

Title: some though on future possibilities for the future of heterogeneous personal and super computation

author: tiago morais morgado **year:** 2016 **independent researcher**

when I personally think about the future of both personal, and super computation, i think that it has, inevitably, to do something with heterogeneous computing. when we think about heterogeneous computing, we think about a paradigm of computation oriented into parallel computing, that can encompass, within a certain extent, things like cpus, apus, gpus, fpgas, asics, and another kinds of processing devices. some years ago, khronos group announced the development of an api called sycl, which was basically bringing OpenCL into an higher level of abstraction, using c++, or at least parts of c++ language and, alongside with it, the possibility of executing parallel computation not only to gpus, but also to another kinds of devices. although there's not an official compiler, some people have managed to develop such a thing. if what you want to do is simply doing gpgpu (general purpose gpu), the only thing you've got to do is to compile the code using something like Vulkan, and creating shaders that can offload your cpu processes to both cpu and gpu. however with sycl (which unfortunately is not a public domain thing, but it will be soon, once it's still in development stage), you can compile your code, and getting it to run on an heterogeneous computation system. with arm technology, and raspberry pi costing around

\$30 it doesn't make sense anymore to think about apus or traditional x86-64 cpus. you can have, with the same level of energy consumption a cluster of a bunch of arm cpus running at a fairly considerable speed. if you bear in mind that a raspberry pi costs around 30€, you can get 10 of those, and getting them connected to each other, building a system with something like 20 cores of arm cpus running at a speed of about 1ghz, so a fair higher amount of processing then with a typical amd or intel cpu, and more scalable towards lower energy consumption and higher performance. if you get a mean using an epci-express bus, to connect a bridge of sli/crossfire cards into this arm system you can get a couple of really fast gpus for a small budget. let's say, for 600€ you can have something like, at the moment, four AMD 460 graphics cards, which is like an hell of a lot of gpu processing. if you hack the raspberry pi circuit, and you add the possibility of getting ssd and sata drives. or simply connecting some external hard drives via modifying the circuit to access ssd and sata hard drives. and if you can get like fpgas and asics binded to the system. you only need to create a kernel, that uses server architecture. to first analyse all the traffic of your system, and properly route crunch it, and translate it to be processed by the desired component in your system. so if you get to create this specific kernel, the only thing you have to do, is to basically build a system adapted to your needs, bearing in mind that you will be able to create machines much faster then you would normally think about. this will be specially interested, specially if you can find a way, within yourself,

to recycle existing hardware and giving it new life. you can for on an instance.
recycle old parts of your old computer, and get them to run straight across
building these new components. more you can upgrade your system, without
making half of the polution you would normally do.