ELMG News Issue 2 July 2006

Welcome to the ELMG News

IN THIS ISSUE
- Partitioning ensures you get it right first time
- Software engineers and FPGAs – compatible or not?
- More Partitioning – people and compliance requirements
- How complex is software? The code complexity model
- From the Compliance FAQ page – Getting started with international compliance
- Compliance services for International Products

If you are not a subscriber become one and receive ELMG News four times a year. Complete the form at www.elmgnz.com/newsletter.html

- Partitioning ensures you get it right first time.

Ever been in a company or on a project where things just seemed to go right? Where the software fitted with the hardware easily and the code was manageable? Or worked with some one who seemed to have fixed your problem before you realised you had it? Or met a marketing person who always specified product that sold really well?

What you saw was probably the result of accurate, timely and suitable partitioning. It is entirely possible that you did not even notice that partitioning was being done as not all companies have realized the value of formal or semi-formal partitioning.

So what is partitioning? It's dividing up the problem or project into smaller parts that can be individually specified and that fit into the organizations strengths and available resources. It is also managing the interfaces between the partitions.

All product developments can (and should) be partitioned. If you consider an embedded system project a suitable partition may be into separately specified hardware and application software. An interface specification that is common to both the hardware and the software then allows the hardware and the software to be developed concurrently. Each of the hardware and software partitions may then be further partitioned by function, complexity or, and this is an important point to realise, by the development team members' individual competence.

In the case of an embedded system the hardware drivers or hardware abstraction layers are effectively the bridge between the hardware and the application. These drivers and hardware abstractions are often badly specified and often not given the development resources needed. The people who create

the drivers are critical to the success of the project. Having management that knows the drivers are critical is the key to getting the project out on time. Managers also need to know that it is possible to design hardware that makes developing the driver very difficult or impossible.

As another example consider a digitally controlled power electronic converter (DCPEC) in which case the partition may not be as clear cut as with an embedded system. Consider partitioning the DCPEC into hardware and software and then allocating the hardware to an electronics engineer and the software to a software engineer without considering whether the hardware or software engineer actually has the skills to perform the underlying task. At this point I can hear people saying that when they work on projects they cannot tell how they should partition the project and so don't. This is an extremely risky approach as without some sort of partition it is not possible to assess and therefore manage the development risk. Any attempt at a partition or reduction of the problem complexity will provide insight to the risk and best project path. It is the process of partitioning the project that generates this insight.

But back to the DCPEC – the typical partition for the project by function may be

1) Power converter hardware
2) Digital controller hardware
3) Software
4) Interface specifications.

The partition by competence in the team may well go along these lines.

Team member 1 - Power converter software control algorithms, EMC, and power converter hardware design.

Team member 2 - Digital controller hardware.

Team member 3 - Software coding.

This is a partition along functional and team member competence lines and best way for the partition to be organized is by the development team itself. Perhaps one of the easiest errors to make as a manager or team leader is to force the partition onto the team rather than let the team partition the project. The FPGA situation below is an example of this.

- FPGA companies push programmable logic design by software engineers.

It seems that the programmable logic and field programmable gate array companies are again suggesting that their products can be used by software engineers. The justification being that this will save on cost. We have seen first hand a company attempt to use their software engineers to design

programmable gate arrays and have seen the large cost of this while helping to pick up the pieces. It showed us first hand that treating hardware as anything but hardware is a very risky approach. As a rule of thumb if an engineer cannot

1) Explain on the spot the mechanism of groundbounce and
2) Explain the advantages of terminating a digital PCB trace and
3) Explain why the common mode needs to be managed when driving differential signals on lengthy lines

Then that engineer should not be let near programmable logic.

- More on Partitions – people and compliance requirements

So why did the place you worked at before get it right more often than not? Some person took care of the partitioning, enforced it through out the project, and let you do your part easily. In other words some person looked for all the rocks and holes in the road and avoided them.

As Sung Tzu said "Victorious warriors win first and then go to war, while defeated warriors go to war first and then seek to win."

The partition is one of the key tools that ensure that projects are successful and as a tool it is used at the beginning of the project. Having an effective partition means your company can contemplate outsourcing part of the project.

Who was it doing the partition if you did not notice a formal or semi-formal documented process? Tell tale signs to spotting them are

1) Everything they did was on time (as they knew when things were required)
2) They took lots of care in what they did and
3) They talked to you informally about what you thought in order to work out what you were capable of so they could fit you into the next partition.

Was this person the R and D manager? Almost definitely not - managers are generally too busy with company management problems. Was it the team leader? Possible but unlikely - they are too busy with scheduling and people problems. Senior or principal engineer on your team? Probably.

Partitions are also extremely useful in designing compliance into a product and ensuring that future product changes can have their compliance managed. This is because it is often the case that compliance requirements apply only to certain parts of the product. Here is a typical plan. Separate the product development into two parts. One part has electrical isolation requirements to be controlled for safety and the other where there are no such requirements. This allows the design of the second part to be changed without effecting the safety certification of the product. Such thinking allows compliance to get onto your product road map. It can even be the case that these partitions become UL recognized components.

EMC compliance is also well served by partitions at a circuit level. The separation of circuits by function and speed leads to cost effective timely EMC compliance.

At this time I can hear some people say – "We have never partitioned our projects". If that is the case then it is probable that your development team could be made significantly more productive by partitioning the product development.

Need some help partitioning your project or want to look at some partitioning methods – contact ELMG at enquiries@elmgnz.com

- How complex is software? And how much will it cost?

Barry Boehm's Constructive Cost Model (COCOMO), which is based on field measurements from software projects, suggests that

$$\text{Effort to create a project} = C \times KLOC^{M}$$

Where the effort is directly translatable to cost, C and M are constants greater than 1 and KLOC is the kilo lines of code. For embedded code developed with reasonable practice C is typically 3.6 and M is 1.4 or greater. So in order to get code working in a cost effective manner make it short.

See www1.jsc.nasa.gov/bu2/COCOMO.html for more information on COCOMO.

- From the ELMG Compliance FAQs page

http://www.elmgnz.com/faq-compliance.html

Q: Are standards deliberately written to be incomprehensible?

A: The first time you read a standard you may conclude that they are. Standards have a specific style, language and body of existing knowledge that is assumed. Generally product specific standards will refer to generic standards and a working knowledge of the generic standard is assumed. The two keys to successfully reading and understanding standards are: accepting that the standard maker has authority (resistance is truly futile); and understanding the intention of the standard, including why it was written.

Contact ELMG at enquiries@elmgnz.com for help with choosing standards, understanding standards and getting compliant product in a cost effective manner.

- Compliance services for International Product Development

If you are looking to start down a path where you will need to know about compliance or develop compliant product then ELMG can help. Our one day Engineer's Introduction to International Compliance training course is the beginning of our Compliance Services Package. This package includes the following services

1) Engineer's Introduction to International Compliance (Seminar 1 Day)
2) A Technology Company Marketing Person's Introduction to International Compliance (Seminar 1 Day)
3) A Board of Director's Introduction to International Product Compliance (Seminar ½ Day)
4) Moving an organization to compliant product developments – change management (Consulting service)
5) Design for compliance – getting the best return on compliance investment (Consulting service)
6) Paths to compliance - Getting compliant product to market competitively (Consulting service)
7) Compliant product development – Turn key development of compliant product

Our most recent Engineer's Introduction Course was run for company in Auckland, New Zealand where it was well received.

Contact ELMG at enquiries@elmgnz.com now for more information on these and other product compliance services.