

Bitvis

Methodology Coordination,
- Is it really needed?

Bitvis

110001011010011110100111011011010011110011

- Independent Design Centre for SW and FPGA (& ASIC)
- 6 designers – from March 1, 2012
 - Expect to increase significantly in 2012/2013
- Building on the Digitas legacy
 - Efficiency/Quality → Methodology
 - Customer partnership relation
- Concept, Specification, Architecture, Implementation, Verification, Test
- Methodology, Reviews, Sparring partner
- Lots of experience on the critical issues
 - Architecture, HW/SW interface, RTOS, Linux, Drivers, Test, ...
 - Design structure, Timing & Clocking, Verification, ...



FPGA Best Practices courses

110001011010011110100111011011010011110011

April 2012

- Bitvis offers the well established course: “FPGA Development Best Practices”
 - Based on the Digitas course previously presented in DK, SE, NO
 - Two full days of learning
 - Major focus on how to improve your FPGA-based projects
 - ◆ Design Structure and Methodology
 - ◆ Verification Structure and Methodology
 - ◆ Focus on Efficiency and Quality improvement
 - ◆ Focus on a Best Practices approach that you can apply today
 - Very good feedback from participants in all countries
- Other courses available on request
 - D&V Reuse, Advanced Verification, Timing and clock domain crossing
 - Basics – to get started in a structured manner.

Agenda

110001011010011110100111011011010011110011

Focus in this presentation primarily on:

- FPGA development
- End quality, end product cost, end development cost & schedule

- Improvement potentials
 - The improvement union and potentials
 - Product delays and deficiencies
 - Management response and actions
 - Roles and Issues
 - Coordination
- Conclusions

Process coordination

Coordination ROI?
(Return On Investments)

110001011010011110100111011011010011110011

Processes

- Sales
- Marketing
- Strategy
- Accountancy
- Finance
- Recruitment
- Purchasing
- Advertising
- Manufacturing
- QA
- Methodology

Focus Parameters

- Management attention
- Customer attention
- Investor attention
- (New) Employee attention
- ISO attention
- Criticality
- Importance (post criticality)
- Effect on income
- Effect on expenses
- Effect on cashflow/capital

Observations and initial questions?

110001011010011110100111011011010011110011

- Most companies have a strong focus on coordinating company critical processes
 - ➔ Sounds really great 😊
- But....
 - ➔ What does this actually mean?
 - ➔ Which processes are critical?
 - ➔ Who is involved when discussing this?
 - ➔ What aspects are they addressing?
 - ➔ Who are they talking to?
 - ➔ From whom do they get their information?
- In the end – Which processes get priority?

Management structure

110001011010011110100111011011010011110011

Most larger companies...

- have external owners and a board
 - with mostly non-technical background
- have perhaps a group mgmnt
 - with mostly non-technical background
- have perhaps a general mgmnt
 - with mostly non-technical background
- have external mgmnt consultants
 - with mostly non-technical background

- ➔ What do they focus on?
- ➔ What info do they get / request ?
- ➔ What do they influence?

**Focus on
ROI !!!
...?**

- ~~CEO/MD~~
- ~~Investments~~
- ~~Marketing~~
- ~~Sales~~
- ~~Other mgmnt~~
- ~~(QA/Processes)~~
- Other mgrs?
- Technical?

Return on investments

110001011010011110100111011011010011110011

- Some critical issues must be taken very well care of
 - Cashflow/Capital, Law and Regulations, (+ Invoicing,)
- For the more general issues
What is the effect of 1 out of 10 failures?
 - In Accountancy?
 - In Finance?
 - In Sales?
 - In Marketing?
 - In Development?
 - In Manufacturing?

My impressions (and experience)

110001011010011110100111011011010011110011

- Many companies fail in determining their critical processes
- Most companies have an unbalanced focus wrt. their needs
- Most companies have insufficient knowledge about what affects product quality and project efficiency the most
- Most companies have a huge potential for improvements
- Most companies invest very little in the major potentials

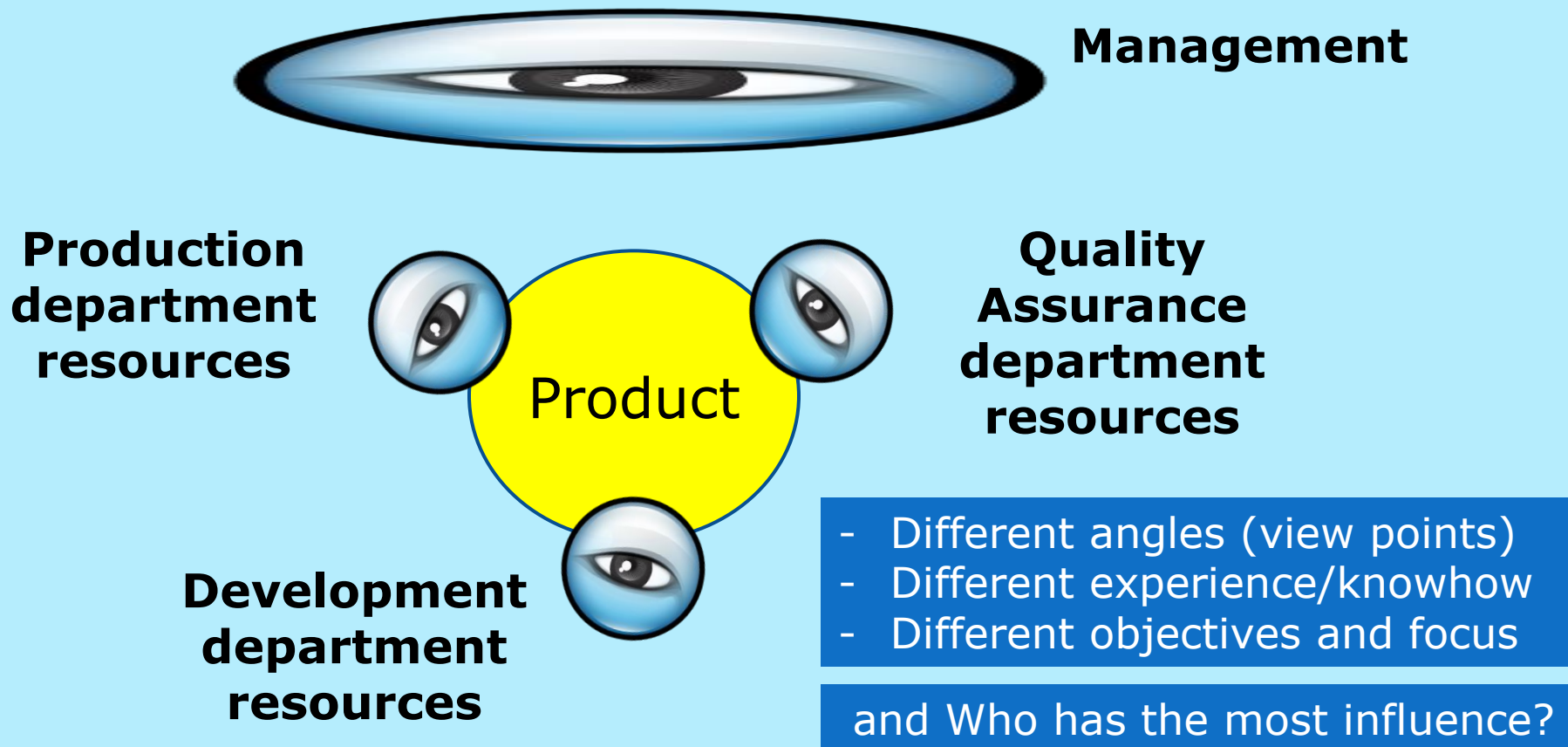
What can we influence?

110001011010011110100111011011010011110011

- Anything related to Product development
 - Development and Prototyping
 - Manufacturing
 - Quality Assurance System
 - Project Management
 - Methodology
- Investments (time and expenses)
- Distribution of investments
- Management focus
- The focus of other related parties

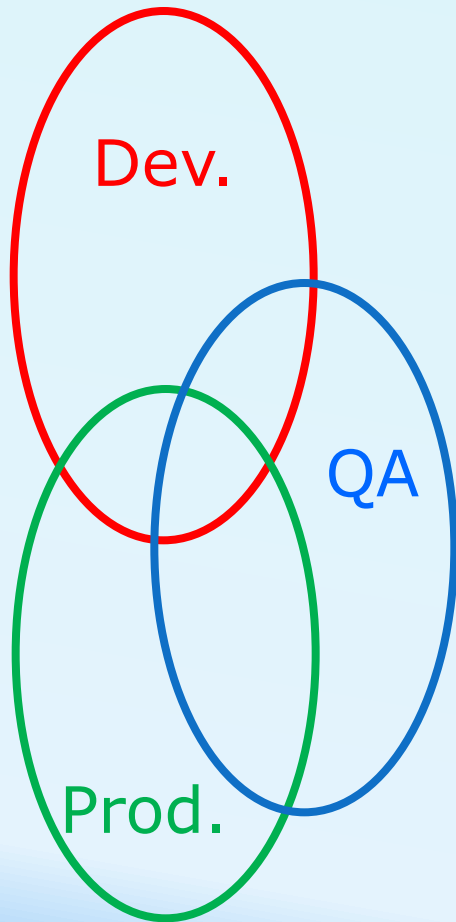
Product development viewing angles

110001011010011110100111011011010011110011



The improvement union

110001011010011110100111011011010011110011



Methodology, Design flow, Architecture, Technical Coordination, Prototyping, Design Knowledge, Pitfall awareness

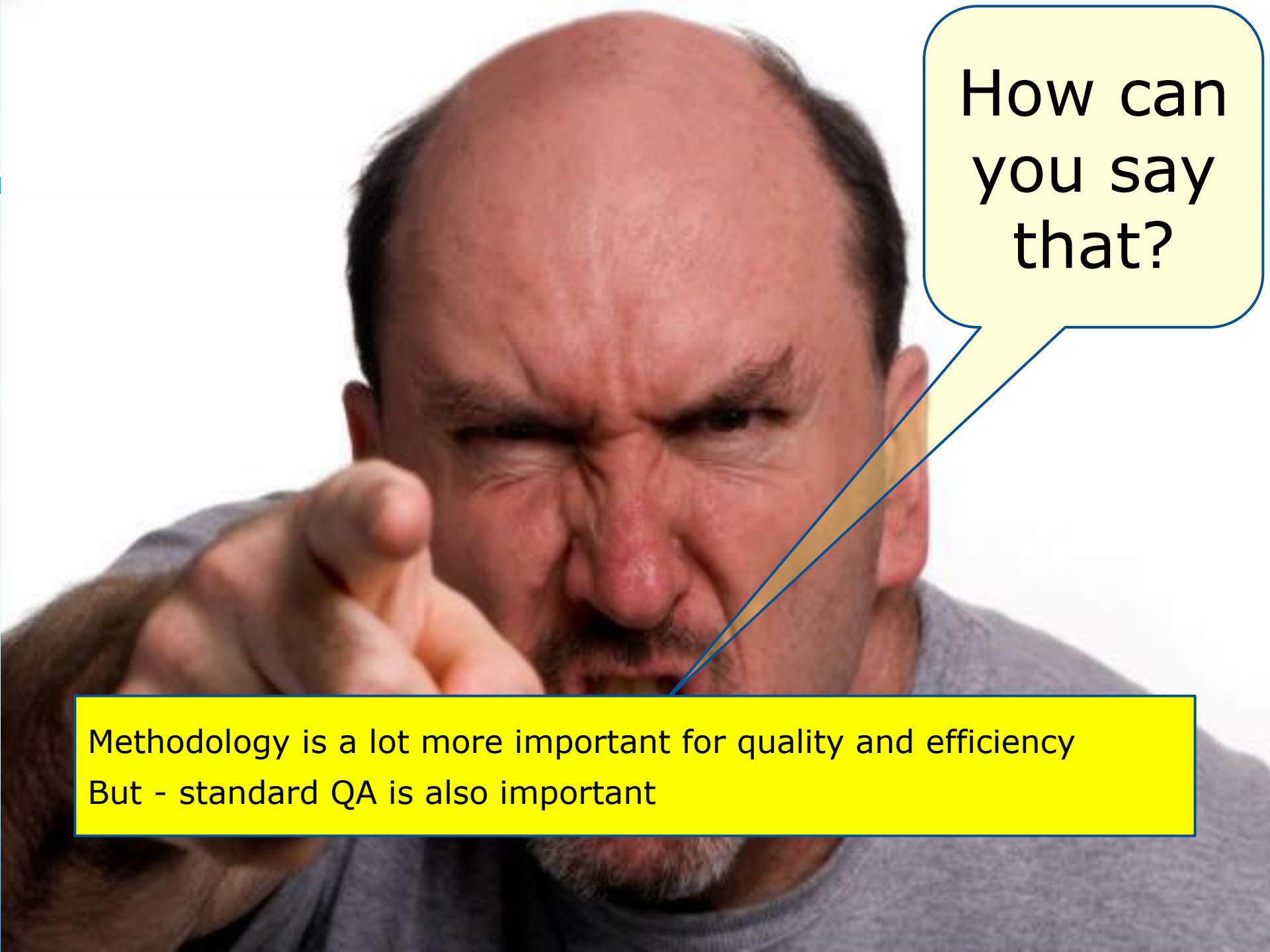


Quality requirements, Non-conformities, Quality and schedule discrepancies, Processes



Subcontractors, Acceptance/Approval, Assembly, Test, Logistics, Engineering, LCC improvements, ...

Focus here primarily on FPGA
(Similar for SW and HW/PCB)

A close-up photograph of a middle-aged man with a serious, almost angry expression. He is pointing his right index finger directly at the camera. His face is wrinkled, and his eyes are narrowed. He has a grey beard and mustache. He is wearing a grey t-shirt. The background is plain white.

How can
you say
that?

Methodology is a lot more important for quality and efficiency
But - standard QA is also important

Product delays – In Norwegian companies

(Real examples - from BP course and FPGA-forum 2008)

110001011010011110100111011011010011110011

- 2 months delay due to sporadic errors (Debug only. Fix took 1 day)
 - 3-4 months delay due to unstable components (debug, total redesign, detailed test)
 - 1 year delay after product was "actual" (patching, testing, patching, testing, patching)
 - 2-4 months delay due to lack of implementation
 - 3-6 months delay due to bug after bug
 - 6 months delay due to terrible state machine
 - "Standard problems"
 - 1-3 months delay due to HW/SW/FPGA
 - 1-6 months delay due to quality of design
- Most of these companies had excellent QA systems and a strong focus on critical processes.
- Most of these issues were caused by good designers or assumed good designers
- None of these companies had a good development methodology.
- A good development methodology would have reduced the total delay to a fraction

Product deficiencies – In Norwegian companies

(Real examples - from BP course and FPGA-forum 2008)

110001011010011110100111011011010011110011

- Communication switch with lots of
 - Did not appear in the lab (or was ignored)
 - Lower throughput. Unhappy customer
- System increasingly failing after a while
 - Multiple functional errors and timing errors
 - More than a year to debug
 - Expensive down time. Unhappy customer
- Customer's application SW not working
 - Starting fine, but problems appearing
 - Did not happen in the lab. Not tested
 - Delayed customer's product
- (Telecom volume product with sporadic failures)
 - Deadlock in hardware → Had to re-design

All of these companies had excellent QA systems and a strong focus on critical processes.

Most of these issues were caused by good designers or assumed good designers

Non of these companies had a good development methodology.

A good development methodology would - with a very high probability - have prevented most of these issues

“Normal” management response

110001011010011110100111011011010011110011

- Involvement depends on criticality
 - and frequency of incidents(For mgmnt level, involvement & follow-up)
- They discuss the issues, but
 - Often with insufficient data from the organisation
 - Often with insufficient group knowledge
 - Initiate actions based on this....
- They hopefully ask for the root cause of problem
 - But often difficult to pinpoint
 - Difficult to see the real root cause
 - Difficult to determine what might have prevented the problem

Typical initiated actions

110001011010011110100111011011010011110011

- Report through QA system (iso req.)
- Increased focus from mgmnt – and QA:
 - Check whether defined processes have been followed
 - ◆ Procedures? Documents? Reviews?
 - Evaluate planning, risk analysis, resource allocation, project management
 - Evaluate and improve processes
 - ◆ In particular on project management....
 - Evaluate potential missing competence
 - State the need for more development knowledge
 - State the need for professional project managers

Project management

110001011010011110100111011011010011110011

■ **Project administration** (Seen from Mgmt and QA)

A very important role, but maybe slightly exaggerated...?

- Some critical issues to handle
 - But most of these can be “easily” followed up
(and properly handled by good procedures and checklists)
- An important effect on schedule, quality and efficiency
 - But limited pot. damage due to other control mechanisms

Please note with respect to “exaggerated role”

- Not at all exaggerated wrt absolute focus from management
- **Only** exaggerated wrt. focus **relative to**
implementatation coordination and development methodology

Implementation coordination

(Project development methodology coordination)

110001011010011110100111011011010011110011

- Needed at all levels
 - Product, Subsystems, SW, HW, FPGA
 - And between parts at the same level
- Typical FPGA/SW issues:
 - Functionality, Protocols, Synchronization
- Designers do not normally coordinate
 - They have a single focus and target
 - Good designers are not necessarily better
- Verification also needs coordination
- Coordination only works with given authority
- Many projects are suffering a lot from lack of implementation coordination

Some typical project problems

110001011010011110100111011011010011110011

& What takes best care of the following problems?

- Missing validation/test of spec. details
- Mismatches in integration at all levels
- Inefficient resource utilisation
- Insufficient progress (e.g. in module)
 - In particular at the end...
- Slow integration
- Slow progress at the end of design
- Slow progress at the end of verification
- Bugs suddenly popping up at the end
 - Or even worse – in the field

Some “non-problem” potentials

110001011010011110100111011011010011110011

- There is a huge potential for improvements not seen by the QA system/department or management
 - Not related to procedure deviations
 - Not related to schedule deviations
 - Not related to final product bugs
- Typically only seen by structured developers
 - Better architecture at all levels
 - Tight implementation coordination
 - More structured handling of typical pitfalls
 - Better verification
 - More structured verification
 - Good reviews
 - Improve competence at all levels on general development

To coordinate or Not to coordinate?

110001011010011110100111011011010011110011

- Meth. is probably the most complex process to coordinate
- Managers often sceptic to methodology coordination
 - Afraid it affects the personal initiative?
 - Afraid it is negative for experienced designers?
 - Afraid it limits the creativity – with too many rules and proc.
 - Afraid it limits the availability of valuable resources

Methodology coordination is critical

- ➔ To improve efficiency and quality in development project
- ➔ To properly structure the improvement work

Conclusions

110001011010011110100111011011010011110011

Some ballpark numbers based on experience and input – taking as an example a budgeted 7000 hour FPGA development project

A good QA system may result in

- Not wasting 200 MH
- Avoiding some stupid pitfalls

A good Pr. Mgmt. may result in

- Not wasting 400 MH
- Avoiding some stupid pitfalls

A good Impl. Coord may result in

- Not wasting 400 MH
- Saving 400 MH
- Avoiding many stupid pitfalls

A good Dev. Meth. may result in

- Not wasting 1000 MH
- Saving 1000 MH
- Avoiding lots of stupid pitfalls

A question of ROI
A question of knowledge & awareness at all levels

Your partner for SW and FPGA (& ASIC)

Methodology Coordination,
- Is it really needed?

→ Only for Quality, Time & Efficiency