Comments on Claim Construction Disputed Terms
John Wawrzynek
July 8, 2015

1. "at least one of ... and ..."

I didn't want to assume that PACT was consistent in their use of this construct, so I looked though all the listed claims for particular instances. Below I've listed all the instances:

181

(1)
"at least one of: 1) an additional processing unit, ii) a memory device, and iii) a peripheral device; and ..."
(2) none
(3, 4, 5, 6) identical to (1)

106

(8)
"at least one of individual lines, buses, and subbuses" within at least one of a unit including "at least one of a data flow processor (DFP), a field programmable gate array (FPGA), a dynamically programmable gate array (DPGA), and a unit having a multidimensional programmable cell architecture," ...
(19, 20, 21, 23) none

077

(1) identical to 106 (8)

242

(1)
a timer arrangement coupled to at least one of the functional parts
and including "at least one of a state machine and a counter," ...
(9)
none

525

(1, 2, 3) none
(11)
some configurable cells output, as a status signal, "at least one of a
carry over signal, a A<B signal, and a zero signal," ...
(18) none

The "at least one of" construction appears as two different types typified by these examples:
type a) "at least one of a hat, a coat, and an umbrella" (specific things)
type b) "at least one of flowers, candies, and fruit" (categories)

Four of the five unique instances are of type a). Type a) constructions imply one or more from the list because the elements in the list are not groups or categories. Otherwise, it does not make sense grammatically. This interpretation also makes the most sense from a technical standpoint.

For 181, consider column 6 paragraph 60, "In addition to cascading identical units (DFPs, FPGAs, DPGAs), memories and peripherals can also be connected as lower-level SLAVE units (SLAVE) to the bus system described here." This indicates to me that the intent is that one or more of the "additional processing unit, memory devices, or peripheral devices" can be added - not all are needed.

From 106, "at least one of a data flow processor (DFP), a field programable gate array (FPGA) a dynamically programable gate array (DPGA), and a unit having a multidim..." It seems clear that they meant not to include all of the categories. I can't see any technical reason to include one of each. So they should have used "or" instead of "and".

Patent 242 has "at least one of a state machine and a counter". This instance is more difficult to interpret. However, in this case, I believe they intended to include at least one of each. While is it possible to implement a "timer arrangement" with only a state machine, almost all implementations would also include a counter because doing so can dramatically improve the implementation cost. Also, a counter by itself is insufficient to provide the control sequence described in the patent.

Patent 525 claim 11 has "at least one of a carry-over signal, a A<B signal, and a zero signal, ...". The signals listed I would consider to be optional outputs from an arithmetic processing unit. Which ones are actually needed dependent on the context and use of the arithmetic unit. I've seen designs where all are implemented and other designs where only a subset is implemented.

The remaining instance of the "at least one of" construction is of my "type b)".

106 has the example above (like 077) plus has "at least one of individual lines, buses, and subbuses within...". Because of the nature of buses and lines, once you have a plurality of lines you have all three of the listed items: individual lines, buses, and subbuses.
So, in this case the choice of interpretation of the "at least" construct makes no substantial difference in the meaning of the claim.

2. "peripheral"

I would drop the "not essential to the basic operation", because that brings into question what is meant by "basic operation". The use of the word "peripheral" goes back to the days when computer systems were centered around a processor (CPU). Back then, generally, peripherals were considered devices external to the CPU and memory, usually, but not always, supplying input/output. One could also consider an accelerator (e.g. cryptography engine) as a peripheral. If it is essential to the "basic operation" depends - sometime yes sometimes no. If the application of the processor is to provide processing in a secure environments - say, for bank transactions - then one could consider a peripheral cryptography engine as essential. However, if the processor where applied in a different less security sensitive setting, then the cryptography engine could be consider nonessential. Substituting FPGAs for CPUs as the central processing unit, confuses the issue somewhat. In this context, I would consider the FPGA the central processor and consider peripherals as devices external to the FPGA.

In Altera's construction, adding "for example, printers, keyboards, graphic digital converters, disks, and tape drives" does change the meaning of the claim. However, I don't understand what they mean by "graphic digital converter". This could mean "graphical display", or graphics accelerator, such as a graphics processing unit (GPU).

3. "bus system"

In any bus system it is necessary to have a "predefined bus protocol", otherwise no communication could be possible. Because some protocol is always necessary, the presence of a the words "predefined protocol" is implied by the definition of a bus.

The argument that was made by Altera is that the protocol is not flexible. However, in my opinion the construction that they propose doesn't necessarily imply lack of flexibility. One can imagine a bus that has a built in fixed protocol but also includes mechanisms to change the protocol as needed.

From the mechanisms described in 181, it is clear that PACT intended some level of flexibility in the bus interface, however, they don't disclose the design details that would be necessary to fully implement a set of bus protocols.

In the tutorial session PACT claimed that flexibility in protocols came from "switching between a set of state machines each of which implements a different protocol". This approach would work. However, I could not find it described in the patent.

Furthermore, from the invention summary in 181: "The present invention includes a permanent implementation of the bus system control. Memories and peripherals can be connected to the bus system without any special measures." The use of the word "permanent" would lead me to think that they have fixed control.

Because of the way the patent is written (lots of details and very little bigger picture) it is difficult to determine for sure, what level of protocol flexibility is intended or provided. Some bus protocols, SCSI for instance, are fairly complex, and require generation and interpretation of a number of commands between devices.
I think their device could implement these commands but not "without any consideration by the programmer". In other words, protocols such as SCSI could be implemented on their device, but only by using the basic mechanisms described in the patent along with extra logic build from reconfigurable cells. From their description, multiple protocols are not built in. The built-in hardware mechanisms, providing some level of flexibility, seem to involve a fairly simple set of functions that can be used for address generation, multiplexing for shared data lines, and storing data in transit, but do not include higher level control. The control aspects of the protocols would require extra logic. I would assume that logic would come through programming the array.

4. "unit"

In Altera's construction, "complete" seems to go against common use.
Commonly in electronic design, engineers would call a unit a part of a larger whole. However, that's not to say that a designer would divide the system up along arbitrary lines. Likely the unit boundary would be draw around some identifiable sub-function.

I believe that it is inappropriate to quote English Language dictionaries as extrinsic evidence in this case. For instance, from Webster's Third New International Dictionary of the English Language: "(unit: “(2) : a single thing (as a magnitude or number) that constitutes an undivided whole.”); The idea of "undivided" does not make sense in the context of electronic design. Every type of component in an electronic design is divisible (ALUs divide into functions blocks which divide into gates which divide into transistors, etc.)

5a. "interface"

In question is the addition of "without any influence by a programmer". As in 4 above, in the PACT patents it seems that their intent was to make it programmable to some degree. However, the abstract says "The bus system control is predefined and does not require any influence by the programmer". A way to reconcile this would be to take this to mean then that the higher level control is fixed, but some lower level aspects of the interface are programmable. Furthermore, I would not interpret "without any influence by a programmer" to mean "can't be influenced by the programmer", so the addition of these words doesn't substantially change the meaning of the claim.

5b. "interface unit"

same as 5a.

6. "Interface elements configured to share the external bus"

In the Altera construction, the phrase "configured to make competing demands for the external bus" is vague. Is this intended to mean "configured to arbitrate competing demands ..."? I don't see why adding "competing demands" is important, any shared medium must include mechanisms for dealing with potential conflicts.

7. "control/controls/controlling"

I don't see the dispute. Both constructions seem okay, but for electronic systems I would use "direct the operation of."

8a. "compiler"

There is no reason to add "Embodied by a microcontroller ...", as in the Altera construction. From the PACT patents, I see nothing precluding the "compiler" unit from being a piece of dedicated logic.

8b. "configuration unit"

same as above

9a. "configuration memory"

Disputed is "valid" and "so that an operable unit is obtained". I can't think of a common reason for providing a non-valid setting - except to test or otherwise characterize the device. So, I don't see how the addition of these words makes much difference.

9b. "configuration signal(s)"

same as above

9c. "configuration interface"

same as above

10a. "configuration"

If configuration bits set the function and interconnection of a logic unit, then it follows that it must be "programmable" - and vice versa. Both constructions seem to have the same meaning.

"set by the configuration memory bits" is implied. The configuration bits must stored somewhere. Again I see no difference in meaning. I don't think it was the intent of PACT to program through some other means than storing a configuration in a memory.

10b. "configure/ing/ed"

Again in this context, it is implied and both constructions have the same meaning to me.

10c. "configurable",
10d. "reconfigure/s/ing/ed",
10e. "reconfigurable",
10f. "reconfigurable cells",
10g. "Dynamically reconfigure/ing",
10h. "Dynamically reconfigurable cells",
10i. "configurable [at|during] runtime

all same as above