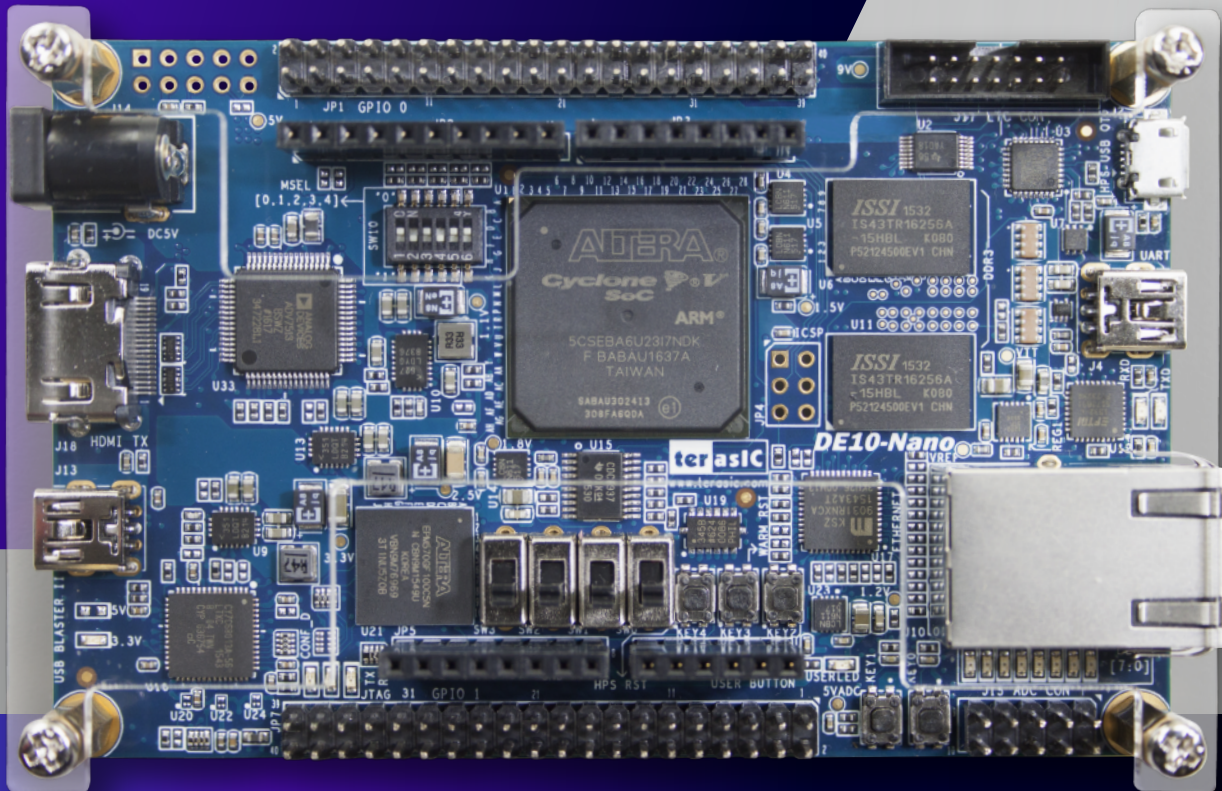


DE10-Nano Development Kit

FPGA-011

The DE10-Nano is the perfect platform to see how an Intel FPGA makes processors better, even if you're not an experienced FPGA designer.



Intel Developer Zone software.intel.com/de10-nano

By

terasic

How do Intel FPGAs make processors better? They can boost performance of critical functions, adapt to changing requirements, and add interfaces not native to the processor.

Intel SoC FPGAs combine the familiarity of an ARM processor with the flexibility of programmable logic.

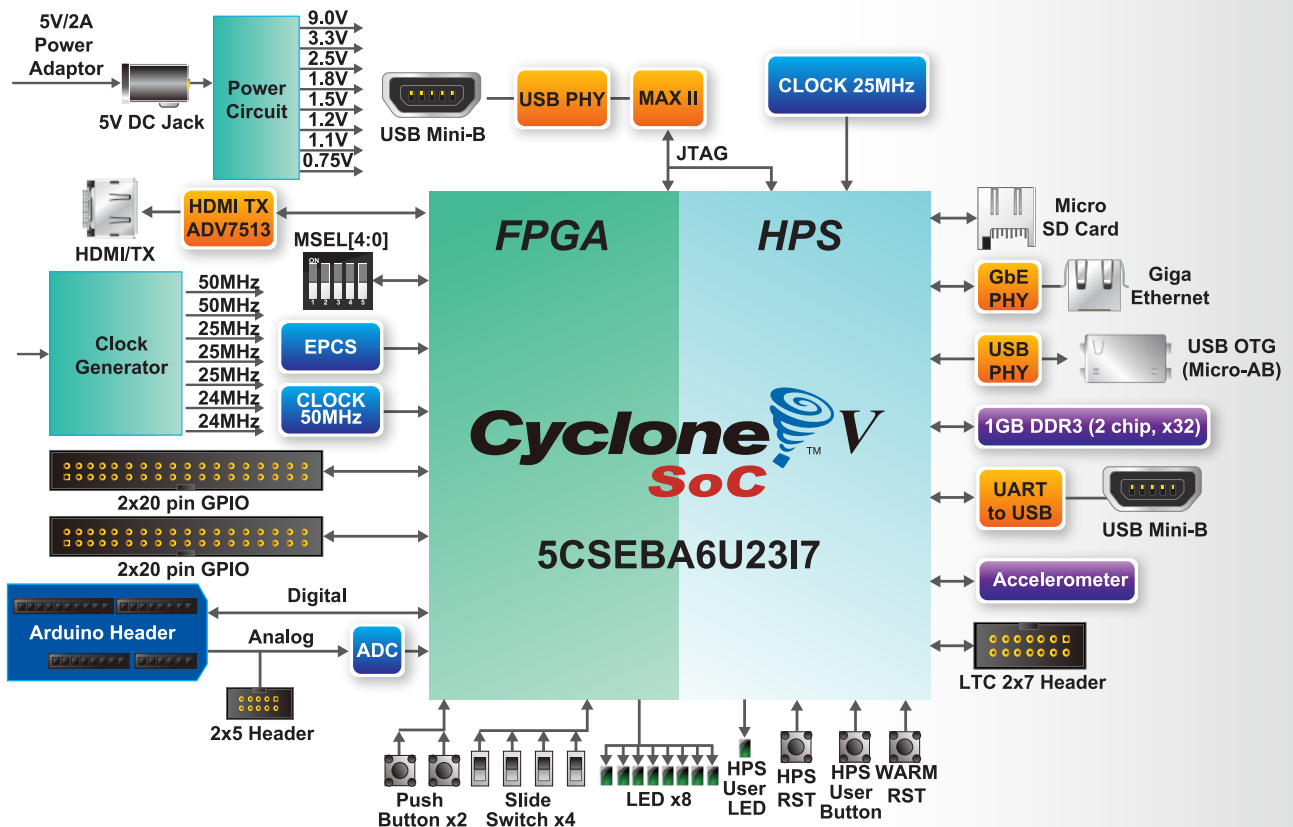
Everything you need to begin your design is either in the box, or free for download.



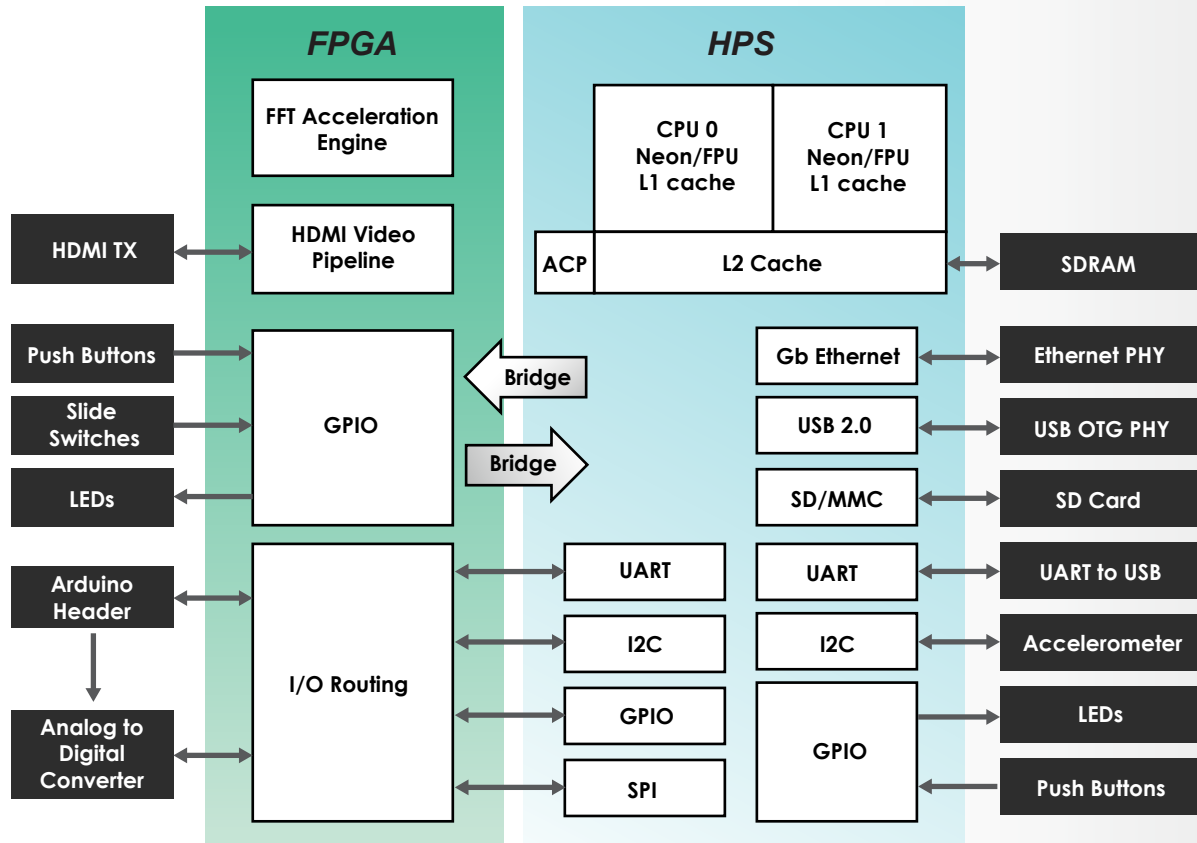
Tools and software available for free download include:

- Intel® Quartus® Prime FPGA Design software
- Intel® SoC Embedded Development Suite
- ARM® Development Studio 5* (DS-5*) Intel SoC FPGA Edition
- FPGA project files (kit reference design)
- Meta-layer (recipes to rebuild Linux software image)
- Compiled binaries and source files
- Example applications and “how-to” articles

Board Block Diagram



System Block Diagram



DE10-Nano Kit

Start

Play

Resources

Welcome to the DE10-Nano Development Kit from Terasic Inc.

This kit will help you learn about Intel® SoC FPGA devices which combine an embedded dual-core ARM® Cortex™ A9 MPCore™ processor system, with user-customizable programmable logic in a single package. This new class of device opens exciting possibilities for the embedded developer. This web site will guide you through the steps of learning about the board from evaluation to full development. Here are some things you can do with this kit:

Play Interact with the board using a web-based interface. Explore sample code, "hands-on" labs, and "how-to" articles.

Resources Learn the fundamentals of SoC FPGA development, access technical documentation and on-line training. Download and install development tools on your PC or workstation. Learn where to get SD Card updates and the source code used in the design running on this board.

Start by getting familiar with the board:

Board Block Diagram and Schematic
SoC System Design
Board Layout
Switches and Push Buttons
LEDs
I/O Headers

Users can connect the DE10-Nano to a computer via the micro USB cable and interact with the hardware through the website served by the board.

The website will guide you through the steps of learning about the board from evaluation to full development.

Specifications

Hard Processor System

Processor

- Dual-core ARM* Cortex*-A9 MPCore processor at 800 MHz
- Neon™ media-processing engine with double-precision floating point unit
- 32 KB L1 instruction cache
- 32 KB L1 data cache
- 512 KB shared L2 cache

Memory

- 64 KB on-chip SRAM
- 1 GB DDR3 SDRAM (32-bit data)

- 8 GB microSD* flash memory card

Processor I/O

- 1 gigabit ethernet PHY with RJ45 connector
- 1 USB 2.0 On-The-Go (OTG) port, USB Micro-AB connector
- microSD* card interface and socket
- Accelerometer (I2C interface plus interrupt)
- UART to USB, USB Mini-B connector

- Warm reset button, cold reset button
- One user button and one user LED
- Expansion header for use with Linear Technology* DC934A dual 16-bit digital-to-analog converter daughter card

Embedded software

- Linux* kernel 4.1.33 LTSI
- Angstrom 2016.12

FPGA

Programmable logic

- Logic elements (LE): 110KLE
- 5,570 kilobits memory
- 224 18 x 19 multipliers
- 112 variable precision DSP blocks
- 6 phased-locked loops (PLL)
- 145 User defined I/O

FPGA configuration sources

- Embedded USB-Blaster* II (JTAG) cable
- Serial configuration flash - EPCS
- ARM* Cortex*-A9 hard processor system (HPS)

FPGA I/O interfaces

- 2 push buttons
- 4 slide switches
- 8 LEDs
- Three 50 MHz clock sources from the clock generator
- Two 40-pin expansion headers with diode protection
- One Arduino expansion header (Arduino UNO* R3 compatibility), can connect with Arduino shields
- One 10-pin analog input expansion header (shared with Arduino analog input)
- 8-channel, 12-bit A/D converter, 500 ksps, 4-pin serial peripheral interface (SPI)

FPGA hardware design

- 32-bit fast Fourier transform (FFT) engine
- HDMI* output (video pipeline)
- GPIO for LEDs, push buttons, and slide switches
- I/F to Arduino shield headers (digital I/O, serial I/O, A/D converter)