

ARM and Time- to- Market

Posted in replay to [TinkerShaw](#) at the New Paradigm Investing board at TMF.

Tinker:

To quickly illustrate, if you charge \$10 per unit and you ship 500,000 units, you have \$5 million in royalties. At 2 million units shipped you now have \$20 million in royalties. Somewhere along this line it just becomes cheaper to do it yourself as you may be able to internally develop the product, say for \$8 million a year, and therefore would decide to do so if more than 800,000 units are expected to ship.

It's not just a question of dollars and cents but a question of total cost. One of the costs often discussed but rarely quantified is time-to-market. How much does it cost to be three months late to market? Is saving three months in time-to-market worth paying an extra 5¢ royalty? In the world of increasing returns, where the winner takes all, where path dependence sets the future, it would seem that the extra royalty is well worth paying. I think the problem lies in the fact that it is very hard to quantify the value of time-to-market so, instead of cost accountants with green shades and spreadsheets, it will be the market that will make the decision on whether to pay or not to pay the higher royalties. Some companies will use ARM and pay the royalty while others will use in-house designed SoCs for their needs. As one group becomes ascendant over the other (path dependence), the pragmatist herd will head this way or that. Here is where the importance of the tornado comes in. In tornado situations, the pragmatist is not given time to think, he is swept up in general flow of the market. When there is no tornado, the buy signal is still present but it is not as strong and the pragmatist is given the time to think and to make mistakes. If the conditions are right, the gorilla will still evolve but it will take him longer to reach full strength and, of course, he will face the risk of challengers for a longer time.

There is very good historical evidence that time-to-market is worth a lot of money. Some of you will remember the time when Intel decided to enter the motherboard business. Before then Intel just sold chips but once it consolidated its gorilla power it decided to bring more business in-house, to make a larger part of the whole PC product. This decision meant challenging its own direct customers such as Compaq, Packard Bell and other box makers. Intel, being privy to the

design specs of the new processors, had a significant time advantage over the box makers when it came to the design of the new PCs based on these new processors. In a word, Intel could start the design several months before the box makers could. Using this advantage, Intel started offering these mother boards to the box makers and the weaker players such as Packard Bell eagerly accepted them as it would allow them to improve their competitive position vs. the kings such as Compaq. Compaq denounced the deal and threatened to boycott Intel. A few short months later Compaq capitulated.

DELL, of course, is also tangible proof that time-to-market is money-in-the-bank. Dell's secret is that they shorten the time lag between assembling a machine and the end customer paying for it. This is a different form of the value of time-to-market but just as valid.

Going forward, will time-to-market increase or decrease its influence over the decision between royalties vs. in-house design? I think it's certain that the complexity of chip design and manufacture will increase rather than decrease over time. When MS first produced DOS, that OS was a relatively simple product which a competitor could emulate in a few months. How long will it take a competitor to create a Windoze look alike today? Certainly a lot longer that it would have taken to build a DOS look alike.

Complexity is not just the fact that things get more complicated in some sort of lineal fashion. There comes a time when a design is so complex that no human brain can keep track of the whole. One way to solve the problem is by breaking down the whole into component parts. But even that solution is only partial. Some components become so complex that it is virtually impossible to build a new design from scratch. What you do then is to add new features to old established designs. Most Windoze were not really Operating Systems, they were window dressing for the underlying OS which continued to be DOS! BTW, that is the way nature also builds, by aggregating complex structures in ever more complex and innovative ways.

As SoCs become more complex, the difficulty of the in-house shops in competing with institutions like ARM will increase and therefore I believe that time is on ARM's side. ARM's job at this time is to dissuade customers from buying from the competition and from designing their own. One way to help promote this

objective is to keep royalties low, specially in a time of financial difficulties for the whole market.

Denny

"Demand creates queues. Supply gets rid of them."

[Software Times](#)