

NeptuneSDR

Zynq7020 AD936X Development Platform (K230) User
Manual

Catalogue

Copyright Notice.....	3
disclaimer.....	4
China Low-Power Short-Range Equipment Declaration.....	5
I. Development Board Overview.....	7
II. ZYNQ Chip.....	8
III. DDR Chip.....	9
4. QSPIFlash.....	10
V. PS System Clock Source.....	10
VI. Structural Diagram.....	11
7. ESP32S3.....	11
8. Gigabit Ethernet Interface.....	12
IX. UART Interface.....	12
10. SD Card Slot.....	13
11. DEBUG Connector.....	13
12. GPS Module.....	14
13. AD936X.....	14

Copyright Notice

Nepu Technology holds the copyright to this document. Without the prior written authorization of Nepu Technology, no company, organization, website, media entity, or individual may modify, distribute, copy, or excerpt any part of this document in any manner or form. Such content includes, but is not limited to, text, images, charts, tables, and other materials. The intellectual property rights to this document belong to Nepu Technology; no person or organization may use this document for commercial purposes through any channel. Nepu Technology reserves the right to take legal action against any party that infringes upon these rights.

We place great importance on intellectual property rights associated with this document – a reflection of our respect for intellectual property laws and for the creators behind it. We hope that all users of this document will respect intellectual property rights and comply with the above statement to avoid any unnecessary legal disputes or risks arising from any infringement. At the same time, we encourage everyone to protect their own intellectual property rights and create more valuable works.

Through our collective efforts, we aim to raise public awareness and appreciation of intellectual property rights, work together to safeguard their legitimate interests, and ensure that intellectual property receives better protection and further development. Thank you for your cooperation and support!

Disclaimer

This product is intended solely for research and educational purposes; the purchaser shall use this product within the scope of legal and regulatory requirements. Any disputes arising from the purchaser's use of this product or any consequences resulting therefrom shall be the sole responsibility of the purchaser and shall not involve Nape Technology. This statement is hereby issued.

Nepu Technology does not provide any form of warranty – whether express or implied, including but not limited to warranties regarding suitability for specific purposes – with respect to the code, software, documentation, or other materials provided herein; users shall bear all risks associated with their use. We fully recognize the inherent uncertainties associated with technology as well as the potential challenges users may encounter during deployment. Therefore, we continuously provide users with enhanced technical support to assist them in overcoming technical difficulties, and leverage our professional expertise to deliver a superior user experience.

Any unauthorized modification or alteration of this product without the prior authorization of Nep Technology or certification by the relevant authorities will render the wireless compliance status invalid and result in the loss of your access to operate this product. We recommend that users thoroughly review the applicable laws and regulations before using this product to avoid any unnecessary complications.

We encourage our users to use this product in a lawful and compliant manner, helping us together to build a healthy technological ecosystem. If you encounter any issues during your use of the product or have any suggestions, please feel free to contact us – we are committed to serving you with the utmost dedication.

China Low-Power Short-Range Equipment Declaration

This product possesses the micro-power, short-range radio transmission capability as stipulated in the Regulations on the Administration of Radio Communications of the People's Republic of China. In accordance with the requirements set forth in Announcement No.52 of 2019 issued by the Ministry of Industry and Information Technology, this is hereby specified as follows:

(1) The product complies with the technical requirements for Class F general-purpose low-power equipment as specified in the "Catalogue of Low-Power Short-Range Radio Transmitter Equipment and Technical Requirements"; it is intended for research and educational purposes only. The product does not come equipped with an antenna or a control switch; please refer to the user manual for further information.

(2) It is prohibited to alter the usage scenarios or conditions without authorization, to expand the transmission frequency range, or to increase the transmission power (including the additional installation of a RF power amplifier); it is also prohibited to modify the transmission antenna without authorization;

(3) No harmful interference shall be caused to other lawful radio stations; nor shall any request for protection against harmful interference be made;

(4) Interference from industrial, scientific and medical (ISM) application equipment that is subject to radio frequency energy exposure, or from other lawful radio station interference;

(5) If harmful interference is detected with any other lawful radio station, the use thereof shall be discontinued immediately; the station may only resume operation after appropriate measures have been taken to eliminate such interference;

(6) When using low-power equipment within an aircraft or within the electromagnetic environmental protection zones of military or civilian radio stations (including radio observatories, meteorological radar stations, satellite earth stations – including tracking and control, ranging, reception, and navigation stations) or airports – as designated in accordance with laws, regulations, relevant national provisions, or standards – such use shall comply with the provisions regarding electromagnetic environmental protection and those issued by the relevant industry authorities;

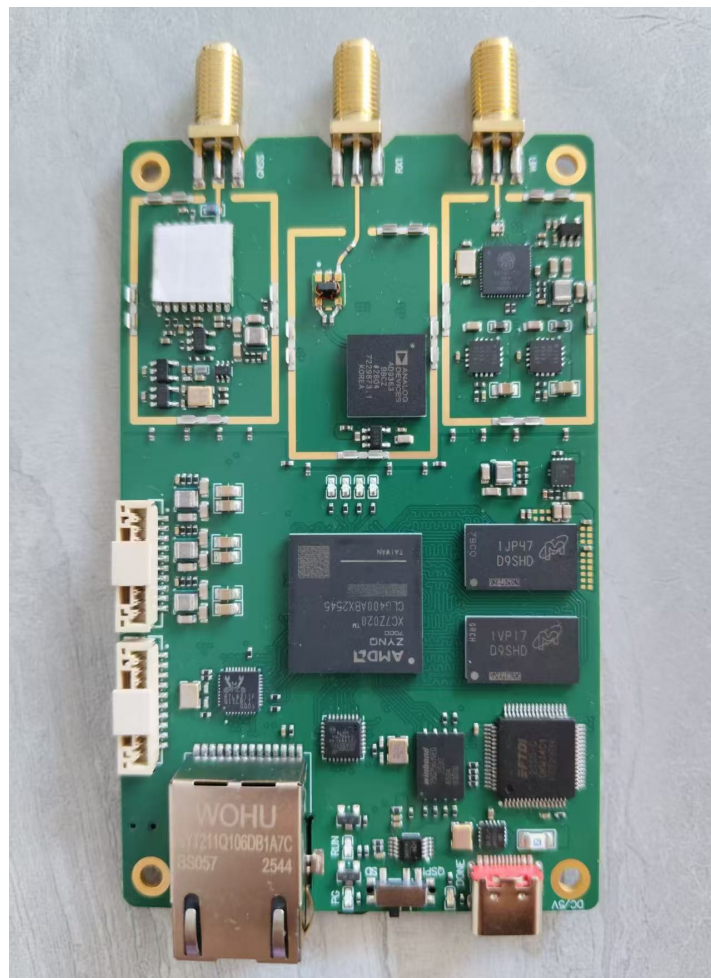
(7) The use of any type of model remote control within a radius of 5,000 meters centered on the central point of an airport runway is prohibited;

(8) The operating voltage of the low-power equipment is 5 V; it operates normally at room temperature.

Nep Technology Co., Ltd. has officially launched the 2024 version of its development board (Model: K230), which is based on the Xilinx Zynq7020+AD936X development platform. To help you quickly get up to speed with this development platform, we have prepared this user manual.

This SDR development platform features an integrated design approach, facilitating easy secondary development and utilization of the development board. The board employs a solution based on the Xilinx Zynq 7020+ AD936X chip, which utilizes Processing System (PS) + Programmable Logic (PL) technology to integrate a quad-core ARM Cortex-A9 processor and FPGA programmable logic onto a single chip. Additionally, the PS side of the core board includes two high-speed DDR3 SDRAM chips (totaling 1024 MB) and one 256 MB QSPI FLASH chip; the PL side of the board is responsible for communicating with the AD936X chip, with both LVDS and CMOS communication modes available.

In terms of development board design, we have provided users with a comprehensive range of peripheral interfaces—including one GH1.25 interface, one Gigabit Ethernet interface, one UART interface, one SD card interface, and more—enabling them to meet diverse requirements for high-speed data exchange, data storage, video transmission and processing, deep learning, artificial intelligence, and industrial control. This is a "professional-grade" ZYNQ SDR development platform that facilitates preliminary verification and subsequent application of high-speed data transmission, exchange, and data processing tasks. We believe this product is ideally suited for students, engineers, and professionals engaged in communication algorithm development.



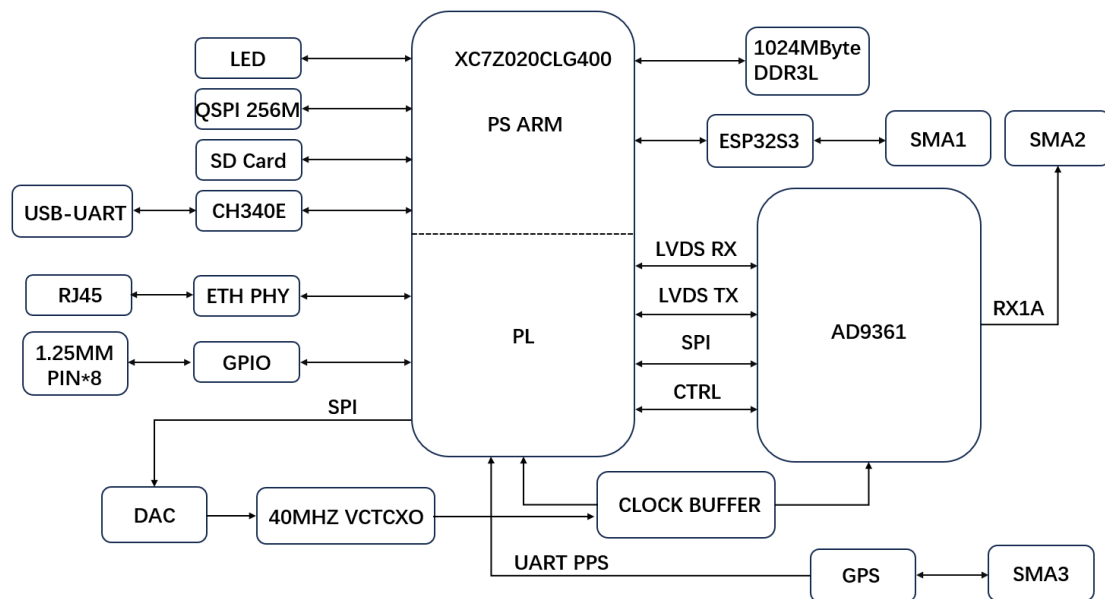
I. Development Board Overview

Here is a brief functional overview of the K230 SDR Development Platform.

The development board is primarily composed of a minimal system featuring an XC7Z020 chip, two DDR3 memory modules, an AD936X chip, and a QSPI FLASH memory device. The main microcontroller is an Xilinx Zynq 7000 series chip, model XC7Z020-1CLG400I. The Zynq 7020 chip is divided into two sections: the Processor System (PS) and the Programmable Logic (PL). Two DDR3 memory modules, each with a capacity of up to 512 MB, are connected to the PS section of the Zynq 7020 chip, enabling the ARM and FPGA systems to independently process and store data. A 256 MB QSPI FLASH memory device connected to the PS section is used for the static storage of the Linux operating system, file system, and user data.

The development board features a comprehensive range of peripheral interfaces, including one Gigabit Ethernet interface, one UART interface, one SD card interface, and several LEDs.

The figure below shows a structural diagram of the entire development system:



Through this diagram, we can see the interfaces and features available on our development platform.

Zynq7020 Minimal System

The system consists of a Zynq7020 processor, a 1024 MB DDR3 memory module (PS), and a 256 MB QSPI flash memory; an additional crystal oscillator provides the clock signal, with a single-ended 33.3333 MHz crystal oscillator supplied to the PS system.

Gigabit Ethernet interface

1 10/100M/1000M Ethernet RJ45 interface, connected to the PS end; used for Ethernet data transmission with computers or other network devices.

USB UART interface

The USB interface is of TYPE-C type; this USB UART interface provides a single UART channel for communication with the computer, facilitating user debugging. The interface chip is the CH340E chip.

Micro SD card slot

1 Micro SD card slot for storing the operating system image and file system.

LEDlamp

3 One LED light, one custom LED light connected to the PS port, one DONE configuration indicator light, and one power indicator light.

DIP switch

1 A 1-bit DIP switch is used for mode switching, supporting three boot modes: QSPI, SD card, and others.

RF SMA Interface

3 One low-loss SMA interface (internal pinhole); one for receiving radio signals; one for the GPS module; one for the ESP32 chip.

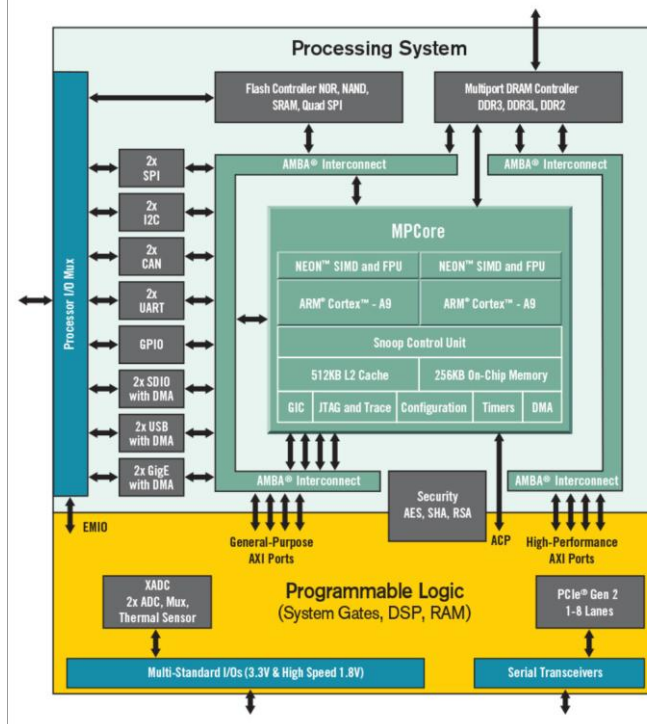
DEBUG Connector:

1 1 bus: includes 1 JTAG bus (3.3 V logic level) + 1 UART bus (3.3 V logic level)

II. ZYNQ Chip

The development board utilizes a Xilinx Zynq7000 series chip, model XC7Z020-1CLG400I. The PS system on the Zynq7000 chip integrates two ARM Cortex-A9 processors operating at up to 1 GHz.

The Zynq7020 chip supports 32-bit DDR3, DDR3L, and LPDDR3 memory chips; the PS side features a comprehensive set of interfaces, including SPI, UART, ETH, USB, QSPI, SD, and others. The PL side incorporates a rich array of programmable logic units, DSPs, and internal RAM. The overall block diagram of the Zynq7020 chip is shown in the figure below:



Logic Cells 85K

Look-Up Tables (LUTs) 53,200

CLB Flip-Flops 106,400

DSP Slices 220

Block RAM 4.9Mb

94 IOs, 8 MIOs

47 Pairs LVDS

100 Voltage Adjustable IOs

AD Converter x 2, 1Mbps

Speed Grade -1, Industrial Grade

Working Temperature -40 °C to 85 °C

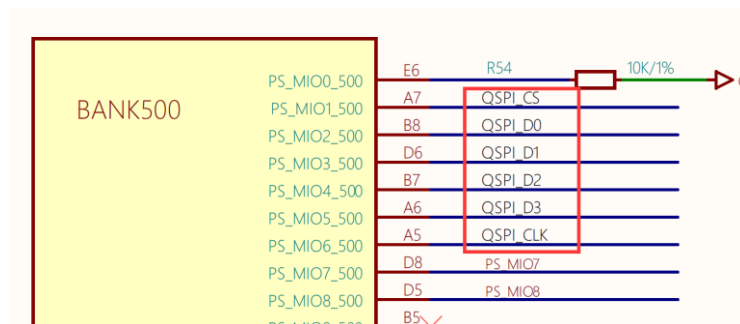
III. DDR Chip

The K230 development board is equipped with two Micron DDR3 memory chips, each with a capacity of 512 MB; model:

The MT41K256M16TW-107 features a 16-bit data bus width and a 512 MB capacity. The DDR3 SDRAM at the PS side supports a maximum operating frequency of 533 MHz (data rate of 1066 Mbps); two DDR3 memory modules are directly connected to the memory interface on the PS's BANK502.

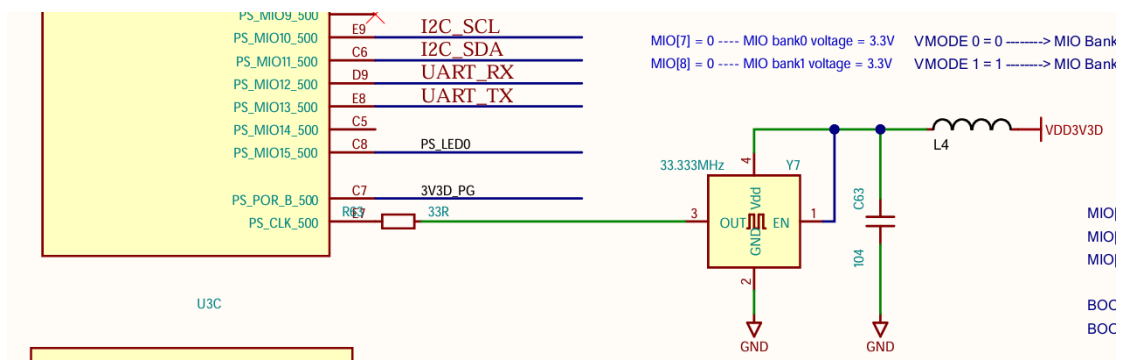
4, QSPIFlash

The K230 development board incorporates a single 256 MB SINGLE-SPI FLASH chip, which supports a 4-bit bandwidth data bus; the FLASH chip model is W25Q256JVEIG and operates under a 3.3 V CMOS voltage standard. Due to the non-volatile nature of QSPI FLASH, it can be used as a system boot device to store the system's boot image. This image primarily includes the FPGA bitstream file, ARM application code, and other user data files. The connection between the QSPI FLASH and the FPGA is shown in the figure below.



V. PS System Clock Source

The X1 crystal oscillator on the Core Board provides a 33.333 MHz clock input to the PS section. This clock input is connected to the PS_REF_CLK_500 pin on BANK500 of the ZYNQ chip. The schematic diagram is shown below:

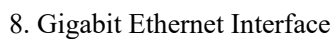


VI. Structural Diagram



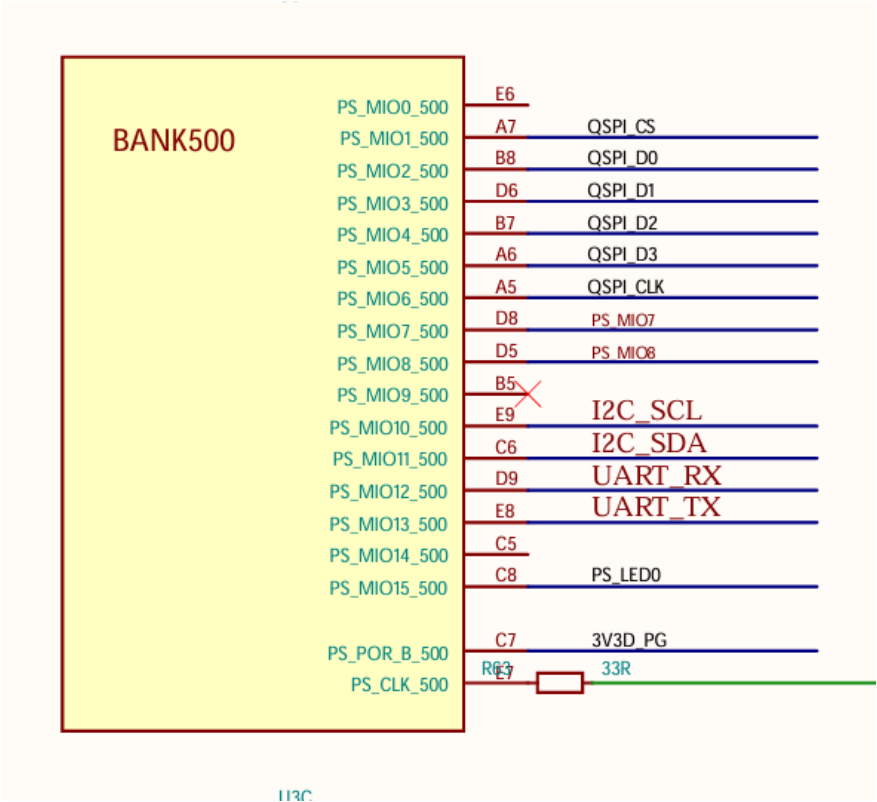
7. ESP32S3

The K230 development board features an onboard ESP32S3 chip with 8 MB of internal FLASH memory; its serial port is connected to the PS side of the ZYNQ. The connection definition is as follows:



CTRL_OUT1		BANK35		CTRL_OUT6	
ETH_TXD3	C20	IO_0_35	IO_L11P_T1_SRCC_35	L16	
ETH_MDC	B20	IO_25_35	IO_L11N_T1_SRCC_35	L17	
PS_PRO_B	B19		IO_L12P_T1_MRCC_35	K17	CLK-40M-FPGA
ETH_MDIO	A20		IO_L12N_T1_MRCC_35	K18	
ETH_RXD1	E17	IO_L1P_T0_AD0P_35	IO_L13P_T2_MRCC_35	H16	ETH_RXCK
ETH_TXCK	D18	IO_L1N_T0_AD0N_35	IO_L13N_T2_MRCC_35	H17	
ETH_TXD2	D19	IO_L2P_T0_AD8P_35	IO_L14P_T2_AD4P_SRCC_35	J18	
ETH_TXD1	D20	IO_L2N_T0_AD8N_35	IO_L14N_T2_AD4N_SRCC_35	H18	
ETH_RXD3	E18	IO_L3P_T0_DQS_AD1P_35	IO_L15P_T2_DQS_AD12P_35	F19	ETH_TXD0
ETH_RXD2	E19	IO_L3N_T0_DQS_AD1N_35	IO_L15N_T2_DQS_AD12N_35	F20	ETH_TXCTL
ETH_RXD0	R138	IO_L4P_T0_35	IO_L16P_T2_35	G17	ETH_RXCTL
ETH_RXD0	R139	IO_L4N_T0_35	IO_L16N_T2_35	G18	
		IO_L5P_T0_AD9P_35	IO_L17P_T2_AD5P_35	J20	
		IO_L5N_T0_AD9N_35	IO_L17N_T2_AD5N_35	H20	
		IO_L6P_T0_35	IO_L18P_T2_AD13P_35	G19	
		IO_L6N_T0_VREF_35	IO_L18N_T2_AD13N_35	G20	
		IO_L7P_T1_AD2P_35	IO_L19P_T3_35	H15	
		IO_L7N_T1_AD2N_35	IO_L19N_T3_VREF_35	G15	
		IO_L8P_T1_AD10P_35	IO_L20P_T3_AD6P_35	K14	
		IO_L8N_T1_AD10N_35	IO_L20N_T3_AD6N_35	J14	
		IO_L9P_T1_DQS_AD3P_35	IO_L21P_T3_DQS_AD14P_35	N15	
		IO_L9N_T1_DQS_AD3N_35	IO_L21N_T3_DQS_AD14N_35	N16	
		IO_L10P_T1_AD11P_35	IO_L22P_T3_AD7P_35	L14	
		IO_L10N_T1_AD11N_35	IO_L22N_T3_AD7N_35	L15	
			IO_L23P_T3_35	M14	
			IO_L23N_T3_35	M15	
			IO_L24P_T3_AD15P_35	K16	CTRL_OUT6
			IO_L24N_T3_AD15N_35	J16	CTRL_OUT2

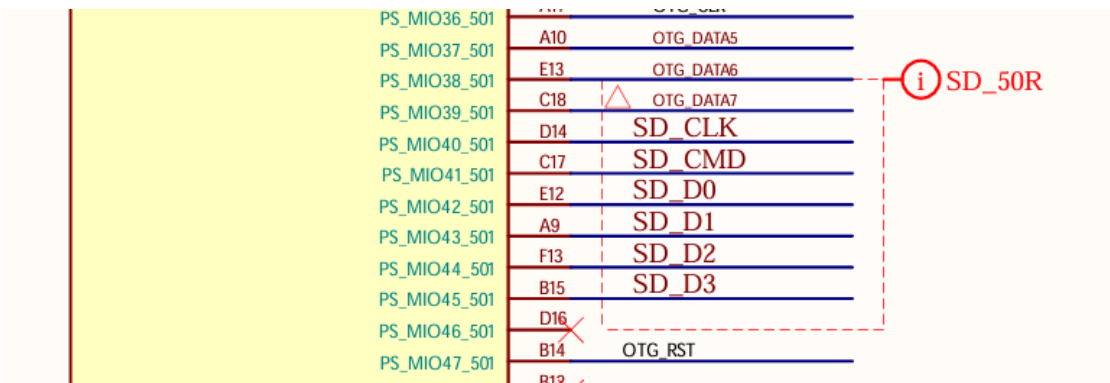
The K230 Development Board features a UART interface on the PS side, designated as UART1; within this interface, TXD is connected to MIO13 and RXD is connected to MIO12; the connection to the Zynq is shown in the figure below:



10. SD Card Slot

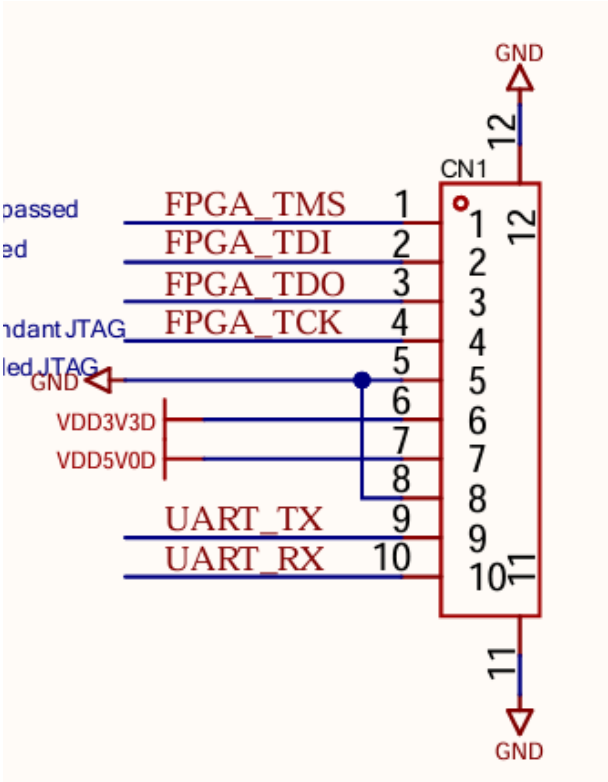
The K230 development board features a MicroSD card interface, enabling users to access SD card storage for storing the Zynq7020 chip's BOOT program, the Linux operating system kernel, the file system, and other user data files.

The SDIO signal is connected to the IO signal of PS BANK501 on the Zynq7020; since the VCCIO for BANK501 is set to 3.3 V, the schematic diagram of the connection between the Zynq7020 PS and the SD card connector is shown in the figure below.



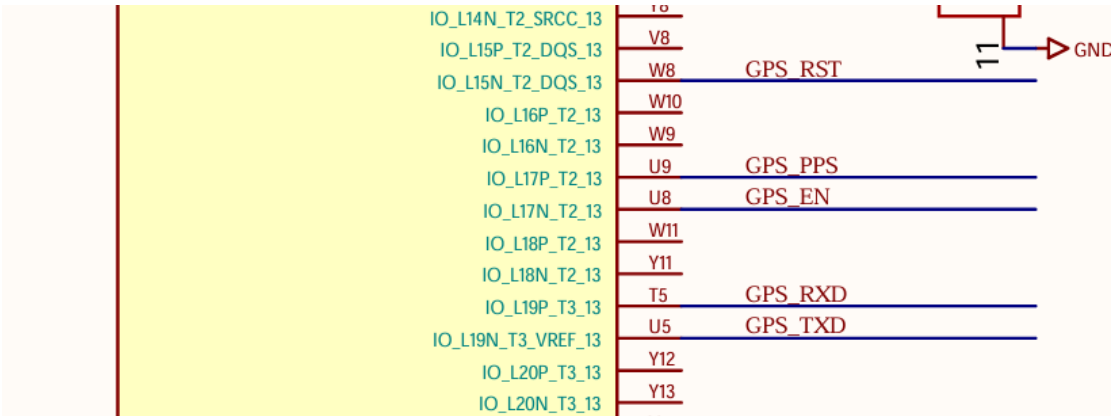
11. DEBUG Connector

The K230 Development Board features a 1.25 mm DEBUG connector; its interface is defined as shown in the figure below, consisting of 10 pins.



12. GPS Module

The K230 board integrates a GPS module for clock synchronization, obtaining GPS time, or tagging IQ data; its connection scheme is shown in the figure below:



13. AD936X

The AD936X is a high-performance, highly integrated RF transceiver designed for RF applications such as 3G and 4G base station and test equipment applications, as well as software-defined radio (SDR) systems. Its programmability and wide bandwidth capabilities make it an ideal choice for a broad range of transceiver applications. This device integrates the RF front-end with a flexible mixed-signal baseband section and integrated circuitry.

NeptuneSDR

The K230 development board and the AD936X support the LVDS18 specification for PCB routing while also being compatible with the LVCMOS18 specification; the SPI interface is used to configure the chip parameters such as transmitter/receiver local oscillator, sampling rate, and bandwidth. Data is transmitted and received using either LVDS or CMOS; communication with the host computer is implemented at the top level using the LibIIO standard library. The LVDS pins are connected to Bank 34 of the Zynq7020, with a VCCO voltage of 1.8 V; the connection diagram is shown below:



6. sampling rate

-Maximum data rate: 122.88 Msps (dual-port half-duplex, DDR mode); -Maximum DATA CLK rate: 61.44 MHz

In LVDS mode:

-Maximum data rate: 122.88 Msps (dual-port full-duplex),

-Maximum DATA_CLK frequency: 245.76 MHz

RF transmission technical specifications:

Local oscillator range: 70 MHz – 6000 MHz

Transmit power: <5 dBm (1 GHz, CW mode)

Bandwidth: 200 kHz – 56 MHz

EVM: <2% 5dBm @20MHz

Gain range:>80 dB

Gain step size: 0.25 dB

Sideband suppression: 40 dBc

Local oscillator suppression: 60 dBc

RF reception technical specifications:

Local oscillator range: 70 MHz – 6000 MHz

Reception sensitivity: -90 dBm @ 20 MHz

Bandwidth: 200 kHz – 56 MHz

EVM: <2% 5dBm @20MHz

Gain range:>60 dB

Gain step size: 1 dB

Noise figure: <8db

IIP3:-25dbm

The `xdc` definition in Vivado is as follows:

1. cmos18

constraints

ad9361

```
set_property -dict {PACKAGE_PIN N20 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_clk_in_p] ; ## IO_L12P_T1_MRCC_35
U1,K17,IO_L12_MRCC_35_DATA_CLK_P
```

```
set_property -dict {PACKAGE_PIN P20 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_clk_in_n] ; ## IO_L12N_T1_MRCC_35
U1,K18,IO_L12_MRCC_35_DATA_CLK_N
```

```
set_property -dict {PACKAGE_PIN Y16 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_frame_in_p] ; ## IO_L7P_T1_AD2P_35
U1,M19,IO_L07_35_RX_FRAME_P
```

```
set_property -dict {PACKAGE_PIN Y17 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_frame_in_n] ; ## IO_L7N_T1_AD2N_35
U1,M20,IO_L07_35_RX_FRAME_N
```

```
set_property -dict {PACKAGE_PIN Y18 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_p[0]] ; ## IO_L1P_T0_AD0P_35
U1,C20,IO_L01_35_RX_D0_P
```

```
set_property -dict {PACKAGE_PIN Y19 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_n[0]] ; ## IO_L1N_T0_AD0N_35
U1,B20,IO_L01_35_RX_D0_N
```

```
set_property -dict {PACKAGE_PIN V17 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_p[1]] ; ## IO_L2P_T0_AD8P_35
U1,B19,IO_L02_35_RX_D1_P
```

```
set_property -dict {PACKAGE_PIN V18 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_n[1]] ; ## IO_L2N_T0_AD8N_35
U1,A20,IO_L02_35_RX_D1_N
```

```
set_property -dict {PACKAGE_PIN W18 IOSTANDARD LVDS_25 DIFF_TERM
```

```
TRUE} [get_ports rx_data_in_p[2]] ; ## IO_L3P_T0_DQS_AD1P_35
U1,E17,IO_L03_35_RX_D2_P
```

```
set_property -dict {PACKAGE_PIN W19 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_n[2]] ; ## IO_L3N_T0_DQS_AD1N_35
U1,D18,IO_L03_35_RX_D2_N
```

```
set_property -dict {PACKAGE_PIN R16 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_p[3]] ; ## IO_L4P_T0_35
U1,D19,IO_L04_35_RX_D3_P
```

```
set_property -dict {PACKAGE_PIN R17 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_n[3]] ; ## IO_L4N_T0_35
U1,D20,IO_L04_35_RX_D3_N
```

```
set_property -dict {PACKAGE_PIN V20 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_p[4]] ; ## IO_L5P_T0_AD9P_35
U1,E18,IO_L05_35_RX_D4_P
```

```
set_property -dict {PACKAGE_PIN W20 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_n[4]] ; ## IO_L5N_T0_AD9N_35
U1,E19,IO_L05_35_RX_D4_N
```

```
set_property -dict {PACKAGE_PIN W14 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_p[5]] ; ## IO_L6P_T0_35
U1,F16,IO_L06_35_RX_D5_P
```

```
set_property -dict {PACKAGE_PIN Y14 IOSTANDARD LVDS_25 DIFF_TERM
TRUE} [get_ports rx_data_in_n[5]] ; ## IO_L6N_T0_VREF_35
U1,F17,IO_L06_35_RX_D5_N
```

```
set_property -dict {PACKAGE_PIN N18 IOSTANDARD LVDS_25} [get_ports
tx_clk_out_p] ; ## IO_L8P_T1_AD10P_35
U1,M17,IO_L08_35_FB_CLK_P
```

```
set_property -dict {PACKAGE_PIN P19 IOSTANDARD LVDS_25} [get_ports
```

```
tx_clk_out_n] ; ## IO_L8N_T1_AD10N_35
U1,M18,IO_L08_35_FB_CLK_N
```

```
set_property -dict {PACKAGE_PIN V16 IOSTANDARD LVDS_25} [get_ports
tx_frame_out_p] ; ## IO_L9P_T1_DQS_AD3P_35
U1,L19,IO_L09_35_TX_FRAME_P
```

```
set_property -dict {PACKAGE_PIN W16 IOSTANDARD LVDS_25} [get_ports
tx_frame_out_n] ; ## IO_L9N_T1_DQS_AD3N_35
U1,L20,IO_L09_35_TX_FRAME_N
```

```
set_property -dict {PACKAGE_PIN T16 IOSTANDARD LVDS_25} [get_ports
tx_data_out_p[0]] ; ## IO_L13P_T2_MRCC_35
U1,H16,IO_L13_35_TX_D0_P
```

```
set_property -dict {PACKAGE_PIN U17 IOSTANDARD LVDS_25} [get_ports
tx_data_out_n[0]] ; ## IO_L13N_T2_MRCC_35
U1,H17,IO_L13_35_TX_D0_N
```

```
set_property -dict {PACKAGE_PIN U18 IOSTANDARD LVDS_25} [get_ports
tx_data_out_p[1]] ; ## IO_L14P_T2_AD4P_SRCC_35
U1,J18,IO_L14_35_TX_D1_P
```

```
set_property -dict {PACKAGE_PIN U19 IOSTANDARD LVDS_25} [get_ports
tx_data_out_n[1]] ; ## IO_L14N_T2_AD4N_SRCC_35
U1,H18,IO_L14_35_TX_D1_N
```

```
set_property -dict {PACKAGE_PIN U14 IOSTANDARD LVDS_25} [get_ports
tx_data_out_p[2]] ; ## IO_L15P_T2_DQS_AD12P_35
U1,F19,IO_L15_35_TX_D2_P
```

```
set_property -dict {PACKAGE_PIN U15 IOSTANDARD LVDS_25} [get_ports
tx_data_out_n[2]] ; ## IO_L15N_T2_DQS_AD12N_35
U1,F20,IO_L15_35_TX_D2_N
```

```
set_property -dict {PACKAGE_PIN V12 IOSTANDARD LVDS_25} [get_ports
tx_data_out_p[3]] ; ## IO_L16P_T2_35
U1,G17,IO_L16_35_TX_D3_P
```

```
set_property -dict {PACKAGE_PIN W13 IOSTANDARD LVDS_25} [get_ports  
tx_data_out_n[3]] ; ## IO_L16N_T2_35  
U1,G18,IO_L16_35_TX_D3_N
```

```
set_property -dict {PACKAGE_PIN T12 IOSTANDARD LVDS_25} [get_ports  
tx_data_out_p[4]] ; ## IO_L17P_T2_AD5P_35  
U1,J20,IO_L17_35_TX_D4_P
```

```
set_property -dict {PACKAGE_PIN U12 IOSTANDARD LVDS_25} [get_ports  
tx_data_out_n[4]] ; ## IO_L17N_T2_AD5N_35  
U1,H20,IO_L17_35_TX_D4_N
```

```
set_property -dict {PACKAGE_PIN V15 IOSTANDARD LVDS_25} [get_ports  
tx_data_out_p[5]] ; ## IO_L18P_T2_AD13P_35  
U1,G19,IO_L18_35_TX_D5_P
```

```
set_property -dict {PACKAGE_PIN W15 IOSTANDARD LVDS_25} [get_ports  
tx_data_out_n[5]] ; ## IO_L18N_T2_AD13N_35  
U1,G20,IO_L18_35_TX_D5_N
```

```
set_property -dict {PACKAGE_PIN T15 IOSTANDARD LVCMOS25}  
[get_ports gpio_status[7]] ;
```

```
set_property -dict {PACKAGE_PIN K16 IOSTANDARD LVCMOS18}  
[get_ports gpio_status[6]] ;
```

```
set_property -dict {PACKAGE_PIN P14 IOSTANDARD LVCMOS25}  
[get_ports gpio_status[5]] ;
```

```
set_property -dict {PACKAGE_PIN P15 IOSTANDARD LVCMOS25}  
[get_ports gpio_status[4]] ;
```

```
set_property -dict {PACKAGE_PIN R14 IOSTANDARD LVCMOS25}  
[get_ports gpio_status[3]] ;
```

```
set_property -dict {PACKAGE_PIN J16 IOSTANDARD LVCMOS18} [get_ports  
gpio_status[2]] ;
```

```
set_property -dict {PACKAGE_PIN J15 IOSTANDARD LVCMOS18} [get_ports  
gpio_status[1]] ;
```

```
set_property -dict {PACKAGE_PIN T10 IOSTANDARD LVCMOS25}  
[get_ports gpio_status[0]] ;
```

```
set_property -dict {PACKAGE_PIN T11 IOSTANDARD LVCMOS25}  
[get_ports gpio_ctl[3]] ;
```

```
set_property -dict {PACKAGE_PIN V13 IOSTANDARD LVCMOS25}  
[get_ports gpio_ctl[2]] ;
```

```
set_property -dict {PACKAGE_PIN T14 IOSTANDARD LVCMOS25}  
[get_ports gpio_ctl[1]] ;
```

```
set_property -dict {PACKAGE_PIN U13 IOSTANDARD LVCMOS25}  
[get_ports gpio_ctl[0]] ;
```

```
set_property -dict {PACKAGE_PIN P16 IOSTANDARD LVCMOS25}  
[get_ports gpio_en_agc] ;
```

```
set_property -dict {PACKAGE_PIN U20 IOSTANDARD LVCMOS25}  
[get_ports gpio_sync] ;
```

```
set_property -dict {PACKAGE_PIN T17 IOSTANDARD LVCMOS25}  
[get_ports gpio_resetb] ;
```

```
set_property -dict {PACKAGE_PIN R18 IOSTANDARD LVCMOS25}  
[get_ports enable] ;
```

```
set_property -dict {PACKAGE_PIN N17 IOSTANDARD LVCMOS25}  
[get_ports txnrx] ;
```

```
set_property -dict {PACKAGE_PIN T20 IOSTANDARD LVCMOS25 PULLTYPE  
PULLUP} [get_ports spi_csn] ; ## IO_L23P_T3_35  
U1,M14,IO_L23_35_SPI_ENB
```

```
set_property -dict {PACKAGE_PIN P18 IOSTANDARD LVCMOS25} [get_ports  
spi_clk] ; ## IO_L23N_T3_35  
U1,M15,IO_L23_35_SPI_CLK
```

```
set_property -dict {PACKAGE_PIN R19 IOSTANDARD LVCMOS25} [get_ports  
spi_mosi] ; ## IO_L24P_T3_AD15P_35  
U1,K16,IO_L24_35_SPI_DI
```

```
set_property -dict {PACKAGE_PIN T19 IOSTANDARD LVCMOS25} [get_ports  
spi_miso] ; ## IO_L24N_T3_AD15N_35  
U1,J16,IO_L24_35_SPI_DO
```

clocks

```
set_property -dict {PACKAGE_PIN C20 IOSTANDARD LVCMOS18} [get_ports  
RGMII_td[3]] ;
```

```
set_property -dict {PACKAGE_PIN D19 IOSTANDARD LVCMOS18} [get_ports  
RGMII_td[2]] ;
```

```
set_property -dict {PACKAGE_PIN D20 IOSTANDARD LVCMOS18} [get_ports  
RGMII_td[1]] ;
```

```
set_property -dict {PACKAGE_PIN F19 IOSTANDARD LVCMOS18} [get_ports  
RGMII_td[0]] ;
```

```
set_property -dict {PACKAGE_PIN E18 IOSTANDARD LVCMOS18} [get_ports  
RGMII_rd[3]] ;
```

```
set_property -dict {PACKAGE_PIN E19 IOSTANDARD LVCMOS18} [get_ports  
RGMII_rd[2]] ;
```

```
set_property -dict {PACKAGE_PIN E17 IOSTANDARD LVCMOS18} [get_ports
```

```
RGMII_rd[1]] ;
```

```
set_property -dict {PACKAGE_PIN F17 IOSTANDARD LVCMOS18} [get_ports  
RGMII_rd[0]] ;
```

```
set_property -dict {PACKAGE_PIN F20 IOSTANDARD LVCMOS18} [get_ports  
RGMII_tx_ctl] ;
```

```
set_property -dict {PACKAGE_PIN D18 IOSTANDARD LVCMOS18} [get_ports  
RGMII_txc] ;
```

```
set_property -dict {PACKAGE_PIN G17 IOSTANDARD LVCMOS18} [get_ports  
RGMII_rx_ctl] ;
```

```
set_property -dict {PACKAGE_PIN H16 IOSTANDARD LVCMOS18} [get_ports  
RGMII_rxc] ;
```

```
set_property -dict {PACKAGE_PIN B19 IOSTANDARD LVCMOS18} [get_ports  
eth_rst_n] ;
```

```
set_property -dict {PACKAGE_PIN A20 IOSTANDARD LVCMOS18} [get_ports  
MDIO_PHY_mdio_io] ;
```

```
set_property -dict {PACKAGE_PIN B20 IOSTANDARD LVCMOS18} [get_ports  
MDIO_PHY_mdc] ;
```

```
create_clock -period 8.000 [get_ports RGMII_rxc]
```

```
create_clock -name rx_clk -period 4 [get_ports rx_clk_in_p]
```

Should you have any further questions, please feel free to contact us at neptunesdr@163.com.

Thank you for supporting Hangzhou Nipe Technology Co., Ltd.! We wish you a successful deployment of your application!

Neptune SDR

©Copyright Neptune SDR 2023

Hangzhou Nipei Technology Co., Ltd.

Address: Hangzhou, Zhejiang, China

Email: neptunesdr@163.com

Taobao: Nepu Technology SDR