INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

DESIGN PATTERNS IN MICROTECHNOLOGY

Albert Albers, Tobias Deigendesch, Tarak Turki
IPEK – Institut für Produktentwicklung, Universität Karlsruhe (TH)

ABSTRACT
System design in microtechnology requires in-depth knowledge about the manufacturing processes. The known models of micro-specific design comprise the integration of production as a main aspect. Design rules are an established means of support for design constraints from production. However, there is design and production knowledge that cannot be formulated by design rules. The authors propose the application of design patterns – a very successful approach in software engineering – in microtechnology for representing knowledge about successful realized design solutions. Basically, a pattern describes the given context, in which the pattern is supposed to be applied, the frequently occurring design problem and the corresponding abstract solution. The solution part is generic and abstract to prevent the prescription of a concrete solution. A method for pattern derivation by correlating function and shape is proposed. The micro-specific patterns are represented in a wiki-system.

Keywords: pattern languages, design patterns, microtechnology, knowledge management, wiki

INTRODUCTION
Today's products follow megatrends such as integration and miniaturization. In this context microsystems are increasingly introduced to markets. Most successful applications e.g. ink jet nozzles, accelerometers or micro-mirror arrays are monolithic two-dimensional silicon-based microelectromechanical systems (MEMS). Miniaturization is also a trend in mechanical engineering. The required three-dimensional mechanical structures e.g. microgears need changed production and design processes, e.g. micro milling or replication techniques like injection molding (tool-based microtechnology). This restricted number of possible processes results in geometrical and procedural restrictions that have to be considered in early stages of design [1,2]. Thus, micro-specific design requires in-depth knowledge of manufacturing and assembly technology. A successful means to ensure the consideration of these restrictions in design are design rules [3,4,5]. However, many aspects of design relevant knowledge cannot be formulated in and represented by design rules.
Design in microtechnology does not require types of knowledge representation systems that are substantially different to those being used in general design. However, to cope with the design restrictions being imposed by production, successful design in microtechnology depends on a well integrated and easy to use knowledge representation. Design patterns are an approach for representation of problem- and solution-oriented knowledge, which first was introduced to architecture by Alexander [6]. The Alexandrian patterns could be successfully applied in software engineering [7] and first approaches were made by Salustri [8] to introduce pattern language to engineering design.
The authors expect the concept of pattern languages to be promising for microtechnology. Thus the research question is, in what way the concepts of patter languages can be adapted to and applied in microtechnology.
The estimated benefits of a micro-specific pattern language are an easy way of knowledge sharing, improved communication among designers themselves and production specialists by a common vocabulary and easier introduction for novel designers to the matter. Further on, reduced design cycles due to optimized knowledge acquisition, representation and communication are seen as chances of the proposed approach.
The application of design patterns in other disciplines is identified. Based on that identification and on the requirements to design in microtechnology, a methodology for design patterns is derivated theoretically and exemplified. Validation has begun in single cases and is subject to future work.

The present contribution describes the specific aspects of design in tool-based microtechnology. Then it introduces the concept of patterns and wiki systems as a tool for computer-supported representation of design knowledge. Based on that background, the format of design patterns for microtechnology, their derivation, representation and integration into the design process are brought into focus.

BACKGROUND

Design in Microtechnology

Microtechnology is relatively new compared to mechanical engineering and does not offer such a multitude of reliable production techniques. Basically, there are three different approaches of manufacturing a microsystem. By adapting the processes for integrated circuits of microelectronics microelectromechanical systems (MEMS) are made. The second way is LIGA, a German acronym that stands for the basic process steps of this technology: lithography, electroplating and molding [9]. The third way is miniaturization of macroscopic technologies, such as milling of molds and subsequent replication by molding. All three have in common, that the number of applicable manufacturing techniques is restricted to few processes which only allow realization of few defined geometries with few defined materials in a defined sequence of process steps. Therefore the designer has to be much more aware of the possibilities and restrictions of production technology.

The micromachining of MEMS and the LIGA process are called mask-based technologies, because relevant structuring steps are made by exposure to radiation through a patterned mask similar to microelectronics.

As the production processes of mechanical microsystems being made of silicon are adapted from microelectronics, Hahn [10] adjusted the Y-model of microelectronics [11] which consists of tripartite representation of design. The Y-model represents functional (desired function), structural (components, connections) and physical (layout planning, masks geometries) design, i.e. the aspects of function, realization and production. The adaptation comprises the three levels of system, component and structure. Shaping MEMS is a product-specific sequence of iterative steps of coating and exposing to radiation, etching or deposing material. Thus, this product-specific manufacturing sequence has to be developed parallel to the product design. All of those three levels have to be considered when developing the manufacturing and vice versa. In another approach, the such-called circle-model [12], four stages of layout design, process development, verification and process modification are passed through repeatedly. Also the “pretzel model” [13] shows the parallelism of developing functional design and production processing sequence. The challenge is to know and consider all the restrictions deriving from the single manufacturing steps and their sequence. In microelectronics and MEMS micromachining design rules and knowledge based engineering are introduced [14,15,16]. Also for designing microsystems in LIGA technology, design rules are applied as a methodological tool of support [3,18,19].

Mechanical micromachining is a tool-based technology that derives from miniaturization of macroscopic production technologies, such as separative (e.g. milling), erosive (e.g. electro discharge machining) or laser ablative processes for structuring original parts or master models. These processes are also used for structuring molds, in which thermoplastic materials but also feedstock with metallic or ceramic powders are injection molded. One unique feature of the tool-based technology is the possibility of manufacturing complex three-dimensional or even free-form geometries. Despite the freedom of the design being offered by this method of micromachining in contrast to the mask-based machining approaches, there are technological conditions and restrictions that have to be stringently considered when designing. The restrictions are e.g. minimum edge radius, aspect ratios or minimum wall thicknesses. These result in a strong orientation on what is producible and – in contrast to macroscopic product design, which is driven by market requirements – in a technology-driven design approach. For designing in a way compatible to production, specific knowledge from the subsequent product life cycle stages has to be made available to the designer. Thus, also tool-based micromaching is supported by knowledge-based engineering [20] and design rules [2,4]. These design rules are detailed instructions for a micro-compatible part design. They are derived from technological requirements and restrictions and represented in databases in general.

Design Patterns

Design patterns are a means of representing knowledge about solutions in a way that is generic and abstract enough to leave the designer freedom of creativity.

The concept of representing frequently recurring solutions in the form of patterns and pattern languages is developed by a group around the architecture theorist Alexander in the seventies [6,21]. For towns and buildings they identify similarities and derivate a total of 253 interconnected patterns. Characteristic of these patterns is the format which basically comprises the descriptions of context, problem and solution. Listing superordinate and subordinate patterns enables to represent the interconnection between the patterns in terms of a language. The description of problem and solution allows the user to access and judge the contents in an easy way and facilitates changes and adaptations without changing central contents.

Alexander et al. define a pattern as follows:

„Each pattern describes a problem which occurs in over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.“ [6]

For the sake of simplicity and clarity, each pattern has the same format. An Alexandrian ones begin with name and an illustration being followed by an introducing paragraph describing the context by depicting the relation to superordinate patterns. The central problem statement is short and emphasized, before an in-depth description of the problem including its background and relevance. Then the solution is described, also in an emphasized style. The solution statement is supposed to be an instruction, how the problem can be resolved by appropriate material or social relations. All relevant elements and relations concerning the problem resolution are specified in a generic and abstract way. The solution statement does not impose any precise solution to the user, but imparts knowledge about all relevant solution elements. These elements are all of those aspects that are identified within all successful solutions as invariant existing. Sketches illustrate the main elements of the solution. The format concludes by pointing out subordinate patterns that support the fulfillment of the solution. A superordinate pattern is only completely realized, if the subordinate patterns are implemented as well. The patterns feature a three-stage assessment of the significance of the problem resolving principle. Maximum significance means all realized solutions have the same invariant features. No identified significance classifies the solution as one possible idea for realization among many others.

It takes a decade until the architectural patterns break new ground in the field of software development. The Alexandrian patterns are taken on by Beck and Cunningham [22] in the form of a small language with few patterns. A software-specific pattern format was developed by a group of conference participants with the objective of preparing a handbook for software architecture [23]. These results shall be the foundation for the 1995 published software pattern language [7] which makes patterns a succeeding concept within the software developer community.

The idea of participative usage of patterns leads to the development of the web-based first wiki [24] for the purpose of an open content management system (CMS), in which all participants can use and improve the stored patterns. The application of patterns for representation of solution-related knowledge spreads to adjacent fields. Pattern languages are developed for the design of human-computer interaction [25], process improvement [26], technology introduction [27], project management [28] or even for the review of conference contributions [29]. All of those follow the Alexandrian format plus specific items like examples of program code.

In the field of engineering design, Salustri [8] introduced design patterns and represents those in a wiki system [30]. Feldhusen and Bungert also apply the pattern concept for integration into product lifecycle management [31].

Wiki Systems

A central challenge today is the ability to control the complexity of innovation and the product development process. A change from time-to-market generation to research and development productivity generation signifies since 2000 the product development process [32]. There are new approaches such as integration of knowledge management and high flexibility. A tool that supports knowledge management are wiki-systems.

Wikis are software tools that work mostly web-based. Wikis are made up of several pages, which are linked together. Contents of different pages in one wiki-system can be created and edited from

multiple users separately and simultaneously. The first web-based wiki was designed in the nineties as a knowledge management tool in the context of the application of design pattern in the field of software engineering [24]. End of the nineties the best-known wiki application was developed: The free encyclopedia Wikipedia. Since several years wikis are more and more used in innovation and product development processes [33]. A survey concerning wikis in business applications concludes that 58% of the surveyed companies use at least one wiki and further 18% plan to use a wiki [34].
Wikis, as a collaborative knowledge management tool allow cooperation of several users. Users can create, modify, update or delete wiki pages over the internet or intranet independent of time and place. The wiki concept allows a form of discussion forum. Contributions can be deposited with own signature. Old contributions are not deleted but they are only made invisible, which automatically creates a record that enables to continuously log and – if necessary – to revoke changes of content. Wiki pages can be used for coordination and planning. Some wikis allow the integration of a project management system into the wiki. It is very simple to create, to edit or to change wiki pages. In general it requires no special knowledge of programming languages like HTML. Furthermore wiki systems are inexpensive to acquire, to set up and to maintain. Wiki pages will be created at a given namespace directly through the browser address bar. Namespaces are designed to categorize pages hierarchically and to create a classification. Namespaces cannot be created alone. They must be always completed with a page. After entering the syntax in the browser's address bar and confirming, the page is created. Access control for user and user groups can be assigned to pages and namespaces. Wikis allow users to find information again supporting different retrieval strategies. Information can be retrieved by browsing the wiki, using a table of contents, by full-text search functions or even by semantic search [17]. Tables, images and files can easily be inserted. Changes in content on a page can be subscribed by interested users to be kept up-to-date. If the page will be changed, the subscriber receives a notification by mail.

DESIGN PATTERNS IN MICROTECHNOLOGY

As presented in the background section design in microtechnology requires mandatorily knowledge especially from life cycle stages being subsequent design. Compared to general design, design in microtechnology does not require different types knowledge representation systems. But to cope with the strong design restrictions being imposed by production (e.g. molding, handling and assembly) quality assurances (e.g. measurement techniques), successful design in microtechnology depends on a well integrated and easy to use knowledge representation. A common representation of technological knowledge from production are design rules. A concept for further reducing the effort and the number of iterations between design and production development is the application of knowledge from successful former solutions. The concept of design patterns allows representing this type of knowledge in a structured way. The content being structured in a standardized format is easy to access. Communication is supported by sharing the same language in the form of patterns. And novel designers get faster an overview on what is realizable with the given technology. This section specifies the integration of patterns into the design process, methods of derivation and the format of the design patterns. It closes with computer-supported representation.

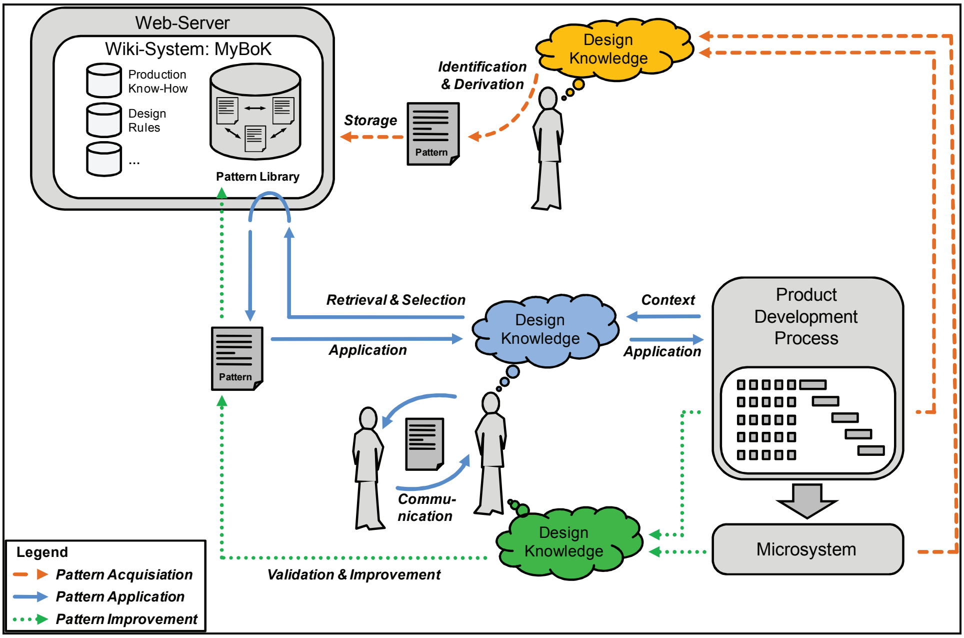

Figure 1: a pattern's life

A Pattern's Life

Basically, a pattern's life has three different situations: a pattern is acquired, applied and improved. When deciding for applying design patterns as a means of support, an experienced designer, better an entire team, has to identify and derive possible patterns. For this knowledge acquisition experience from former development projects is required in order to be able to identify what patterns led to successfully designed product features (cp. Figure 1). Having identified the successful solution elements, problem statement and the context have to be described. Finally the pattern is stored in some knowledge representation system (here: MyBoK wiki system).

During application in a development project, the designer analysis the target system and extracts the design task and its corresponding context. The computer-supported knowledge representation system enables him to retrieve and select possible design patterns. Having found an appropriate set of patterns, the given generic abstract solutions have to be adapted to the concrete design task and to the conditions and constraints given by the target system. Simultaneously, the patterns support communication among designers and other stake holders within the development process. The explicit knowledge enables to discuss with common-sense terms, because the patterns with their standardized structure allow an alignment of individual mental models. When designing, the patterns temper creativity [30] by describing the beneficial and unfavorable aspects of a solution within that context. Concurrently the same patterns demand to be creative in order to adapt the generic and abstract solution to the conditions and constraints at that design task and in the corresponding target system.

Having applied the pattern in a development project, there is a chance to use the gained knowledge to validate and improve the pattern. This participative application of patterns interweaves design activity and enterprise learning.

Derivation of Patterns

Basically, patterns are derived by groups of experts that share experience and knowledge in the same context, e.g. microsystem technology. They have the greatest potential for identification of recurring problems and their solutions. The identification process can be organized with one team at the same place and time synchronously, e.g. by intuitive methods of creativity like brainstorming. It also can be organized locally and temporally distributed in an asynchronous communication way. This is called opportunistic approach. The latter is the constant preoccupation with the identification of new design patterns, which requires the continuous availability of a system for sharing knowledge, respectively patterns, e.g. a web-based wiki.

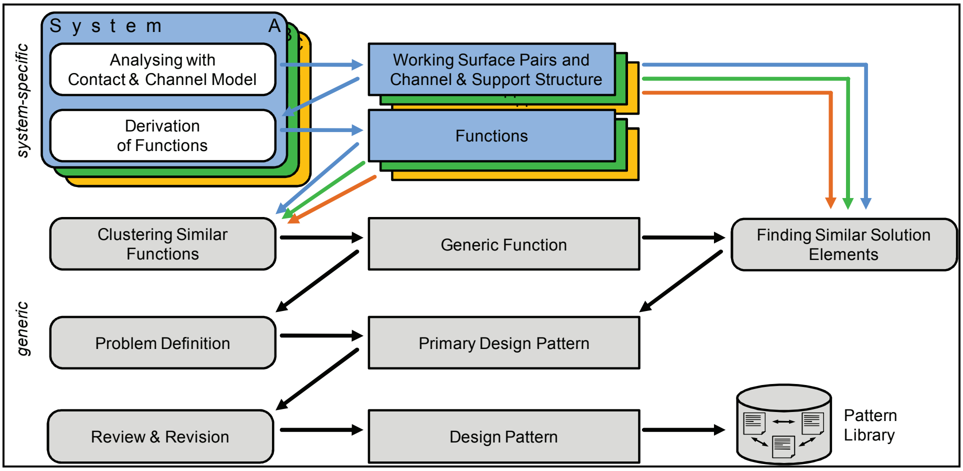

Figure 2: concept for derivation of design patterns

A more systematic approach for identification of possible design patterns is analysis of several realized design solutions regarding the correlation of design and function by an appropriate model (cp. Figure 2). On possible way of modeling offers the Contact & Channel Model (C&CM) [35], which relates technical functions and shape in a single model. This is achieved through the localization of functions on the shape in at least two working surface pairs (WSP) and a connecting channel and support structure (CSS) [35,39]. To identify new design patterns, analysis by C&CM is first applied to real systems within one context. WSP and CSS are identified and assigned to functions. Results are the relations of shape and function within these systems. In a subsequent step, similar functions are clustered for identification of a generic function (e.g. bear shaft). Based on this generic function, similar solution elements are extracted from the original analyzed systems. The similar solution elements and the corresponding problem being solved by those form a prototype design pattern in this context. During review and revision the prototype pattern is checked for fulfillment of characteristic patterns properties, such as generativity or recurrence (cp. [7,6,30]). Finally, the revised patterns are integrated into an appropriate pattern library or catalogue.
For the creation of design patterns, these approaches were applied by the authors in a first case study on micro-specific design patterns.
As an example, two microsystems are analyzed. The first one is a micro pump, which can deliver and dose mediums. The second one is a combination of a micro turbine and a planetary gear, which can convert air pressure in torque. CSS and WSP are indentified, elementary functions derived and two C&C-models created. The functions are listed and clustered according to similarity. There are system-specific functions e.g. to deliver a medium, which are not considered farther. Similar functions in the two different systems e.g. axial and radial fixation are analyzed for invariant features. These invariant features form the generic solution to the corresponding functional problem. The system context, the functional problem and the generic invariant features as solution build together a primary design pattern. This has to be subject to further steps of improvement and validation. Of course, beside this example, more than two systems have to be considered to achieve a significant pattern.

The Format of Design Patterns

For the purpose of manageability and clarity, each design pattern has the same form. A balance between information content and clearness rather user friendliness is assured in a design pattern form. There are several design pattern forms, which are only negligible different from each other [7,6,36]. To represent design patterns in microtechnology, the authors propose the following type of form:
Name: a descriptive name, which captures as much as possible information about the design pattern.

Context: the context describes, where and in what way the problem occurs. Superordinate patterns are listed if appropriate.
Problem: the problem part contains a description of the undesirable situation and of the requirements to be met. An example from product development can clarify the problem.
Solution: the solution part describes how the problem will be resolved. Examples, pictures and diagrams help here.
Consequences: the positive and negative effects of pattern application are noted here.
Chances & Risks: the estimated chances and risks are noted here.
Related patterns: Referencing on similar and related or subordinate design pattern facilitates the work within a pattern language.
Examples: Referencing on successful applications of the pattern facilitates the understanding by the user and reinforces the pattern. Ideally, three different examples should be given, each as different as possible from the others.
References: Sources give the design pattern more credibility. Users can gain in-depth knowledge about the background and can improve the design pattern.
Figures 3 and 5 give examples of the main parts of a pattern, the authors derived for microtechnology.

Computer-supported Representation

The computer supported representation of knowledge is introduced to a cooperative research center working on primary shaping of microcomponents. Several research projects contribute to the conjoint objective of understanding and improving development, production and quality assurance of primary-shaped microcomponents from metallic and ceramic materials. The research center is set up similar to a producing company, to a microfactory: There are divisions for design and material development, prototyping, tool making, production and quality assurance. As explicated in the background section, designers need in-depth knowledge from all the other divisions to be able to develop a producible design. For this reason of sharing knowledge and for improved cooperation within the collaborative research center, a platform for knowledge representation is necessary. Amongst others, the main requirements for such a system are accessibility via internet, access control for namespaces, simple administration and user management, simple information retrieval in namespaces, simple navigation through namespaces, simple syntax and editing and low cost.
The chosen system is based on the dokuwiki engine [37] and is called Micro (μ) Book of Knowledge (MyBoK, cp. Figures 4 and 5) [5,38]. This wiki features very simple syntax, interwiki links, upload and integration of files and images, toolbars and access keys, fulltext search, page recovery, and the possibility of extending features by software plugins. The software is open source and free.

CONCLUDING REMARKS

Microsystem technology is a knowledge-intensive and requires tools for support [1-5,20,22]. The approach of design patterns is successful in software engineering [7] and seems to be promising for engineering design [8,30,31].
The authors propose patterns for application in microtechnology. Beside design patterns, design knowledge can be represented in several forms, e.g. design principles or guidelines. The emergent benefit can only be achieved, if methods and tools are not used in an isolated way but constitute an integrated methodological framework. The proposed set of methods for derivation of patterns, the characteristic pattern format as well as the computer-supported representation allow the integration in typical design activities and processes.
The acquisition, application and improvement of patterns in a product development context are shown. An approach based on the analysis of the relation of function and shape (C&CM [35]) is proposed for supporting the identification of design patterns. The patterns are stored, shared, cultivated and administered in an appropriate knowledge representation system, i.e. a wiki-system.
Based on the experience of working with patterns in microtechnology, the pattern approach has to further improved with respect to derivation, integration into the design process and validation. The “living” patterns are going to be improved continuously during application.

Name:	Non-functional surface for injection and ejection
Context:	In microtechnology, the same principles and solution statements cannot be used as in the macroworld due to changed conditions. Many technologies cannot be used in smaller dimensions, or with significant restrictions on the production of microparts or systems. The primary-shaping technology is an economic alternative for the manufacture of microcomponents.
Problem:	The casting equipment has mostly macrodimensions. Knockout pins are usually made of hardened steel. When the microcomponent is ejected from the mold, the pin leaves a mark, which is compared to macrocomponents relatively bigger. This leads to malfunction in case the affected surface is a functional one. The injection point (gate) is in macrodimensions negligible, but it is not in microsystems. Injection and subsequent ejection result in elevated or rough gate marks. A subsequent processing of microcomponents (e.g. sanding) is difficult to be realized, because of the minimal dimensions and the materials used (e.g. ceramics).
Solution:	These problems should be early identified and should be avoided the latest by adding special areas to the design, that are reserved to injection or ejection. The part shall be designed in a way, that the knockout pin mark area or the gate mark area are non-functional surfaces. The desired functions will not be impaired. Even in the case of a local damage, function losses can be avoided.
Consequences:	The design provides additional surfaces making components bigger.
Chances & Risks:	Chances: Production and assembly errors or malfunction during application are reduced. Design requires less iterations. Risks: Reduced potential of miniaturization.

Figure 3: Extract from pattern in micro technology ("Non-functional surface for injection and ejection")

Figure 4: Micro Book of Knowledge (MyBoK): view mode (engine: dokuwiki [37])

Non-functional surface for injection and ejection

Inhaltsverzeichnis

Context [Bearbeiten]

In microtechnology, the same principles and solution statements cannot be used as in the macroworld due to changed conditions. Many technologies cannot be used in smaller dimensions, or with significant restrictions on the production of microparts or systems. The primary-shaping technology is an economic alternative for the manufacture of microcomponents.

Problem [Bearbeiten]

The casting equipment has mostly macrodimensions. Knockout pins are usually made of hardened steel. When the microcomponent is ejected from the mold, the pin leaves a mark, which is compared to macrocomponents relatively bigger. This leads to malfunction in case the affected surface is a functional one. The injection point (gate) is in macrodimensions negligible, but it is not in microsystems. Injection and subsequent ejection result in elevated or rough gate marks. A subsequent processing of microcomponents (e.g. sanding) is difficult to be realized, because of the minimal dimensions and the materials used (e.g. ceramics).

Solution [Bearbeiten]

These problems should be early identified and should be avoided the latest by adding special areas to the design, that are reserved to injection or ejection. The part shall be designed in a way, that the knockout pin mark area or the gate mark area are non-functional surfaces. The desired functions will not be impaired. Even in the case of a local damage, function losses can be avoided.

Consequences [Bearbeiten]

The design provides additional surfaces making components bigger.

Chances & Risks [Bearbeiten]

Chances [Bearbeiten]

- Production and assembly errors or malfunction during application are reduced.
- Design requires less iterations.

Risks [Bearbeiten]

- Reduced potential of miniaturization.

Related patterns [Bearbeiten]

- Loop-in allows to overcome deviations deut to marks.

Examples [Bearbeiten]

- Mikrodispenser

the gate mark area are non-functional surfaces

Mikrodispenser: Detail A: the gate mark , Detail B: the knockout pin mark area

Figure 5: Micro Book of Knowledge (MyBoK): Design Pattern: Non-functional surface for injection and ejection

REFERENCES

[1] Albers A., Marz J. Knowledge-based design environment for primary shaped micro parts. *5th International Workshop on High Aspect Ratio Micro-Structure Technology HARMST 03*, Monteray, California, USA, June 15-17, 2003

[2] Albers A., Marz J. Restrictions of production engineering on micro-specific product development. *Microsystem Technologies* 2004 (10) p.205-210

[3] Leßmöllmann C. *Fertigungsgerechte Gestaltung von Mikrostrukturen für die LIGA-Technik*. University of Karlsruhe (TH), 1992, doctoral thesis

[4] Marz J. *Mikrospezifischer Produktentwicklungsprozess (µPEP) für werkzeuggebundene*

Mikrotechniken. In: Albers A. (ed) Forschungsberichte des Instituts für Produktentwicklung, Vol.17 University of Karlsruhe (TH), 2005, doctoral thesis
[5] Albers A. Burkardt N. and Deigendesch T. Knowledge-based Support of Decision Making at the Example of Microtechnology. *Journal of Automation, Mobile Robotics & Intelligent Systems JAMRIS*, 2007 (1) 4, p.16-20
[6] Alexander C., Ishikawa S., Silverstein M., Jacobson M., Fiksdahl-King I. and Angel, S. *A Pattern Language: Towns, Buildings, Construction*, 1977 (Oxford University Press, New York)
[7] Gamma E., Helm R., Johnson R. and Vlissides J. *Design Patterns. Elements of Reusable Object-Oriented Software*, 1995 (Addison Wesley)
[8] Salustri F. A. Using Design Patterns to Promote Multidisciplinary Design. *CSME Int. Conf. on Multi-disciplinary Design Engineering*, 2001, Montreal, Canada
[9] Menz W., Mohr J. and Paul O. *Microsystem Technology*, 2000 (Wiley-VCH, Weinheim)
[10] Hahn K. *Methoden und Werkzeuge zur fertigungsnahen Entwurfsverifikation in der Mikrotechnik*. In: Fortschritt-Berichte VDI, 1999 (20) 286 (VDI-Verlag, Düsseldorf, Germany).
[11] Gajski D. and Kuhn R. New VLSI-Tools. *Computer* 1983 (16) 12, pp.11-14.
[12] Brück R. and Schumer C. INTERLIDO – Web-basierte Werkzeuge für den Mikrostrukturentwurf. *Workshop Multimedia und Mikrotechnik*, 1998 (1), Lüdinghausen, Germany, pp.82-102
[13] Wagener A. and Hahn K. Eine Entwurfsmethodik für die Mikrosystemtechnik und Post-CMOS. *Austrochip 2003*, Linz, Austria
[14] Mead C. and Conway L. *Introduction to VLSI Systems*. 2nd ed., 1980 (Addison-Wesley, Reading, USA)
[15] Rabaey J. M., Chandrakasan A. and Nikolic B. *Digital integrated circuits—a design perspective*. 2nd ed., 2003 (Prentice Hall, New Jersey, USA)
[16] Scherer K. P., Buchberger P., Eggert H., Stiller P. and Stucky U. Knowledge based support for manufacturing of microstructures. *Microsystem Technologies* 1996 (2) pp.167-170
[17] Albers A., Sauter Ch., Meboldt M. and Deigendesch T. Semantic Wikis in Knowledge Management for Multidisciplinary Product Development. *DAAAM Symposium*, 24-27th October 2007, Zadar, Croatia
[18] Buchberger P., Eggert H. and Scherer, K.-P. MIDAS: *Ein wissensbasiertes System zur Unterstützung des fertigungsfertigungsgerechten Entwurfs von LIGA-Mikrostrukturen*. Wissenschaftliche Berichte 5843, 1997 (Forschungszentrum Karlsruhe, Germany)
[19] Solf C., Janssen A., Mohr J., Ruzzu A. and Wallrabe U. Incorporating design rules into the LIGA technology applied to a Fourier transformation spectrometer. *Microsystem Technologies*, 2004 (10) 10, pp.706-710
[20] Albers A., Burkardt N., Hauser S. and Marz J. Knowledge-based design environment for primary shaped micro parts. *Microsystem Technologies*, 2005 (11) 4-5, pp.254-260
[21] Alexander C. *The Timeless Way of Building*. 1979 (Oxford University Press, New York)
[22] Beck K. and Cunningham W. Using Pattern Languages for Object-Oriented Programs. *ACM Conference on Object-Oriented Programs, Systems, Languages, and Applications (OOPSLA)*, 1987
[23] Coad P. and Mayfield M. Workshop Report – Patterns. Addendum to the Proceedings. *Conference on Object-Oriented Programs, Systems, Languages, and Applications OOPSLA*, 1992, 5-10 October, Vancouver, British Columbia, Canada, , pp.93-95
[24] WikiWikiWeb. *http://www.c2.com/cgi/wiki?WikiWikiWeb* [02.01.2009]
[25] Bayle E., Bellamy R., Casaday G., Erickson T., Fincher S., Grinter B., Gross B., Lehder D., Marmolin H., Moore B., Potts C., Skousen G. and Thomas J.: Putting it all Together: Towards a Pattern Language for Interaction Design. *SIGCHI Bulletin* 1998 (30) 1, pp.17-24. *http://www.sigchi.org/bulletin/1998.1/erickson.html* [02.01.2009]
[26] Appleton,B. Patterns for Conducting Process Improvement. *Procs. Pattern Languages of Programs PLoP 1997*, September 3-5, 1997, Allerton Park, Monticello, Illinois, USA
[27] DeLano E. and Rising L. Introducing Technology into the Workplace. *Procs. Pattern Languages of Programs PLoP 1997*, September 3-5, 1997, Allerton Park, Monticello, Illinois, USA
[28] Beedle M., Devos M., Sharon Y., Schwaber K. and Sutherland J. SCRUM: An extension pattern language for hyperproductive software development. *Procs. Pattern Languages of Programs, PLoP 1998*, August 11-14, 1998, Allerton Park, Monticello, Illinois, USA

[29] Harrison N. The Language of the Shepherds – A Pattern Language for Shepherding. *Procs. Pattern Languages of Programs, PLoP 1999*, August 15-18, 1999, Allerton Park, Monticello, Illinois, USA
[30] Salustri F. A. Using Pattern Languages in Design Engineering. *Procs. Int. Conf. on Engineering Design ICED 05*, August 15-18, 2005, Melbourne, Australia
[31] Feldhusen J. and Bungert F. PLM Pattern Language: An Integrating Theory of Archetypal Engineering Solutions. *14th CIRP Conference on Life Cycle Engineering*, June 11th–13th, 2007, Waseda University, Tokyo, Japan, pp.125-130
[32] McGrath M. E. *Next Generation Product Development: How to Increase Productivity, Cut Costs and Reduce Cycle Times*. 2004 (McGraw-Hill Companies)
[33] Weerasinghe J. S. and Salustri F. A. Use of Wikis as an engineering collaborative tool. *Procs. Int. Conf. on Engineering Design ICED2007*, 28-31 August, 2007, Paris, France
[34] Bartel T. Wikis in Unternehmen. *http://wikipedistik.de/umfrage/* [12.01.2009]
[35] Matthiesen S. A contribution to the basis definition of the element model 'Working Surface Pairs & Channel and Support Structures' about the correlation between layout and function of technical systems. In: Albers A. (ed) mkl-Forschungsberichte, Vol.6, University of Karlsruhe (TH), 2002, doctoral thesis
[36] Cunningham & Cunningham, Inc. Therefore but. *http://c2.com/cgi/wiki?ThereforeBut* [13.01.2009]
[37] Gohr A. DokuWiki. *http://www.dokuwiki.org/* [12.01.2009]
[38] Albers A., Deigendesch T. and Hauser S. Design Rules for Microcomponents. In: Ritzhaupt-Kleissl, H.-J. and Johander, P. (eds.) *Ceramics Processing in Micro Technology*. 2009 (Whittles Publishing)
[39] Albers A., Alink T., Matthiesen S. and Thau S. Support of system analysis and improvement in industrial design through the contact & channel model. *International Design Conference – Design*, May 19 - 22, 2008, Dubrovnik, Croatia

ACKNOWLEDGEMENTS
We are grateful for the support provided by the German Research Foundation (DFG) within the collaborative research center (SFB) 499 "Development, Production and Quality Assurance of Primary Shaped Microcomponents from Metallic and Ceramic Materials".

Contact:
Prof. Dr.-Ing. Dr. h.c. Albert Albers
Karlsruhe Institute of Technology (KIT)
Universität Karlsruhe (TH)
IPEK –Institut für Produktentwicklung
Kaiserstraße 10
76131 Karlsruhe
Germany
Phone: +49 721 608 2371
Fax: +49 721 608 6051
E-Mail: albers@ipek.uka.de
URL: http://www.ipek.uni-karlsruhe.de