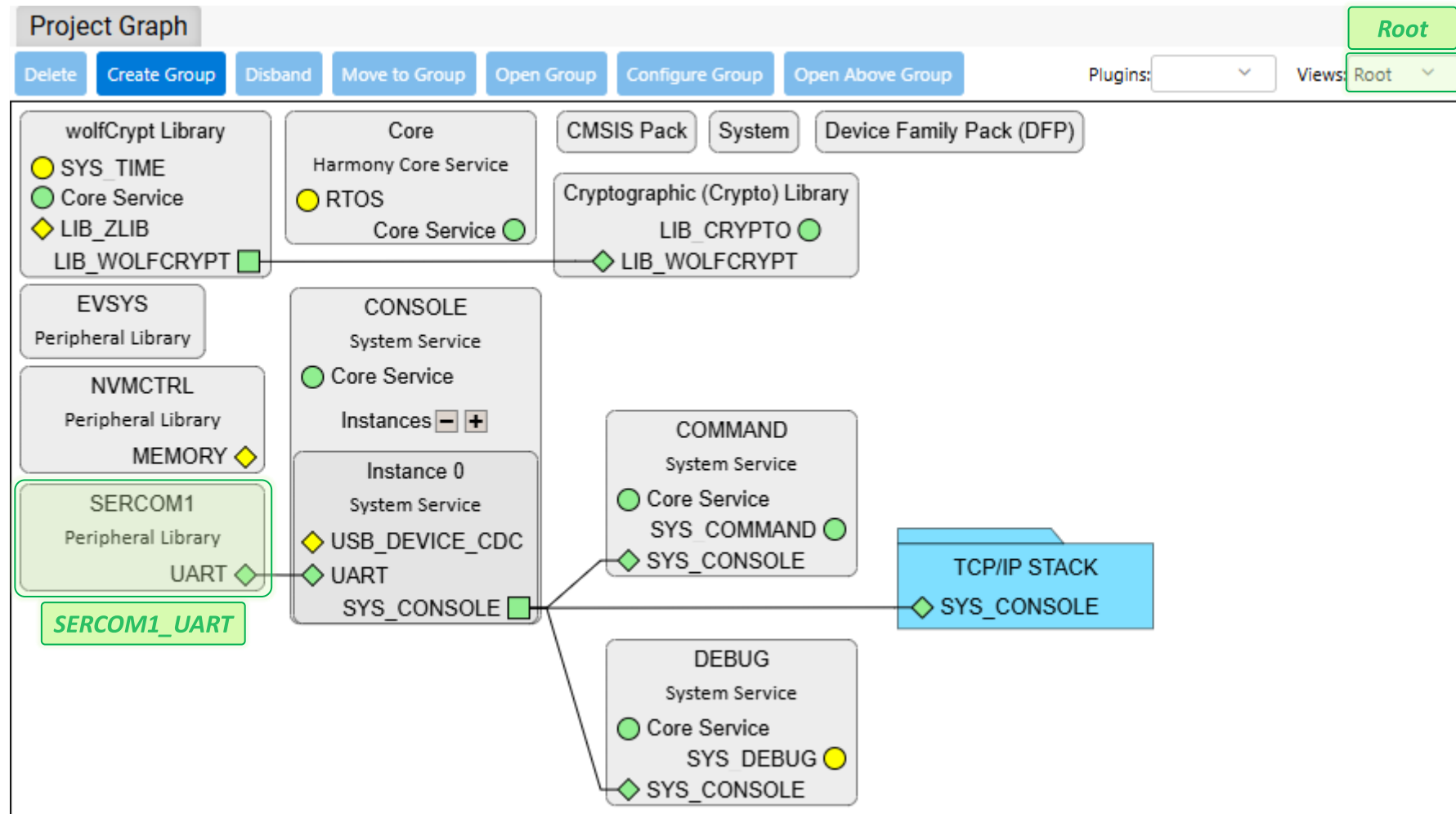
- Right click on  "UART" of **CONSOLE System Service**, select "**SERCOM1**" as Satisfiers of it.



Added TCP/IP Dependency, System Driver & Peripheral Library

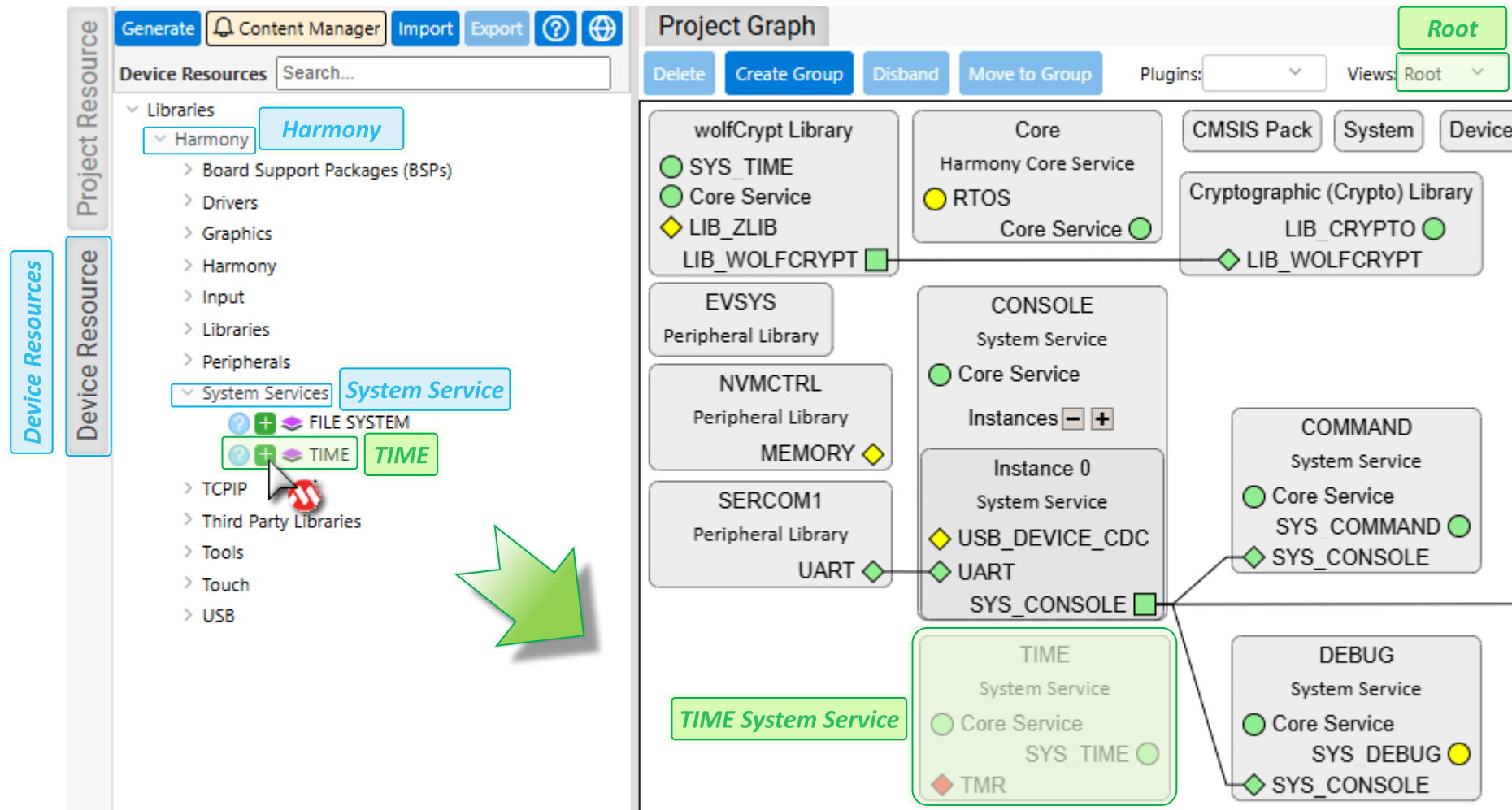
Step 16

- Successfully added "**SERCOM1_UART**" to satisfy "**CONSOLE System Service**" dependency.



Step 17

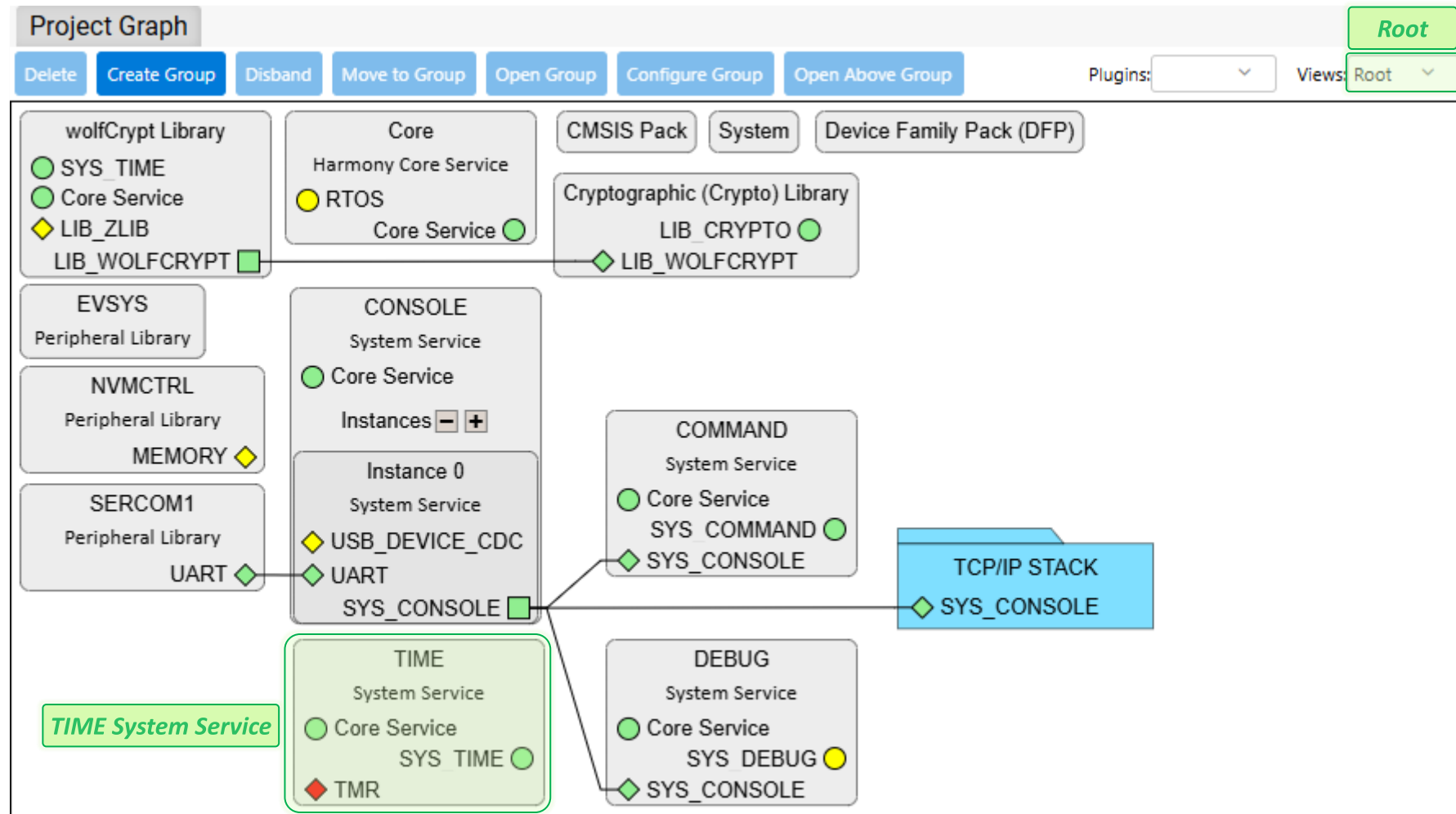
- Add "TIME System Service" from Device Resources of MCC By click  TIME.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 18

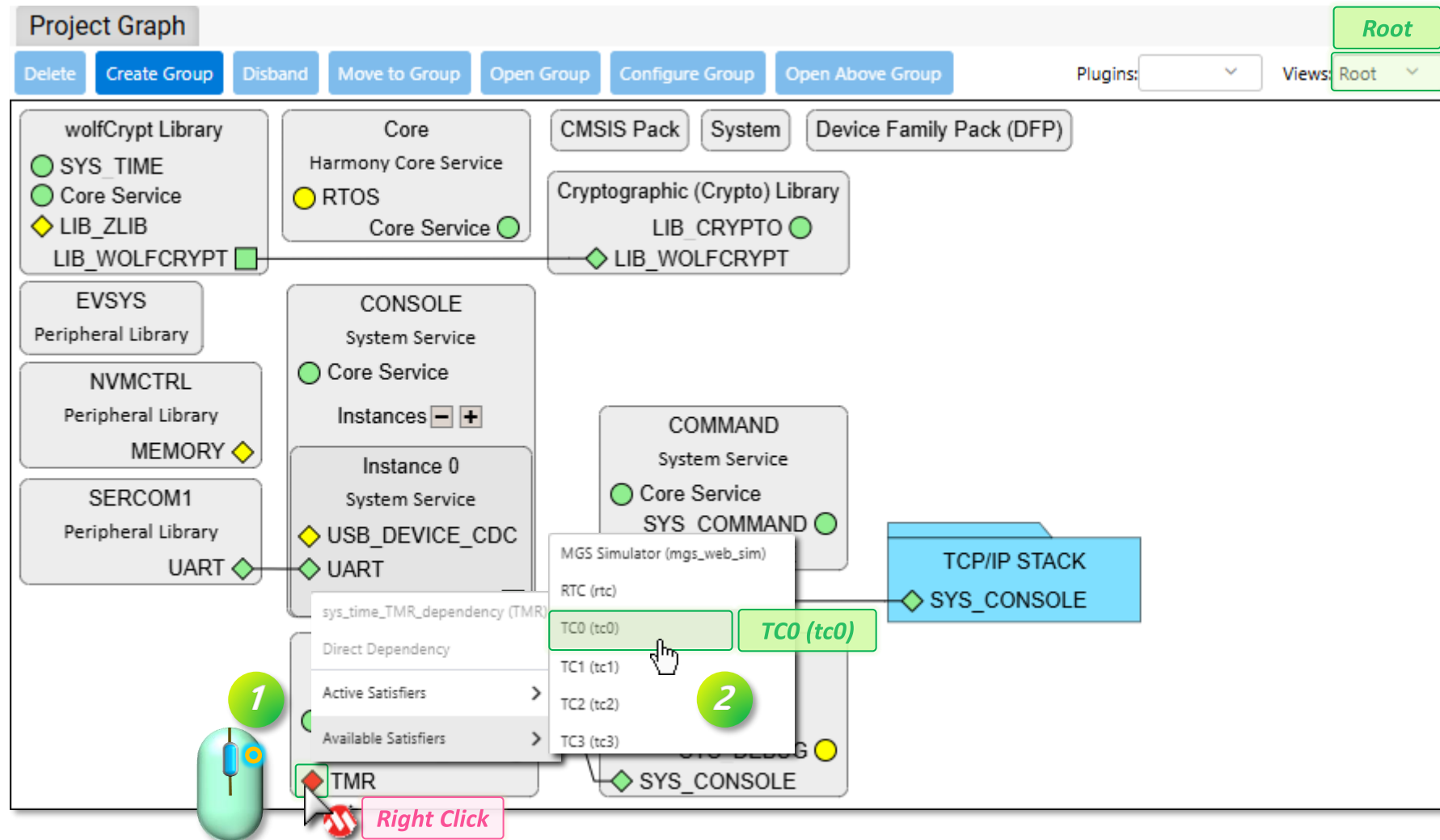
- Successfully added "**TIME System Service**" from **Device Resources** of MCC.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 19

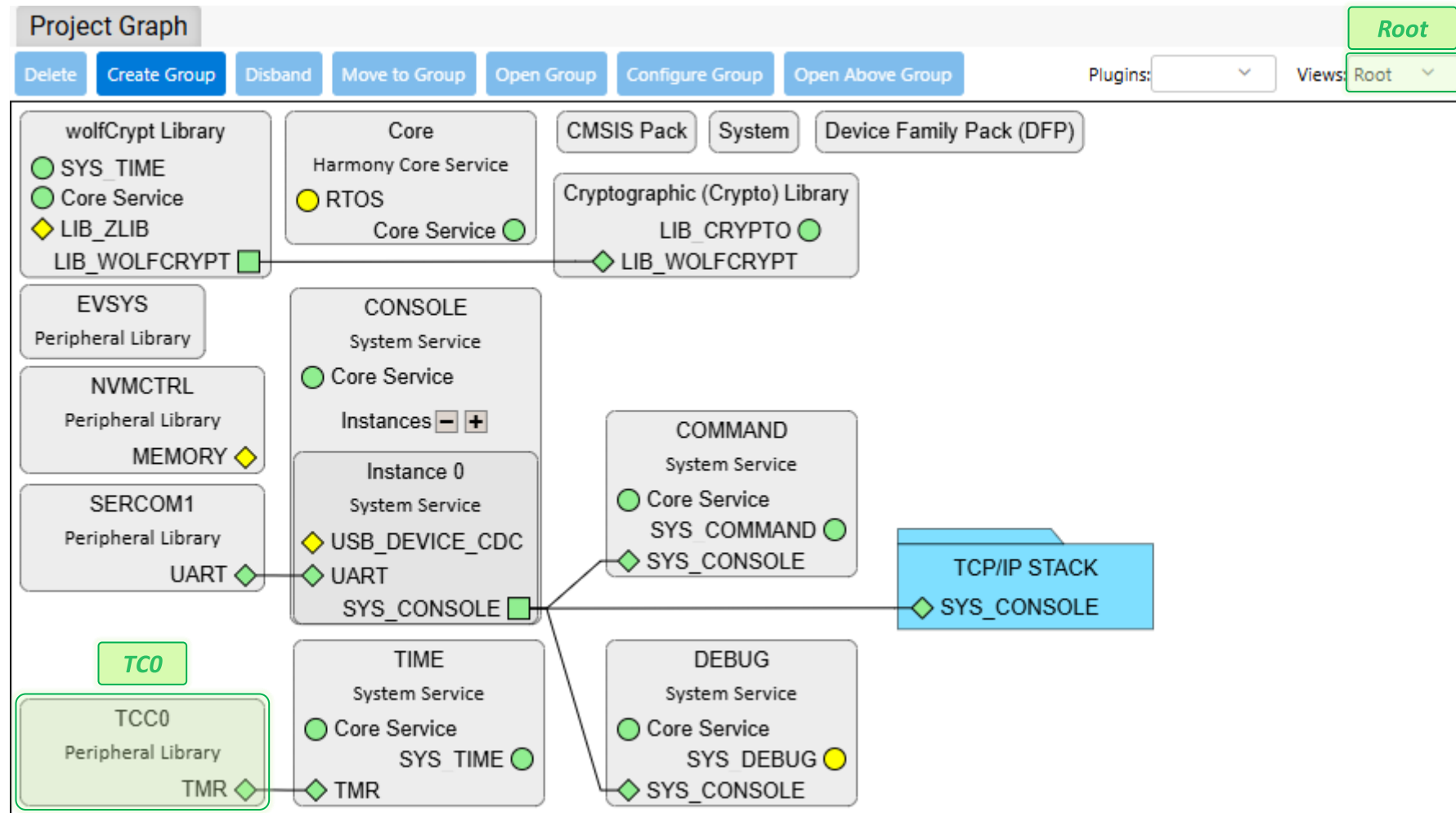
- Right click on  "TMR" of TIME System Service , select "TC0" as Satisfiers of it.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 20

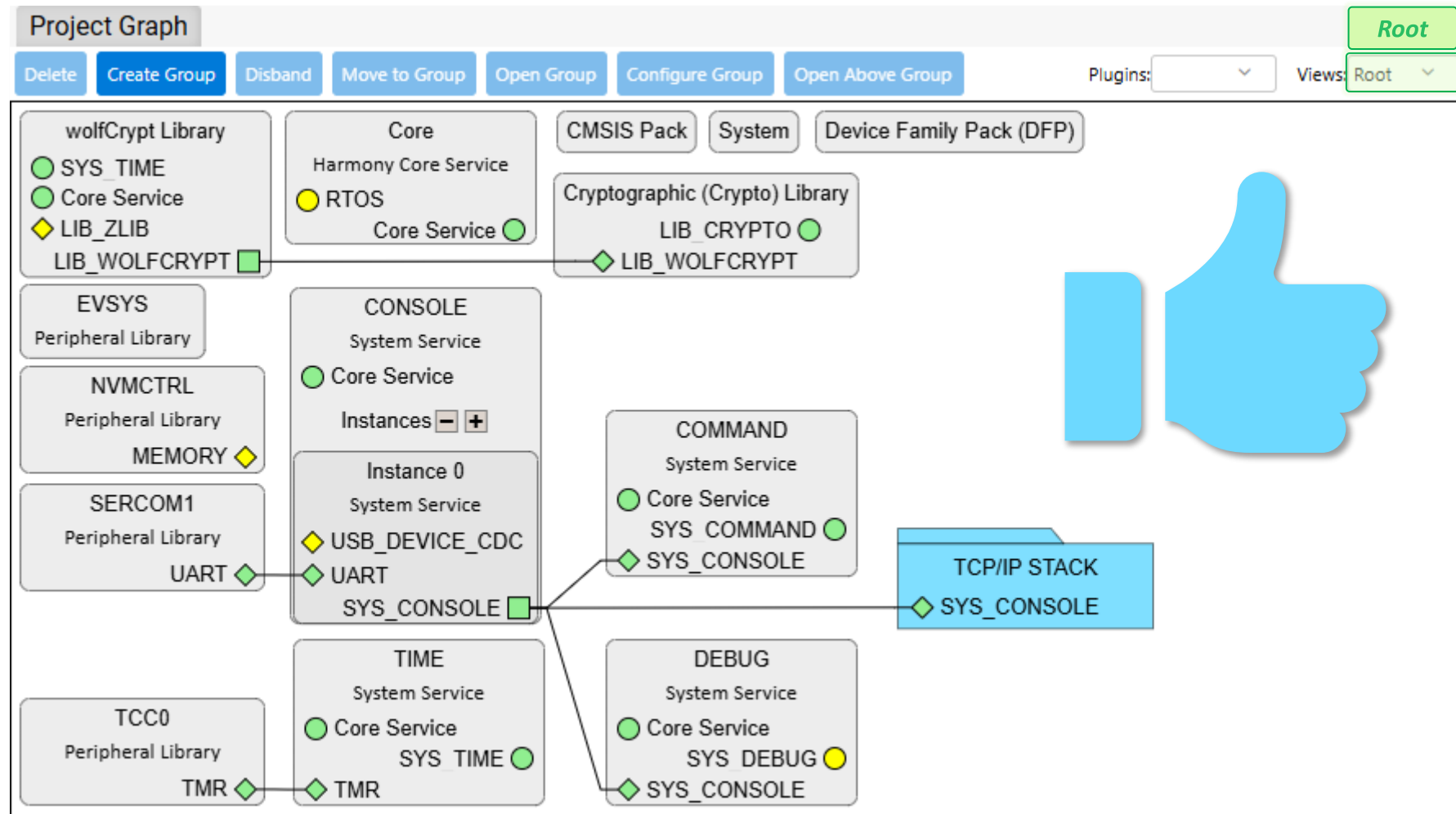
- Successfully added "TC0" to satisfy "TIME System Service" dependency.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 21

- Finally, the "Root" layer should look like shown below.



Added TCP/IP Dependency, System Driver & Peripheral Library

Step 22

- Open the "TCP/IP Configurator" and click the **Config Summary** to view dependency status.

The image shows two screenshots of the TCP/IP Configurator interface, illustrating the process of resolving dependencies.

Left Screenshot (Initial State):

- Configuration Summary:**
 - TCP/IP Stack Configuration Status:**
 - Application: DNS CLIENT
 - Presentation: None
 - Transport: TCP, UDP
 - Network: ARP, ICMPv4, IPv4
 - Data Link & Physical: LAN865x-0, NETCONFIG-0
 - Basic Configuration: TCPIP CMD, TCPIP CORE
 - Unsatisfied Dependency Status:**
 - Required Dependency Status:**
 - LAN865x-0 : DRV_SPI (Driver)
 - TCPIP CMD : SYS_COMMAND (System Service)
 - TCPIP CORE : SYS_TIME (System Service)
 - Optional Dependency Status:**
 - TCPIP CORE : SYS_CONSOLE (System Service)

Some Required Dependency Not Ready yet !

Right Screenshot (Final State):

- Configuration Summary:**
 - TCP/IP Stack Configuration Status:**
 - Application: DNS CLIENT
 - Presentation: None
 - Transport: TCP, UDP
 - Network: ARP, ICMPv4, IPv4
 - Data Link & Physical: LAN865x-0, NETCONFIG-0
 - Basic Configuration: TCPIP CMD, TCPIP CORE
 - Unsatisfied Dependency Status:**
 - Required Dependency Status:**
 - Optional Dependency Status:**

All required Dependencies are ready now.

A green arrow points from the left screenshot to the right screenshot, indicating the transition. A green box labeled "Config Summary" points to the "Config Summary" button in the bottom right corner of the right screenshot.

MPLAB® Code Configurator (MCC)

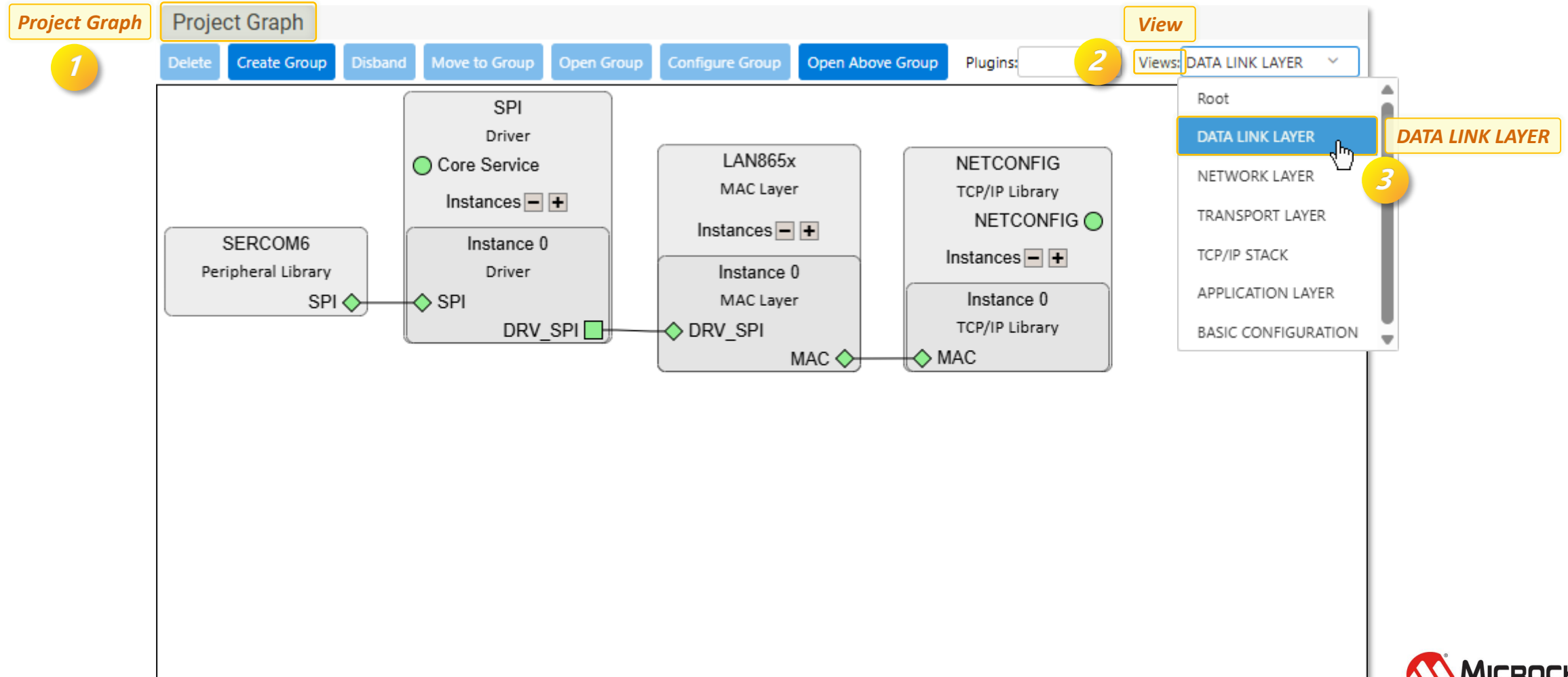
Configure the Added Modules

MCC Harmony Configuration

Configure the Added Modules

Step 1

- Return "Project Graph" of MCC Harmony, Select "DATA LINK LAYER" from the "View".



Configure the Added Modules

Step 2

- Select **Instance 0** of **LAN865x** and Click **Expand All** to expand options of configuration.

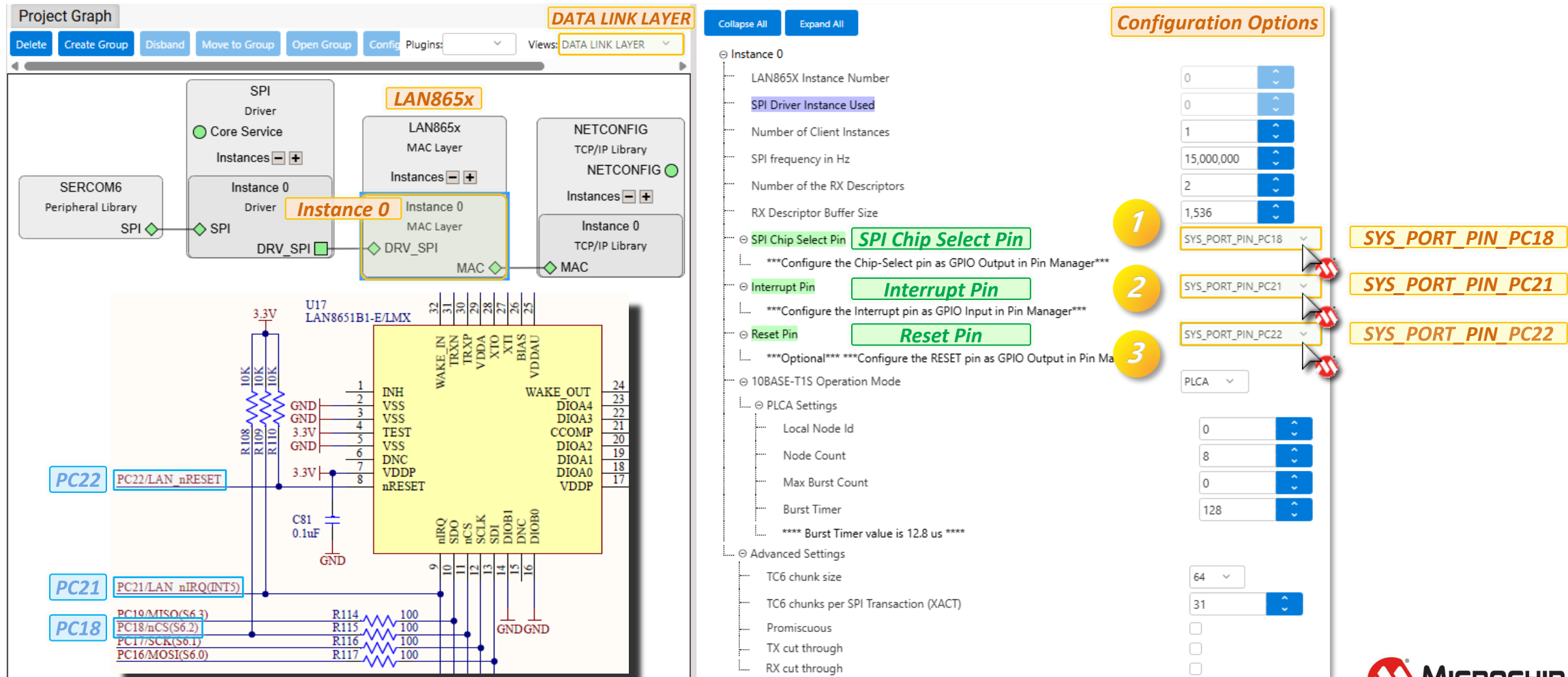
The screenshot displays the Microchip Studio configuration interface. On the left, the **Project Graph** shows the hardware and software components. The **LAN865x** module is highlighted with a yellow box and a red cursor. A yellow circle with the number **1** points to the **Instance 0** of the **LAN865x** module. On the right, the **Configuration Options** panel is shown. A yellow circle with the number **2** points to the **Expand All** button. The **Configuration Options** panel lists various settings for the LAN865x module, including:

- LAN865X Instance Number: 0
- SPI Driver Instance Used: 0
- Number of Client Instances: 1
- SPI frequency in Hz: 15,000,000
- Number of the RX Descriptors: 2
- RX Descriptor Buffer Size: 1,536
- SPI Chip Select Pin: SYS_PORT_PIN_NONE
- Interrupt Pin: SYS_PORT_PIN_NONE
- Reset Pin: SYS_PORT_PIN_NONE
- 10BASE-T1S Operation Mode: PLCA
- PLCA Settings:
 - Local Node Id: 0
 - Node Count: 8
 - Max Burst Count: 0
 - Burst Timer: 128
- Advanced Settings:
 - TC6 chunk size: 64
 - TC6 chunks per SPI Transaction (XACT): 31
 - Promiscuous: ☐
 - TX cut through: ☐
 - RX cut through: ☐

Configure the Added Modules

Step 3

- Configure GPIO pins (**PC18**, **PC21**, **PC22**) for **LAN865x** chip (**SPI-CS**, **INT**, **Reset**) as below.



Configure the Added Modules

Step 5

- Configure **PLCA Settings**, ensure the **Local Node Id** is **unique** within T1S network chain.

The image shows the Project Graph and Configuration Options for the LAN865x module. The Project Graph on the left shows the connection between the SERCOM6 Peripheral Library, SPI Driver, LAN865x MAC Layer, and NETCONFIG TCP/IP Library. The Configuration Options on the right show the settings for Instance 0, including LAN865X Instance Number, SPI Driver Instance Used, Number of Client Instances, SPI frequency in Hz, Number of the RX Descriptors, RX Descriptor Buffer Size, SPI Chip Select Pin, Interrupt Pin, Reset Pin, 10BASE-T1S Operation Mode, and PLCA Settings. The PLCA Settings section is highlighted, showing the Local Node Id and Node Count. The Local Node Id is set to 9, and the Node Count is set to 255. A pink arrow points from the text 'Unique Node ID' to the Local Node Id field. A yellow box highlights the Node Count field with the text 'Change Node Count to 255'. A pink box highlights the Local Node Id field with the text 'Ensure the Node Id is unique'.

Project Graph

DATA LINK LAYER

Configuration Options

Instance 0

- LAN865X Instance Number: 0
- SPI Driver Instance Used: 0
- Number of Client Instances: 1
- SPI frequency in Hz: 15,000,000
- Number of the RX Descriptors: 2
- RX Descriptor Buffer Size: 1,536
- SPI Chip Select Pin: SYS_PORT_PIN_PC18
- Interrupt Pin: SYS_PORT_PIN_PC21
- Reset Pin: SYS_PORT_PIN_PC22
- 10BASE-T1S Operation Mode: PLCA
- PLCA Settings
 - Local Node Id: 9
 - Node Count: 255
 - Max Burst Count: 0
 - Burst Timer: 128
- Advanced Settings
 - TC6 chunk size: 64
 - TC6 chunks per SPI Transaction (XACT): 31
 - Promiscuous: ☐
 - TX cut through: ☐
 - RX cut through: ☐

Unique Node ID

Ensure the Node Id is unique

Change Node Count to 255

Configure the Added Modules

Step 6

- Configure **Advanced Settings**, enable the ☒ **TX cut through**.

Project Graph

Buttons: Delete, Create Group, Disband, Move to Group, Open Group, Config, Plugins: [v], Views: DATA LINK LAYER [v]

LAN865x

Instance 0

MAC

Configuration Options

Instance 0

- LAN865X Instance Number: 0
- SPI Driver Instance Used: 0
- Number of Client Instances: 1
- SPI frequency in Hz: 15,000,000
- Number of the RX Descriptors: 2
- RX Descriptor Buffer Size: 1,536
- SPI Chip Select Pin: SYS_PORT_PIN_PC18
 - ***Configure the Chip-Select pin as GPIO Output in Pin Manager***
- Interrupt Pin: SYS_PORT_PIN_PC21
 - ***Configure the Interrupt pin as GPIO Input in Pin Manager***
- Reset Pin: SYS_PORT_PIN_PC22
 - ***Optional*** ***Configure the RESET pin as GPIO Output in Pin Manager***
- 10BASE-T1S Operation Mode: PLCA
 - PLCA Settings
 - Local Node Id: 9
 - Node Count: 255
 - Max Burst Count: 0
 - Burst Timer: 128
 - **** Burst Timer value is 12.8 us ****
- Advanced Settings
 - TC6 chunk size: 64
 - TC6 chunks per SPI Transaction (XACT): 31
 - Promiscuous: ☐
 - TX cut through: ☒ **Tx cut through**
 - RX cut through: ☐

1 **Check to Enable**

Configure the Added Modules

Step 7

- Select **Instance 0** of **NETCONFIG** and Click **Expand All** to expand options of configuration.

The screenshot displays the Microchip configuration tool interface. On the left, the **Project Graph** shows a hierarchical structure of modules. The **NETCONFIG** module is highlighted, and its **Instance 0** is selected. A yellow circle with the number **1** points to the **Instance 0** label. On the right, the **Configuration Options** panel is shown. A yellow circle with the number **2** points to the **Expand All** button. The configuration options are expanded, showing various settings for the network configuration.

Project Graph

- SERCOM6 Peripheral Library
- SPI
- SPI Driver
- Core Service
- Instances
- Instance 0 Driver
- DRV_SPI
- LAN865x MAC Layer
- Instances
- Instance 0 MAC Layer
- DRV_SPI
- MAC
- NETCONFIG TCP/IP Library
- NETCONFIG
- Instances
- Instance 0 TCP/IP Library
- MAC

Configuration Options

- Instance 0
- Network Configurations Index
- Interface
- Host Name
- Mac Address
- IPv4 Static Address
- IPv4 SubNet Mask
- IPv4 Default Gateway Address
- IPv4 Primary DNS
- Network Configuration Start-up Flags
- DHCP Flag
- DNS Flag
- Multicast Enabled on this Interface
- Advanced Settings
- IPv4 Secondary DNS
- Network MAC Driver
- Power Mode
- VLAN ID
- VLAN PCP

Configure the Added Modules

Step 8

- Ensure **IPv4 static** and **MAC** are **unique** within the T1S network and disable **DHCP Flag**.

Project Graph

Buttons: Delete, Create Group, Disband, Move to Group, Open Group, Config, Plugins: [v], Views: DATA LINK LAYER

NETCONFIG

Instance 0 TCP/IP Library

Instance 0

Configuration Options

1. Mac Address: 10:04:25:1C:A0:09 (Ensure the Mac Address is unique)

2. IPv4 Static Address: 192.168.100.9 (Ensure the IPv4 Address is unique)

3. DHCP Flag: None (None)

Modify the last number of MAC and IP to the number same as your EVB

Unique Node ID

Mac Address: 00:04:25:1C:A0:09

IPv4 Static Address: 192.168.100.9

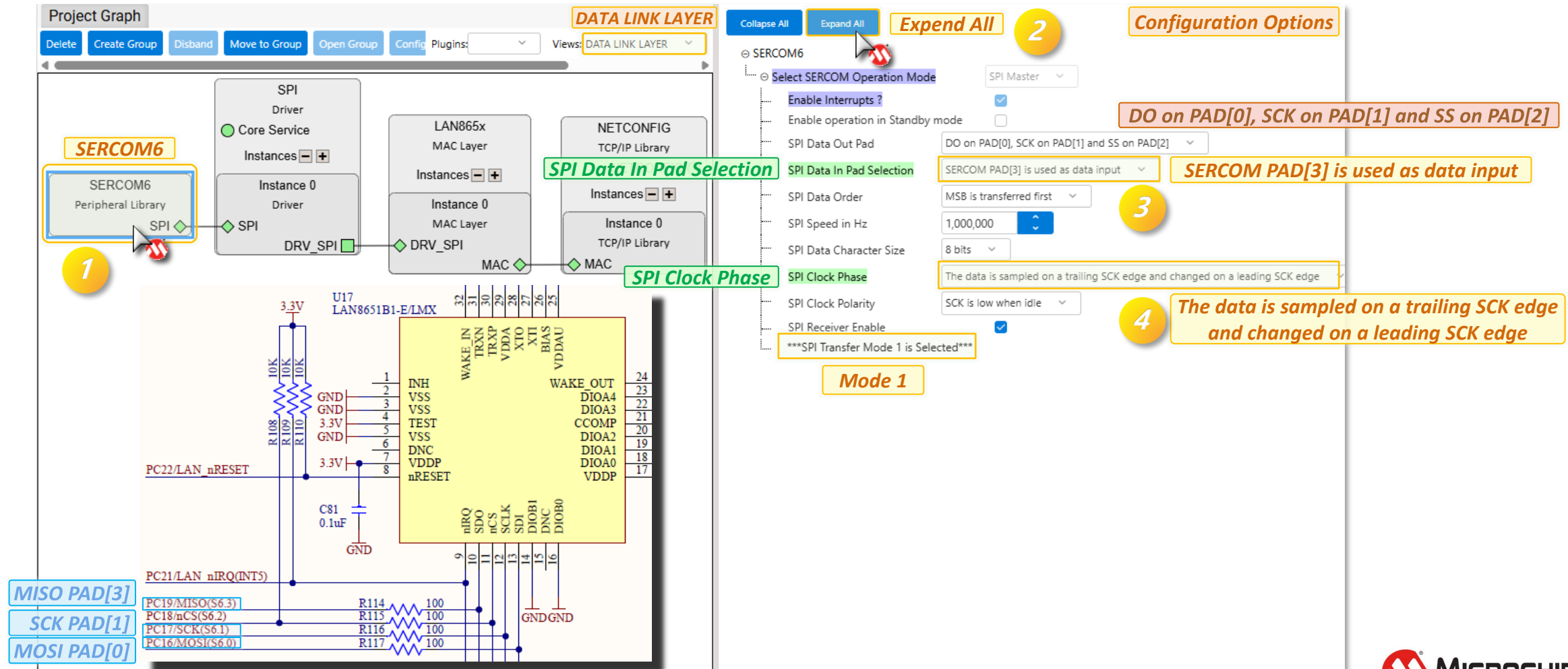
1. Convert to HEX (Ex. 192 = C0)

2.

Configure the Added Modules

Step 9

- Configure the PLIB **SERCOM6_SPI** for **LAN865x** chip as below.



Configure the Added Modules

Step 10

- Configure **Instance 0** of **SPI Driver**, modify **Transfer Queue Size** to **1** and ☒ **Use DMA**.

The screenshot shows the Project Graph and Configuration Options for the SPI Driver Instance 0. The Project Graph on the left shows the connection between the SERCOM6 Peripheral Library, the SPI Driver Instance 0, the LAN865x MAC Layer Instance 0, and the NETCONFIG TCP/IP Library Instance 0. The Configuration Options on the right show the settings for Instance 0, including the Transfer Queue Size and the Use DMA for Transmit and Receive checkbox.

Project Graph:

- SERCOM6 Peripheral Library (SPI) is connected to the SPI Driver Instance 0.
- The SPI Driver Instance 0 is connected to the LAN865x MAC Layer Instance 0.
- The LAN865x MAC Layer Instance 0 is connected to the NETCONFIG TCP/IP Library Instance 0.

Configuration Options:

- Instance 0
 - PLIB Used
 - Number of Clients
 - Transfer Queue Size: 1 (labeled 2)
 - Use DMA for Transmit and Receive? (checked, labeled 3)
 - DMA Channel For Transmit
 - DMA Channel For Receive

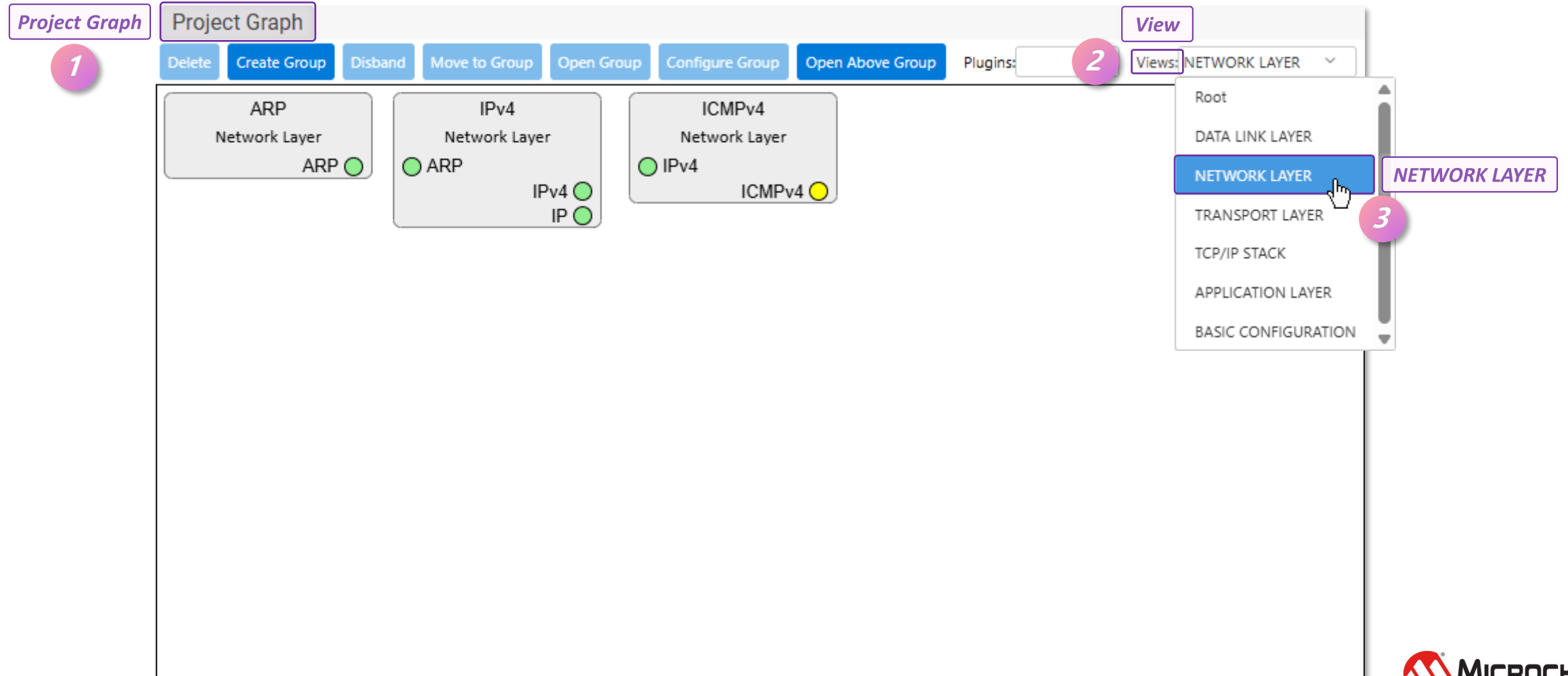
Annotations:

- 1: SPI Driver
- 2: Transfer Queue Size
- 3: Use DMA for Transmit and Receive
- Transfer Queue Size : 1
- Use DMA for Transmit and Receive

Configure the Added Modules

Step 11

- Return "Project Graph" of MCC Harmony, Select "NETWORK LAYER" from the "View".



Configure the Added Modules

Step 12

- Configure **ICMPv4**, check ☒ **Use ICMPv4 Client** and ☒ **Enable ICMP Client Console Commands**.

The screenshot displays the Microchip Studio configuration interface for the Network Layer. The left pane shows the Project Graph with three modules: ARP, IPv4, and ICMPv4. The ICMPv4 module is highlighted with a red box and a red circle with the number 1. The right pane shows the configuration options for the ICMPv4 module, with a red circle with the number 2 pointing to the 'Expand All' button. The configuration options are listed in a tree view, with 'Use ICMPv4 Client' and 'Enable ICMP Client Console Commands' highlighted in green. A red circle with the number 3 points to the 'Use ICMPv4 Client' checkbox, and a red circle with the number 4 points to the 'Enable ICMP Client Console Commands' checkbox. The 'Configuration Options' pane on the right shows the values for these options: 'Use ICMPv4 Client' is checked, 'Maximum Number of Supported Client Requests' is 4, and 'Echo request timeout (in msec)' is 500.

Project Graph

NETWORK LAYER

ICMPv4

Configuration Options

- ☒ **Use ICMPv4 Client**
- ☒ **Enable ICMP Client Console Commands**

ICMP Client Task Rate (in msec)

Maximum Number of Supported Client Requests

4

Number of ICMP Echo requests

ICMP Request Delay (in msec)

1,000

ICMP Give Up Time-out (in msec)

5,000

Echo Request Data Buffer size - bytes

2,000

Echo request Default Data Size - bytes

100

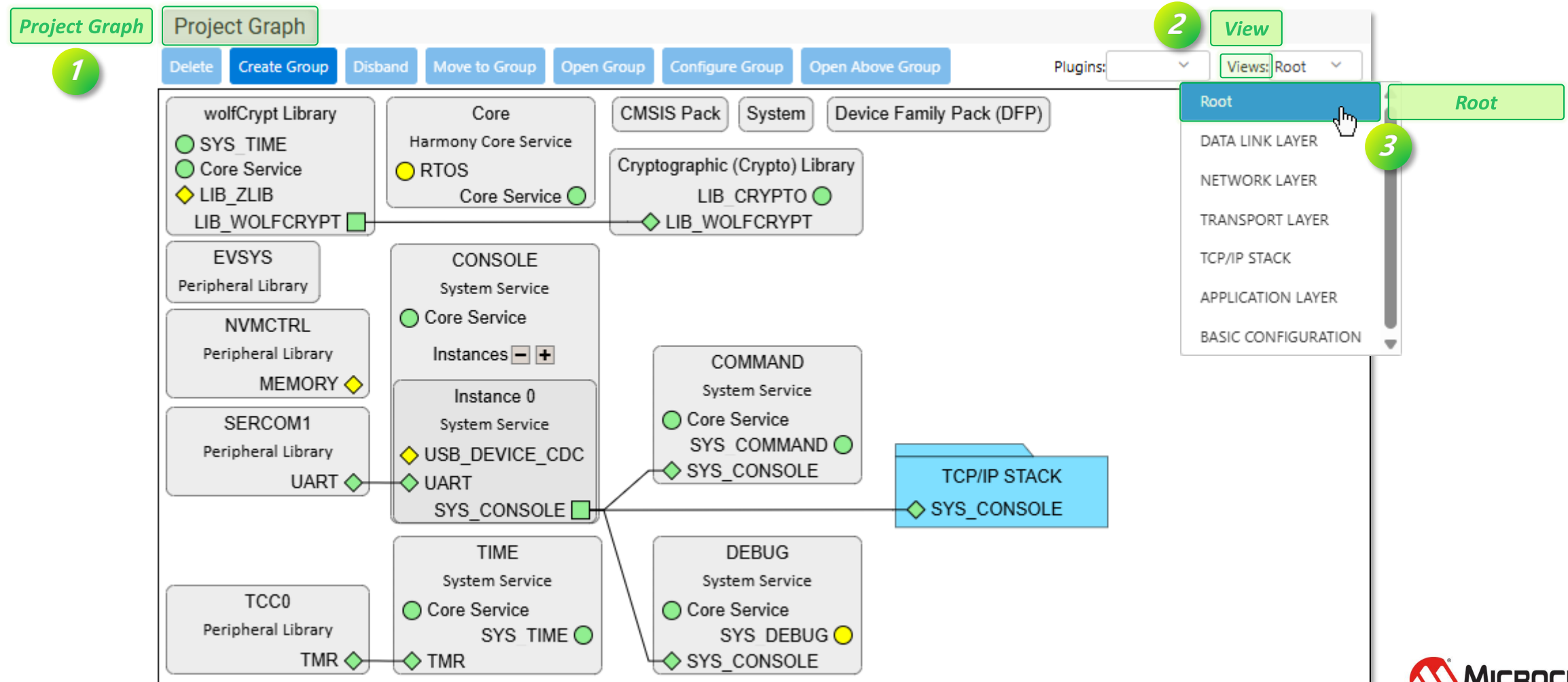
Echo request timeout (in msec)

500

Configure the Added Modules

Step 13

- Select "Root" from the "View".



Configure the Added Modules

Step 14

- Configure the PLIB **SERCOM1_UART** for **USB CDC** Debug output as below.

Project Graph

1 **SERCOM1**

2 **Configuration Options**

3 **Select SERCOM Operation Mode**

4 **Configure Ring Buffer Size**

5 **Receive Pinout**

Configuration Options

Select SERCOM Operation Mode: USART with internal Clock

Operating Mode: Ring buffer mode

Configure Ring Buffer Size:

- TX Ring Buffer Size: 1,024 (1024 (TX Ring Buffer Size))
- RX Ring Buffer Size: 1,024 (1024 (RX Ring Buffer Size))

Receive Enable: ☒

Transmit Enable: ☒

Frame Format: USART frame

Baud Rate in Hz: 115,200 (Baud Rate is 115200 Hz)

Use fractional baud?: ☐

Parity Mode: No Parity (No Parity Check)

Character Size: 8 Bits (8 Bits data)

Stop Bit Mode: One Stop Bit (One Stop Bit)

Start-of-Frame Detection Enab: ☐

Receive Pinout: SERCOM PAD[1] is used for data reception (SERCOM PAD[1] is used for data reception)

Transmit Pinout: PAD[0] = TxD; PAD[1] = XCK (PAD[0] = TxD; PAD[1] = XCK)

Enable Run in Standby: ☐

Project Graph

Root

Plugins: Views: Root

wolfCrypt Library

- SYS_TIME
- Core Service
- LIB_ZLIB
- LIB_WOLFCRYPT

Core

- Harmony Core Service
- RTOS
- Core Service

CMSIS Pack

- System
- Device Family Pack (DFP)

Cryptographic (Crypto) Library

- LIB_CRYPTO
- LIB_WOLFCRYPT

EVSYN

- Peripheral Library

NVMCTRL

- Peripheral Library
- MEMORY

SERCOM1

- Peripheral Library
- UART

TCC0

- Peripheral Library
- TMR

CONSOLE

- System Service
- Core Service
- Instances: - +
- Instance 0
- System Service
- USB_DEVICE_CDC
- UART
- SYS_CONSOLE

COMMAND

- System Service
- Core Service
- SYS_COMMAND
- SYS_CONSOLE

TCP/IP STACK

- SYS_CONSOLE

DEBUG

- System Service
- Core Service
- SYS_DEBUG
- SYS_CONSOLE

TIME

- System Service
- Core Service
- SYS_TIME

U1

PIC32CX1025SG41128T-I/Z2X

Signal	Pin	Function
VDDCORE	114	nRESET
nRESET	113	PA27/TFT_RS
PA27	112	PC28/RX(S1.1)
PC28	111	PC27/TX(S1.0)

RX PAD[1]

TX PAD[0]

Configure the Added Modules

Step 15

- Configure the **Heap** size to **44,960** in **System** to reserve memory for TCP/IP stack.

The image shows two side-by-side screenshots from a development tool interface.

Left Screenshot (Project Graph):

- The **Project Graph** window is open, showing a tree structure of project components. The **System** component is highlighted with a green box and a green circle with the number **1**.
- Below the **System** component, the **TCP/IP STACK** is expanded, showing the **SYS_CONSOLE** component.
- Other components visible include **wolfCrypt Library**, **Core**, **CMSIS Pack**, **Device Family Pack (DFP)**, **EVSYN**, **NVMCTRL**, **SERCOM1**, **TCC0**, **CONSOLE**, **COMMAND**, **TIME**, and **DEBUG**.

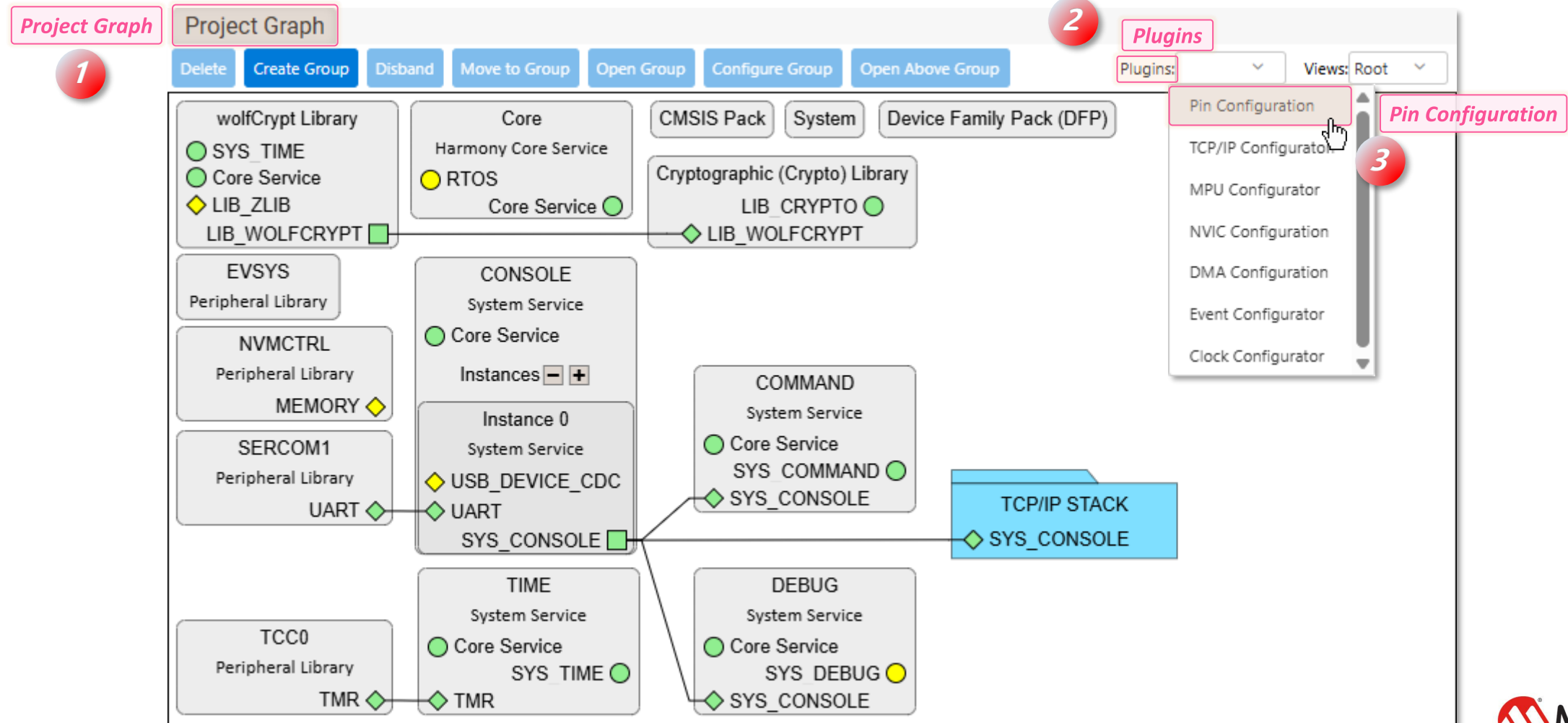
Right Screenshot (Configuration Options):

- The **Configuration Options** window is open, showing a tree structure of configuration options. The **System** component is highlighted with a green box and a green circle with the number **2**.
- Under the **System** component, the **Heap Size (bytes)** option is highlighted with a green box.
- The **Heap Size (bytes)** value is set to **44,960**, which is highlighted with a green box and labeled **44,960 (Heap Size)**.
- Other configuration options visible include **Device & Project Configuration**, **Project Configuration**, **Tool Chain Selection**, **XC32 Global Options**, **Linker**, and **General**.

Configure the Added Modules

Step 16

- Open the **Pin Configuration** in **Plugin** to configure the **I/O pins**.



Configure the Added Modules

Step 17

- Double-click the **Pin Settings** tab to enlarge it and **Order** the pins by **Port**.

Pin Settings

Order: Pins ☒ Easy View

1

2

3

Double-click

Port

Pin Number	Port	Custom Name	Function	Mode	Direction	Latch	Pull Up	Pull Down	Drive Strength
1	PA0		Available	Digital	High Impedance	Low	☐	☐	NORMAL
2	PA01		Available	Digital	High Impedance	Low	☐	☐	NORMAL
3	PC00		Available	Digital	High Impedance	Low	☐	☐	NORMAL
4	PC01		Available	Digital	High Impedance	Low	☐	☐	NORMAL
5	NC			Digital	High Impedance	Low	☐	☐	NORMAL
6	AVDD			Digital	High Impedance	Low	☐	☐	NORMAL
7	PC02		Available	Digital	High Impedance	Low	☐	☐	NORMAL

Configure the Added Modules

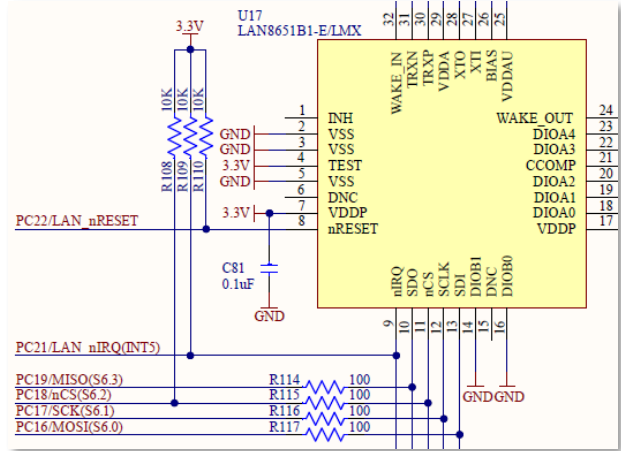
Step 18

- Configure the **SERCOM6_SPI** pins as below. (Custom Name of **SPI** signal is optional)

Pin Number	Pin ID	Custom Name	Function	Mode	Direction	Latch	Pull Up	Pull Down	Drive Strength
		Custom Name	Function	Mode	Direction	Latch			
70	PC16		SERCOM6_PAD0	Digital	High Impedance	n/a	☐	☐	NORMAL
71	PC17		SERCOM6_PAD1	Digital	High Impedance	n/a	☐	☐	NORMAL
72	PC18	SPI_CS	GPIO	Digital	Out	High	☐	☐	NORMAL
73	PC19		SERCOM6_PAD3	Digital	High Impedance	n/a	☐	☐	NORMAL

Peripheral	Pin ID	Custom Name	Function	Direction	Latch
SPI (SERCOM6)	PC16		SERCOM6_PAD0		
	PC17		SERCOM6_PAD1		
	PC18	SPI_CS	GPIO	Out	High
	PC19		SERCOM6_PAD3		

Must match the entire word's case without spaces



Configure the Added Modules

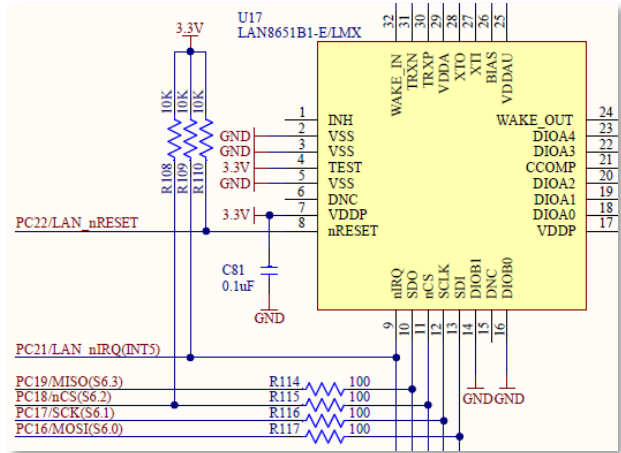
Step 19

- Configure the LAN865x GPIO pins as below. (Custom Name is important)

Pin Number	Pin ID	Custom Name		Function		Mode	Direction		Latch	Pull Up	Pull Down	Drive Strength
		Custom Name		Function			Direction					
75	PC21	TC6_INT	TC6_INT	GPIO	GPIO	Digital	In	In	Low	☐	☐	NORMAL
76	PC22	TC6_RESET	TC6_RESET	GPIO	GPIO	Digital	Out	Out	High	☐	☐	NORMAL

Peripheral	Pin ID	Custom Name	Function	Direction	Latch
GPIO	PC21	TC6_INT	GPIO	IN	
	PC22	TC6_RESET	GPIO	OUT	High

Must match the entire word's case without spaces



Configure the Added Modules

Step 20

- Configure the **SERCOM1_UART** pins. (Custom Name of **UART** signal is optional)

Pin Number	Pin ID	Custom Name	Function	Mode	Direction	Latch	Pull Up	Pull Down	Drive Strength	
		Custom Name	Function		Direction	Latch				
111	PC27	PC27	UART_TX	SERCOM1_PAD0	Digital	High Impedance	n/a	☐	☐	NORMAL
112	PC28	PC28	UART_RX	SERCOM1_PAD1	Digital	High Impedance	n/a	☐	☐	NORMAL

Peripheral	Pin ID	Custom Name	Function	Direction	Latch
UART (SERCOM1)	PC27		SERCOM1_PAD0		
	PC28		SERCOM1_PAD1		

U1		PIC32CX1025SG41128T-I/Z2X	
VDDCORE	114	nRESET	
nRESET	113	PA27/TFT_RS	
PA27	112	PC28/RX(S1.1)	
PC28	111	PC27/TX(S1.0)	
PC27			

Configure the Added Modules

Step 21

- Configure the LED1, LED2 and LED3 on GPIO pin output PA12, PA13 and PA15.
- Configure the user buttons BT1 and BT2 on GPIO pin input PD00 and PD01.

Pin Number	Pin ID	Custom Name	Function	Mode	Direction	Latch	Pull Up	Pull Down	Drive Strength
60	PA12	LED1	GPIO	Digital	Out	Low	☐	☐	NORMAL
61	PA13	LED2	GPIO	Digital	Out	Low	☐	☐	NORMAL
62	PA14		Available	Digital	High Impedance	Low	☐	☐	NORMAL
63	PA15	LED3	GPIO	Digital	Out	Low	☐	☐	NORMAL
13	PD00	BT1	GPIO	Digital	In	Low	☐	☐	NORMAL
16	PD01	BT2	GPIO	Digital	In	Low	☐	☐	NORMAL

Peripheral	Pin ID	Custom Name	Function	Direction	Latch
LED1	PA12	LED1	GPIO	Out	Low
LED2	PA13	LED2	GPIO	Out	Low
LED3	PA15	LED3	GPIO	Out	Low
BT1	PD00	BT1	GPIO	In	
BT2	PD01	BT2	GPIO	In	

Configure the Added Modules

Step 22

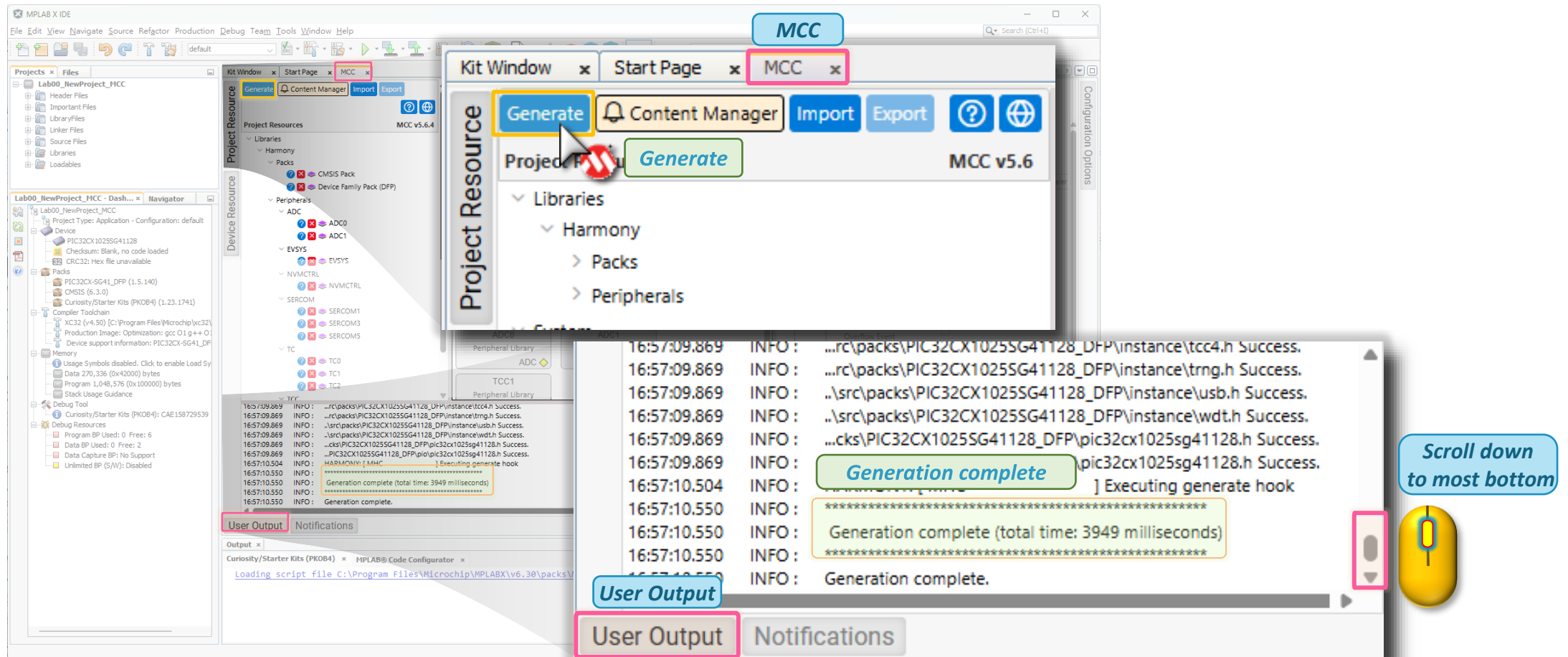
- Overview of completed **Pin Configuration** result.

Pin Number	Pin ID	Custom Name	Function	Mode	Direction	Latch	Pull Up	Pull Down	Drive Strength
60	PA12	LED1	GPIO	Digital	Out	Low	☐	☐	NORMAL
61	PA13	LED2	GPIO	Digital	Out	Low	☐	☐	NORMAL
62	PA14		Available	Digital	High Impedance	Low	☐	☐	NORMAL
63	PA15	LED3	GPIO	Digital	Out	Low	☐	☐	NORMAL
70	PC16	SPI_MOSI	SERCOM6_PAD0	Digital	High Impedance	n/a	☐	☐	NORMAL
71	PC17	SPI_SCK	SERCOM6_PAD1	Digital	High Impedance	n/a	☐	☐	NORMAL
72	PC18	SPI_CS	GPIO	Digital	Out	High	☐	☐	NORMAL
73	PC19	SPI_MISO	SERCOM6_PAD3	Digital	High Impedance	n/a	☐	☐	NORMAL
74	PC20		Available	Digital	High Impedance	Low	☐	☐	NORMAL
75	PC21	TC6_INT	GPIO	Digital	In	Low	☐	☐	NORMAL
76	PC22	TC6_RESET	GPIO	Digital	Out	High	☐	☐	NORMAL
111	PC27	UART_TX	SERCOM1_PAD0	Digital	High Impedance	n/a	☐	☐	NORMAL
112	PC28	UART_RX	SERCOM1_PAD1	Digital	High Impedance	n/a	☐	☐	NORMAL
13	PD00	BT1	GPIO	Digital	In	Low	☐	☐	NORMAL
16	PD01	BT2	GPIO	Digital	In	Low	☐	☐	NORMAL

Code Generation & Application

Step 23

- Click on **Generate** button in top of **MCC** to **generate** your code after configuration.

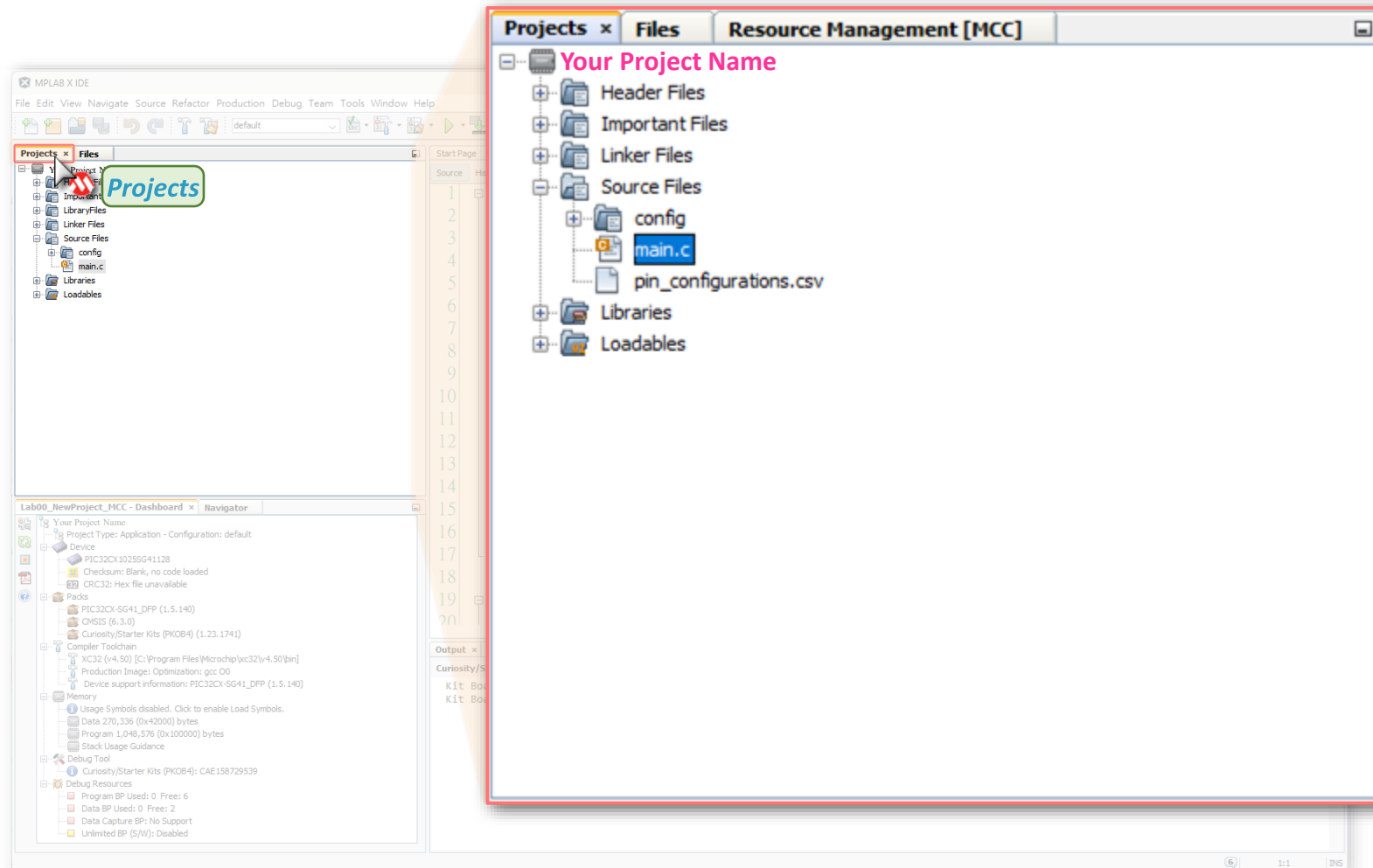


MPLAB® Code Configurator (MCC) Project Properties & Application

MPLAB X IDE Project Manager

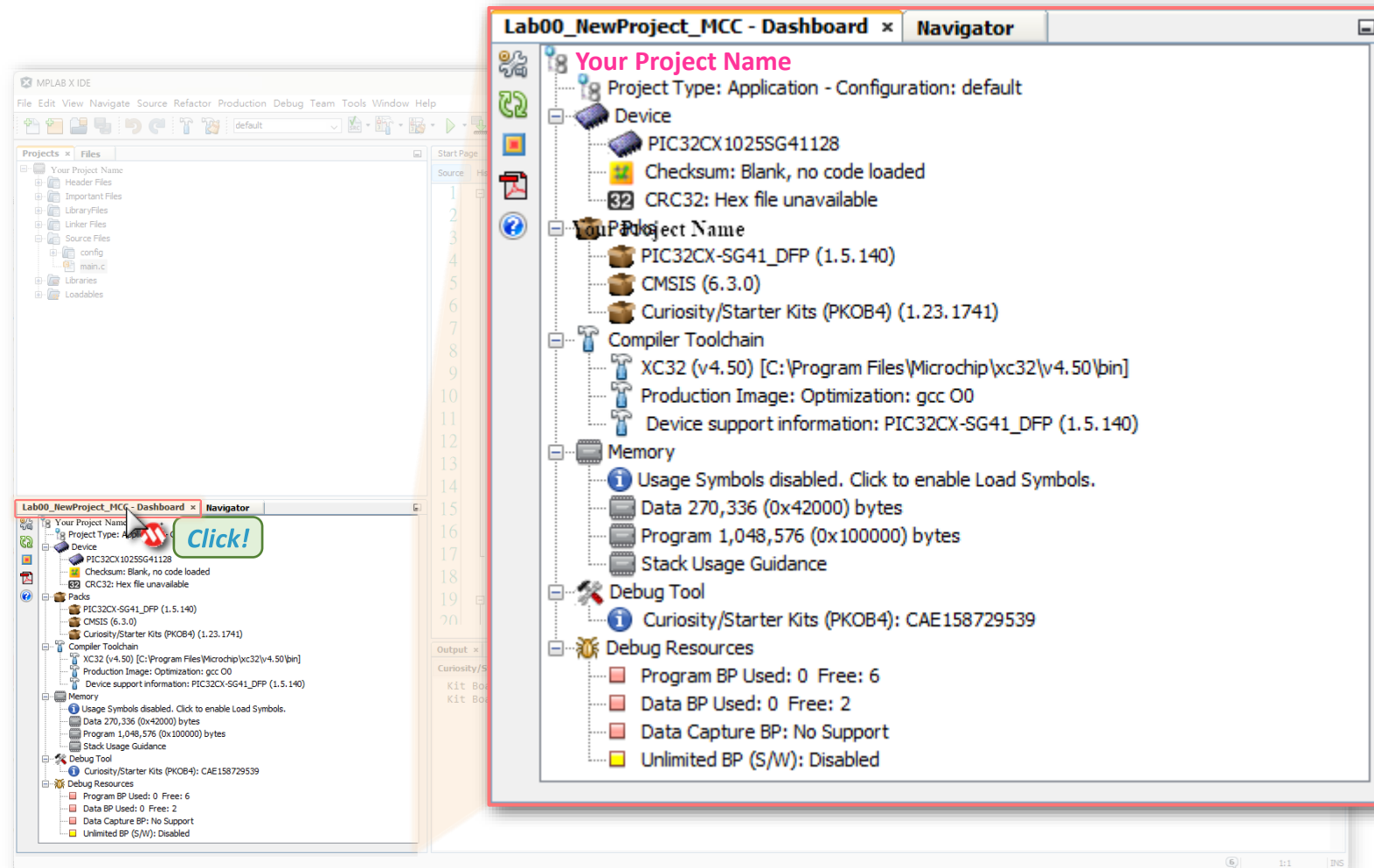
MPLAB X IDE Interface

- Project window.



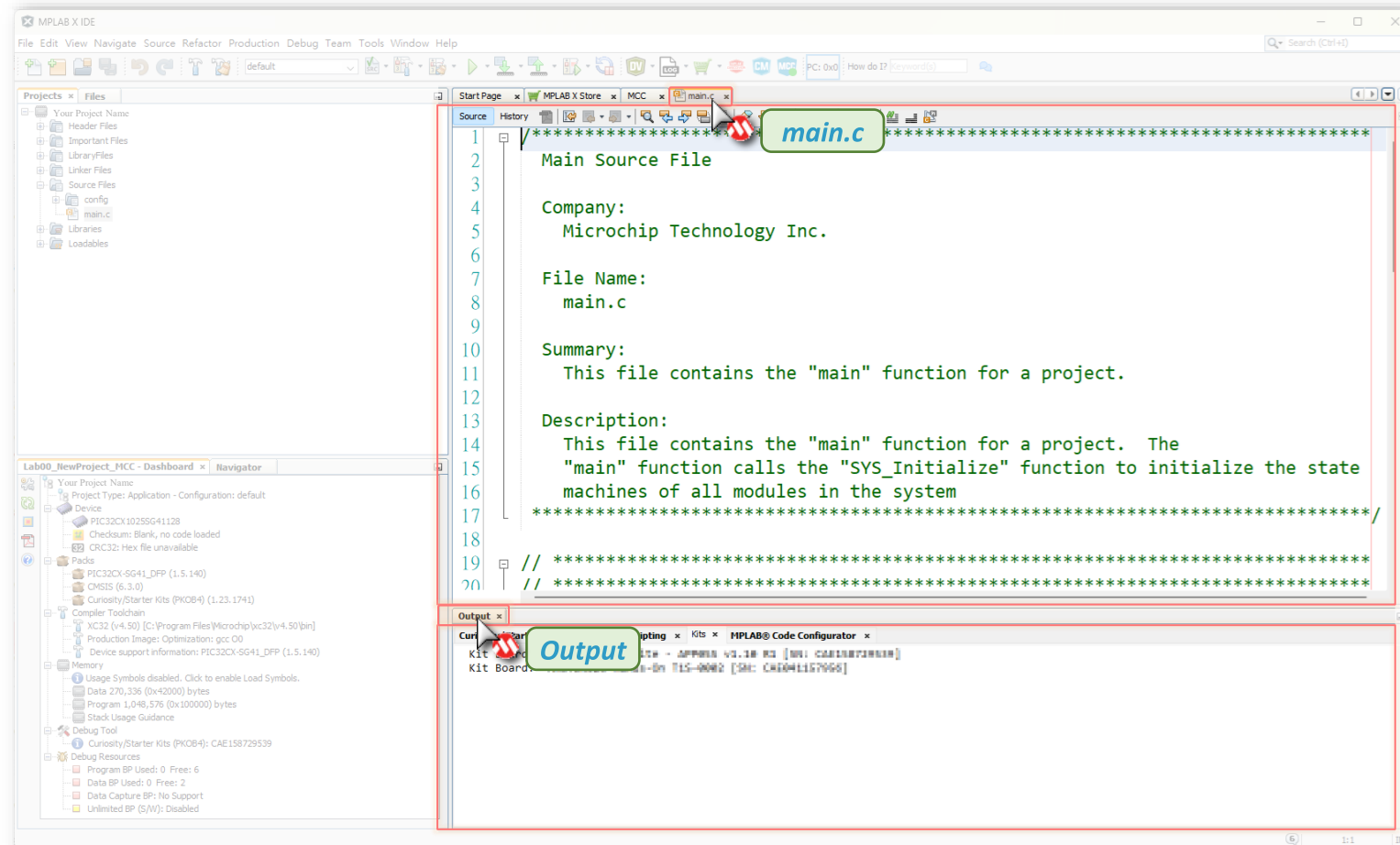
MPLAB X IDE Interface

- Dashboard window. (The version of modules might be different to yours.)



MPLAB X IDE Interface

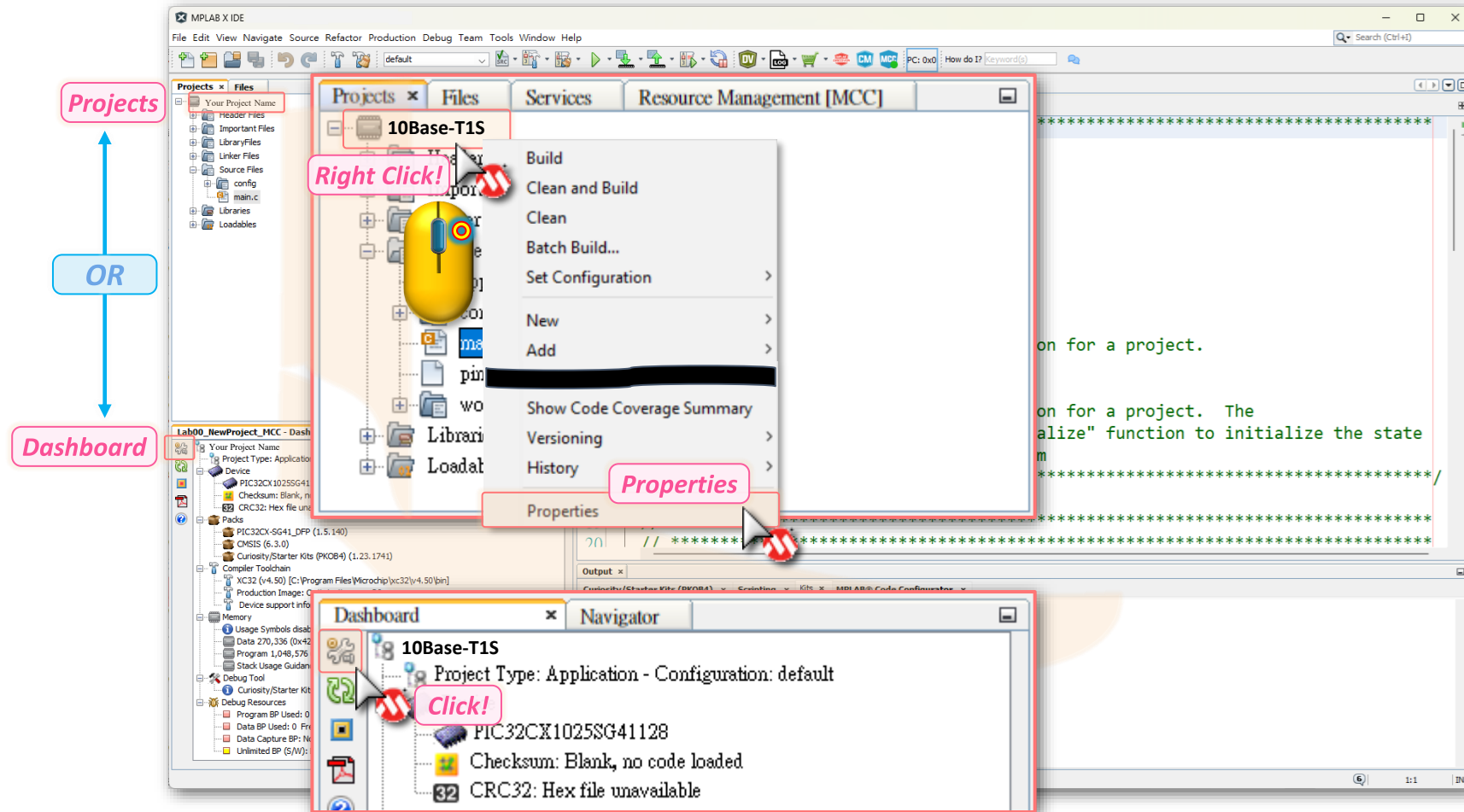
- Source code and Output window.



Project Properties & Application

Step 1

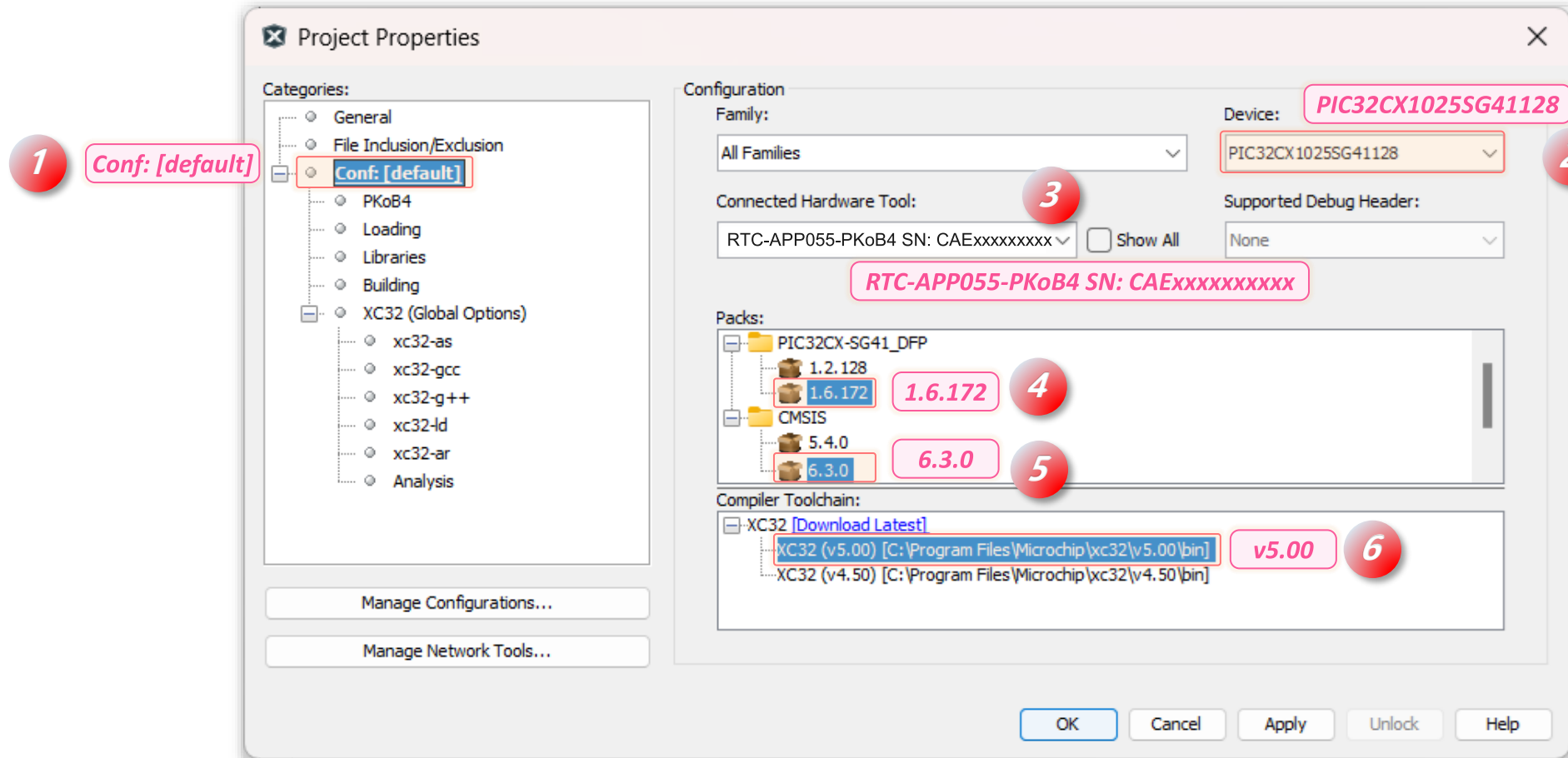
- Click  on **Dashboard** or Right click on **Project Name** and select the **Properties**.



Project Properties & Application

Step 2

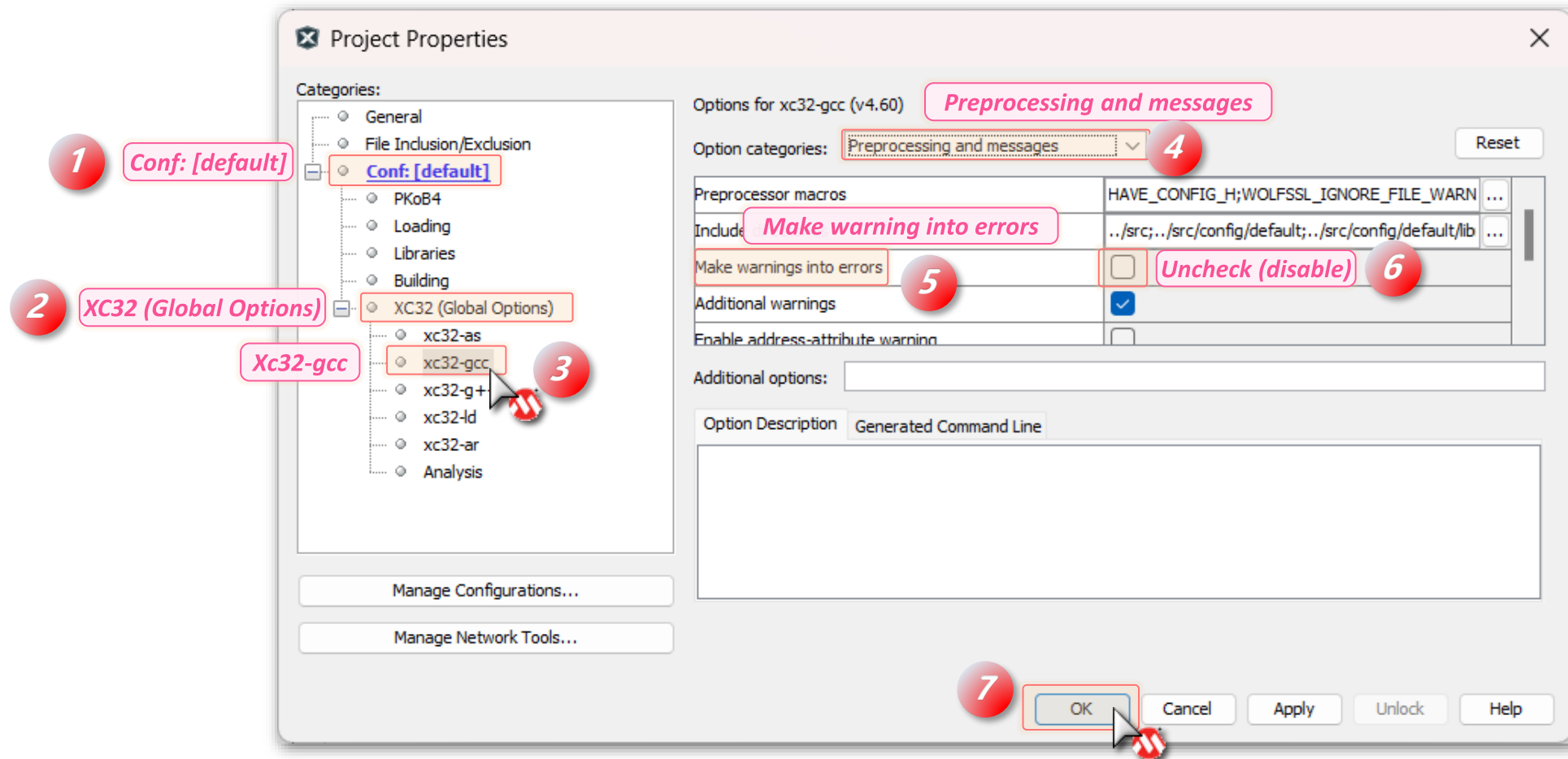
- Under **Conf: [default]**,
Check the **Device**, **Connected Hardware Tool**, **Packs** and **Compiler Toolchain** versions.



Project Properties & Application

Step 3

- Click left configure tree **Conf:[default]** > **XC32 (Global Options)** > **xc32-gcc** and In **Preprocessing and messages** > **uncheck** ☐ **Make warning into errors** to disable it.

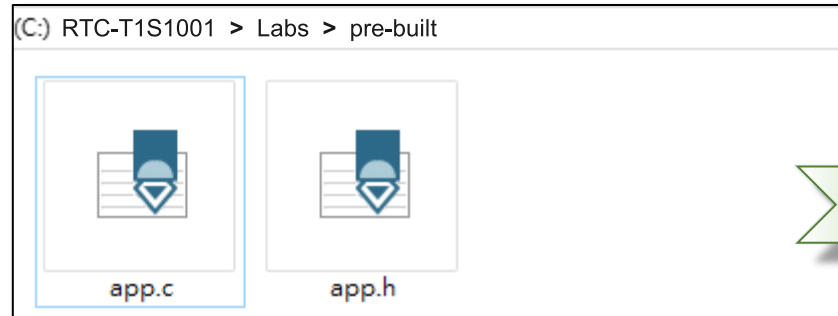


Project Properties & Application

Step 4

- Copy the pre-built **APP.c** and **APP.h** to overwrite the files in the **\src** folder.

C: > RTC-T1S1001 > Labs > pre-built



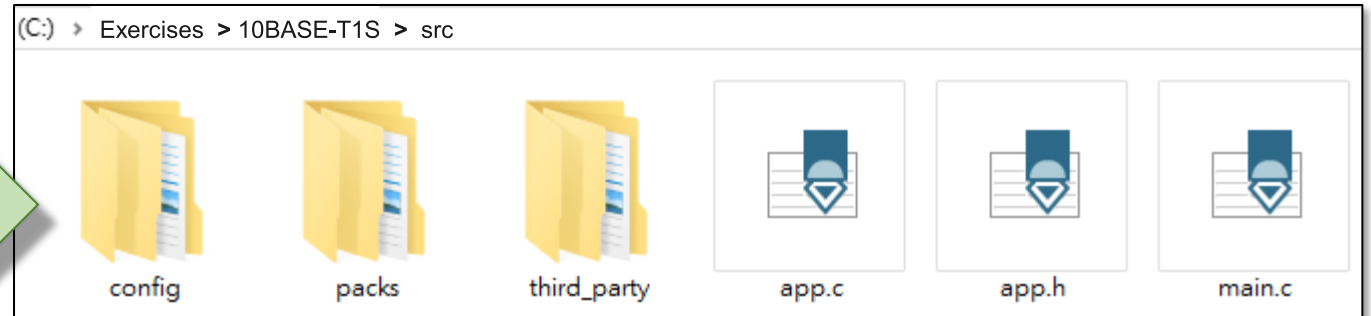
Select All



Copy



> [Your Project] > src >



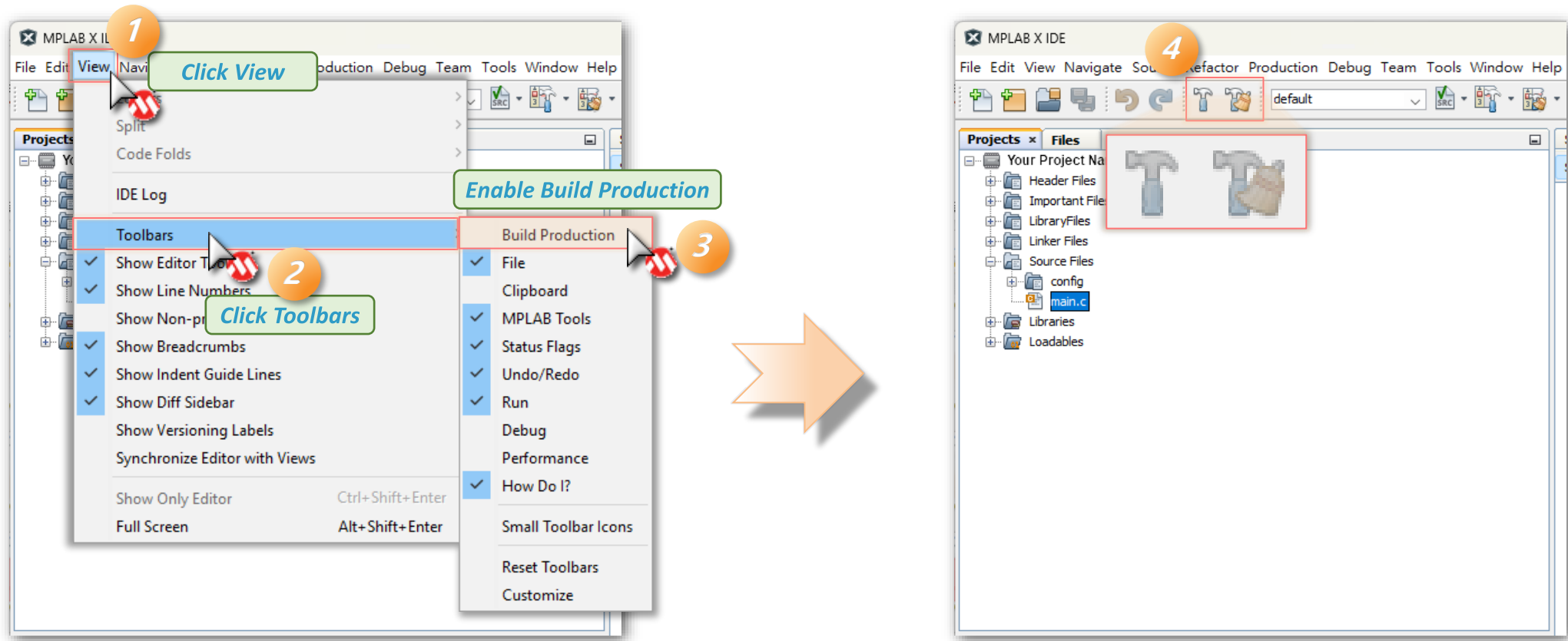
Paste



Project Properties & Application

Step 5

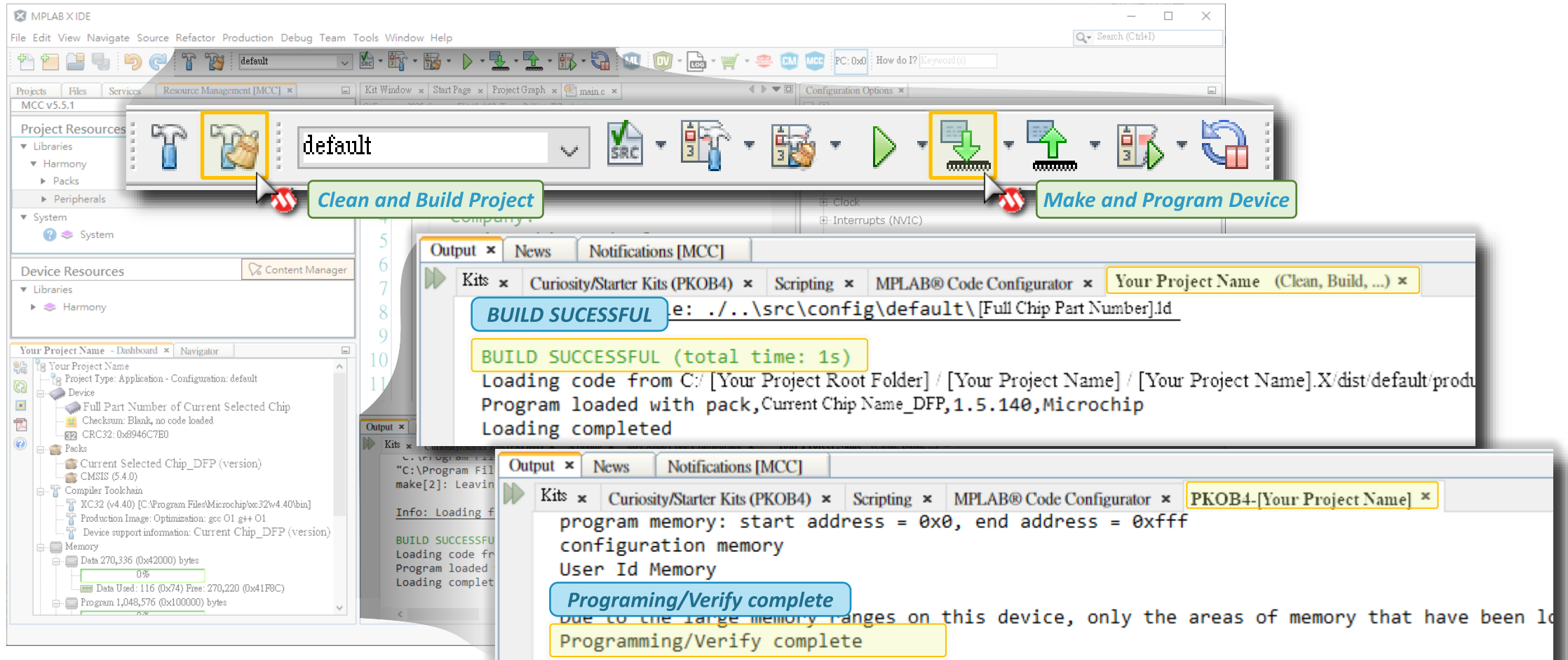
- Add the Build Production to Toolbars to make it easy to Clear and Build projects.



Project Properties & Application

Step 6

- Click on icon  to **Clean and Build Project** and then  to **Program code to EVB.**



The screenshot displays the MPLAB X IDE interface. The top toolbar contains icons for various actions. Two specific icons are highlighted with red circles and arrows pointing to callout boxes:

- Clean and Build Project:** Indicated by a red circle and arrow pointing to the icon showing a hammer and a wrench.
- Make and Program Device:** Indicated by a red circle and arrow pointing to the icon showing a green arrow pointing down to a chip.

The bottom of the screenshot shows the Output window with the following text:

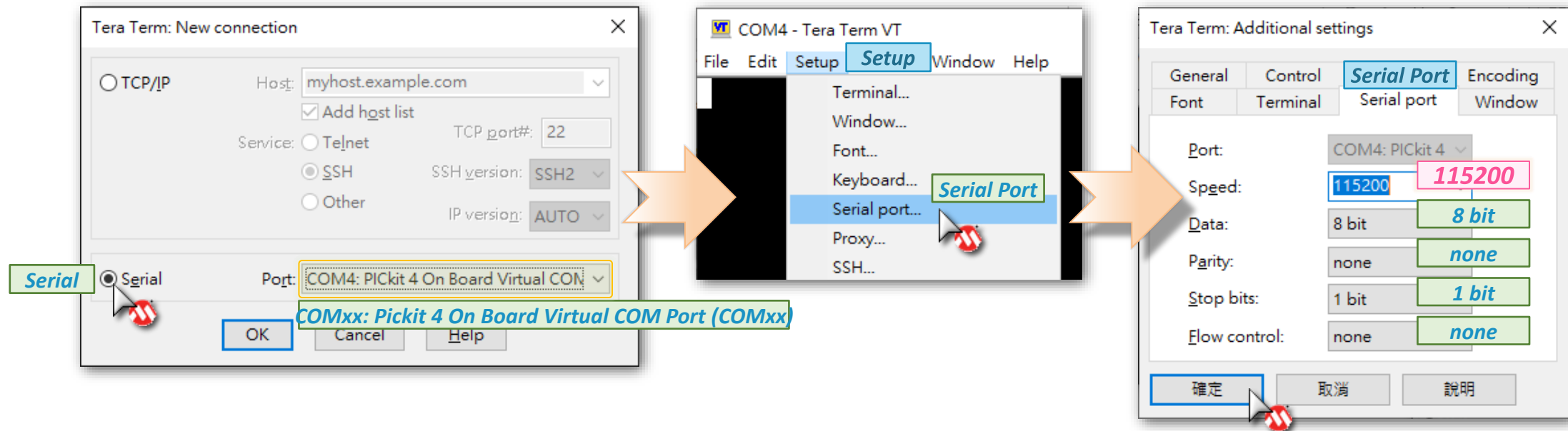
```
Output x News Notifications [MCC]
Kits x Curiosity/Starter Kits (PKOB4) x Scripting x MPLAB® Code Configurator x Your Project Name (Clean, Build, ...) x
BUILD SUCCESSFUL e: ./.../src/config/default/[Full Chip Part Number].ld
BUILD SUCCESSFUL (total time: 1s)
Loading code from C:/[Your Project Root Folder]/[Your Project Name]/[Your Project Name].X/dist/default/produ
Program loaded with pack, Current Chip Name_DFP,1.5.140, Microchip
Loading completed

Output x News Notifications [MCC]
Kits x Curiosity/Starter Kits (PKOB4) x Scripting x MPLAB® Code Configurator x PKOB4-[Your Project Name] x
program memory: start address = 0x0, end address = 0xffff
configuration memory
User Id Memory
Programming/Verify complete
Due to the large memory ranges on this device, only the areas of memory that have been lo
Programming/Verify complete
```

Project Properties & Application

Tera Term : The Terminal Console Software

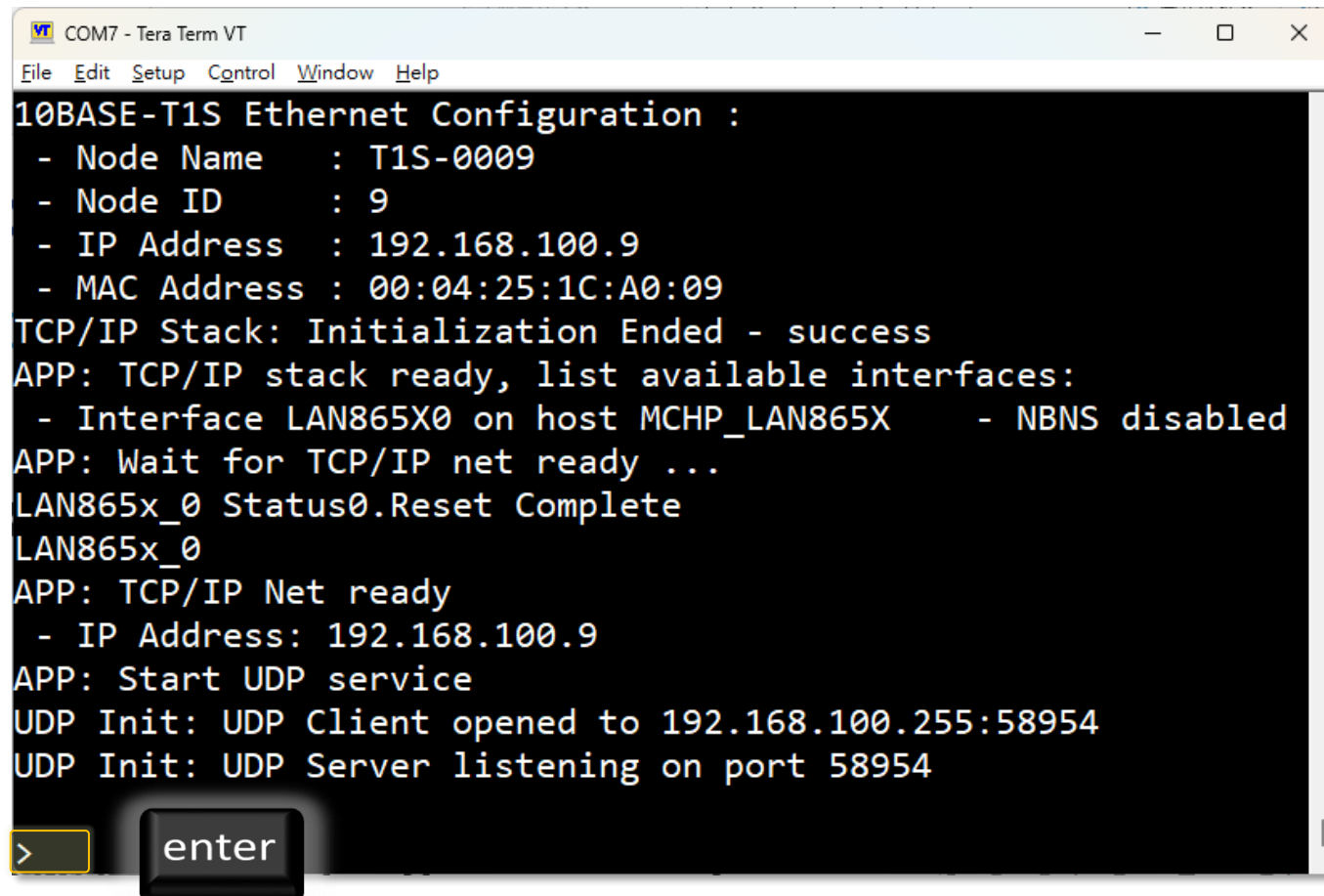
- Tera Term is a popular and free to use terminal console software, you could download it from <https://teratermproject.github.io/index-en.html>
- Execute the Tera Term by double Click on the Icon  on your PC's desktop.
- Choice a New Connection “Serial” and select correct COM Port of your EVB,
 - It might show as **COMxx: Pickit 4 On Board Virtual COM Port (COMxx)**
- Go to the top menu and enter “Setup” tab and config the “Serial Port” to **115200-n-8-1**.



Project Properties & Application

TCP/IP Command Console

- Open Tera Term, you will get the UART debug message output as below.
Type  to display the command prompt character ">" to allow you to type.



The screenshot shows a Tera Term VT window titled "COM7 - Tera Term VT". The menu bar includes File, Edit, Setup, Control, Window, and Help. The main text area displays the following output:

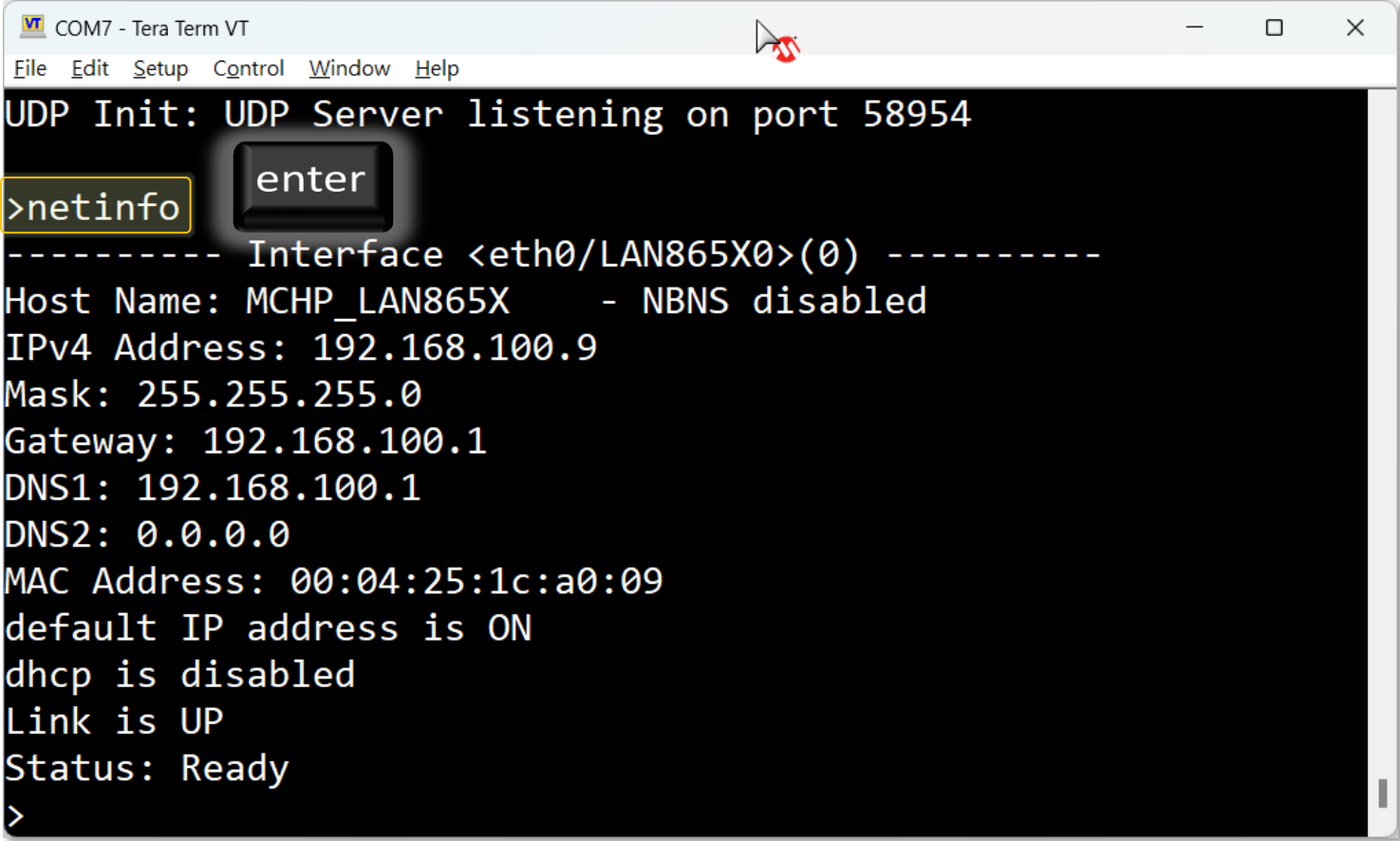
```
10BASE-T1S Ethernet Configuration :  
- Node Name   : T1S-0009  
- Node ID     : 9  
- IP Address  : 192.168.100.9  
- MAC Address : 00:04:25:1C:A0:09  
TCP/IP Stack: Initialization Ended - success  
APP: TCP/IP stack ready, list available interfaces:  
- Interface LAN865X0 on host MCHP_LAN865X - NBNS disabled  
APP: Wait for TCP/IP net ready ...  
LAN865x_0 Status0.Reset Complete  
LAN865x_0  
APP: TCP/IP Net ready  
- IP Address: 192.168.100.9  
APP: Start UDP service  
UDP Init: UDP Client opened to 192.168.100.255:58954  
UDP Init: UDP Server listening on port 58954
```

At the bottom of the window, there is a command prompt character ">" and an "enter" key icon.

Project Properties & Application

TCP/IP Command Console - netinfo

- Typin in : “**netinfo**”  will output current network configuration of your EVB.



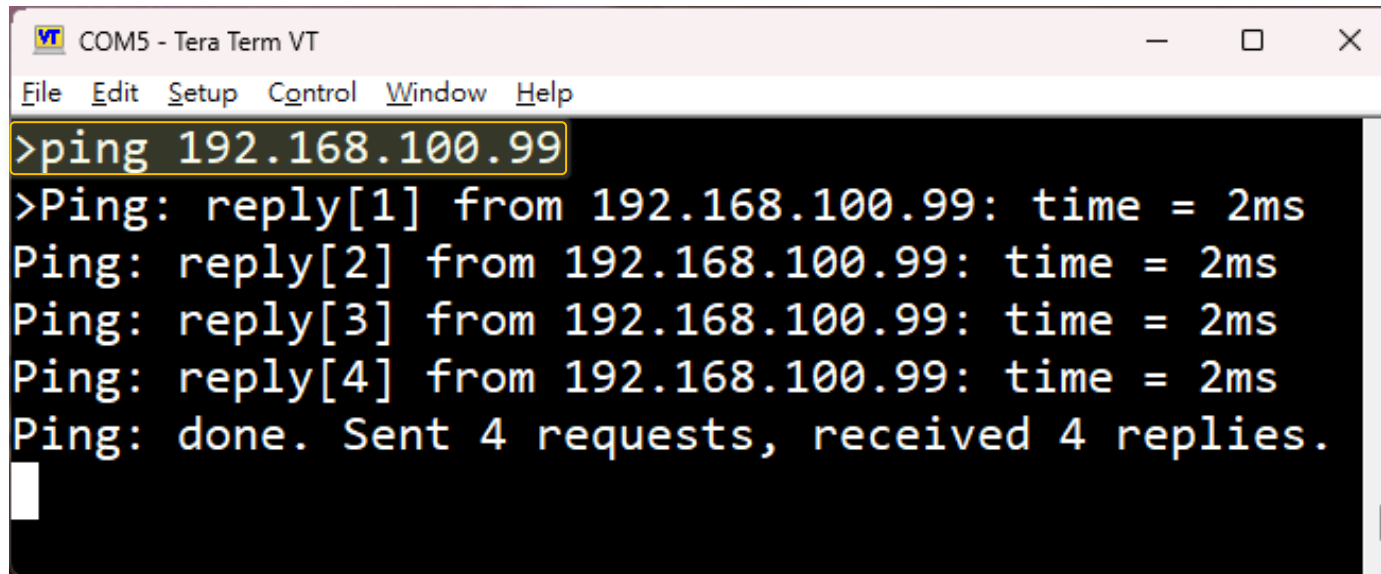
```
COM7 - Tera Term VT
File Edit Setup Control Window Help
UDP Init: UDP Server listening on port 58954
>netinfo
----- Interface <eth0/LAN865X0>(0) -----
Host Name: MCHP_LAN865X      - NBNS disabled
IPv4 Address: 192.168.100.9
Mask: 255.255.255.0
Gateway: 192.168.100.1
DNS1: 192.168.100.1
DNS2: 0.0.0.0
MAC Address: 00:04:25:1c:a0:09
default IP address is ON
dhcp is disabled
Link is UP
Status: Ready
>
```

Project Properties & Application

TCP/IP Command Console - ping

- Typing “cls”  to clean console screen.
- “ping 192.168.100.99”  will attempt to ping the instructor's host.

T1S Host IP



```
VT COM5 - Tera Term VT
File Edit Setup Control Window Help
>ping 192.168.100.99
>Ping: reply[1] from 192.168.100.99: time = 2ms
Ping: reply[2] from 192.168.100.99: time = 2ms
Ping: reply[3] from 192.168.100.99: time = 2ms
Ping: reply[4] from 192.168.100.99: time = 2ms
Ping: done. Sent 4 requests, received 4 replies.
```

Project Properties & Application

TCP/IP Command Console - send

- Typing “**send Message**”  to broadcast a message to everyone.
- If your message contains spaces, please use **"Message from Microchip"**.

1

```
COM7 - Tera Term VT 192.168.100.9
File Edit Setup Control Window Help
>send "Message from 192.168.100.9"
- UDP Client TX Done!
T1S-0009 ( 3 ) -> Message from 192.168.100.9
```

4

```
> - UDP Server RX Data Signal!
T1S-0009 ( 1 ) <- T1S-0002, Feedback from 192.168.100.2
```

2

```
COM6 - Tera Term VT 192.168.100.2
File Edit Setup Control Window Help
> - UDP Server RX Data Signal!
T1S-0002 ( 5 ) <- T1S-0009, Message from 192.168.100.9
```

3

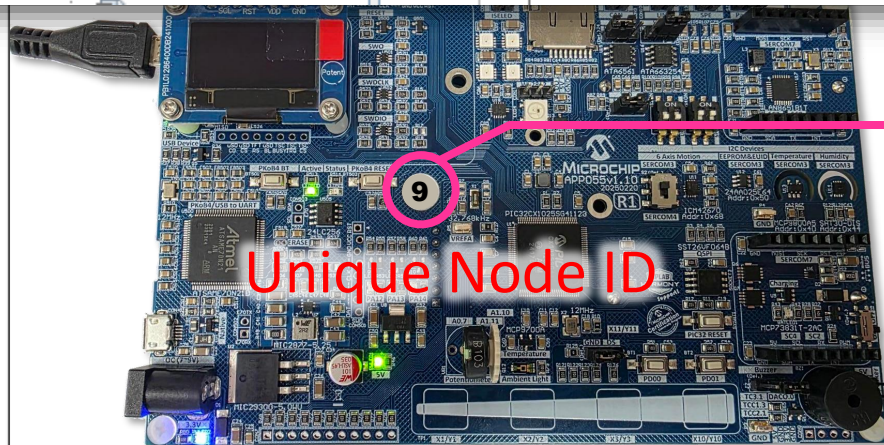
```
send "Feedback from 192.168.100.2"
- UDP Client TX Done!
T1S-0002 ( 2 ) -> Feedback from 192.168.100.2
>
```

Project Properties & Application

Change the MCC configuration by directly modifying the program code.

- Modify the **Node ID** and **Node Count** in line 144 and 145 of **configuration.h**

```
141 #define DRV_LAN865X_CHUNK_SIZE_IDX0 64
142 #define DRV_LAN865X_CHUNK_XACT_IDX0 31
143 #define DRV_LAN865X_PLCA_ENABLE_IDX0 true
144 #define DRV_LAN865X_PLCA_NODE_ID_IDX0 9
145 #define DRV_LAN865X_PLCA_NODE_COUNT_IDX0 255
146 #define DRV_LAN865X_PLCA_BURST_COUNT_IDX0 0
```



Unique Node ID

Optional Configure the RESET pin as GPIO Output in Pin Manager***

10BASE-T1S Operation Mode

PLCA Settings

Local Node Id **Local Node Id**

Node Count **Node Count**

Max Burst Count

Burst Timer

**** Burst Timer value is 12.8 us ****

Advanced Settings

TC6 chunk size 64

TC6 chunks per SPI Transaction (XACT) 31

Promiscuous

TX cut through

RX cut through

Project Properties & Application

Change the MCC configuration by directly modifying the program code.

- Modify the **MAC Address** and **IP Address** in line 258 and 260 of **configuration.h**

The screenshot displays the Microchip Studio interface. The top section shows the 'Project Graph' with the 'DATA LINK LAYER' selected. The 'NETCONFIG' block is highlighted, and its 'Instance 0' is selected. The 'Configuration Options' panel on the right shows the 'Mac Address' and 'IPv4 Static Address' fields. The bottom section shows the 'configuration.h' file with the following code:

```
257 #define TCPIP_NETWORK_DEFAULT_HOST_NAME_ID0 "MCHP_LAN865x"
258 #define TCPIP_NETWORK_DEFAULT_MAC_ADDR_IDX0 "00:04:25:1C:A0:09"
259
260 #define TCPIP_NETWORK_DEFAULT_IP_ADDRESS_IDX0 "192.168.100.9"
261 #define TCPIP_NETWORK_DEFAULT_IP_MASK_IDX0 "255.255.255.0"
262 #define TCPIP_NETWORK_DEFAULT_GATEWAY_IDX0 "192.168.100.1"
```

Annotations indicate that the MAC Address and IPv4 Static Address should be modified to ensure uniqueness. The 'Mac Address' field is highlighted in green, and the 'IPv4 Static Address' field is highlighted in green. The 'Instance 0' label is highlighted in orange. The 'Configuration Options' panel is highlighted in orange. The 'configuration.h' file is highlighted in orange. The 'Mac Address' and 'IPv4 Static Address' fields are highlighted in green. The 'Instance 0' label is highlighted in orange. The 'Configuration Options' panel is highlighted in orange. The 'configuration.h' file is highlighted in orange.

1 Ensure the Mac Address is unique
2 Ensure the IPv4 Address is unique

1 "MCHP_LAN865x"
2 "00:04:25:1C:A0:09"
2 "192.168.100.9"
2 "255.255.255.0"
2 "192.168.100.1"

IPv4 Static Address 192.168.100.9 2

EVB-LAN8670-USB Setup

10BASE-T1S Ethernet Host

EVB-LAN8670-USB Setup - 10BASE-T1S Ethernet Host

- The EVB-LAN8670-USB interconnects an USB interface with a 10BASE-T1S Ethernet network interface. The adapter serves as a network card that connects applications via USB 2.0 to the 10BASE-T1S network interface. Document and drivers could download from <https://www.microchip.com/en-us/development-tool/ev08l38a>



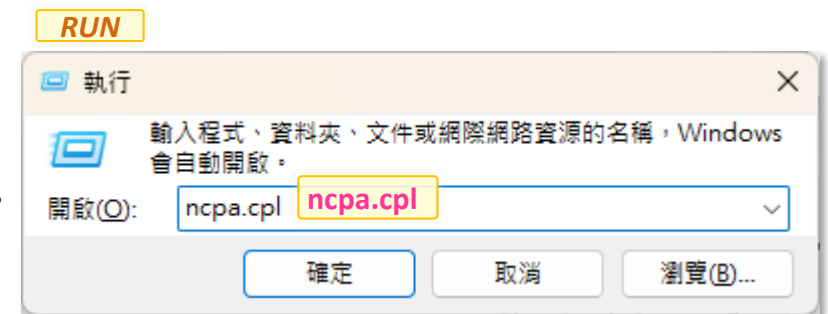
- Go **Control Panel\All Control Panel Items\Network Connections** or RUN **ncpa.cpl**

Control Panel\All Control Panel Items\Network Connections



Network Card of EVB-LAN8670-USB

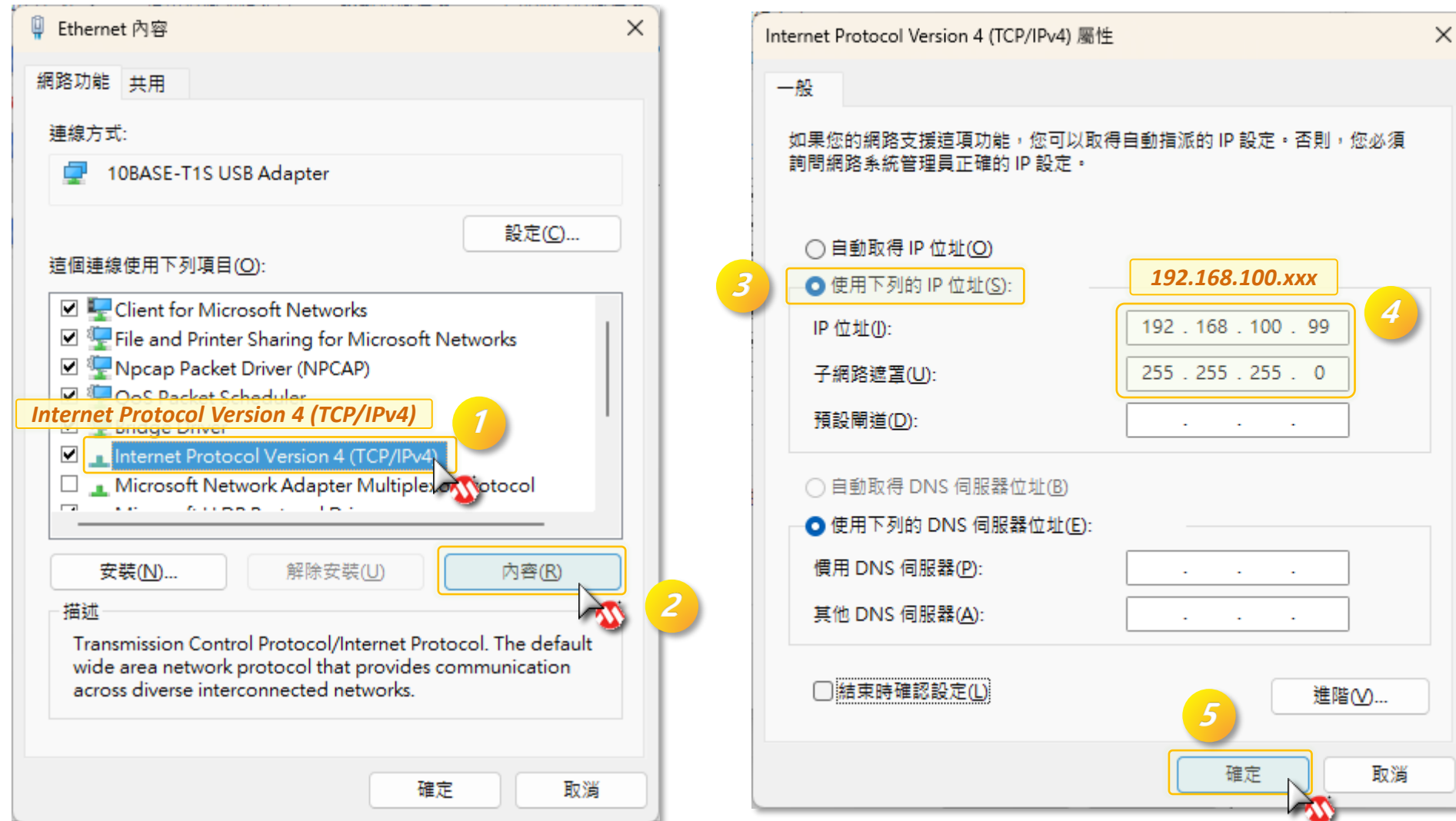
or



* The Network Card Name could be customized to "10BASE-T1S USB Adapter" to easy find the device

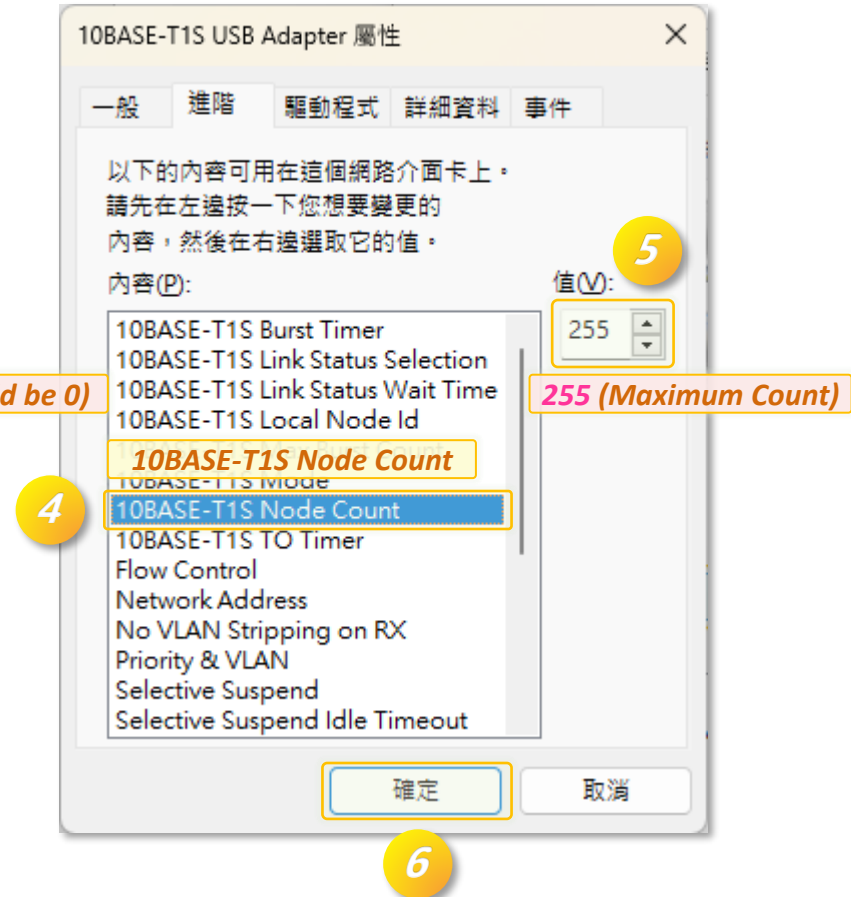
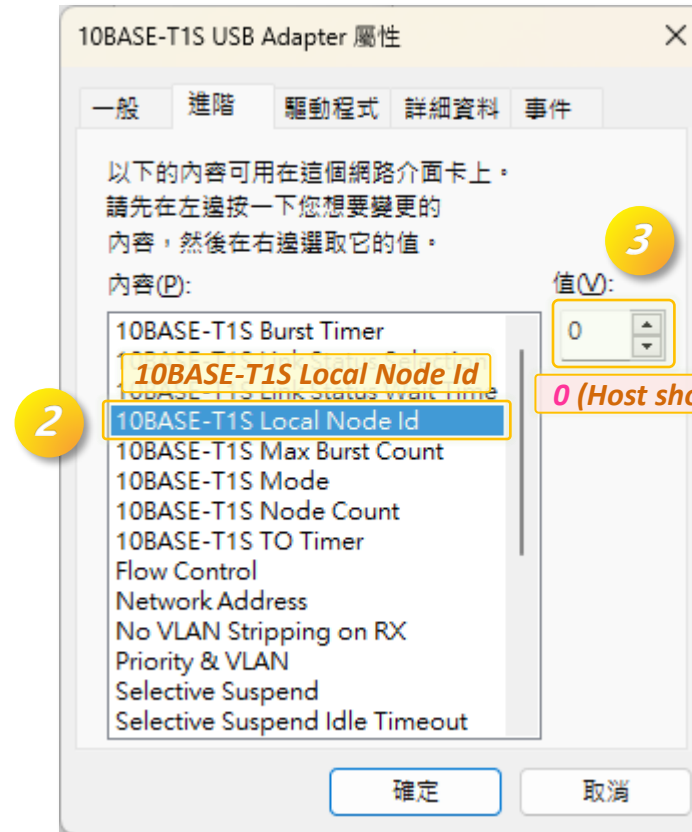
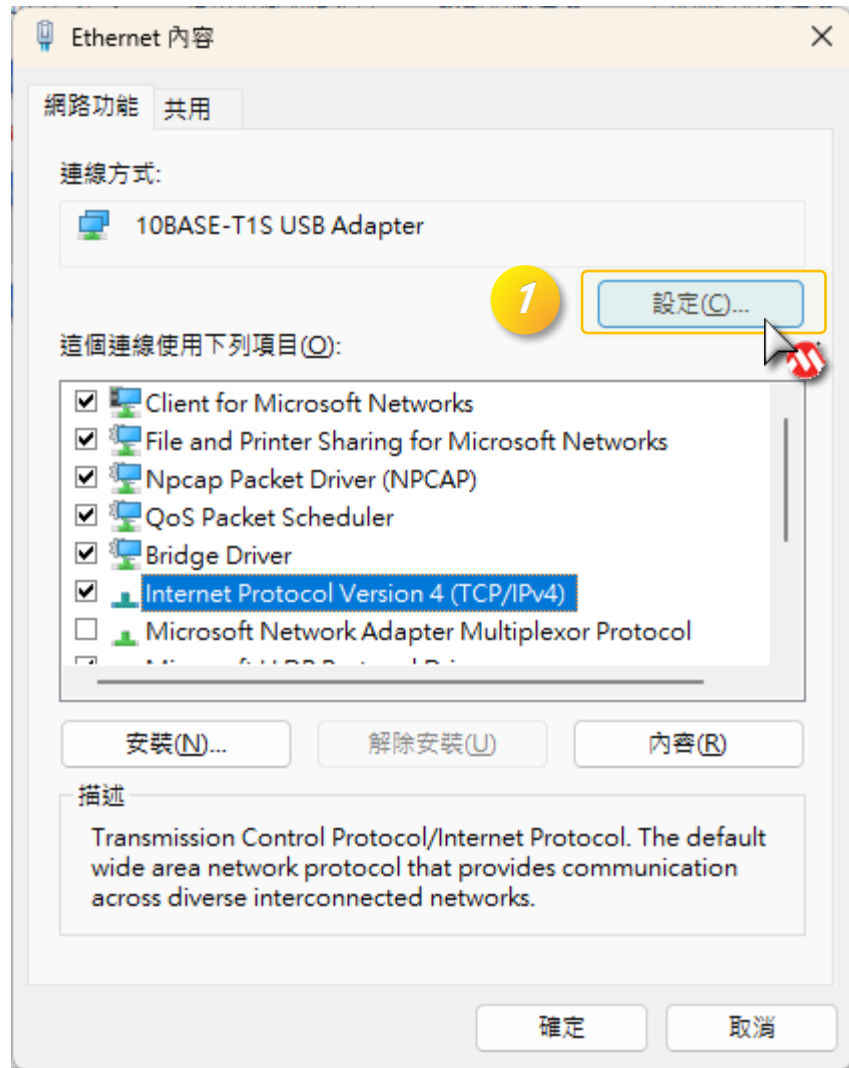
EVB-LAN8670-USB Setup - 10BASE-T1S Ethernet Host

- Setup Fixed IPv4 address of Network Card of the EVB-LAN8670-USB.



EVB-LAN8670-USB Setup - 10BASE-T1S Ethernet Host

- Configure the 10BASE-T1S Node ID (Host should be 0) and maximum Node count 255.



Microchip Trademarks

<https://www.microchip.com/en-us/about/legal-information/microchip-trademarks>

Trademark	Description/Appropriate Name	Notes
ClockWorks™	configurator, device, family, product, series	US only
CodeGuard™	security	
CryptoAuthentication™	development library, device, family, IC, library	
CryptoAutomotive™	development kit, security IC, technology	
CryptoCompanion™	chip	
CryptoController™	solution, TPM, trusted platform module	
CryptoMemory™	cryptographic security IC, IC	
CryptX™	device, transponder	
dsPIC™	DSC, digital signal controller	
dsPIC™ logo	dsPIC™ logo	
dsPICDEM™	demonstration board, development board	
dsPICDEM.net™	board	
Dynamic Averaged Matching™ (DAM™)	architecture	
ECAN™	solution, technology	
Espresso T15™	device	
EtherGREEN™	family, platform, technology	
EtherSynch™	family, platform, technology	US only
EyeOpen™	cable compensation	
Flashtec™	controller, family, product	US only
FlexPWR™	technology	
Frequency on Demand™		Registered in other countries (not US)
Gecko™	controller, sensing, technology	
GridTime™	device, GMS time server, time server	
HELDO™	family, regulator	
Hyper Speed Control™	architecture, family	US only
HyperLight Load™	mode	US only
In-Circuit Serial Programming™ (ICSP™)	programming capability	
IsiBridge™	dual MOSFET bridge inverter, rectifier	
ICat™	board, demonstration board	
IGLOO™	device, family, FPGA, series	
INCEnt™	sniffer, technology	
Note: *Always check with the Legal Department for the most current trademarks.		

Trademark	Description/Appropriate Name	Notes
IntDrive™	driver board, integrated solution, power module	
Intelligent Paralleling™	technology	
IntelliMOS™	family, power stage family	US only
Inter-Chip Connectivity™	technology	
JitterBlocker™	attenuator, family	
Jukeblox™	platform, technology	
jukeblox™	jukeblox™ registered word mark with a design	Image is not registered; this is a graphic treatment with the registered word mark
KeeLoq™	protocol, security IC, technology	
Kleer™	technology	
KLEER™	Kleer™ registered word mark with a design	Image is not registered; this is a graphic treatment with the word mark
Knob-on-Chip™ (Knob™)	device, family, HMI, knob designer, position wizard, solution, technology, touch controller, user interface, widget configurator	
LANCheck™	online design review, online review, review	
Libero™	design file, designer, IDE, SoC design tools, software, tool	US only
LINK™	cable diagnostics, diagnostics engine, family, technology	
MagneTAP™	signal integrity testing	
maxCrypto™	controller-based encryption, encryption	
RealStyle™	solution	
multiTouch™	family, studio, technology, touchscreen controller	
modbus™	driver, storage manager, tool	
ModuS™	bus, controller, device	
megaBaud™	device, MCU, microcontroller	
mem Sloan™	neuromorphic memory product, product, solution, technology	
Microchip logo	Microchip logo	Please do not use this logo; always use the Microchip name and logo.
Note: *Always check with the Legal Department for the most current trademarks.		

Trademark	Description/Appropriate Name	Notes
	Microchip name and logo	This is the vertical representation; the horizontal representation is also available.
	Microsemi logo	Do not use
Microsemi®		Do not use
Mind™	analog simulator, analog simulator tool	
Mini™	application, coordinator, mesh, network, protocol, software, stack, wireless networking protocol	
MOST™	board, cooperation, design, device, evaluation kit, NetServices, network, platform, product, specification, system, technology	
	MOST® logo	
ProtonBench™	development suite	US only
MPX™	assembler	
MPX™	device, product	
MPLAB™	Names: C compiler, compiler, development ecosystem, ICD, ICE, IDE, PE, in-circuit debugger, in-circuit emulator, integrated development environment, integrated programming environment, library, programmer, server, starter kit, universal device programmer, XC compiler Tool Names: AI Edge, Analog Designer, Analog Test Suite, Code Configurator, Connect Configurator, Data Visualizer, Discover, Extensions, Harmony, Mind™, Network Creator, PICkit™, Programmer-to-Go, Programmer-to-Go Mobile App, PPS, SoC Power Simulator, Snap, Xpress	
	MPLAB® Certified logo	
MPLIB™	libraries, object libraries	
Note: *Always check with the Legal Department for the most current trademarks.		

Trademark	Description/Appropriate Name	Notes
MPLINK™	linker, object linker	
mSC™	bare die, diode, gate driver, module, MOSTET, product, solution, technology	
miTouch™	development kit, evaluation kit, library, sensing solution, solution, technology	US only
MultiTRAC™	technology	
NetDetach™	technology	
On-Chip Code Generation™ (OCG)	compilation technology	
OptoLite™	network analysis tool, studio, tool	
pic™	assembler, device, MCU, microcontroller	
	PIC32 logo	
	PIC64 logo	
PICDEM™	demonstration board	
PICDEM.net™	Internet/Ethernet demonstration board	
PICkit™	debugger, development tool, in-circuit debugger, on-board, programmer, starter kit, tool	
picPower™	technology	
PICSTART™	development programmer	
PICtail™	board, daughter board	
PolarFire™	device, family, FPGA, kit, SoC, System-on-Chip	
PowerSmart™	designer, development suite, digital control library	
Precision Edge™	family, product family	US only
ProASIC™	architecture, board, family, FPGA, kit	US only
ProASIC Plus™	architecture, device, family, FPGA, switch	US only
Prochip Designer™	software suite	
PureEditor™	editor, technology	
QMatrix™	device, sensor	
QTouch™	board, compiler, configurator, device, development platform, library, project builder, sensor, solution, technology, tool	
Quiet-Wire™	enhanced EMI technology, family, technology	
Note: *Always check with the Legal Department for the most current trademarks.		

Trademark	Description/Appropriate Name	Notes
Single Block™	attenuator, family of linear regulation, product line, technology	
REAL ICE™		
RT5A™	FPGA	
RT5A™	FPGA	
SAM 9A™	applet, application, host, in-system programmer, ISP, monitor, software emulator	
SAM-ICE™	product, sensor	
SenseMemory™	family, flash device, product	
Serial Quad I/O™ (SQI™)		Japan only
Silicon Storage Technology®		
SimpleLink™	protocol	
SimpleLink™	device, family, product	
SmartFusion™	device, IC	
SmartFusion™	evaluation kit, FPGA, SoC, System-on-Chip	US only
SmartLink™	compiler, compiler software, IDE, software	
SMART-LS™	technology	
SpAHC™	device	
SST™	Acronym for Silicon Storage Technology	
	SST logo	
A Microchip Technology Company		
STX™	starter kit	
storCust™	encryption technology, technology	
SuperFlash™	device, IC, matrix, memory, technology	
SuperSwitcher™	bus regulator, family, regulator	
SuperSwitcher II™	family	
Switchbox™	family, product line, switch, technology	
Symmetron™		China only
Symmetron™		
SynchroPac™	device, family, product	
SyncServer™	clock, device, instrument, oscillator, server	
Note: *Always check with the Legal Department for the most current trademarks.		

Trademark	Description/Appropriate Name	Notes
SynWorld™	ecosystem program, timing	US only
Tachyon™	controller, family, platform, product, technology	
The Embedded Control Solutions Company™		US only; should be in italics
TimeExam™	primary frequency reference, product	US only
TimeLink™	platform, system	US only
TimePilot™	platform, software suite, synchronization management system	US only
TimeProvider™	clock, enhanced PRTC, APRTC, extension, system, grandmaster, kit, module, product, series, series, shelf	US only
TimeSource™	PIC, PIC, primary reference clock, primary reference source, shelf, system	
tinyAUC™	MCU, microcontroller	
tinyAUC™		
Total Endurance™	software model	
TSP™	controller, touchscreen controller	
Tuning™	hardware security module, program	
UNIC®	box, communication protocol, firmware, hardware port, memory chip	
USBCheck™	design review	
VarSense™	technology	
VectorBox™	accelerator, intellectual property, IP, platform, SDK, software development kit, solution, tool, tool chain, tool flow, oscillator, products	
Vibron™		
Vibron™	CT, configuration tool, product, solution, software platform, SP	
ViewSpan™	cable diagnostics algorithm, cable diagnostics software suite	
ViewSpan™	technology	
WaterLock™		
XMEGA™	Custom Logic (XCL) device, family, MCU, microcontroller, series	
XpressConnect™	retimer	
ZENA™	analyzer, software, wireless adapter	
ZL™		US only
Note: *Always check with the Legal Department for the most current trademarks.		

Trademark	Description/Appropriate Name	Notes
	Adaptive™ logo	Registered in other countries (not US)
Adapter™	adapter, cable, controller, driver, family, product, series, software, solution	
Adjacent Key Suppression™	technology	
AgileSwitch™	device, digital programmable gate driver, driver, family, gate driver	US only
AKS™	technology	
Asking for the Digital Age™	APD family tagline	
Any Capacitor™	stable	
AnyCap™	capability, feature, structure	
AnyDut™	capability, feature	
Augmented Switching™	technology	
AVR™	architecture, CPU, device, family, MCU, microcontroller	
	AVR™ logo	
AVR Freaks™	forum	
BackTime™	technology	
BeaCust™	SDK, software development kit	
BlackSky™	firewall, subscription service, technology	
BodyCom™	development kit, system, technology	
chipKIT™	IO shield, board, hardware, network, platform, software	
chipKIT™	chipKIT logo	
ClockStudio™	application, program, software, utility	
Note: *Always check with the Legal Department for the most current trademarks.		