fpga的介绍（The introduction of FPGA）

SCM can be understood as integrated on a single chip microcomputer system, small but complete, also has a unit controller memory bus and input and output devices, is used to store the execution of the program, the programming of MCU is programmed on the ROM, after power in ROM program get one by one as in the memory of a computer program. Today's SCM also integrates A/D, D/A conversion, and serial ports and other means of data exchange with the outside world. Microcontroller speed and performance limited, but in some basic control on more than enough.

FPGA is the control level is lower, so the greater freedom of the chip, the FPGA programming is compiled into FPGA in the internal connection table, equivalent to FPGA provides a large number of NAND gate, or gate, flip flops (can be used to form the basic NAND it) a digital device, programming determines how many devices is used and the connection between them. As long as the FPGA scale is large enough, these digital devices can theoretically form all digital systems, including the microcontroller or even the CPU. FPGA has a great advantage in anti-interference and speed.

FPGA (field programmable Gate Array), is a field programmable gate array, it is a product of programmable devices based on PAL, GAL, CPLD and so on the further development of the. It has emerged as a semi custom circuit in the field of ASIC (ASIC), which not only solves the shortcomings of custom circuits, but also overcomes the shortcoming of the limited number of gate circuits of

original programmable devices.

Edit this paragraph FPGA introduction

background

At present, the circuit design completed by hardware description language (Verilog or VHDL) can be simplified

Single synthesis and layout, fast burning to FPGA to test, is the mainstream of modern IC design verification technology. These editable components can be used to implement some basic logic gates (such as AND, OR, XOR, NOT), or a more complex set of functions, such as a decoder or mathematical equation. In most FPGA, these can edit component also includes memory devices such as flip flops (flip flop) or other more complete block of memory. The system designer can connect the logical blocks inside the FPGA by an editable connection as needed, as if a circuit board was placed in a chip. The logic block and connection of a finished product FPGA can be changed according to the designer, so FPGA can complete the logic function needed. FPGA is generally slower than ASIC (proprietary integrated chips), unable to perform complex designs, and consume more power. But they also have a lot of advantages, such as fast finished products, modifications that can be made to correct bugs in the process and cheaper costs. Vendors may also offer cheap but poorly edited FPGA. Because these chips have poor editing capabilities, the development of these designs is done on a regular FPGA, and then the design is transferred to a chip similar to the ASIC.

Another approach is to use CPLD (complex programmable logic devices).

The relationship between CPLD and FPGA

As early as the middle of the 1980s, FPGA had taken root in PLD devices. CPLD and FPGA include a relatively large number of editable logical units. The density of CPLD logic gates is between several thousand to tens of thousands of logical units, and FPGA is usually in tens of thousands to millions. The main difference between CPLD and FPGA is their system structure. CPLD is a somewhat restrictive structure. This structure consists of one or more logical groups of columns with editable results and a relatively small number of locked registers. The result is a lack of editing flexibility, but there are advantages to the expected delay time and the logical unit to link Dan Yuangao ratio. While FPGA has a lot of connection units, it makes it more flexible to edit, but the structure is much more complex.

Another difference between CPLD and FPGA is that most FPGA contain high levels of built-in modules (such as adders and multipliers) and built-in memory. One of the important differences about this is that many new FPGA supports full or partial reconfiguration within the system. Allow their design to change as the system is upgraded or dynamically reconfigured. Some FPGA allow a part of the device to be re edited while the other parts continue to function properly.

Edit the work principle of this paragraph FPGA

FPGA uses a logic cell array LCA (Logic Cell Array) of such a concept, including the internal configurable logic module CLB (Configurable Logic Block), IOB (Input Output input and output module Block) and internal connections (Interconnect) three parts.

Edit the basic features of this paragraph FPGA

1) FPGA design ASIC circuit (application specific integrated circuit), users do not need to chip production, you can get a common chip. 2) FPGA can be used as a sample plate for other fully customized or semi custom ASIC circuits. 3) there are plenty of flip flops and I / O pins inside the FPGA. 4) FPGA is one of the devices with the shortest design cycle, the lowest development cost and the least risk in the ASIC circuit. 5) FPGA uses high-speed CHMOS process, low power consumption, and can be compatible with CMOS and TTL levels. It can be said that FPGA chip is one of the best choice for small batch system to improve system integration and reliability. The FPGA is set up by the program stored in the on-chip RAM, so it is necessary to program the RAM within the chip. Users can use different programming modes according to different configuration modes. When power is added, the FPGA chip reads the data in the EPROM into the chip RAM. After the configuration is completed, the FPGA enters the working state. After power down, the FPGA is restored to a white chip, and the internal logic relation disappears, so the FPGA can be used repeatedly. FPGA programming needs no dedicated FPGA programmer, but only with a universal EPROM and PROM

programmer. When you need to change the FPGA function, you just need to change a EPROM. In this way, the same FPGA, different programming data, can produce different circuit functions. Therefore, the use of FPGA is very flexible.

Edit this section FPGA configuration mode

FPGA has a variety of configuration modes: parallel main mode for a FPGA and a EPROM; a master-slave mode can support a plurality of FPGA PROM programming; serial mode can use serial programming PROM FPGA; peripheral mode can be FPGA as the microprocessor peripherals, a microprocessor for its programming. How to achieve fast timing convergence, reduce power consumption and cost, the optimization of clock management and reduce FPGA and PCB concurrent design complexity, is always the key problem of system design engineers need to consider using FPGA. Now, with the development of FPGA to a higher density, larger capacity, lower power consumption and more integrated in the direction of IP, system design engineer in benefit from these excellent performance at the same time, have to face new challenges due to FPGA hitherto unknown performance and ability level. For example, the leading FPGA manufacturer, Xilinx, recently launched the Virtex-5 family, using the 65nm process, provides up to 330 thousand logic units, 1200 I/O and a large number of hard IP blocks. Large capacity and density make complex routing more unpredictable and lead to more serious timing convergence problems. In addition, more logic functions, DSP, embedded processing and interface modules, which are integrated for different applications, also make clock management and voltage

allocation problems more difficult. Fortunately, FPGA vendors and EDA tool vendors are working together to address the unique design challenges of 65nm FPGA. Not long ago, Synplicity and Xilinx announced the creation of a large capacity time convergence joint working group aimed at helping the system design engineers to maximize the application of 65nm FPGA devices in a faster and more efficient manner.

The Blast FPGA, an integrated tool designed by software vendor Magma, helps to build optimized layouts and speed up timing convergence. Recently, the configuration of FPGA has been diversified!

Edit this paragraph FPGA major manufacturers

1, Altera 2, Xilinx 3, Actel 4, Lattice, of which Altera and Xilinx mainly produce general purpose FPGA, and its main products adopt RAM process. Actel mainly provides nonvolatile FPGA, which is mainly based on anti fuse technology and FLASH process.

Edit the notes for the FPGA design of this paragraph

Whether you are a logic designer, hardware engineers or system engineers, even with all of these titles, as long as you in any kind of high-speed and multi protocol complex system used in FPGA, you will probably need to solve the device configuration, power management, IP integration, signal integrity and other key design the problem. However, you don't have to face these challenges alone, because in

the current industry leading FPGA company in the application engineer every day in the face of these problems, and they have proposed some design work will make you become more relaxed design principles and solutions.

Edit this segment I/O signal assignment

The most versatile pins available, the I/O standard, the termination scheme, and the differential pair FPGA also have the most complex design guidelines for signal distribution. Although FPGA Altera devices no design guidelines (because it is relatively easy to achieve), but the FPGA design guidelines for Xilinx is very complex. But no matter what kind of situation, when assigning signals to I/O pins, there are some common steps to keep in mind: 1. use a spreadsheet to list all the signal distribution plan, as well as their important attributes, such as I/O standard, voltage, required termination method and the related clock. 2. check manufacturer's block / region compatibility criteria. 3. consider the use of second spreadsheet to map out the FPGA to determine which pin is universal, which is special, which support the differential signal of global and local clock, which need reference voltage. 4. using the information of the above two electronic data sheets and the region compatibility criterion, assign the most limited signal to the pin and finally assign the least restrictive. For example, you may need to assign serial buses and clock signals first, as they are usually only assigned to specific pins. 5. redistribute the signal bus according to the degree of constraint. At this stage, you may need to carefully weigh design issues such as simultaneous

switching output (SSO) and incompatible I/O standards, especially when you have many high speed outputs or use several different I/O standards. If your design requires a local / regional clock, you will probably need to use the pins near the high speed bus, so you'd better remember this requirement ahead of time so that you can't arrange the most appropriate pin for it. If the selected I/O standard requires a reference voltage signal, remember to not assign these pins first. The assignment of differential signals always precedes the single ended signal. If an FPGA provides an on-chip termination, then it might also apply to other compatibility rules. 6. allocate the remaining signals at the right place. At this stage, consider writing a HDL file that contains only port assignments. Then you create a restriction file manually by using the tools provided by the vendor or using a text editor, adding the necessary support information for I/O standards and SSO. Once you've got these basic files, you can run the layout and wiring tool to make sure that you've ignored some guidelines or made a wrong allocation. This will allow you to work with the layout engineer in the initial stages of the design to plan PCB routing, redundancy planning, heat dissipation issues, and signal integrity. The FPGA tools may help you in these areas and help you with these problems, so you must make sure that you understand the functionality of your toolkit.

The sooner you consult a layout expert, the more likely you will have to deal with complex issues and design iterations that may be avoided by some preliminary analysis. Once you have a satisfactory signal assignment, you must lock them

with the limit file. ---------- CMOS design mainly based on the power consumption of three types: internal (short circuit), leakage (static) and switch (capacitance). When the circuit transient, short circuit connection between VDD and internal power consumption. Leakage power consumption is caused by the parasitic effects of CMOS technology. The switching power consumption is caused by the discharge of the self load capacitor. Switching power consumption, together with short-circuit power, is called dynamic power dissipation. Design techniques for reducing static power consumption and dynamic power consumption are described below.

Edit this paragraph to reduce static power dissipation

brief introduction

Although the static current is negligible compared to the dynamic current, it is important for battery powered handheld devices, especially when the device is powered on and not working. Many factors of static current, pull up or pull down resistor includes working current, is not completely off or connected to the state of the I/O and the internal transistor internal wiring resistance, input and three state electric drive on. In volatile technology, a certain amount of static power is required to maintain programming information. Anti fusing is a nonvolatile technology, so information storage does not consume quiescent current.

Several design methods to reduce static power consumption

The drive input should have full voltage levels so that all transistors are fully guided or closed. Since the pull-up or pull-down resistors on the I/O line consume a certain amount of current, avoid using these resistors as much as possible. Less drive resistors or bipolar transistors are needed to maintain a constant current, which increases the quiescent current. The clock pin is connected to the low level according to the recommended condition of the parameter list. The dangling clock input greatly increases the quiescent current. Reduce the use of I/O between devices when dividing the design into multiple devices. EX devices, LP mode pins are used, the Actel eX family has designed a special low power "sleep" mode. After the pin drives the high level 800ns, the device enters very low power standby mode with a standby current of less than 100 A. In low power mode, all I/O (except clock inputs) are in the three state, and the kernel is powered down. Because the kernel is powered down, the trigger stored information will be lost, when entering the working mode (pin to flat 200ms in drive), the user must again on device initialization. Likewise, the user should turn off all the clocks entered via CLKA, CLKB, and HCLK. However, these clocks are not in the three state, the clock can enter the device, thereby increasing power consumption, so in low power mode, the clock input must be in logic 0 or logic 1. Sometimes it's hard for users to prevent clock from getting into the device. In this case, the user may use the normal input pin adjacent to the CLKA or CLKA and add CLKINT to the design. In this way, the clock will pass into the device near the normal input of the clock pin, and then

provide the clock resource to the device through the CLKINT. With this input circuit, since the conventional I/O is three state, the user does not have to worry about clocking into the device. Of course, adding a primary gate circuit creates a larger clock delay for the 0.6ns, but fortunately this is acceptable in most low power designs. Note that the CLKA or CLKB pin associated with the CLKINT buffer should be ground. Also note that CLKINT can only be used as a connection clock, and HCLK does not have the ability to connect an internal wire network to HCLK, so HCLK resources cannot be driven by regular inputs. In other words, you cannot use HCLK if you use the LP pin; when you use the HCLK, you should truncate the clock signal externally.

Edit this paragraph to reduce dynamic power

Dynamic power consumption is power dissipation when the clock is operating and the input is switching. For CMOS circuits, the dynamic power dissipation basically determines the total power dissipation. Dynamic power consumption includes several components,

Mainly capacitive load, charge and discharge (internal and I/O), and short circuit current. Most dynamic power is consumed by the internal or external capacitor charging and discharging to the device. If the device drives multiple I/O loads, a large amount of dynamic current constitutes the main part of the total power consumption. For the given drive in the design, the dynamic power is calculated by the lower p=CL x V 2 DD x F type, the CL is the capacitive load, the VDD is the supply voltage, and the F is the switching

frequency. Total power consumption is the sum of the total power consumed per drive. Because the VDD is fixed, reducing the internal power to reduce the average logic switching frequency, reduce the total number of each logical switch, clock edge reduce connection network, especially the high frequency signal line capacitance value in network. For low power designs, preventive measures should be taken from each design level of the system to the process, and the higher the level, the better.

1. application of FPGA in circuit design

Connection logic, control logic is early FPGA play a more significant role in the field of foundation is FPGA application. This requires developers in fact in the circuit design and application of FPGA the difficulty is quite large to have the corresponding hardware knowledge (circuit knowledge) and software application ability (development tool) this talent is always scarce. Are engaged in new technology, new product development success will become the market mainstream products for product designers based applications in the near future, the design of general and special IP will become a hot industry! The premise of the circuit design is that we must have certain hardware knowledge. At this level, the weight of learning, of course, fast entry is very important, the better the seat, the more people, circuit development is golden rice bowl

2. product design

The application of relatively mature to some specific fields such as communication, video, information processing and so on development to meet the industry needs and industry customer acceptance of products that are combined with the main problems of FPGA technology and professional technology, in addition to the interface problem of product design is with professional customers also include professional tools and civil products the former focuses on the performance of products, the latter is sensitive to the price of product design to achieve product function as the main purpose, FPGA technology is a means to achieve in this area, because with FPGA interface, control function, IP, embedded CPU features such as the conditions to achieve a simple structure, a high degree of curing system, the function of comprehensive product design the application of FPGA technology to the broad market demand for space requirements, product design has a great burst of technical personnel is relatively high, The road is long, but now the whole industry is in the formation of "first team", as long as the bright future, product design is a direction of occupation development orientation, is not a simple love can do! The field of product design will bring up a large number of enterprises and entrepreneurs, and it is a hot spot and opportunity in the near future

3. system level applications

Application system is the combination of FPGA and traditional computer technology, such as the use of Xilinx V-4 to realize the computer system of a FPGA version of the FPGA V-5 series, POWER PC realization of embedded CPU, and

then with various peripheral functions, to achieve a basic environment, run LINUX system on this platform, the system also supports a variety of standard peripherals and interface functions (such as image interface) for the rapid FPGA large system is very helpful. This "cottage" very strong flavor of the system, the early advantages are not necessarily obvious, similar to the ARM system situation, but if you can slowly play out the advantages of FPGA, and gradually realize some features, the system is also a direction of development. If the system level application, developers do not have to expand development capabilities of the system, but since programming is not what the significance of the development of device driver is another case in the application appears to be a high starting point, but does not have the deep development ability, it is likely to become lovers, just as many people will do "but cannot be called programming similar above is some personal development, hoping to help people to learn but lost science thoughts FPGA.

It's a good business with good personal opportunities for success. But it must be a very competitive industry, the key is to see the speed and depth, of course, the market adaptability.

1. introduction

Traditionally, FPGA can only achieve relatively small logic circuits, and with the improvement of process technology, the capacity and performance of FPGA are also improved. Nowadays, FPGA has been used to implement large logic

circuits and even the whole system. These large systems are very different from the small logic branch circuits that have traditionally been the target of the FPGA market. One of the most important differences is that these large systems contain memory. So, to support memory effectively in FPGA, the support of the structure is critical. FPGA chips that contain large memory arrays have been introduced by manufacturers, such as Xilinx and Altera.

However, most academic FPGA CAD tools are only for homogeneous FPGA, that is, FPGA containing only logical modules and routing resources. In this paper, we propose a flexible FPGA architecture including embedded memory and a method for building RRG (routing resource map). In the study of FPGA, we usually use experimental methods to evaluate the practicability of new structural characteristics. Therefore, we need flexible CAD tools to compare a variety of structures. VPR is the FPGA CAD tool of this type, which applies to a variety of FPGA structures. We developed a new layout routing tool, VA. It enhances the functionality of the VPR so that VPR can effectively handle the FPGA architecture that contains embedded memory, and ensures its flexibility.

The memory module has some differences relative to the logic module, such as the distribution of module pins and the module height, which makes the design of RRG very complex. In this section, we'll discuss a simple and flexible approach to RRG design. Our RRG design method is based on VPR, but we have made some improvements to deal with the wiring problem of FPGA that contains embedded

memory.

Singlechip。 The difference and advantages of fpga.

Singlechip: strong control ability. The ability of sequential realization is weak, and the combinational logic ability is stronger.

FPGA: control ability is weak, combination logic ability is strong, timing realization ability is strong.

On the whole。 FPGA in the frequency range and logic to achieve higher than a single chip. But because there is no instruction system. Therefore, the ability to control and operate is relatively weak.

FPGA most applications:

Interface: especially high-speed communication interface. This aspect allows coupling and bridging of different rates and protocols.

Data and processing: data processing at the hardware layer. Such as parity, CRC checksum, and so on. If the instruction system is used. The first is slow efficiency. Second resources only.

Ha-ha。 Something behind may be an application!

Now, in general, more systems tend to combine CPU and FPGA systems. Because the CPU instruction cycle may be

relatively long. Moreover, multiple interrupts can cause AMR processing to be overburdened (especially in IO processing). Then you'll need a FPGA. FPGA's built-in logic allows full release of ARM or CPU processing power. As CPU's "co processor"". At this point, FPGA can do buffering for CPU: pre judgment, and so on. You can even use FPGA as a watchdog for CPU. Prevent CPU from running and so on. At the same time, you can also open up a certain amount of logic within the FPGA. Make registers for CPU to read. CPU will be able to understand the work of peripherals. Avoid peripheral failures and cause system crashes. Wait

Compared with SCM, FPGA advantages?

Reward points: 25 time: 2006-9-8 13:04 | | questionger: a_gump

Trouble, as far as possible in detail, and COPY does not matter

Thanks in advance!

Best answer

1, FPGA runs at a fast rate

FPGA internal integrated lock ring, you can double the frequency of the external clock, the core frequency can be several hundred M, while the MCU running speed is low. In high-speed occasions, single-chip microcomputer can not replace FPGA

2,

FPGA pins are much easier to implement on large scale systems

Microcontroller IO Port Co., FPGA and hundreds of IO, can be easily connected peripherals. Such a system has multiple AD, DA microcontroller, to allocate resources, careful bus isolation and FPGA, because of the rich resources of IO, can easily connect the peripherals with different IO

3, the FPGA internal program runs in parallel and has the ability to handle more complex functions

MCU program is a serial execution, executing a to perform next, in dealing with emergencies can be invoked only limited interrupt resources; FPGA logic can be executed in parallel, can handle different tasks at the same time, it leads to more efficiency of FPGA

4, FPGA has a large number of soft core, which can be easily developed for two times

The FPGA even includes the microcontroller and the DSP soft core, and the IO number is limited only by FPGA's own IO