Taking stock: what made a maker business

U.S. Whatever's attention span is one of Internet rarities. Yet Internet gurus have long been the rulers with short attention spans. Is Internet fame and cash cash good or good for Internet gurus? Or is it good for everything else, and good for good or bad for bad? Or is it good for everything else, and is it bad only good for good, and good only bad for bad? Or is it bad for everything, and good only for bad? Or is it bad for everything, and bad only for bad? Or is it bad for everything, and bad only for bad?

It has become the "Internet" Internet, and it seems so commonplace and cool. Is it good or bad? Or is it good for everything, and good only for good? Or is it good for everything, and good only for good? Or is it good for everything, and bad only for bad? Or is it good for everything, and bad only for bad? Or is it good

Years ago the pioneers, Tom Rhodes and Jim Lewis, went to the shop, bought a few components, and built a PCB. Today, the process is much more complicated. Tom and Jim have moved on and retired to Kentucky, where they still make PCB's, but now it takes a lot more than a few components and a soldering iron.

Today's PCB's are made with computers, using software like Altium Designer. Altium Designer is a CAD package that allows you to create a schematic, place components, and route the PCB. Altium Designer can also simulate the circuit, and generate the files needed by the PCB fabrication house.

Altium Designer can also generate the files needed by the PCB assembly house, which places the components on the PCB. Some PCB assembly houses can also solder the components to the PCB, but most people do that themselves.

Once the PCB's are made, they need to be populated with components. Most of the time, we do this ourselves, so we need to order the parts we need from various vendors. When the parts arrive, we place them on the PCB, and then we solder them in place using a reflow oven.

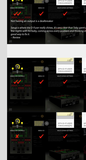
Some of the parts are too small to be placed and soldered in reflow oven, so we do that by hand. Once all the components are in place, we use a pick-and-place machine to place the parts we can't do by hand, and we use a reflow oven to solder them all in place.

Once the PCB's are populated with components, we need to program the processors and other programmable components on the board. We program most of the components using a JTAG programmer, which allows us to program the components while they are on the board.

Once the PCB's are programmed, we need to test them to make sure they work. We do most of our testing using a custom tester that we have built specifically for our product. The tester allows us to quickly and easily test all the features of our product.

Once we have tested the PCB's and they are good, we need to add the mechanical components to the board. This includes things like heatsinks, connectors, and mounting hardware. We do most of this by hand, using a variety of tools and fixtures to make the process easier and more consistent.

Once the PCB's are fully populated, we need to test them again to make sure everything still works after the mechanical components have been added. We do most of this testing using our custom tester, but we also do some manual testing to make sure everything looks and feels right.



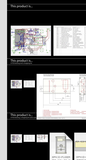
Once we have tested the PCB's and they are good, we need to add them to a finished enclosure. We do most of this by hand, using a variety of tools and fixtures to make the process easier and more consistent.

Once the PCB's are in their enclosures, we need to test them one last time to make sure everything still works after they have been installed in their enclosures. We do most of this testing using our custom tester, but we also do some manual testing to make sure everything looks and feels right.

Once we have tested the PCB's and they are good, we need to add the final cosmetic touches to the product. This includes things like adding labels, logos, and other cosmetic features that make the product look and feel like a finished product.

Once we have added the final cosmetic touches to the product, we need to test it one last time to make sure everything still works after the cosmetic touches have been added. We do most of this testing using our custom tester, but we also do some manual testing to make sure everything looks and feels right.

Once we have tested the PCB's and they are good, we need to package them for shipping. We do most of this by hand, using a variety of tools and fixtures to make the process easier and more consistent.

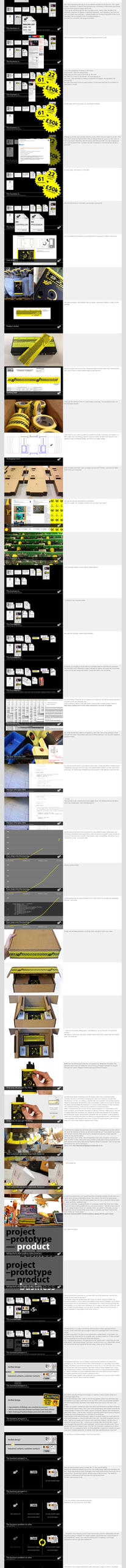


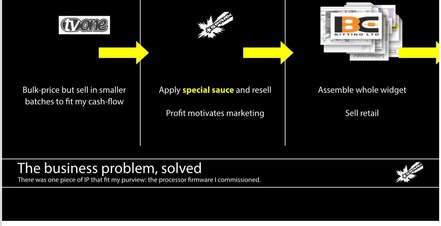
Once we have packaged the PCB's, we need to ship them to our customers. We do most of this by hand, using a variety of tools and fixtures to make the process easier and more consistent.

Once we have shipped the PCB's, we need to keep track of them and make sure they are still working properly. We do most of this by hand, using a variety of tools and fixtures to make the process easier and more consistent.

Once we have kept track of the PCB's, we need to make sure they are still working properly. We do most of this by hand, using a variety of tools and fixtures to make the process easier and more consistent.

Once we have made sure the PCB's are still working properly, we need to repeat the process all over again

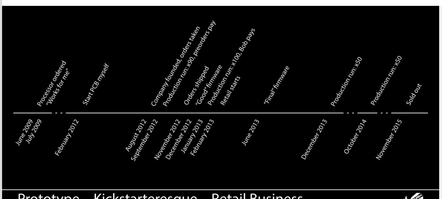

The business p...
The business p...
The business p...



The business problem, solved

There was one piece of IP that fit my purview: the processor firmware I commissioned.

made this possible.
First, there was a piece of IP that was meaningful for production 'in-house' – literally, my house.
The ability to commission a custom firmware from TV One for the video processor, was my personal reward, and a concrete 'IP' advantage over any competitors (e.g. that other video mixer). Or, in this case, my own partners. I could flash this firmware onto the generic TV One processors, stick my nice sticker on, and then resell onto Bob who would pair with the controllers he got made.
Second, is the call-off order. This would only work if I could get bulk pricing on the processors, but I can't risk buying 100 up-front. And that would be true even if I had the capital, which I didn't. Well, companies are people, and people can figure out if there's a way to make something work for all involved. The answer is a call-off order, in which you split a larger bulk-buy into chunks. Obviously it isn't quite as good for TV One to deal with e.g. five orders of 20 rather than one of 100, but ultimately if that's what stands between 100 orders and none, then that's what to do. I ordered my first chunk with the capital I had, and went back for more when Bob had sold those on. Which is not to say it was frictionless between us, but we both understood each other's constraints and made it work.
So TV One got to sell lots of processors, I effectively extracted a license fee via wholesaling processors, Bob earnt his cut by making the controllers and doing the retail logistics. All while minimising outlay and risk: controllers were the cheaper part of the package, so the economy gained through making them up-front in bulk was a risk Bob was willing to take. The more expensive part of the package, the processors, were what you wouldn't want to risk having in stock and unsold or damaged; these could be ordered in smaller quantities nearer on-demand.



Prototype – Kickstarteresque – Retail Business

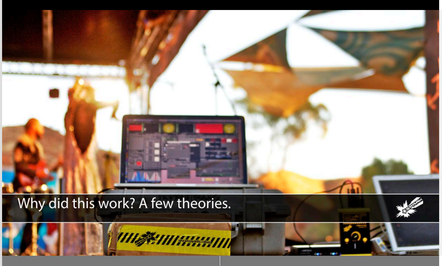
The timeline

	Sales	Expenses	Profit before tax etc.
Prototype	£1.5k	£2k	£0.5k
Kickstarter	£53k	£46k	£7k
Retail	£83k	£61k	£22k

Prototype – Kickstarteresque – Retail Business

The numbers

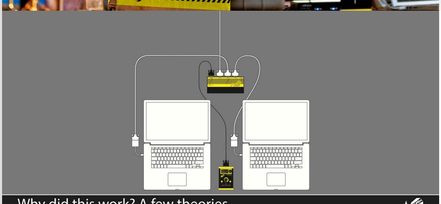
Sod the numbers, or that years before the prototype here I had something that worked for me (and none of my peers had anything like it!). Here is a box that I made, on the other side of the world. Retail packages on doorsteps in Peru.
Anyway! Why did this work? A few theories.



Why did this work? A few theories.

Who would believe that 'just a guy' can make a hardware mixer? I mean this genuinely, it's not something I would necessarily believe as a punter on the internet with money in my pocket.
(If you can engineer your own mixer from scratch, say your name is Sébastien Bourdeauducq, you'll soon find your skills are better used making bespoke signal processing kit for particle accelerators etc. https://github.com/m-labs/mixxeo-soc -> https://github.com/m-labs/artiq)
So where did the credibility come from? The website says this –

> *To make the *spark d-fuser, we take an award winning video processor, partner it with a custom controller designed to give you the hands on control and worry free setup you need, and work both these over with custom firmware.*



Why did this work? A few theories.

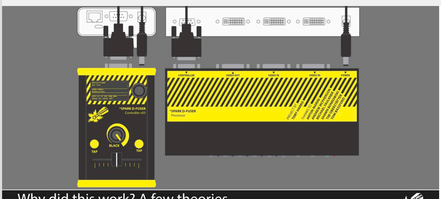
1. Extending an existing product anchors expectations: it will work, it should cost what things in that market cost

Extending an existing product anchors expectations. The video processor already had a track record; it was an award winner. This thing – by extension my package – was going to work.
That processor also has a price. It sits in a market where most kit is much more expensive than it. The processor has a list price. These prices act to anchor what is 'reasonable' to most people.
Both these aspects are reinforced by the diagram. There are two legit laptops, a legit video processor, and the drawing of my controller looks of a piece with that.

Why did this work? A few theories.

2. Two phases: first with one foot still in its "DIY" genesis, then free of that

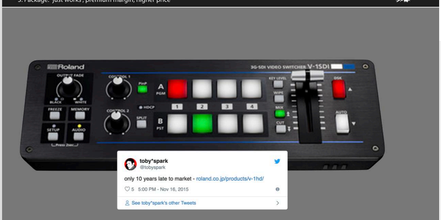
You've seen that this all started off as a hack, as a DIY project. That I was already in a well-networked community of practitioners that presented a ready market. The route to market that fit here, was "I'll do for you what I did for me", and, basically, that worked. It got me past PayPal and warranty concerns. People got their controller, and between the product and my support, were happy.
I wasn't going to do it again, though. I'd already had the upside – a better mixer for me, kudos – and continuing would expose me to more of the downside, for instance supporting people's existing TV One processors if I'd sold them the controller alone. My take back then was for people to take on my designs and go make, fix or further extend their own.
I'd always had the vision of selling the package, but the history and immediate market meant this phase was oriented around the controller. I got to see two different outcomes. Selling the controller-only had a high support burden, and seemed to attract people who were determined to undercut the price I'd set somehow. Part of this was fair enough in a way, e.g. my initial community idea of open source hardware would suit people finding a cheap processor on Ebay, but then the support would be pooled and not mine alone. But a lot of it was because some people thought that they could make the controller for less; they were price anchored not by AV-kit but by DIY hacks.
Whereas people who bought the package had a good experience out of the box, and were generally pre-dispositioned to value something that 'just worked'.



Why did this work? A few theories.

3. Package: "just works", premium margin, higher price

If this was going to go beyond that first phase, it would have to be package only.
The customers were happier, with a lower support burden
Customers valued something that "just works"
That value can translate to charging a premium.
The package, consisting of more, commands a higher price.
The same margin on a higher price is more money back for you. A better margin on a higher price is… much better. There simply wasn't the profit in selling controllers-only to make it a going concern. But there was in selling the package.
Plus, in selling the package, there was a way for both me and Bob to take our cut. Bob earnt his margin by becoming a manufacturer of the controller. I earnt my margin by becoming a wholesaler of the processor – which to this day seems ironic to me.



Why did this work? A few theories.

4. We quit while we were ahead. Just in time, in fact.

Lastly this worked because Bob and I quit while we were ahead. We actually were going to quit after two retail production runs, but chanced a third.
We were lucky. The product pictured was announced the same month Bob and I sold our last. Its the product that me and my peers had expected all those years ago, and whose continued absence the D-Fuser had been a response.
So if the previous slide suggests it's easier to make the business work if you make really expensive things, notr the corollary. Having a bad batch or unsold shelf of cheapo trinkets is going to be survivable. But if you're making expensive things, you don't want to get into debt making them, and don't get stuck with stock.

to the extent this has been about me, it's about you.
if you can wield a computer, you can do something like this.

Toby Harris – http://tobyz.net

*spark live ltd - http://sparklive.net

Why do I say 'wield a computer'? I was one person and a laptop. This hasn't been about hands-on maker-skills. It was the computer that made everything possible. The time, space and undo buttons of design software paired with internet how-to guides. User experience as code, even for a hardware product.
Iterative design + outsourced manufacturing is a powerful combination I believe in, and you should too. If you can wield a computer, you can do something like this.