PIONIERKRAFT

STM32 H7 43 ZI Ethernet Implementirung

Bewerbung Ing. Marinero

Ethernet Implementierung

- Allgemeines:
  - Zu verwendende Entwicklungsumgebung: STM32CubeIDE
  - Implementierung in neuem STM32-Projekt
  - Implementierung **ohne** Operation System (RTOS, o.ä.), falls möglich
- Aufgabenstellung:
  - Implementierung von Ethernet auf einem STM32H743ZIT6 Microcontroller
    - Entweder TCP oder UDP. Please implement the safer one.
  - Senden von Daten (25 Werte (float 32 bit)) zu einer Adresse
    - Werte vor dem Senden in JSON-Format umwandeln (besser für Backend, Web)
  - Empfangen von Daten (25 Werte (float 32 bit)) von einer Adresse
    - Werte wieder in float zurück wandeln
- Zusätzliche Fragestellungen:
  - Unterschied TCP und UDP erklären
  - Aufbau JSON-Nachricht
  - Wie können die im JSON-Format gesendeten Daten in einem Backend abgelegt werden (Schrittfolge aufzeigen, kein Code)?
- Präsentation (in Powerpoint):
  - Vorstellung der Aufgabenstellung in STM32CubeIDE
  - Vorstellung der zusätzlichen Fragestellungen

STM32 H7 43 ZI T6

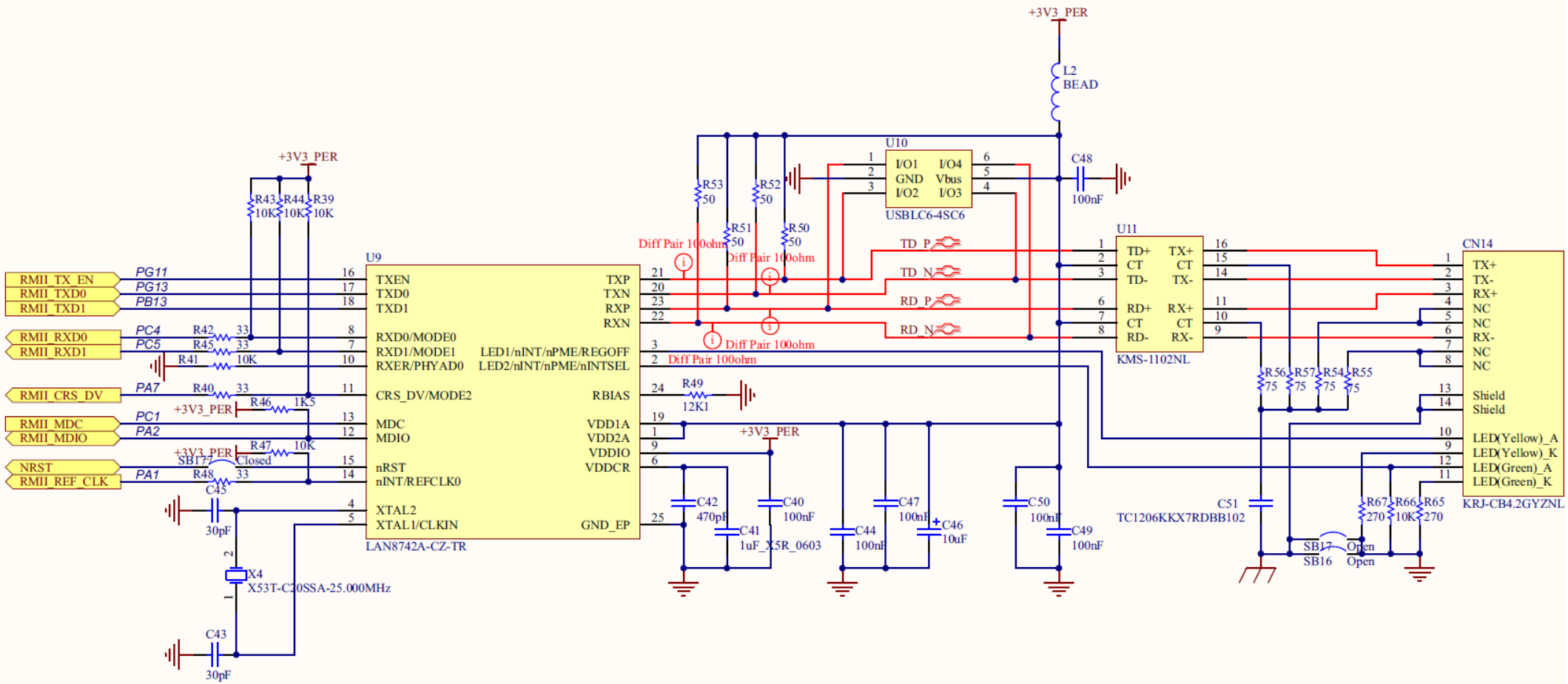
Order code	Board reference	User manual	Target STM32	Differentiating features
NUCLEO-F767ZI	MB1137	UM1974	STM32F767ZIT6U	• Arm® Mbed Enabled™ • Ethernet • On-board USB OTG • USB OTG FS on Micro-AB connector • ST-LINK/V2-1
NUCLEO-H743ZI			STM32H743ZIT6U	• Arm® Mbed Enabled™ • Ethernet • USB OTG FS on Micro-AB connector • ST-LINK/V2-1
NUCLEO-H743ZI2	MB1364	UM2407	STM32H743ZIT6U	• Ethernet • USB OTG FS on Micro-AB connector • STLINK-V3E
NUCLEO-H753ZI			STM32H753ZIT6U	• Ethernet • USB OTG FS on Micro-AB connector • STLINK-V3E • Cryptography

https://www.st.com/resource/en/data_brief/nucleo-h743zi.pdf

STM32 H7 43 ZI T6

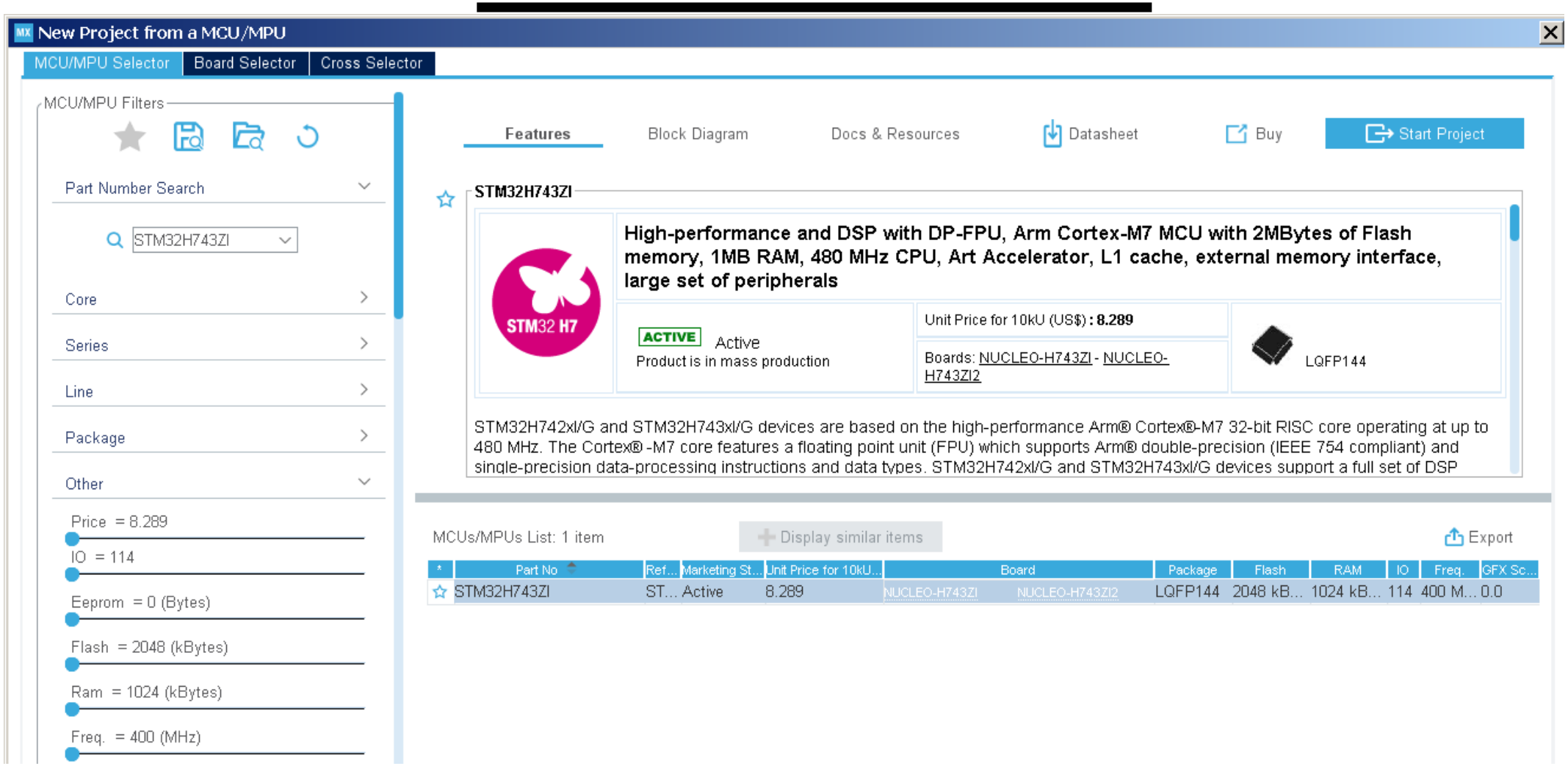
Board MB1137

Figure 19. Ethernet PHY with RJ45 connector



PIONIERKRAFT

STM32 H7 43 ZI



PIONIERKRAFT

Ethernet – main bibliography

RM0433
Reference manual

STM32H742, STM32H743/753 and STM32H750 Value line advanced Arm®-based 32-bit MCUs

https://www.st.com/content/ccc/resource/technical/document/reference_manual/group0/c9/a3/76/fa/55/46/45/fa/DM00314099/files/DM00314099.pdf/jcr:content/translations/en.DM00314099.pdf

Chapter 57: **Ethernet (ETH): media access control (MAC) with DMA controller**

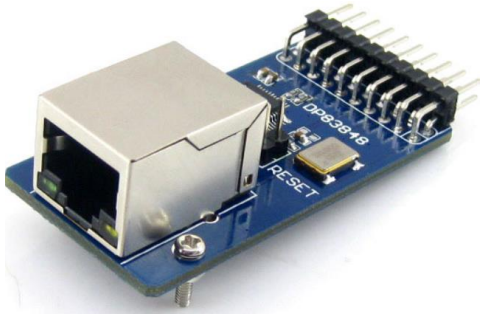
Ethernet vocabulary

ETH: Ethernet

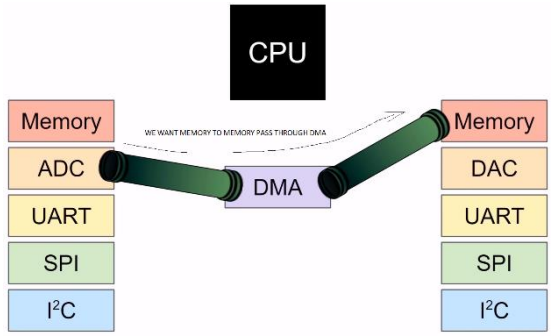
IP: volatile address

MAC: fixed address

PHY: physical device port



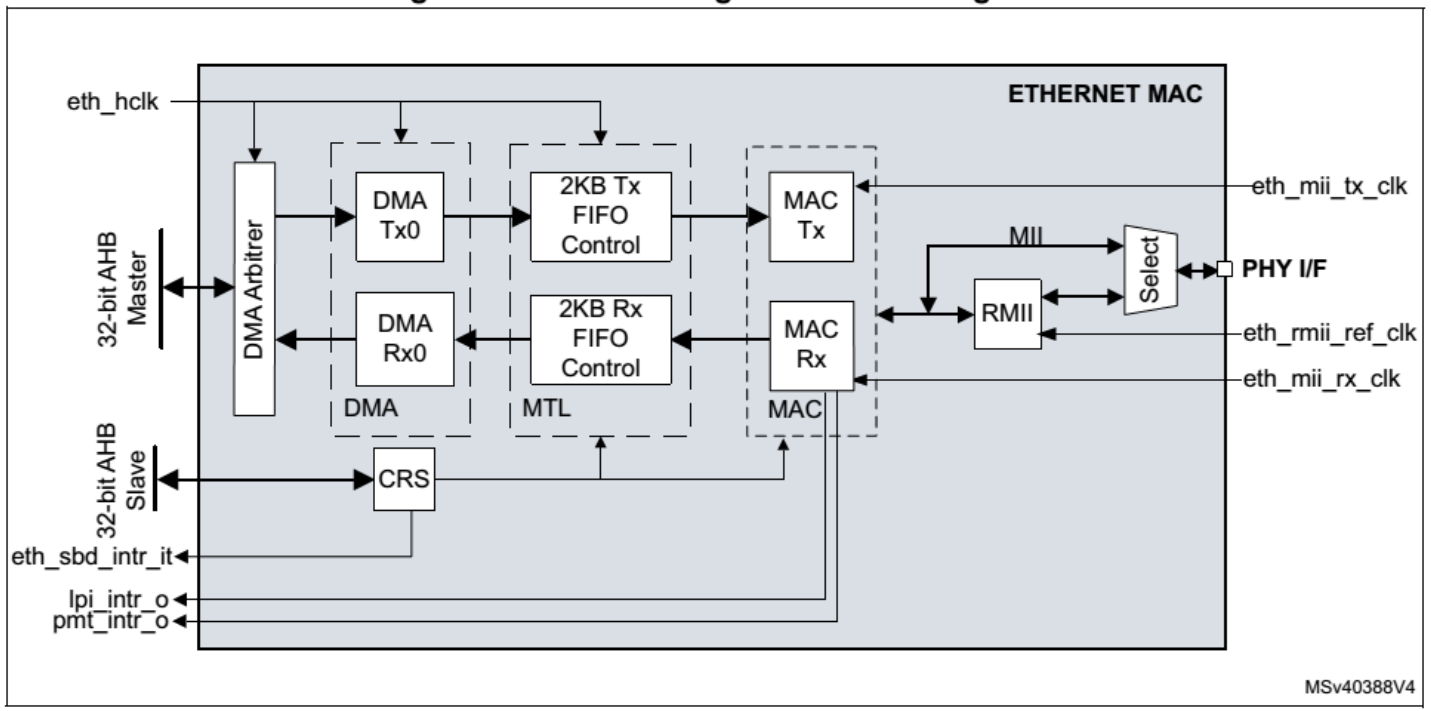
PHY

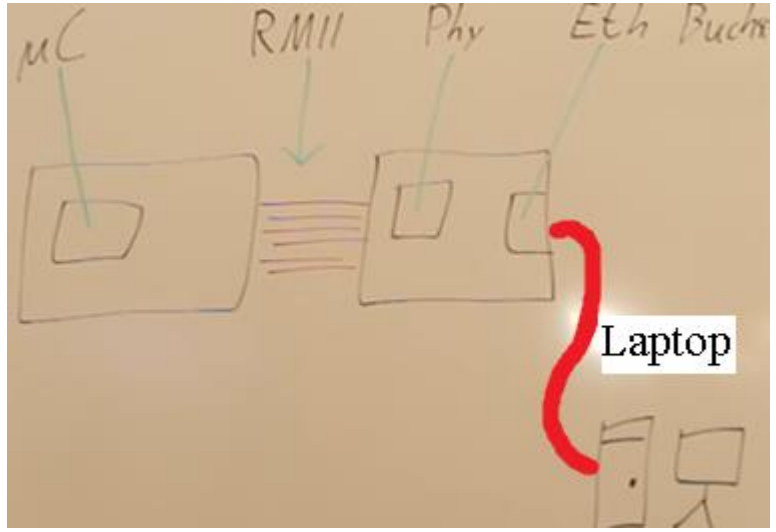


PIONIERKRAFT

Ethernet

Figure 776. Ethernet high-level block diagram





MAC - media access control modul: to implement Eth protocol

MTL - mac transaction layer: to control data flow between application and MAC

RMII: Reduced media independent interface

DMA: direct memory access interface, to communicate without bodering the CPU

Ethernet pins

57.3 Ethernet pins and internal signals

Table 506 lists the Ethernet inputs and output signals connected to package pins or balls, while *Table 507* shows the internal Ethernet signals.

Table 506. Ethernet peripheral pins

Pin name	Alternate function name (mapped on AF11)
PA0	ETH_MII_CRS
PA1	ETH_MII_RX_CLK/ETH_RMII_REF_CLK
PA2	ETH_MDIO
PA3	ETH_MII_COL
PA7	ETH_MII_RX_DV/ETH_RMII_CRS_DV
PA9	ETH_TX_ER
PB0	ETH_MII_RXD2
PB1	ETH_MII_RXD3
PB2	ETH_TX_ER
PB5	ETH_PPS_OUT
PB8	ETH_MII_TXD3
PB10	ETH_MII_RX_ER
PB11	ETH_MII_TX_EN/ETH_RMII_TX_EN
PB12	ETH_MII_TXD0/ETH_RMII_TXD0
PB13	ETH_MII_TXD1/ETH_RMII_TXD1
PC1	ETH_MDC
PC2	ETH_MII_TXD2
PC3	ETH_MII_TX_CLK
PC4	ETH_MII_RXD0/ETH_RMII_RXD0
PC5	ETH_MII_RXD1/ETH_RMII_RXD1
PE2	ETH_MII_TXD3
PG8	ETH_PPS_OUT
PG11	ETH_MII_TX_EN/ETH_RMII_TX_EN

Table 506. Ethernet peripheral pins (continued)

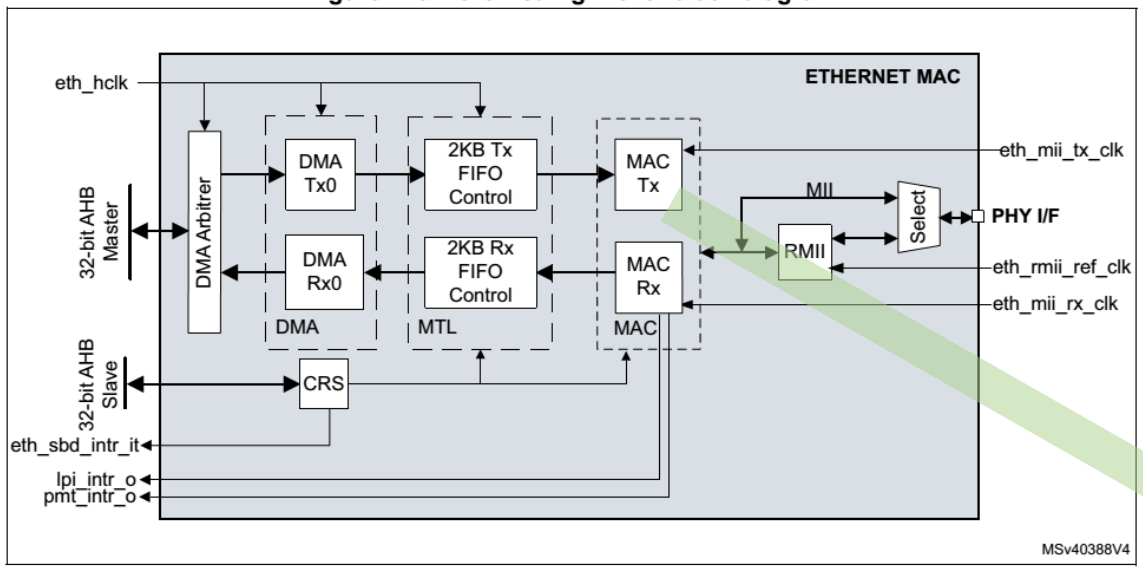
Pin name	Alternate function name (mapped on AF11)
PG12	ETH_MII_TXD1/ETH_RMII_TXD1
PG13	ETH_MII_TXD0/ETH_RMII_TXD0
PG14	ETH_MII_TXD1/ETH_RMII_TXD1
PH2	ETH_MII_CRS
PH3	ETH_MII_COL
PH6	ETH_MII_RXD2
PH7	ETH_MII_RXD3
PI10	ETH_MII_RX_ER
PI12	ETH_TX_ER

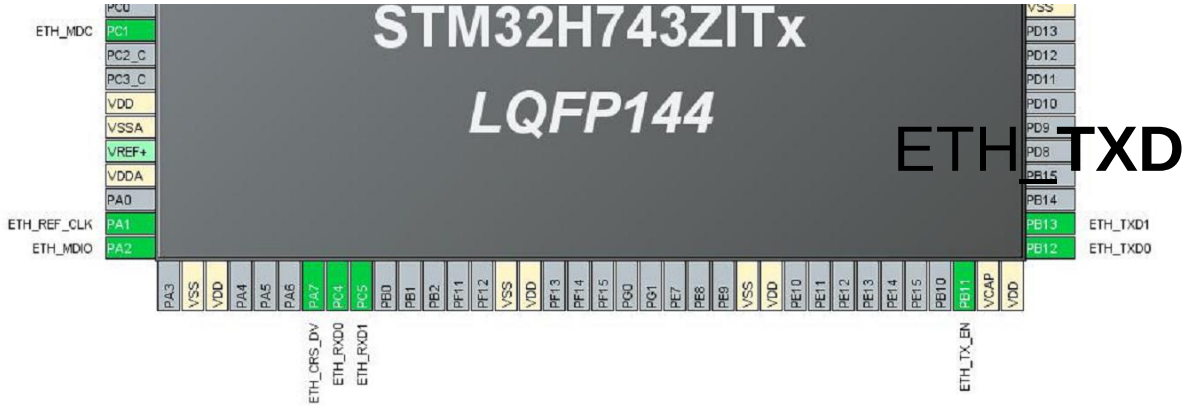
Table 507. Ethernet internal input/output signals

Signal name	Signal type	Description
eth_hclk	Digital input	AHB clock
eth_sbd_intr_it	Digital output	Main Ethernet interrupt
lpi_intr_o	Digital output	Sideband signal generated when the transmitter or receiver enters or exits the LPI state.
pmt_intr_o	Digital output	Sideband signal generated when a valid remote wakeup packet is received
eth_mii_tx_clk	Digital input	MII Tx kernel clock
eth_mii_rx_clk	Digital input	MII Rx kernel clock
eth_rmii_ref_clk	Digital input	RMII reference kernel clock

Ethernet pins

Figure 776. Ethernet high-level block diagram

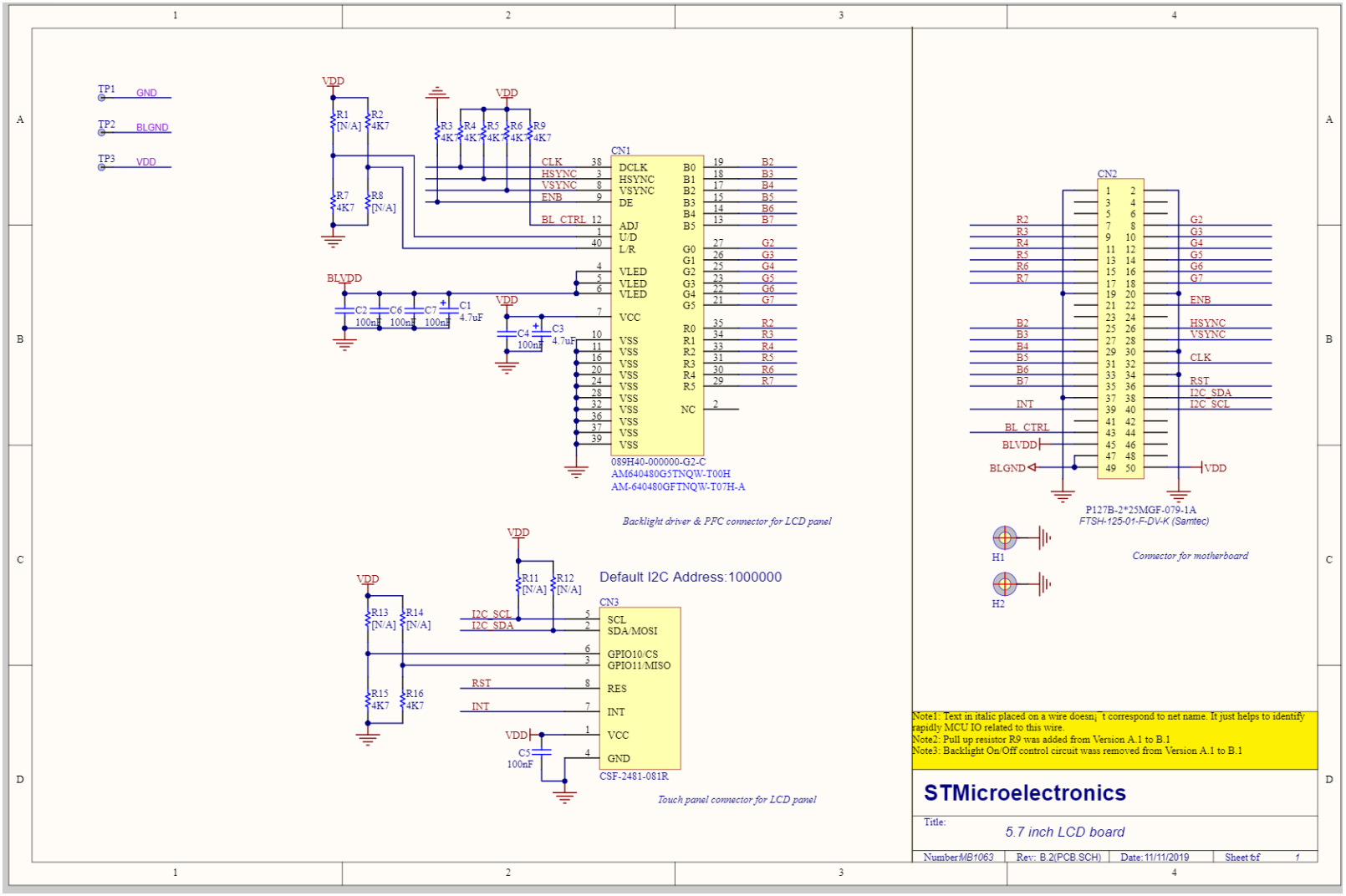




ETH_**TXD**

ETH_**RXD**

Ethernet pins



PIONIERKRAFT

DMA

Figure 777. DMA transmission flow (standard mode)

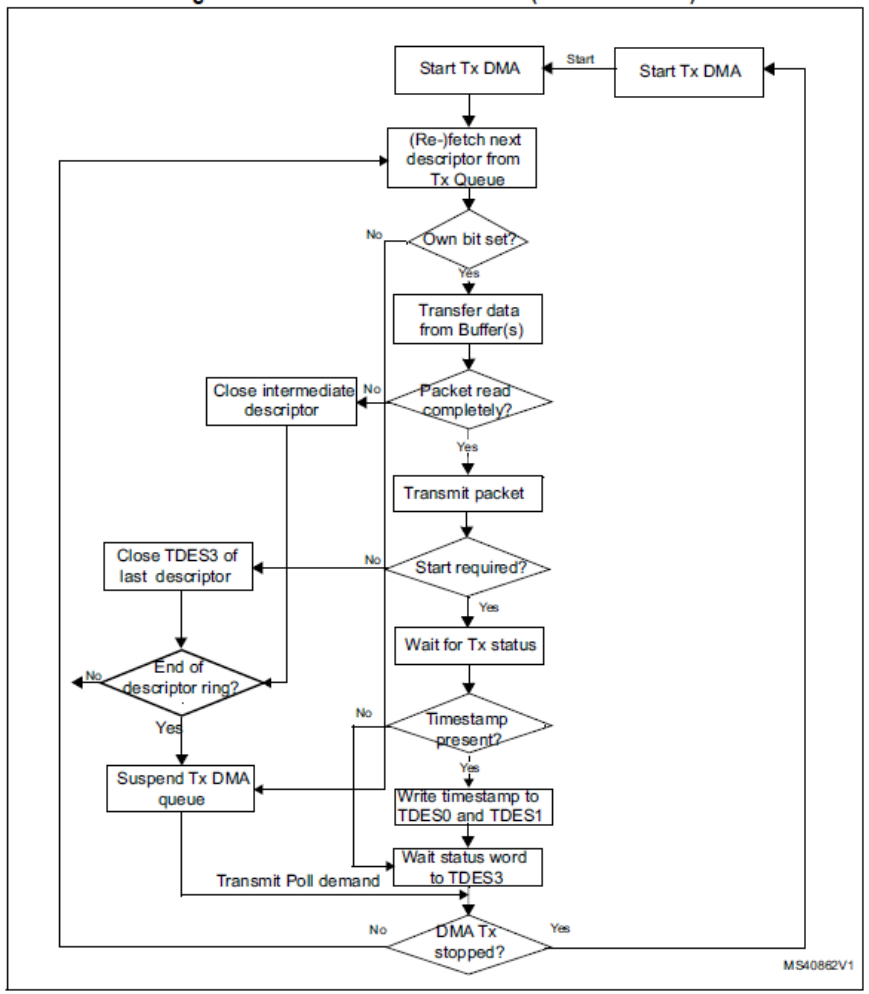
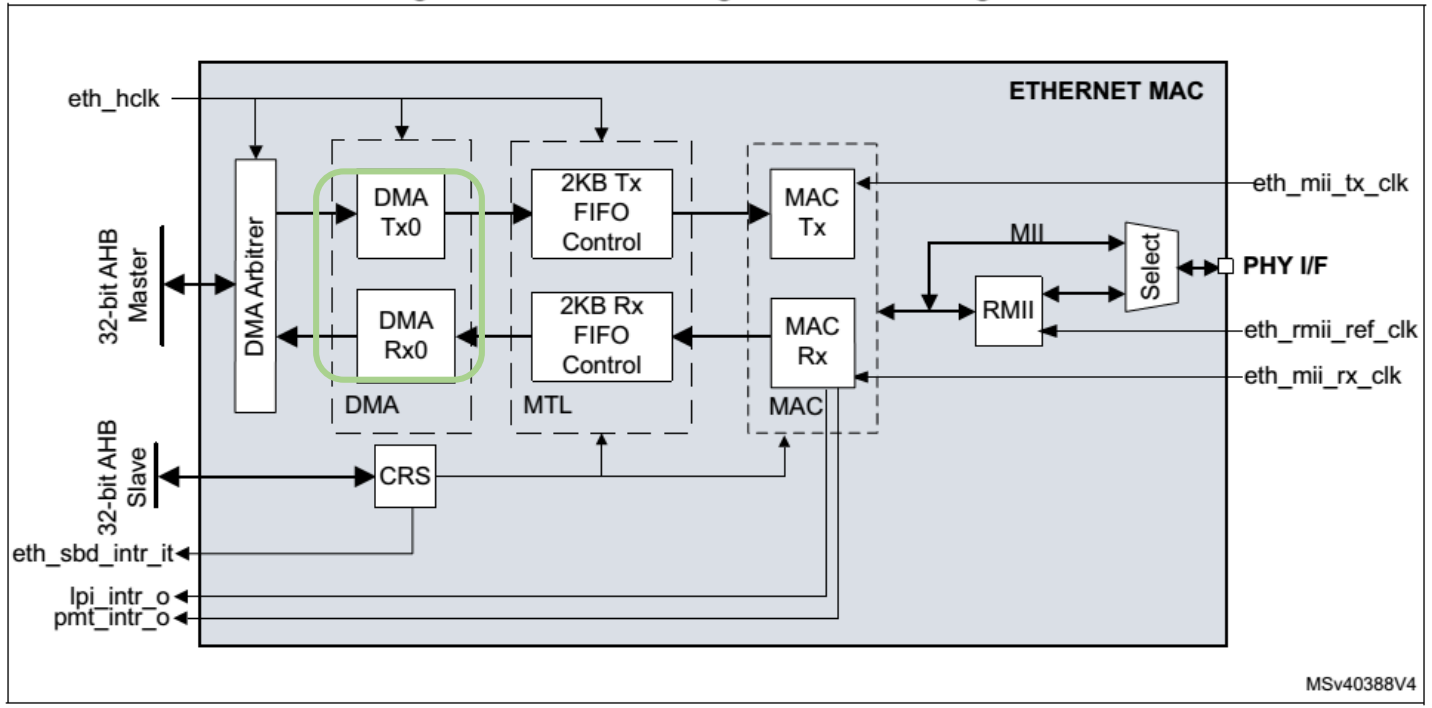
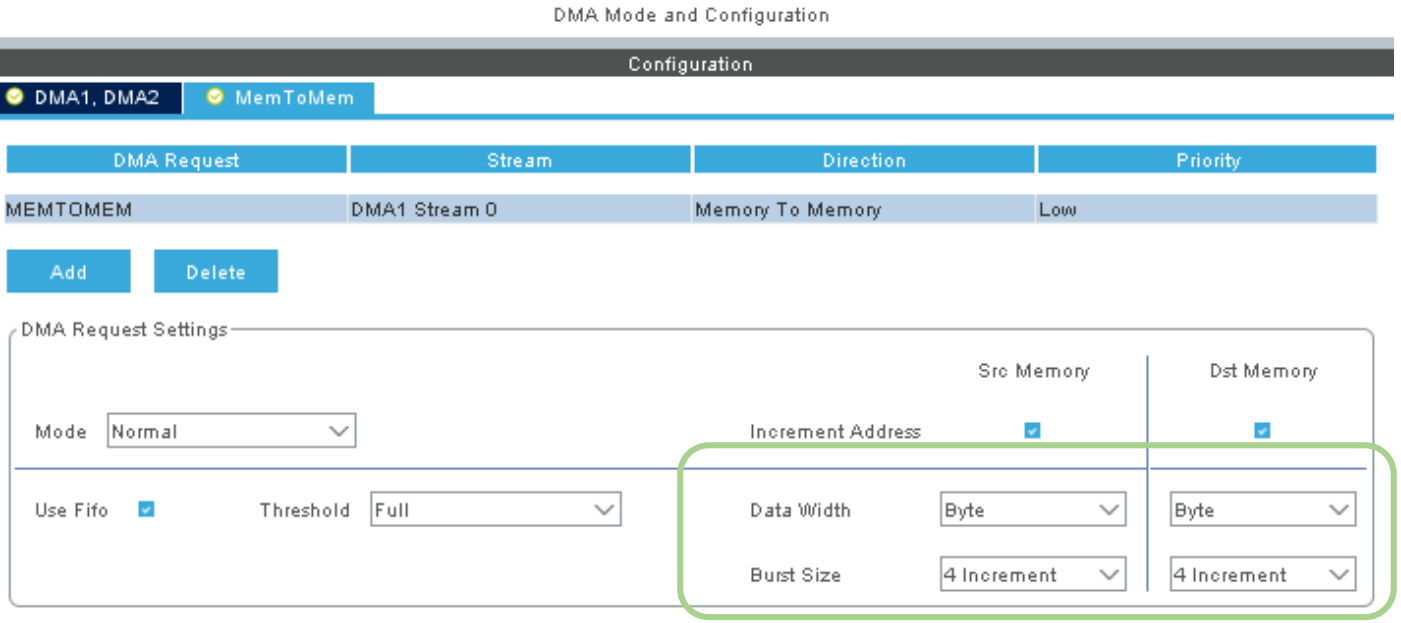


Figure 776. Ethernet high-level block diagram



DMA



FIFO: Threshold & Burst mode

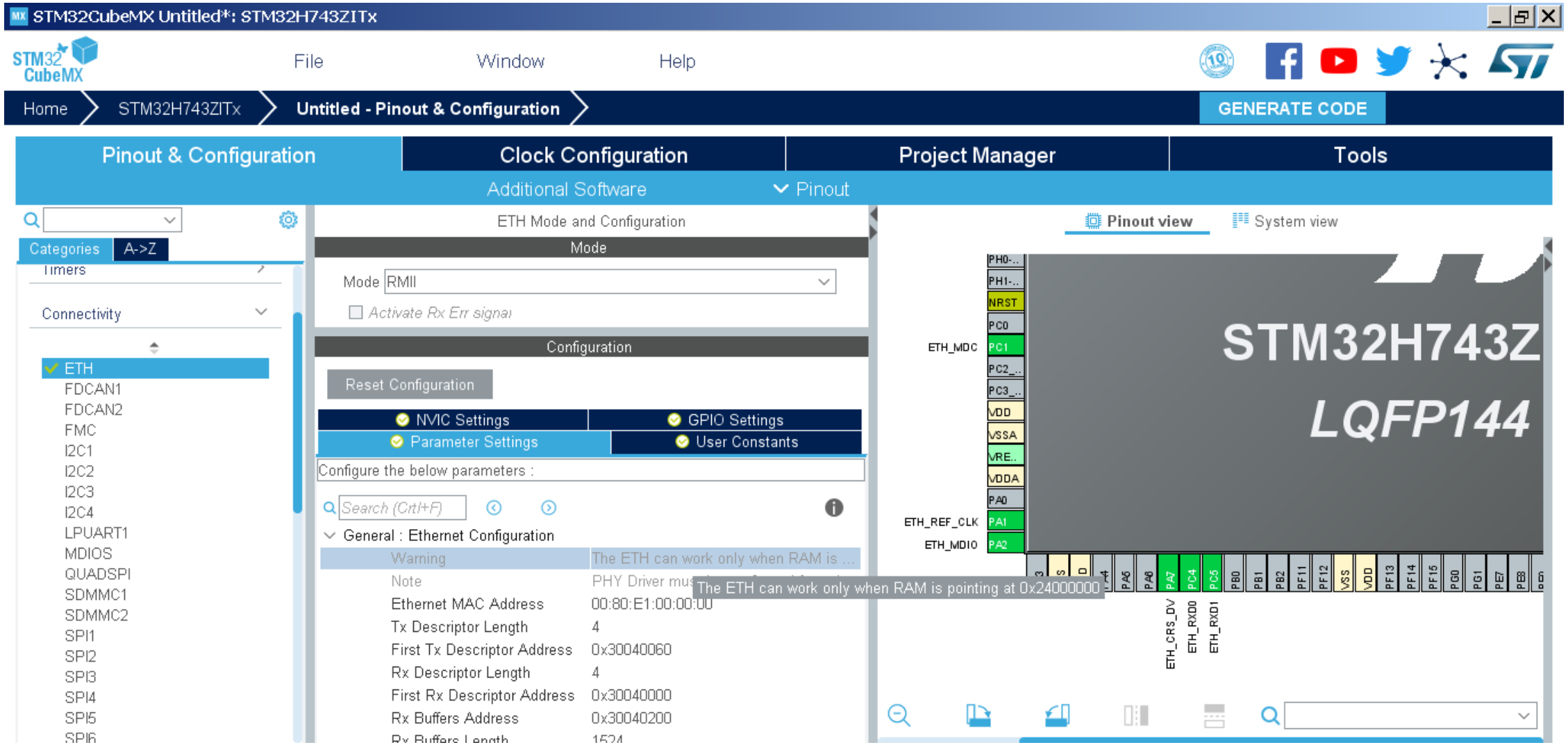
- Threshold level triggers the data transfers to/from Memory.
- Each stream has independent configurable threshold levels:
  - Four threshold levels available: ¼ FIFO Full ,1/2 FIFO Full, ¾ FIFO Full, and FIFO Full
- Burst mode:
  - Burst mode is available only when FIFO mode is enabled (Direct mode disabled)
  - Available Burst modes:
    - INC4: 1 burst = 4 beats (4 Words, 8 Half-Words or 16 Bytes)
    - INC8: 1 burst = 8 beats (8 Half-Words or 16 Bytes)
    - INC16: 1 burst = 16 beats (16 Bytes)
- FIFO threshold should be compatible with Burst size

Memory data size	Burst Size	Allowed threshold levels
Byte	4 beats (INC4)	¼, ½, ¾ and Full
	8 beats (INC8)	½ & Full
	16 beats (INC16)	Full
Half-Word	4 beats (INC4)	½ & Full
	8 beats (INC8)	Full
Word	4 beats (INC4)	Full

25 DWORDS = 25*4bytes

(imposed specification)

Ethernet + RMII

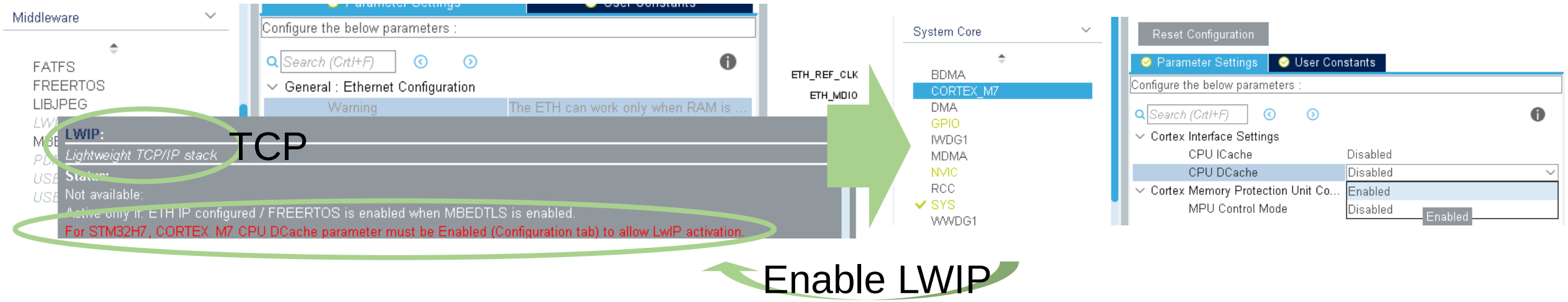


PIONIERKRAFT

TCP vs UDP

- UDP: no guarantee you are getting all the packets.
- TCP: guarantees the recipient will receive the packets in order by numbering them.

TCP is safer → LWIP

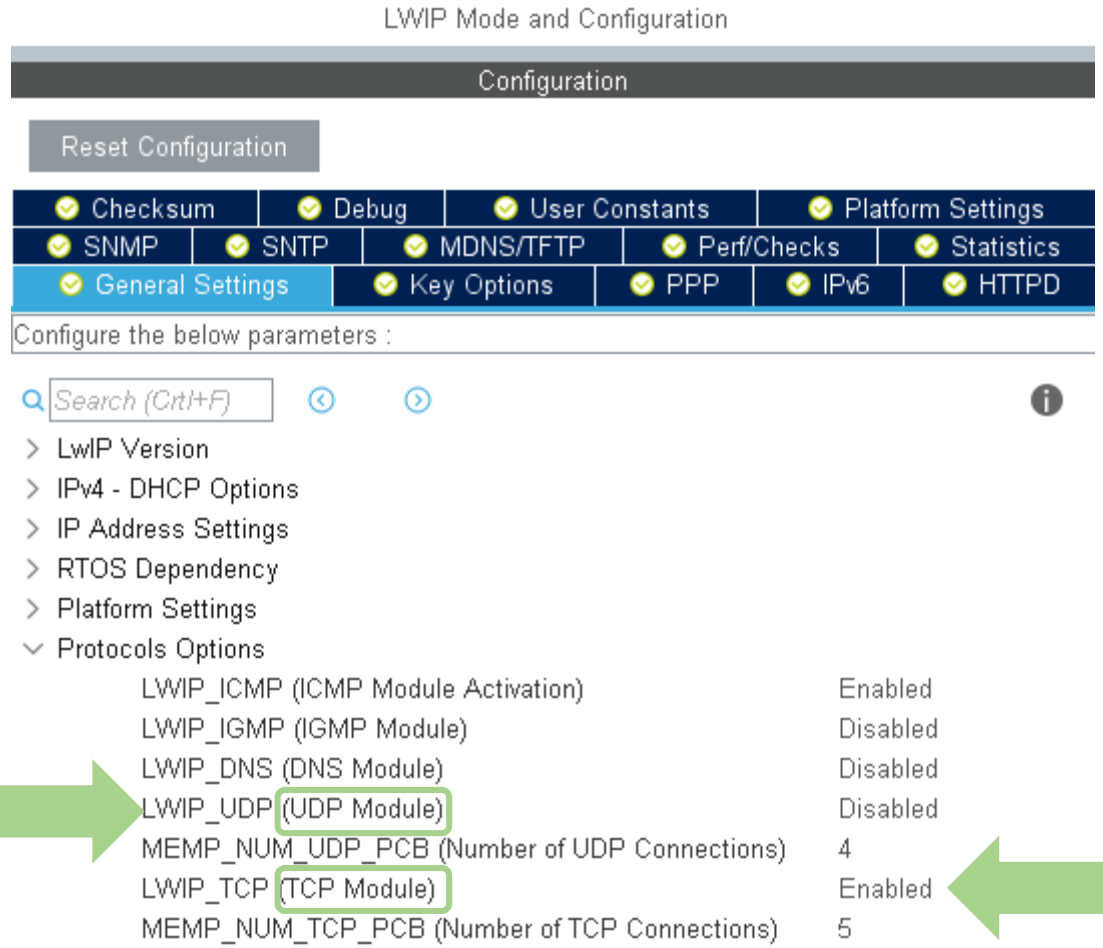


Enable LWIP

PIONIERKRAFT

TCP vs UDP

In LwIP General Settings…

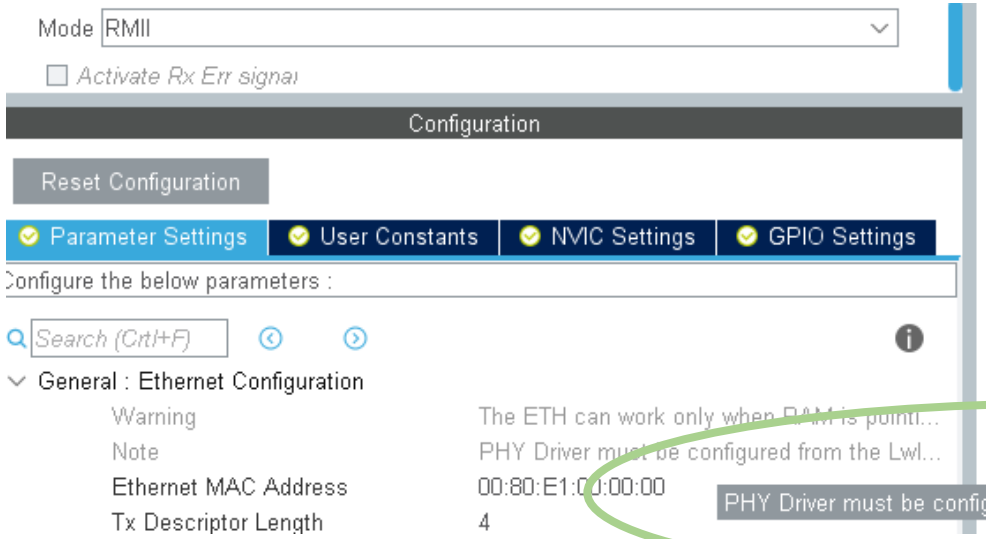


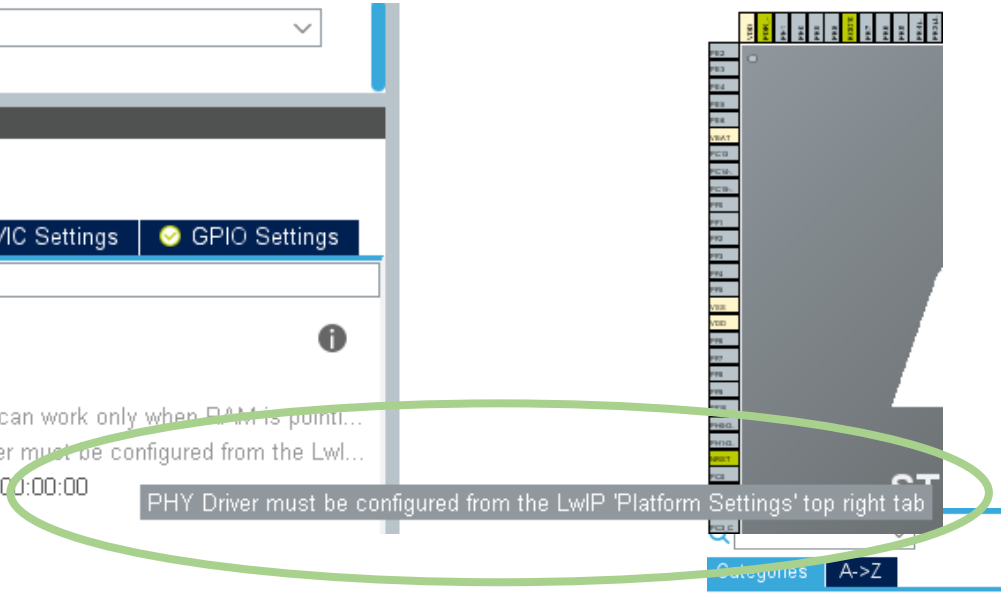
Disable: UDP

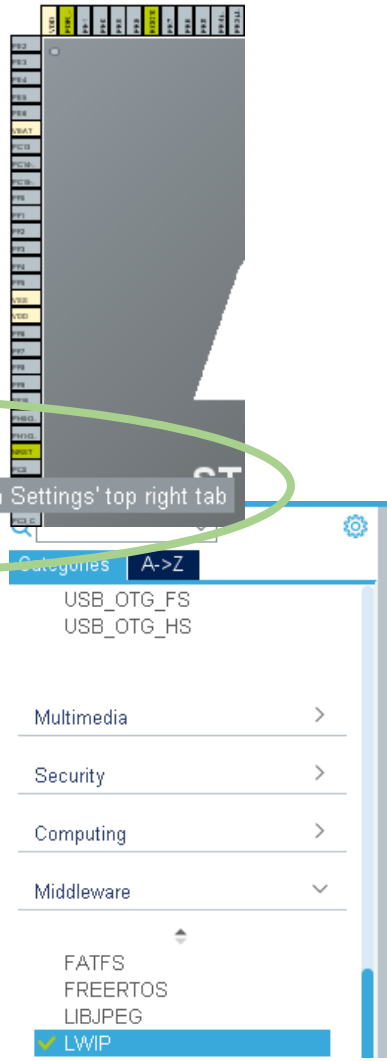
Enable: TCP

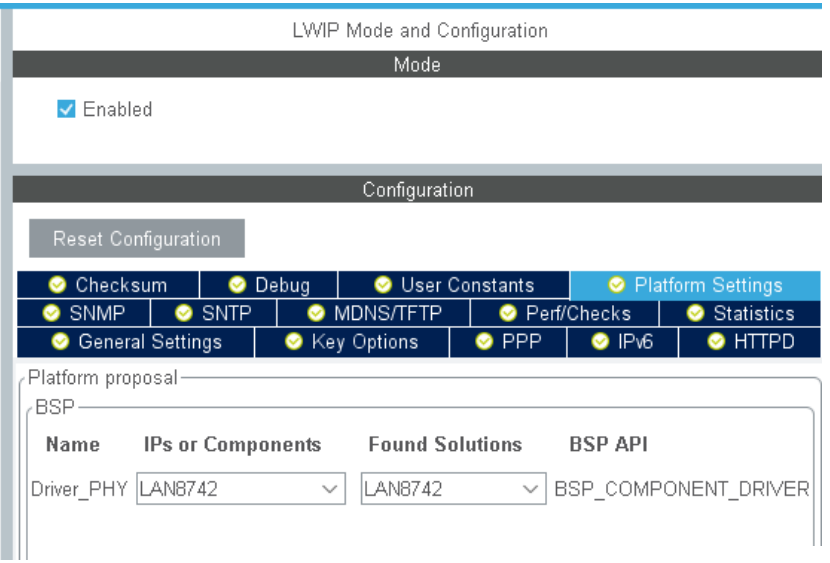
PIONIERKRAFT

PHY configure

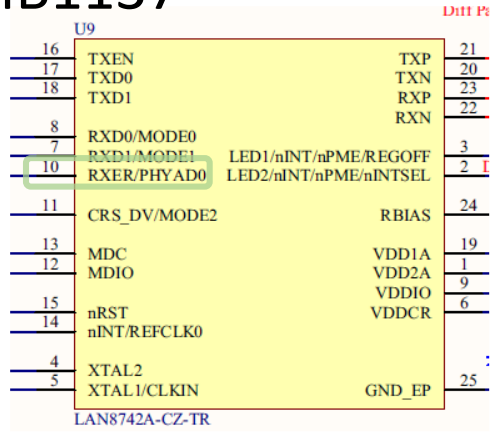








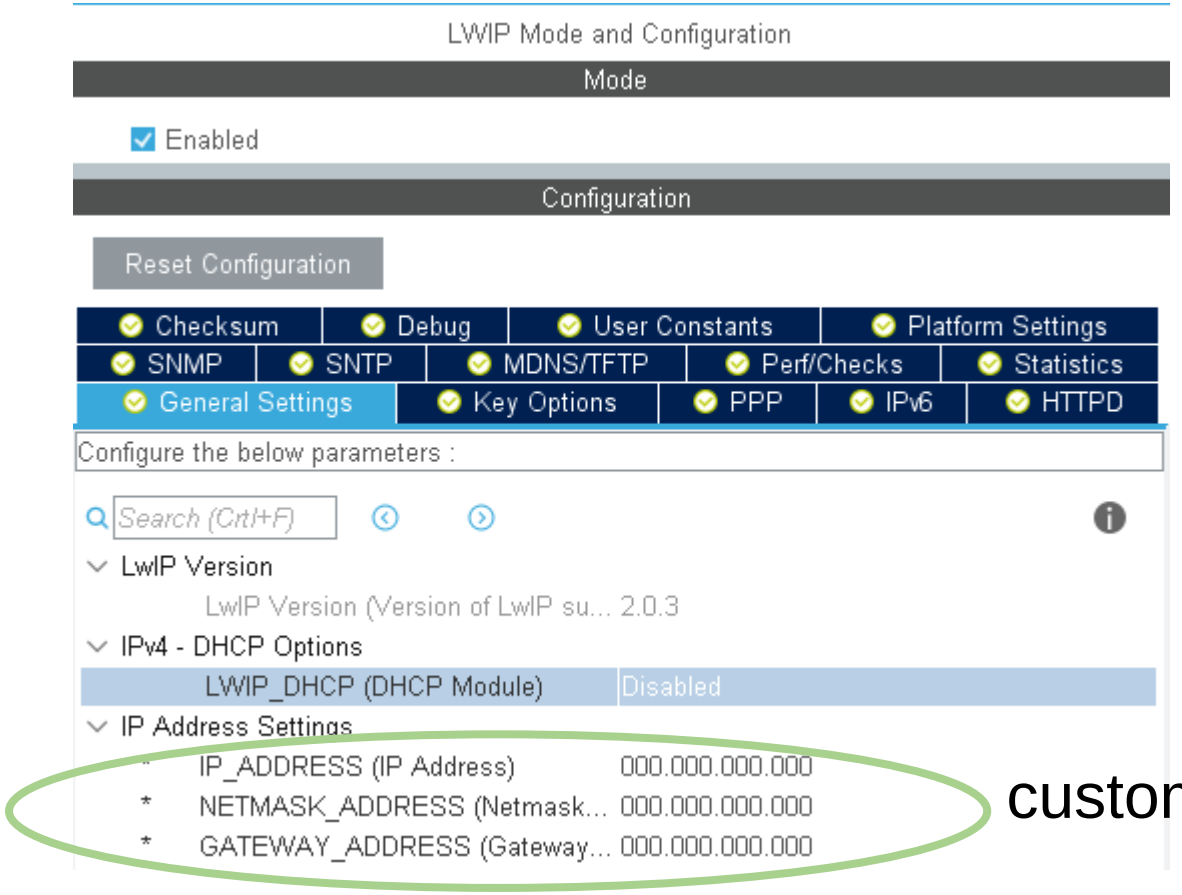
Board MB1137



PIONIERKRAFT

IP

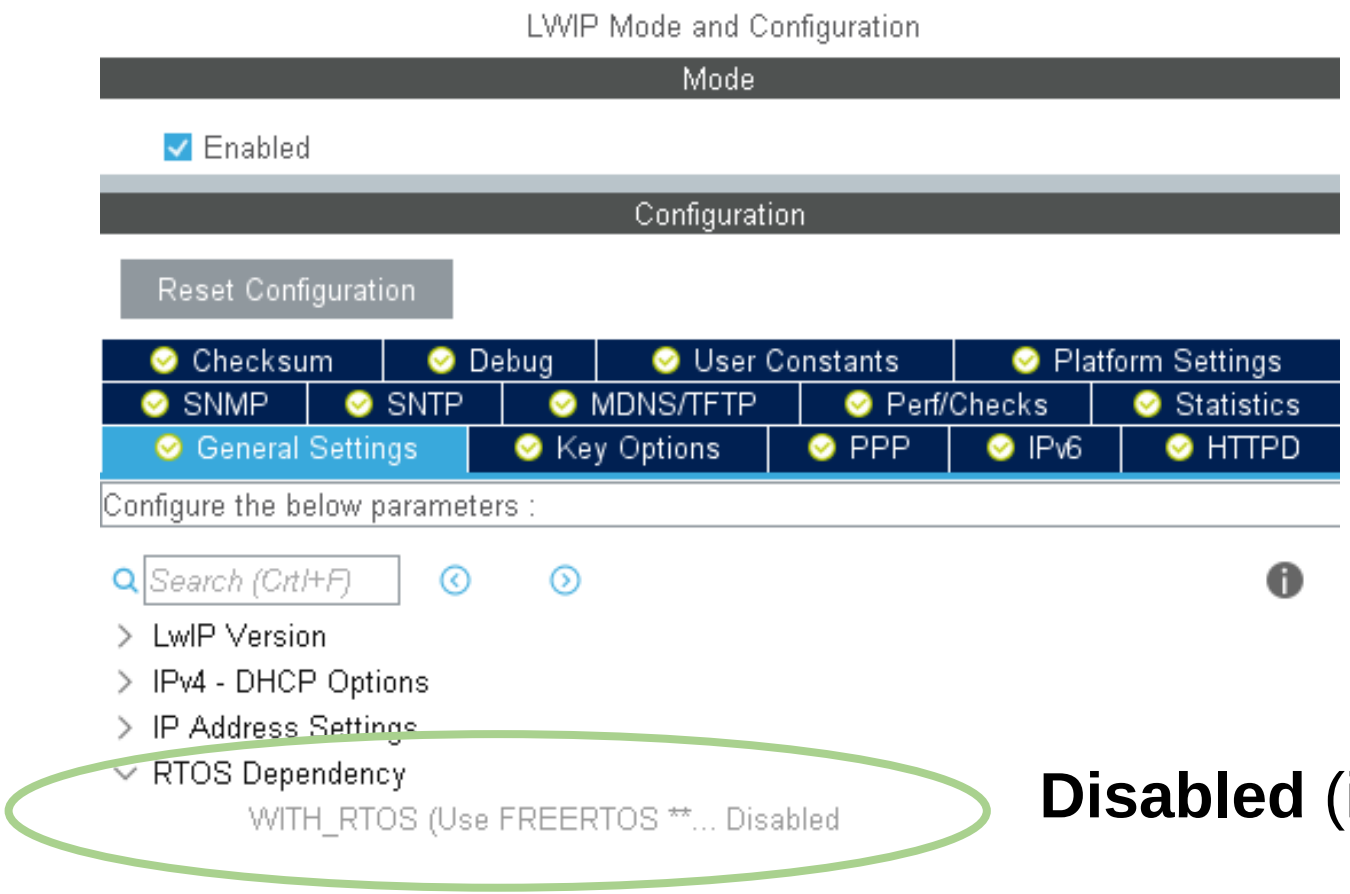
DCHP disabled → insert IP address



customize !!

PIONIERKRAFT

Real-time optimization (RTO)



Disabled (imposed specification)

PIONIERKRAFT

C code generator → STM32CubeIDE

Follow STM32CubeMX “User Manual” (tutorial 1):

Tutorial 1: From pinout to project C code generation using an MCU of the STM32F4 Series

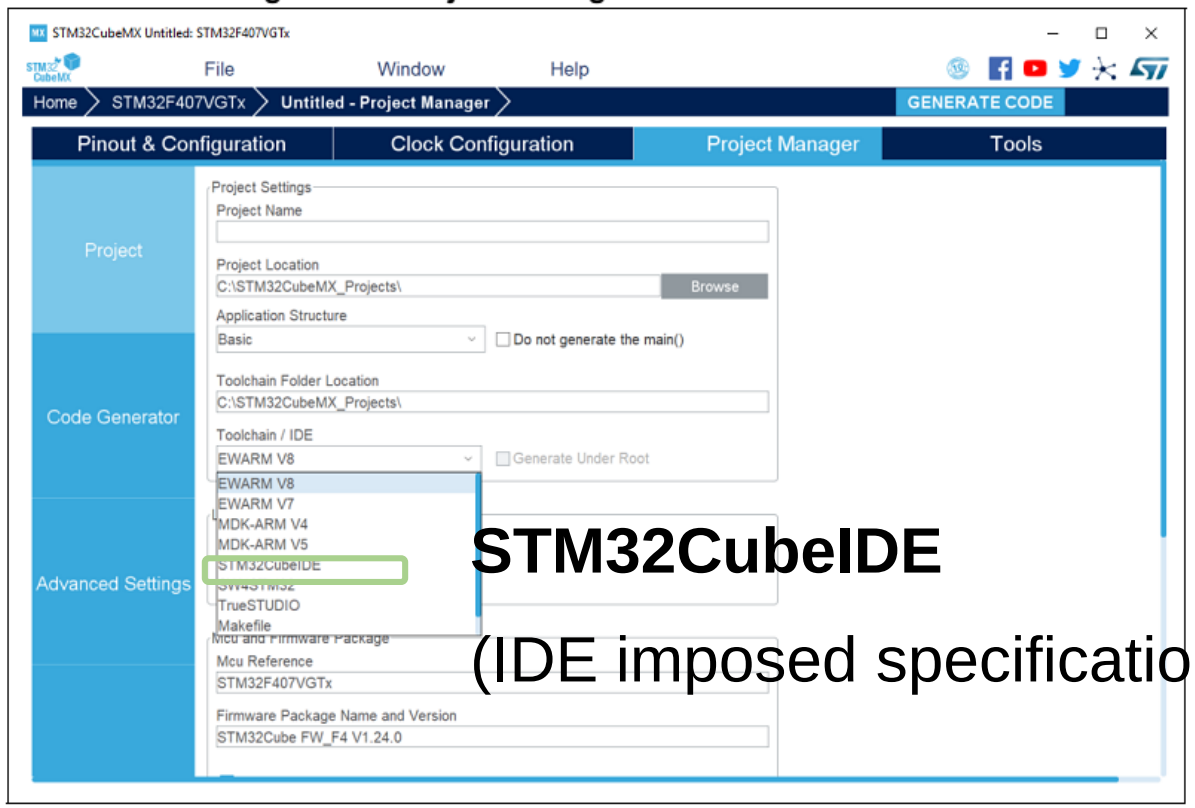
10.7 Generating a complete C project

10.7.1 Setting project options

Default project settings can be adjusted prior to C code generation as shown in *Figure 216*.

1. Select the **Project Manager** view to update project settings and generation options.
2. Select the **Project Tab** and choose a Project **name**, **location** and a **toolchain** to generate the project (see *Figure 216*).
3. Select the **Code Generator** tab to choose various C code generation options:
   - The library files copied to *Projects* folder.
   - C code regeneration (e.g. what is kept or backed up during C code regeneration).
   - HAL specific action (e.g. set all free pins as analog I/Os to reduce MCU power consumption).

Figure 216. Project Settings and toolchain selection

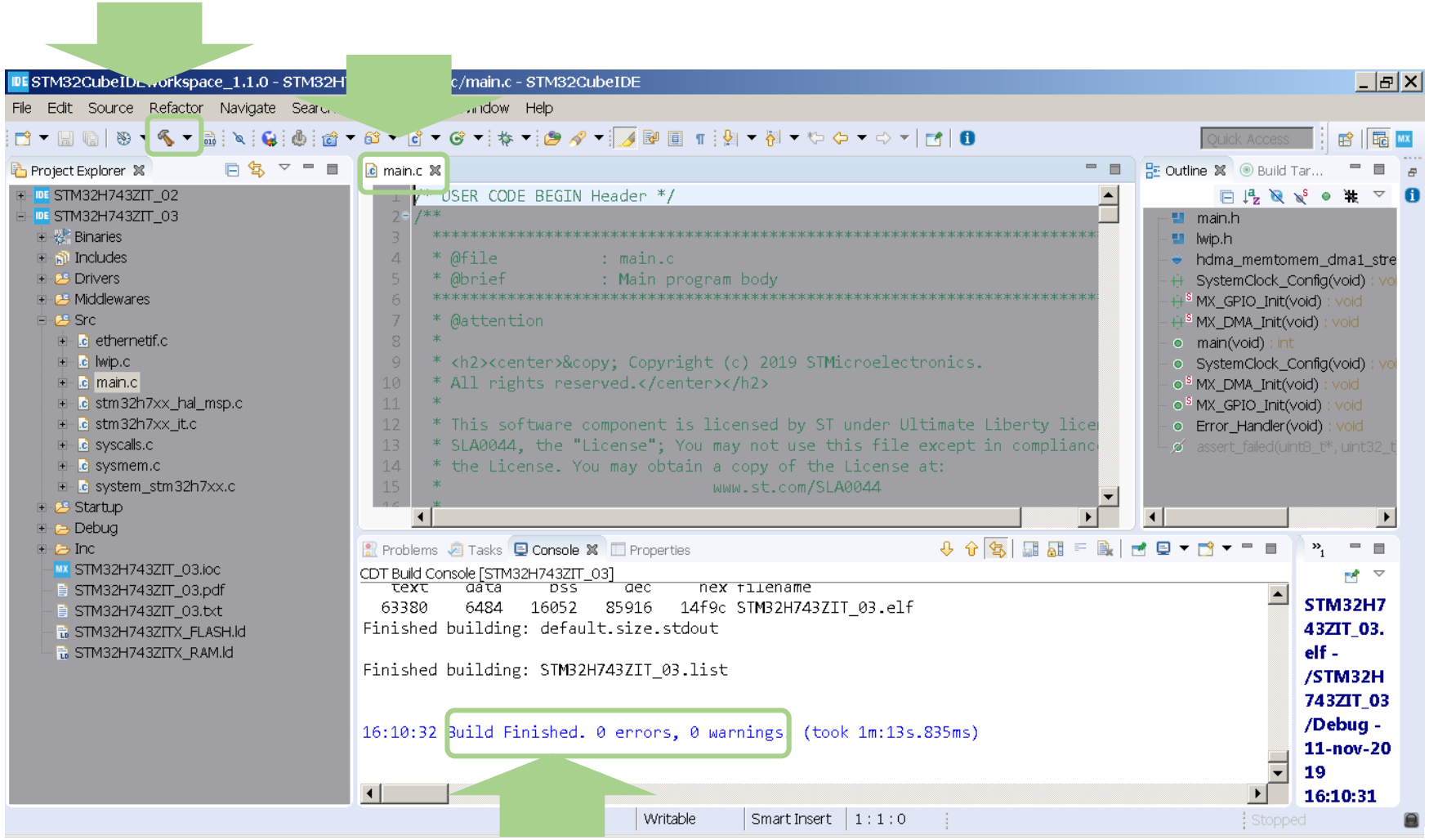


STM32CubeIDE

(IDE imposed specification)

PIONIERKRAFT

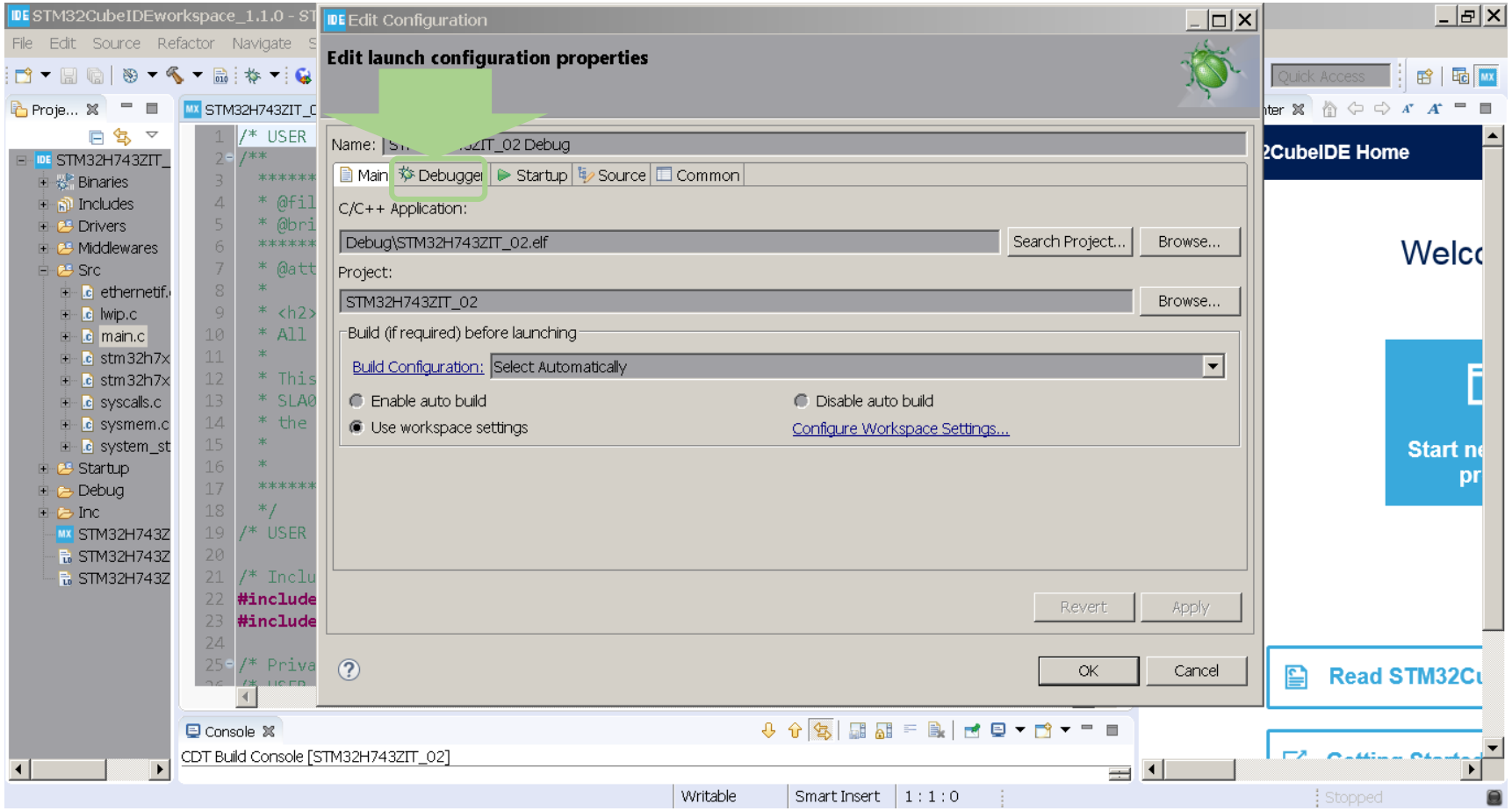
code build

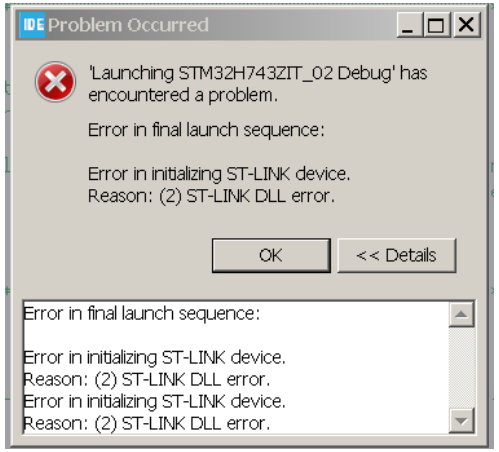


PIONIERKRAFT

code debug

Run/Debug





Es fehlt noch

- Allgemeines:
  - Zu verwendende Entwicklungsumgebung: STM32CubeIDE
  - Implementierung in neuem STM32-Projekt
  - Implementierung **ohne** Operation System (RTOS, o.ä.), falls möglich
- Aufgabenstellung:
  - Implementierung von Ethernet auf einem STM32H743ZIT6 Microcontroller
    - Entweder TCP oder UDP. Please implement the safer one.
  - Senden von Daten (25 Werte (float 32 bit)) zu einer Adresse
    - Werte vor dem Senden in JSON-Format umwandeln (besser für Backend, Web)
  - Empfangen von Daten (25 Werte (float 32 bit)) von einer Adresse
    - Werte wieder in float zurück wandeln
- Zusätzliche Fragestellungen:
  - Unterschied TCP und UDP erklären
  - Aufbau JSON-Nachricht
  - Wie können die im JSON-Format gesendeten Daten in einem Backend abgelegt werden (Schrittfolge aufzeigen, kein Code)?
- Präsentation (in Powerpoint):
  - Vorstellung der Aufgabenstellung in STM32CubeIDE
  - Vorstellung der zusätzlichen Fragestellungen