Let's talk about how to learn a single chip.

To learn a single chip, good methods, good tools, good teachers are necessary,Because they can help you get more done and improve your learning efficiency.I am often asked, "is there any good way to learn computer programming?"I answered without thinking: "yes!"This method is: "case study method".

Our traditional education method is to let you memorize a bunch of things because they are useful for the exam.The same is true of English.This education method forgets the basic of education: learn to use it.It is now clear that the best way to learn English is to study in a good language environment.The advantage of learning in an application environment is that for a certain point of knowledge, not only is it understood quickly, but it is also learned how to use it.More importantly, you will find that this knowledge point is related to other knowledge points, and is used in conjunction with each other!

So, I suggest that you carefully read an actual product case,and you are going to be able to figure out every single line and know why the author wrote this.Don't let any of the problems pass.Basically, when you're done with this case, you know how to program, you know how to design a product.What an efficient way to learn!It shines with the light of the philosophy of "unity of knowledge and practice", which I call "case study". This is the way I learned C, students who are interested can read this blog “ how I learned C language”.In addition to enabling you to master basic knowledge of procedural grammar,it also has the following advantages:

1、 You can learn product design idea and software architecture.Software architecture can be said to be the author's programming ideas.It is critical that software architecture design is reasonable.The design is reasonable, the programming is more favorable, the development is easy to succeed.Instead, programming is awkward, and it takes a lot of time and quality to change.Only by mastering the software architecture can you actually write qualified product programs.

2、 You can learn normalized programming.Beginners usually don't know how to write a program, usually they want to make it up,So the readability, portability and maintainability of the program are not available.The software writers of formal products are more experienced engineers generally.The program writed by them is also standardized.Beginners can learn a lot about programming techniques and techniques.

3、 You can learn production-level programs.Often, the software that a beginner is exposed to is easily accessible, primary learning material, and only a functional demonstration.These are not even lab products.However, there is a great difference between lab products and actual products. This is mainly about the robustness and anti-interference of the software and the quality of the products.These are not available on the regular learning board.

However, the source of the actual product is a trade secret.The average beginner can't get it.In view of this situation, our series of single-chip learning board is suitable for this "case study method".First, it makes the experimental routines of each functional module in a small "case" form;Second, it integrates some typical cases from actual projects.When students learn these cases, you can do the project themselves.Of course, achieving these goals is all about the TreeOS 1.0 operating system, a real-time operating system that can be easily mastered by a beginner.

WeiboID：@TreeOS Lin Tianxiao