Customer Value-based HW/SW Partitioning Decision in Embedded Systems

Neunghoe Kim, Taek Lee, Donghyun Lee, Hoh Peter In

Department of Computer Science and Engineering
College of Information and Communications
Korea University, Korea
{nunghoi, comtaek, tellmehenry, hoh_in}@korea.ac.kr

Keun Lee
Software Laboratory
Samsung Electronics, Korea
gskeun.lee@samsung.com

Abstract
In launching a product, requirement change is always risk to an embedded system designer. Including requirements change, limited resources (e.g. development time and cost) are also risk factors to designing a system. However, the time to market constraint among several limitations is most critical especially to embedded system products (e.g. cellular phone, PDA, etc.). For that reason, many strategy makers try to find effective way to a HW/SW partitioning decision in early designing phase. Only ideal way to reduce requirements change and satisfy customers is reflecting their value to decision making steps in early phase prior to fixing HW and SW component design. In this paper, Customer Value based Partitioning Decision (CVPD) method is proposed to identify, analyze, and calculate the value of the customers’ requirements, reflecting the value on the partitioning decision making process.

1. Introduction
Recently embedded software industry is developing in a high speed. Specially, by the increase of embedded software industry, project complexity, quality, and time to market problems are happening. So, methods for developing embedded software have to be considered.

If problems on customer requirements occur after the HW/SW partitioning, the loss will be much bigger than in normal software development. We need partitioning method with accurate hardware and software implementation separation to avoid these problems.

It is hard for developers, who decide the partitioning, to understand the customers’ point of view. That’s why they plan and develop with a developer’s point of view. And for problems that occur after partitioning decision step, developers allocate enough time and resource to all problems that may occur. But this time and resource is unnecessary.

Making requirements accurately before the partitioning decision is difficult because the point of view is different. Customers consider external requirements and they expect them to be selected but developers consider internal requirements like ROI, standard problems for future techniques, marketability, and monopolization. But customer requirements have a value hard to ignore.

However produces good product, value of product decreases rapidly over time. So, to success in the market, time to market is important. To purchase ideal product in a proper time, components have to have an order of priority. A component of high priority includes thinking importantly by both customers and developers. But, to order these components, customer’s requirements have to be analyzed carefully first.

Because of the reason above, we need a method to accurate customers’ and developers’ requirement before partitioning decision and select proper components for software and hardware. In this way, you can minimize the feedback process after the partitioning decision. Moreover, it is possible to increase customer’s satisfaction by creating customers’ value.

In this paper, we propose a method for HW/SW partitioning decision involving customer’s values (CVPD: Customer Value based Partitioning Decision). In the proposed method, it is possible to decide scores of importance by the analysis of customer’s
requirement and identify the value by the customer’s requirements, components, and scores of importance. And by this value, we can select components, which is temporary chosen to hardware and software components and derives interrelationship developer requirements. Finally, to consider interrelationship and identified value, it is determine successfully the option of hardware and software in time to market situation.

The rest of paper comprises as follows. Section 2 explains background knowledge about embedded design life cycle and Kano model, and Customer Value based Partitioning Decision (CVPD) method is proposed in Section 3. Section 4 takes a hypothetical application example for easy understanding of how to be utilized in practical. Finally, Section 5 concludes this paper with introducing some future work.

2. Background

2.1 Embedded Design Life Cycle

To reduce time on developing an embedded system, hardware and software is designed at the same time and after that, it integrates. There are 7 phases in the embedded design life cycle. The first phase is specification of the product and the next is HW/SW partitioning. The third phase is iteration and implementation. The fourth phase is to detail the HW/SW design. The fifth and sixth phase is to integrate HW/SW and acceptance testing. The last phase is maintaining and upgrading [1].

The important part is partitioning decision for HW/SW before the phase of designing software and hardware. Partitioning decision is a very important phase for cost, development time and risk management.

2.2 Kano Model

Kano model was proposed by Kano[2] and the model classifies product requirements into three major attributes which affect customer satisfaction level. There are basically three attributes of requirements.

- Must-be Requirements: these requirements must be covered by developers for customer satisfaction. If not, the degree of satisfaction will be enormously dropped down.
- One-dimensional Requirements: customer satisfaction of some requirements has one-dimensional relationship against the degree of requirement fulfillment. Hence, if the degree of fulfillment goes up, the level of satisfaction linearly follows up.
- Attractive Requirements: If some set of requirements are implemented, it raises the level of customer satisfaction. However, even in the case of that the requirements are not accepted, customers don’t mind.

Kano questionnaire consists of some questions asking the level of satisfaction and dissatisfaction for each product requirements under what-if scenario (function form versus dysfunction form). Basing on the results from Kano questionnaire, one classifies requirement attributes and determines each priority of customer requirements by referring to the attributes.

3. Customer Value based Partitioning Decision (CVPD)

Partitioning decision based on customer value determines the importance by analyzing customers’ required attributes and identifies their own value by referring to the importance. Hence, the value is used as a critical factor to make a partitioning decision. First considering the value at the partitioning decision step makes both of customers and developers be able to reflect their preference to the others in advance. As a result, it causes less feedback from customer requirements change with well organized hardware and software components. In this manner, CVPD can reduce the time to market with reflecting effectively customer needs.
Figure 1 is the overview picture of our proposed CVPD method. The proposed five steps cover the preprocessing tasks prior to an embedded designing life cycle.

Firstly, in Step 1, the extracted customer requirements are classified by major attributes following Kano model and scored to requirements importance. The scores of importance are used as weight values in Step 2. The purpose of Step 2 is to extract customer values from the matrix of customer requirements versus components. The outputs of Step 2 are numerically presented customer values for each component with their ranks. Step 3 sorts and picks out influential components which should be considered first of all. Only the components selected in Step 2 are placed on the top of the matrix as HW and SW columns in Step 3. Step 4 is a core part to make a decision of partitioning. So, some of criteria (i.e. developer perspective requirements) to partition components into HW and SW categories are considered being placed on left-side rows of the matrix. The purpose of the matrix in Step 4 is to measure the reasonableness about whether a particular component should be implemented to HW type or SW type. The customer values discovered from value identification step (Step 2) are utilized as influential inputs in Step 4. As soon as the results from Step 4 come out, final decision making in partitioning is done in Step 5. Each of the temporary HW/SW component candidates considered in Step 4 is finally determined and partitioned to HW or SW by selecting a high score marked type. The rest of process follows embedded designing life cycle.

Our proposed CVPD method was motivated by HOQ (House of Quality) model[3], which is a mapping tool to analyze customer and developer requirements against functional components consisting of a product.

3.1 What is Customer Value?

In our context, customer value is a kind of preference about utility to product functionality. In our CVPD method, customer value reflects the relation between customer requirements and functional components and its degree of relation which is ultimately represented as some quantitative numeric values (i.e. degree of strength or weakness in relationship between requirements and components). To measure customer value, we used Kano model in requirement attribute classification and have an expert score attribute importance in his or her opinion. Our proposed HW/SW partitioning method considering the customer value can ultimately raise the degree of customer satisfaction for a product under designing.

3.2 Detail Steps of CVPD Method

3.2.1 Importance Decision (Step 1)

This step extracts importance evaluation from customer requirements by using Kano model. Kano model classifies customer requirements into 6 attributes such as ‘Questionable’, ‘Attractive’, ‘One-dimensional’, ‘Indifferent’, ‘Must-be’, and ‘Reverse’. Each of attributes has different level of importance in terms of requirements preference.

Table 1. Kano questionnaire

<table>
<thead>
<tr>
<th>Function form of the question</th>
<th>I like it that way</th>
<th>It must be that way</th>
<th>I am neutral</th>
<th>I can live with it that way</th>
<th>I dislike it that way</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dysfunction form of the question</td>
<td>I like it that way</td>
<td>It must be that way</td>
<td>I am neutral</td>
<td>I can live with it that way</td>
<td>I dislike it that way</td>
</tr>
<tr>
<td>Customer Requirements</td>
<td>Dysfunctional form of the question</td>
<td>I like it</td>
<td>I must be</td>
<td>I am</td>
<td>I can live with it</td>
</tr>
</tbody>
</table>

Table 1 is the questionnaire used in Kano model. The level of satisfaction and dissatisfaction for individual requirements is evaluated by the questionnaire. For a given requirement, the participants are asked to select one of five options: “I like it that way”, “It must be that way”, “I am neutral”, “I can live with it that way”, or “I dislike it that way”. The selection is conducted under the two ‘what-if’ scenario perspectives. One is the situation assumed that a given requirement is implemented and served. The other one is the opposite situation (not implemented).

Table 2. Mapping Table for Requirements

<table>
<thead>
<tr>
<th>Customer Requirements</th>
<th>I like it that way</th>
<th>It must be that way</th>
<th>I am neutral</th>
<th>I can live with it</th>
<th>I dislike it</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function form of the question</td>
<td>Q</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>O</td>
</tr>
<tr>
<td>Dysfunctional form of the question</td>
<td>R</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>M</td>
</tr>
<tr>
<td>Customer Requirements</td>
<td>I like it</td>
<td>I must be</td>
<td>I am</td>
<td>I can live with it</td>
<td>I dislike it</td>
</tr>
<tr>
<td>Function form of the question</td>
<td>O</td>
<td>R</td>
<td>R</td>
<td>R</td>
<td>O</td>
</tr>
</tbody>
</table>

Table 2 plays an important role in interpreting and determining the attribute of a given requirement. Thus, it is an attribute mapping table to which a given customer requirement is referred as one of six attribute indicators (M, A, I, O, Q, R).
Table 3. Frequency Marking Sheet

<table>
<thead>
<tr>
<th>CR 1</th>
<th>CR 2</th>
<th>CR 3</th>
<th>CR n</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>O</td>
<td>M</td>
<td>I</td>
</tr>
</tbody>
</table>

Table 3 is to count and write down the number of attribute indicators appeared in requirements questionnaire. As a result, most frequently mapped indicator is assigned to category column and importance of each requirement is calculated depending on the category indicator marked. The preference priority of indicators is ‘M > O > A > I’. For quantitative measuring, numeric value 5 is assigned to M; 4 to O; 3 to A; 2 to I; and 1 to R. Q is omitted because it is inconsistent and doubtful answer. The reason why the assigned number starts from one (R) is the customer requirement marked as R should not be weighed in importance column in Step 2. The attribute ‘Reverse’ means that customers have no interesting.

3.2.2 Value Identification (Step 2)

This step analyzes correlation between component (CP) and customer requirement (CR) and identifies the customer value by using customer importance (CI). Component value evaluation matrix consists of functional components, customer requirements, and customer importance. Only the components able to be implemented as both of HW type and SW type are selected as candidates under consideration because either HW dependent implementation or SW dependent implementation should be anyway developed regardless of customer needs.

The customer requirements (CR) refined by expert group are put on the left side rows. The functional component elements under consideration are enumerated on top of the matrix. An analyst group scores the correlation degree between customer requirements and functional components by asking “How strongly is CP, related to CR?, or How much can CP, accomplish CR,?”. As a result, it can be evaluated with marks: High, Medium, and Low. Each of marks has respectively numeric values: 9, 3, and 1. By using customer importance (CI) as weights, customer importance (CI) is calculated on step 1. the final customer value (CVj) can be calculated as follows.

\[
CV_j = (CI_1 \times R_{ij}) + (CI_2 \times R_{3j}) + \cdots + (CI_i \times R_{nj})
\]

After sorting customer values, value rank (VRj) is determined in it priority, and then the value ranks are used to select influential functional components in Step 3.

3.2.3 Component Selection (Step 3)

This step aims at selecting influential functional components which should be considered prior to any other one. Because of resource limitations (e.g. time and cost), it is impossible to consider all of components and customer requirements, so some of components are selected in customer value priority (VR). Those of selected things are most valuable components in the viewpoint of customers. However, in order to prevent customer group from making an unrealistic or unachievable decision for setting up components, evaluation process is conducted again with developer requirements in the next step. Hence, the value balance between customer perspective and developer perspective is achieved.

3.2.4 CVPD Analysis (Step 4)

This step supports an analyst group to complete CVPD analysis and draw a core decision for partitioning. Component (C), developer requirements (DR), and customer value (CV) in the previous step are used here.

Table 5. CVPD Analysis Model

<table>
<thead>
<tr>
<th>DR 1</th>
<th>DR 2</th>
<th>DR 3</th>
<th>DR n</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>R2</td>
<td>R3</td>
<td>Rn</td>
</tr>
</tbody>
</table>

Table 5 is a table type model to make a portioning decision. In this model, components are enumerated at both side of HW and SW column. The component set here are some of selected ones from Step 3, so they are subset of component set in Step 2. The reason why same components are placed twice at each side of HW
and SW is because they should be evaluated simultaneously in implementation suitableness. Like Step 2, R_ij is evaluated and marked at the same time with correlation degree in respective sides (HW/SW). The following calculation generates TS and the TSs are used to determine HW type or SW type depending on its high score.

$$TS_j = (R_{ij} + R_{i2} + R_{i3} + \cdots + R_{in}) + CV_j$$

After all, TS_j plays a core role as a critical variable in making the final decision in partitioning.

3.2.5 Partitioning Decision (Step 5)

This is the last step to make a final decision in partitioning. This step finally decides which component should be implemented to HW type or SW type. The derived TS values have the meaning of implementation suitableness in HW type or in SW type as well as the degree of customer needs. This value gives designer group the picture about an achievable co-design strategy in proper time.

Table 6. Partitioning Decision List

<table>
<thead>
<tr>
<th>Component</th>
<th>Design Type</th>
<th>Development Time</th>
<th>Total Score</th>
<th>Time-to-Market</th>
</tr>
</thead>
<tbody>
<tr>
<td>CV_k1</td>
<td>HW</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CV_k2</td>
<td>SW</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CV_k3</td>
<td>HW</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CV_k4</td>
<td>SW</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CV_k5</td>
<td>HW</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

After making the partition decision. By using information come from CVPD, partitioning is done. Selected components and required time in HW or SW type of implementation are written down on the table. This partitioning decision table partitions the functional components depending on the constraint of time to market. First of all, most implementable components within time constraint are promisingly selected and the final partitioning decision is made by considering each of the total scores in their design type (HW or SW).

Table 7. Mobile Phone Example

Table 8. Value Identification in Mobile Phone

Table 9. CVPD Analysis for Mobile Phone

4. A Hypothetical Application Example

This Chapter, in light of ascertaining the validity of CVPD method proposed mentioned above and getting closer to quick understanding of its applications, is to provide an example of scenario plotted supposing that a firm A be under decision making to differentiate the priorities in its mobile phone development.

In this regard, out of 15 interviewees in total composed by customers’ and developers’ representatives well recognized with the flow of this current scenario, a prepared Kano questionnaire was distributed to gain principal data from them for further application to the aforesaid CVPD method.
Table 10. Partitioning Decision List

<table>
<thead>
<tr>
<th>Component</th>
<th>Design Type</th>
<th>Development Time</th>
<th>Total Score</th>
<th>Time-to-Market</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component 1</td>
<td>HW</td>
<td>2 month</td>
<td>23</td>
<td>5 month</td>
</tr>
<tr>
<td></td>
<td>SW</td>
<td>5 month</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>Component 2</td>
<td>HW</td>
<td>3 month</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SW</td>
<td>4 month</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>Component 4</td>
<td>HW</td>
<td>4 month</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SW</td>
<td>7 month</td>
<td>42</td>
<td></td>
</tr>
<tr>
<td>Component 5</td>
<td>HW</td>
<td>1 month</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SW</td>
<td>3 month</td>
<td>12</td>
<td></td>
</tr>
</tbody>
</table>

Table 10 shows the result by way of CVPD method. Provided that the time-to-market factor takes 5 months, the successful development could be made, except for SW of the component 4, over the all periods. Thus, it is of utter importance the component 4 must be designed as a form of HW. Taken into account the total score in the rest of component, the component 1 must be divided to HW/SW, and likewise the component 2 SW, component 5 HW, respectively. In case of same value of total scores like the component 2, the expert shall determine at his discretion those two opposite alternatives that HW comparatively faster in development but with bigger cost needed, and SW slower in development but with smaller cost on the contrary.

5. Conclusion

How to divide HW/SW in consideration of combining the requirements of developers and customers in the aspect of embedded design life cycle gives rise to a big burden in making a right decision. This study aims at identifying, through a Kano model, the importance by putting a priority over a spectrum of the customer requirements against the embedded design life cycle. Moreover, it provides some possible ways to make HW/SW divided, being kept in mind the values identified over the time-to-market variables situation. The ultimate goal in this study is to allow HW/SW to be divided in a manner of rationale and in such a way of developing the goods most timely and toward minimization of damage provoked by the customer's feedback by laying much emphasis on the two; identification of component importantly influential to both of developers and customers and adequate perception of time-to-market factor.

The study, taking an example for application of the proposed method to determine the option of hardware and software, successfully shows possibility of effective selection in most-fit timeliness to the market by quantitatively reflecting the value of the customer.

It is of great need from now on to evaluate the way proposed through deeper review on real cases to be developed and, at the same time, to improve method in partitioning decision of HW and SW.

Acknowledgements

“This research was supported by the MKE(Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Advancement)” (IITA-2008-C1090-0801-0032)

References