Brian Clegg wonders whether our schools are placing the right emphasis on computer education.

Think of the children

It's knee-jerk time. Computers in schools? Great. It gets our young people ready for the real world. But is it a good thing? (For that matter, are schools there to get people ready for the real world? Anyway...) Let's travel back to the Stone Age. When I was a lad, computing didn't exist. Actually, that's not true; but it might as well have been, because there weren't any microcomputers and Littleborough County Primary School didn't run to a mainframe. Similarly, my illustrious grammar school hadn't even a calculator (it was, however, hot on slide rules). That's where I got my introduction to computing, though. We punched cards by hand, bundled them up, and posted them to a London university. A week later, a listing came back, showing a punching error in the second card. With luck, you could get a program running at least once a term.

This might have been extreme, but it was preparation for real life. Out in the business world there was quicker turnround, and you could punch your cards by machine (or have them produced by the punch room). Even so, the basics were the same. Now let's skip forward. As micros emerged, computers started to penetrate primary education. Now there was a conflict of interest. Should we opt for a school computer that was user-friendly, or one that prepared you for the real world? In the US, ease of use (and excellent marketing) won. While business equipped itself with IBM compatibles, American education chose the friendly face of the Mac. In the UK, thanks to politics and cost, we ended up with Acorn's BBC Micro — a typical British compromise that was neither relevant to business nor easy to use.

> **Acorn's BBC was sophisticated when compared with the home offerings of the time, BUT IT LOCKED THE EARLY YEARS OF OUR EDUCATIONAL SYSTEM INTO A BACKWATER from which it is only now escaping**

Before I get flooded with emails, I ought to qualify that. The BBC machine was sophisticated when compared with the home offerings of the time, and it came out a couple of years before the Mac. But it locked the early years of our educational system into a backwater from which it is only now escaping. When you see how five-year-olds can use a Windows PC, there's little argument for going any other way now, and PCs are at last taking over; but software remains an issue. Not software for teaching maths or spelling, but the software that introduces computing itself. Here, the traditional values of buying British and not worrying about the real world still hold sway.

I've recently been sent a package called ToonTalk, made by Logotron, a British software house specialising in software for education. ToonTalk is a programming language, letting youngsters learn the rudiments of controlling a computer using an on-screen construction set with the help of a friendly robot. ToonTalk is brilliant, and adequately covers modelling and control, two requirements of the curriculum. Admittedly, my junior-school teacher friend initially found it slow going, but her pupils are now getting on well. And ToonTalk is great, seen just as a modelling tool. But it's certainly not preparation for the world, since the vast majority of those children won't be programming computers.

So what's missing from computer education? How about starting with the basics of handling, like putting CDs in without getting fingerprints all over them. Then the ABC of Windows, understanding how to interact with the user interface to get things done. How to explore menus and help systems, all done in a jolly, fun and games fashion, of course. If we can ensure that our children's natural instinct to explore is used in the right way, they'll never need training in Word or Notes when they get older — it will come naturally. As for programming, if we must teach it, then why not use something as universal as Visual Basic? Those crafty UK educational software houses could build nice add-ins to make VB even easier, though even without it, eight-year-olds are capable of writing programs, are learning a lot about control, and are gaining a useful skill into the bargain. By the time they get to work, VB may not be in existence, but it's a better bet for the basis of future development environments than a construction kit.

Brian@cul.co.uk