

Hệ thống nhúng - Thiết kế và Phát triển .....	3
Những điều cần tìm... ..	3
12.0 Giới thiệu .....	4
12.1 Thiết kế và phát triển hệ thống .....	6
12.1.1 Chuẩn bị sẵn sàng - Bắt đầu Tư duy .....	7
12.1.2 Bắt đầu .....	số 8
12.2 Mô hình vòng đời .....	9
12.2.1 Mô hình thác nước .....	10
12.2.2 Mô hình chu kỳ V .....	11
12.2.3 Mô hình xoắn ốc .....	13
12.2.4 Tạo mẫu nhanh - Tăng dần .....	14
12.3 Giải quyết vấn đề .....	15
12.3.1 Năm bước để thiết kế .....	15
12.4 Quy trình thiết kế .....	16
12.5 Xác định các yêu cầu .....	18
12.6 Xây dựng đặc điểm kỹ thuật yêu cầu .....	21
12.6.1 Môi trường .....	22
12.6.2 Hệ thống .....	23
12.7 Đặc điểm kỹ thuật thiết kế hệ thống .....	35
12.7.1 Hệ thống .....	35
12.8 Thông số kỹ thuật hệ thống so với yêu cầu hệ thống .....	50
12.8 Thông số kỹ thuật hệ thống so với yêu cầu hệ thống .....	51
12.9 Phân vùng và phân rã hệ thống .....	52
12.9.1 Suy nghĩ ban đầu .....	52
12.9.2 Khớp nối .....	55
12.9.3 Sự gắn kết .....	57
12.9.4 Xem xét thêm .....	59
12.10 Thiết kế chức năng ... ..	59
12.11 Thiết kế kiến trúc .....	66
12.11.1 Đặc điểm kỹ thuật và thiết kế phần cứng và phần mềm .....	67
12.12 Mô hình chức năng so với mô hình kiến trúc .....	71
12.12.1 Mô hình chức năng .....	72
12.12.2 Mô hình kiến trúc .....	72
12.12.3 Sự cần thiết của cả hai mô hình .....	72
12.13 Nguyên mẫu .....	73

12.13.1 Thực hiện .....	73
12.13.2 Phân tích thiết kế hệ thống .....	74
12.14 Các cân nhắc khác .....	77
12.14.1 Viết hoa và tái sử dụng .....	77
12.14.2 Yêu cầu Truy xuất nguồn gốc và Quản lý .....	78
12.15 Lưu trữ dự án .....	79
12.16 Tóm tắt .....	81

## Chương 12

# Hệ thống nhúng - Thiết kế và Phát triển

## Những điều cần tìm...

- Những điều cần xem xét trong một thiết kế. Chu kỳ
- sống của sản phẩm.
- Năm bước để thiết kế.
- Sự cần thiết phải hiểu môi trường và hệ thống đang được thiết kế.
- Sự khác biệt giữa định nghĩa yêu cầu và đặc điểm kỹ thuật. Động cơ và mục tiêu
- khi phân vùng hệ thống. Khớp nối và sự gắn kết và tại sao chúng lại quan trọng.
- 
- Sự khác biệt giữa mô hình chức năng và mô hình kiến trúc của một hệ thống.
- Động lực và thời gian phân tích tĩnh và động của một thiết kế Viết hoa và tái sử dụng các thiết kế.
- Yêu cầu truy xuất nguồn gốc.

## 12.0 Giới thiệu

Trong chương này, chúng ta sẽ nghiên cứu các giai đoạn chính của quá trình phát triển hệ thống nhúng. Các khía cạnh chi tiết hơn của quá trình đó sẽ được khám phá cùng với việc thiết kế và kiểm tra các yếu tố phần cứng và phần mềm cụ thể của hệ thống.

Trong chương này, chúng ta sẽ tìm hiểu rằng thiết kế là quá trình chuyển các yêu cầu của khách hàng thành một hệ thống hoạt động và sự phức tạp của các hệ thống đương đại đòi hỏi một cách tiếp cận chính thức và các phương pháp chính thức. Làm việc từ một đặc tả chính thức của một vấn đề, chúng ta sẽ xem xét các cách phân vùng hệ thống như một bước mở đầu để phát triển một thiết kế chức năng. Sau đó, chúng tôi sẽ kiểm tra quá trình ánh xạ mô hình chức năng đối với một cấu trúc kiến trúc và cuối cùng là một nguyên mẫu đang hoạt động. Để giúp đảm bảo độ bền của sản phẩm cuối cùng, chúng tôi sẽ minh họa cách phân tích phê bình thiết kế cả trong và sau khi phát triển.

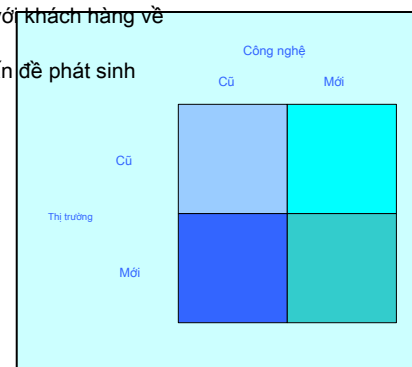
Chúng ta cũng sẽ xem xét một số cân nhắc quan trọng khác trong vòng đời thiết kế. Chúng sẽ bao gồm tài sản trí tuệ, tái sử dụng thành phần / mô-đun, quản lý yêu cầu và quy trình lưu trữ.

Khi chúng ta bắt đầu nghĩ về một sản phẩm mới hoặc thêm các tính năng mới vào một sản phẩm hiện có, chúng ta phải nhìn mọi thứ từ nhiều quan điểm khác nhau. Điều quan trọng nhất trong số này là của khách hàng vì họ tài trợ cho việc phát triển sản phẩm trực tiếp thông qua hợp đồng đã thỏa thuận hoặc gián tiếp thông qua mua hàng.

Thiết kế tốt nhất có giá trị nhỏ nếu không có ai sẵn sàng mua. Vì vậy, chúng tôi đặt ra câu hỏi: Những loại điều gì nên được xem xét?

Nếu chúng ta xem xét sản phẩm, chúng ta phải biết cách đo lường *chi phí* và *đặc trưng*. Chúng ta phải có khả năng xác định và phân biệt giữa *thực tế* và *nhu cầu nhận thức*. Thông thường khi chúng tôi nói chuyện với khách hàng về các sản phẩm mới, “yêu cầu” thiết yếu đối với sản phẩm thế hệ tiếp theo là điều bị thiếu khi một vấn đề phát sinh sáng nay.

Điều quan trọng là phải học cách đánh đổi thị trường và công nghệ. Cách đây vài năm, bảng rất đơn giản sau đây đã được đề xuất. Đang lấy *cũ* công nghệ vào và *cũ* thị trường là hợp lý và an toàn



chiến lược. Đây là những thị trường thích hợp và thường cung cấp sự hỗ trợ và phát triển theo hướng phát triển cho các sản phẩm không còn nằm trong cung cấp chính thống của nhà cung cấp. Đang lấy *Mới* công nghệ thành *Mới* thị trường khó khăn và rủi ro. Đồng thời, phần thưởng có thể rất cao. Máy tính cá nhân là một ví dụ rất tốt. Xerox và Apple đều không thành công với những sản phẩm đầu tiên của họ. Con người và công nghệ đầy đủ chỉ đơn giản là chưa sẵn sàng. Đang lấy *Mới* công nghệ trong một *hiện có* khu vực hoặc *hiện có* công nghệ trong một *Mới* khu vực dễ dàng hơn. Ít nhất một phần của vấn đề - thị trường hoặc công nghệ

- được hiểu rõ và phát triển tốt.

Chúng ta phải hiểu tầm quan trọng của *thời hạn cuối cùng* và *chi phí*. Việc phát triển sản phẩm dựa trên hợp đồng được thương lượng (trực tiếp hoặc gián tiếp) giữa chúng tôi và (các) khách hàng. Việc không tôn trọng chi phí hoặc lịch trình phát triển và giao hàng dẫn đến mất doanh thu, thị phần và uy tín.

Chúng ta cũng phải luôn xem xét *độ tin cậy*, *an toàn*, và *chất lượng* trong các sản phẩm chúng tôi thiết kế. Chúng tôi sẽ nghiên cứu những điều này rất chi tiết ngay sau đây. Ngoài nhu cầu rõ ràng để hoạt động bình thường, sản phẩm phải *cường tráng*. Đó chỉ đơn giản là, 'Nó có làm được những gì nó phải làm không?' và 'Nó hoạt động như thế nào với các đầu vào không mong muốn?' Tuy nhiên, mạnh mẽ có nghĩa là nhiều hơn thế này. Mạnh mẽ cũng ngụ ý rằng hệ thống hoạt động ngay cả khi nó bị hư hỏng một phần, hoặc trong điều kiện nhiệt độ khắc nghiệt, hoặc nếu nó bị rơi. Nếu một sản phẩm làm được những gì nó phải làm nhưng dễ vỡ và có lỗi, thì sản phẩm đó không mạnh.

Các *tài liệu* chúng tôi sản xuất để đi kèm với sản phẩm phải rõ ràng và dễ hiểu. Sản phẩm phải dễ sử dụng - trực quan thay vì phản trực quan.

*Hỗ trợ đăng bán hàng*, bao gồm cả việc sửa lỗi, là rất quan trọng. Thiếu chất lượng có hai cái giá phải trả. Chi phí đầu tiên là hiển nhiên và ngay lập tức, chi phí để sửa chữa, thường là nhỏ. Thứ hai, chi phí ẩn, sự mất niềm tin của khách hàng và doanh số bán hàng, có thể rất lớn. Một khi sự tự tin mất đi rất khó lấy lại.

**Một ví dụ đơn giản:**

Nhiều năm trước, khi phát triển một số hệ thống nhúng dựa trên bộ vi xử lý ban đầu, chúng tôi sẽ gặp phải sự cố khi gỡ lỗi phần cứng và phần mềm. Vào thời điểm đó, các công cụ còn rất ít và xa. Đây là một lĩnh vực mới.

Một công cụ rất mạnh để giúp theo dõi các vấn đề như vậy được gọi là máy phân tích logic. Nó cho phép người ta làm theo các hướng dẫn mà bộ xử lý đang thực thi (trong thời gian thực) và tìm hiểu lý do tại sao mọi thứ đi vào và không bao giờ xuất hiện. Chúng tôi phải có một chiếc, vì vậy, công ty chúng tôi đã mua hai chiếc từ hai nhà cung cấp khác nhau.

Máy phân tích từ nhà cung cấp A đến được đặt ngoài hộp, trên băng ghế dự bị, được kết nối với hệ thống và thực hiện các phép đo hữu ích trong vòng 10-15 phút. Chỉ vài ngày sau, có ai nghĩ rằng hãy xem qua sách hướng dẫn. Máy phân tích của nhà cung cấp B có giao diện người dùng cạnh tranh với biểu mẫu thuế 1040. Sách hướng dẫn dày một inch của nó cũng khó hiểu không kém và đòi hỏi nhiều giờ nghiên cứu trước khi thực hiện các phép đo đơn giản nhất.

Đoán xem nhạc cụ nào luôn có một hàng người chờ đợi để sử dụng nó và đoán xem nhà cung cấp nào đã bán cho chúng ta nhiều nhạc cụ hơn nữa.

**12.1 Thiết kế và Phát triển Hệ thống**

Thiết kế và phát triển hệ thống là một vấn đề đầy thách thức. Điều làm cho nó thú vị và thú vị là nó có một thành phần sáng tạo rất lớn. Không có quy tắc nào, không có bước nào phải tuân theo để tạo một quảng cáo. Tuy nhiên, có một tập hợp lớn các quy tắc để đảm bảo điều ngược lại. Xem xét một đứa trẻ mới. Mỗi người bước vào thế giới này, đôi mắt mở to với hàng triệu câu hỏi. Tại sao bầu trời màu xanh? Tại sao mặt trời có màu vàng? Tại sao chúng ta không thể nhìn thấy không khí? Không khí đến từ đâu? Chúng ta làm gì? Chúng tôi đưa chúng vào trường học. Chúng tôi dạy họ các quy tắc. Đi vào bất kỳ nhóm nhỏ nào và hỏi xem có bao nhiêu bạn có thể hát? Có bao nhiêu bạn có thể vẽ? Hầu hết mọi bàn tay nhỏ bé đều nháy lên. Đi vào bất kỳ nhóm người lớn nào giống nhau và đặt những câu hỏi tương tự. Mọi người bỗng nhiên bị mê hoặc với những đôi giày của họ. Một tay có thể đưa lên từ từ. Tại sao? Chúng ta đặt ra quá nhiều hạn chế trong suy nghĩ của mình. Chắc chắn, chúng tôi có thể cần thiết bị điện tử trị giá 10 triệu đô la để cung cấp cho giọng nói của chúng tôi cao độ hoàn hảo, nhưng, thì sao. Chúng ta cần xóa bỏ những hạn chế giả tạo mà chúng ta áp đặt lên suy nghĩ của mình.

Nhìn những đứa trẻ đang vẽ hoặc tô màu. Chúng ta nói gì với họ? Không, mọi người không có màu tím. Bò không bay được. Cá không có chân - nữa.Ồ, nhân tiện, hãy luôn tô màu trong các đường nét... và chúng ta cũng hãy học cách sáng tạo.

### 12.1.1 Chuẩn bị sẵn sàng - Bắt đầu Suy nghĩ

Ok, bắt đầu thôi. Lái xe luôn là một nơi tốt để bắt đầu. Các quy tắc rất dễ dàng. Giữ vạch vàng bên trái và vạch trắng bên phải - ngoại trừ ở Anh và một số nơi khác. Bây giờ là cơ hội để sáng tạo. Vào mùa thu ở các vùng phía bắc của thế giới, ngày ấm áp, nhưng, đêm bắt đầu lạnh hơn. Thường có một chút sương mù cũng làm xuất hiện. Vào buổi sáng, sương mù và cái lạnh đã kết hợp lại tạo thành một lớp băng rất mịn trên đường. Chúng tôi gọi đây là băng đen; nó cho chúng ta cơ hội để sáng tạo. Lên xe và chạy ra đường. Điều này vô nghĩa về việc ở lại trong dòng là gì?

Bây giờ có lẽ chúng tôi đã quyết định rằng chúng tôi có thể sáng tạo một chút, hãy bắt đầu khám phá. Khi chúng ta bắt đầu nghĩ về một thiết kế mới, chúng ta sẽ phát hiện ra rằng có rất nhiều điều cần được xem xét. Vấn đề có thể không phải lúc nào cũng là sự đồ mạt lúc đầu. Roger van Oech trong *Một vết nứt ở phía bên của đầu*, Warner Brooks, nói "Luôn luôn tìm kiếm câu trả lời đúng thứ hai." Anh ấy nói đúng. Khi chúng ta bắt đầu, điều quan trọng là phải hiểu vấn đề để đảm bảo rằng chúng ta đang giải đúng vấn đề. Hãy xem xét hình bên cạnh. Hình nào là hình chính xác? Đó là bà già hay trẻ?

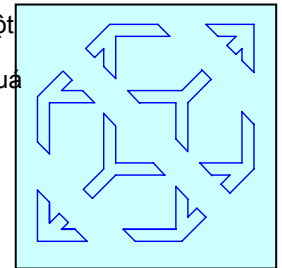


Khi chúng ta bắt đầu cố gắng giải quyết một vấn đề, điều quan trọng là phải nói chuyện với tất cả những người có liên quan; để lắng nghe các ý kiến khác nhau; để xem thiết kế có thể ảnh hưởng như thế nào đến những người phải làm việc với nó. Chúng ta phải dành thời gian để xem xét các quan điểm khác nhau về vấn đề; nhìn

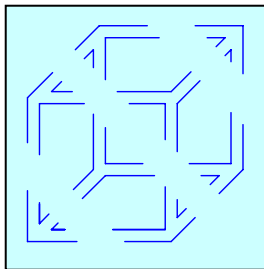


tại nó từ cả bên trong và bên ngoài. Dựa trên quan điểm của chúng tôi, chúng tôi có thể có một số cách hiểu khác nhau về vấn đề sau đây. Chúng ta đang xây một cái cốc hay chúng ta đang xây hai bức tượng?

Sẽ luôn có những trường hợp chúng ta có quá nhiều thông tin, quá nhiều ý kiến hoặc quá nhiều chi tiết. Nhớ biểu cảm xưa không thấy rừng lấy cây. Điều này cũng đúng khi chúng ta bắt đầu cố gắng hiểu một vấn đề trong giai đoạn đầu của thiết kế. Nhìn vào bản vẽ tiếp theo này. Bạn thấy gì? Một thiết kế thú vị; nó trông có lẽ giống như một bông tuyết. Đây là một trường hợp mà chúng ta có quá nhiều thông tin.



Hãy xóa một số thông tin - nếu chúng ta có cái nhìn trừu tượng hơn về



vấn đề, giải pháp dễ thấy hơn.

Bây giờ chúng ta đã bắt đầu, chúng ta hãy xem xét vấn đề thiết kế. Hãy xem mỗi thiết kế như một cơ hội để khám phá.

### 12.1.2 Bắt đầu

Thiết kế và phát triển các hệ thống nhúng đặt ra một số thách thức thú vị và đòi hỏi một số lượng lớn các quyết định. Một số quyết định yêu cầu kiến thức về vấn đề, những quyết định khác về các công cụ và kỹ thuật có thể có sẵn, và vẫn có những quyết định khác chọn phương pháp tiếp cận giải pháp. Thường sẽ vẫn còn nhiều điều cần suy nghĩ mà không liên quan đến phần kỹ thuật của vấn đề. Tập hợp những điều chúng tôi làm khi chuyển từ yêu cầu sang ứng dụng thường được gọi là *chu kỳ sống của sản phẩm*.

Giống như rất nhiều thứ khác trong cuộc sống, có lẽ có nhiều mô hình vòng đời sản phẩm khác nhau cũng như việc người thiết kế các hệ thống này. Ai nói rằng không có bất kỳ sự sáng tạo? Mỗi mô hình này đều có những người ủng hộ nó và mỗi mô hình cũng có những nhóm người gièm pha. Mục tiêu trong vài trang tiếp theo là giới thiệu một số điều quan trọng hơn mà người ta nên nghĩ đến khi thực hiện một thiết kế, trình bày một số mô hình vòng đời phổ biến hơn và trình bày một số hướng dẫn về những điều đã hoạt động trong các dự án thành công. Bất chấp những gì họ nói với bạn, không có quy tắc nào khó và nhanh cả. Rất tiếc, tôi đã nói dối. Có ba điều: học hỏi được nhiều điều với mỗi dự án, vui vẻ và làm đúng công việc... với khả năng tốt nhất của bạn. Bắt đầu nào.



## 12.2 Mô hình vòng đời

Vòng đời sản phẩm của một ứng dụng nhúng hoàn toàn là một biểu diễn mô tả. Nó phá vỡ quá trình phát triển thành một loạt các hoạt động có liên quan lẫn nhau. Mỗi hoạt động đóng một vai trò chuyển đổi đầu vào (đặc tả) của nó thành đầu ra (một giải pháp được lựa chọn). Việc tổ chức các bước được thực hiện theo một *mô hình quá trình thiết kế* - *mô hình vòng đời*. Hình thức trong thiết kế cung cấp cấu trúc cho sự phát triển cho phép người ta khám phá thiết kế một cách sáng tạo trong khi sử dụng các công cụ để quản lý một số vấn đề cơ học hơn. Chúng tôi sử dụng cấu trúc như một sự hỗ trợ hơn là một thứ gì đó mã hóa thiết kế.

Như chúng tôi đã nhận xét, trong các tài liệu liên quan, người ta có thể thấy rằng nhiều cách tiếp cận và mô hình khác nhau đã được đề xuất. Vào cuối ngày, tất cả đều có cùng một mục tiêu cơ bản; chúng đều có các giai đoạn tương tự nhau. Có lẽ, chúng ta có thể nói chính xác hơn rằng tất cả họ đều có nhu cầu hoặc mục tiêu hoặc mục tiêu giống nhau. Những nhu cầu này rất đơn giản,

- Tìm hiểu những gì khách hàng muốn
- Nghĩ ra cách để cung cấp cho họ những gì họ muốn Chứng minh những gì bạn đã làm
- bằng cách xây dựng và thử nghiệm nó Xây dựng thật nhiều để chứng minh rằng đó
- không phải là một sự tình cờ Sử dụng sản phẩm để giải quyết vấn đề của khách hàng
- 

Một số mô hình hoặc cách tiếp cận phổ biến hơn trong lịch sử bao gồm,

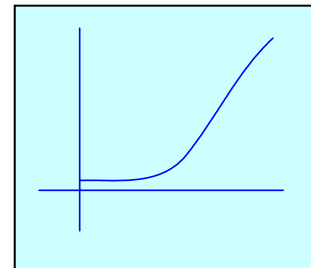
- Thác nước
- Chu kỳ V
- Xoắn ốc
- Nguyên mẫu nhanh

Ngày nay, chúng tôi liên tục phát triển những cái mới. Dù người ta chọn mô hình nào, thì điểm quan trọng nhất là hiểu được ý nghĩa và mục đích hoặc mục tiêu của từng giai đoạn hoặc bước trong quy trình. Hiểu các sản phẩm phân phối cho mỗi bước cũng như các đầu ra và đầu vào cần thiết được yêu cầu để di chuyển, kết thúc hoặc bước vào mỗi giai đoạn

trong mô hình đã chọn. Sau đó, hãy làm theo những điều đó - không đi đường tắt. Chúng ta sẽ xem xét ngắn gọn từng mô hình

trong số bốn mô hình này ngay lập tức. Trước khi làm như vậy, hãy xem

ở một mô hình khác phù hợp với bất kỳ giai đoạn kỹ thuật nào; nó trông giống như thế trong hình đi kèm.



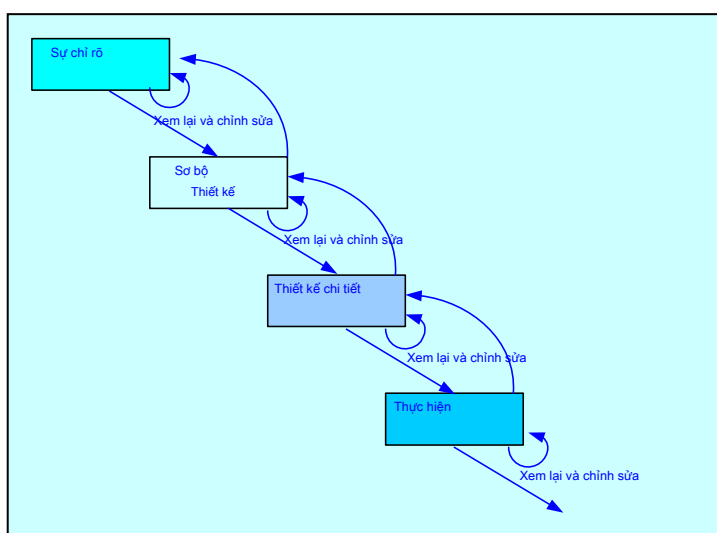
Đây được gọi là *mô hình gậy khúc côn cầu* hoặc đường cong; hình dạng của nó rất gợi ý về nơi mà cái tên đó xuất phát. Chúng tôi đã nói về tầm quan trọng của việc giải quyết độ tin cậy và an toàn sớm trong

đặc điểm kỹ thuật yêu cầu và các giai đoạn thiết kế của vòng đời. Đường cong của gậy khúc côn cầu mang lại cảm giác trực quan về lý do tại sao. Nếu chúng ta gắn nhãn trục hoành là thời gian và trục tung là chi phí và áp dụng nó ở đây, chúng ta thấy rằng chúng ta càng trì hoãn trong việc giải quyết những vấn đề đó, thì chi phí sẽ càng lớn. Chi phí không phải là tiền đơn thuần.

Hãy bắt đầu với *Thác nước* mô hình. Sử dụng khả năng sáng tạo nghệ thuật của bạn ở đây. Tên của nó gợi lên âm thanh của nó gợi lên triết lý và cách tiếp cận được tạo ra trong mô hình.

### 12.2.1 Mô hình thác nước

Các *Mô hình thác nước* đại diện cho một chu kỳ; một loạt các bước xuất hiện giống như một thác nước, tuần tự, một bên dưới bước tiếp theo như chúng ta thấy trong hình sau.



Các bước được đưa ra như,

- Sự chỉ rõ
- Thiết kế sơ bộ
- Đánh giá thiết kế
- Thiết kế chi tiết
- Đánh giá thiết kế
- Thực hiện
- Ôn tập

Cùng nhau, chúng nắm bắt từng nhu cầu mà chúng tôi đã xác định trước đó. Các bước kế tiếp được liên kết theo một cách liên kết.

Một liên kết như vậy có xu hướng nói: *Hoàn thành giai đoạn này và chuyển sang giai đoạn tiếp theo.*

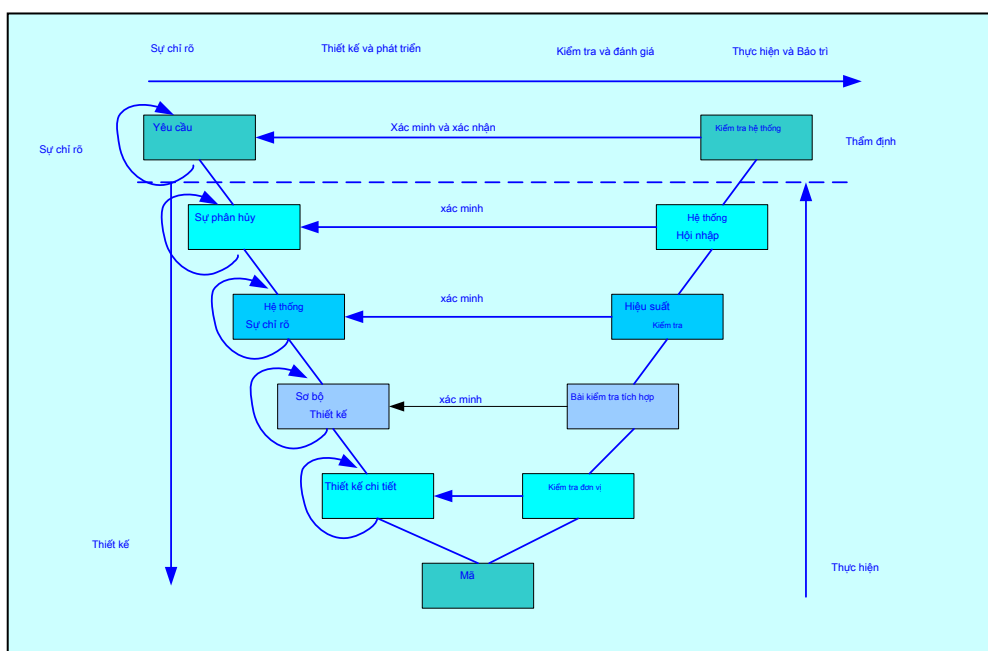
Quan sát rằng mỗi pha cũng được kết nối trở lại pha trước đó. Kết nối ngược đó cung cấp một liên kết xác minh thiết yếu ngược lại để đảm bảo rằng giải pháp (ở dạng hiện tại) đồng ý và tuân theo thông số kỹ thuật. Với mô hình Waterfall, việc ghi nhận các vấn đề có thể bị trì hoãn cho đến khi các trạng thái phát triển sau này có chi phí sửa chữa cao hơn (đường cong khúc côn cầu). Mô hình Waterfall bị giới hạn ở chỗ nó không xem xét tính chất lặp đi lặp lại điển hình của thiết kế trong thế giới thực.

### 12.2.2 Mô hình chu kỳ V

Các *Chu kỳ V* tương tự như mô hình Waterfall ngoại trừ việc nó nhấn mạnh nhiều hơn vào tầm quan trọng của việc giải quyết các hoạt động thử nghiệm trước thay vì sau đó trong vòng đời. Mỗi giai đoạn liên kết hoạt động phát triển cho giai đoạn đó với một bài kiểm tra hoặc xác nhận ở cùng cấp độ. Mỗi giai đoạn thử nghiệm được xác định với giai đoạn phát triển phù hợp của nó như chúng ta thấy trong hình sau.

Trong sơ đồ, chúng ta có

- Yêu cầu ↔ Kiểm tra hệ thống / chức năng
- Thiết kế cấp cao ↔ Kiểm tra tích hợp
- Thiết kế chi tiết ↔ Kiểm tra đơn vị

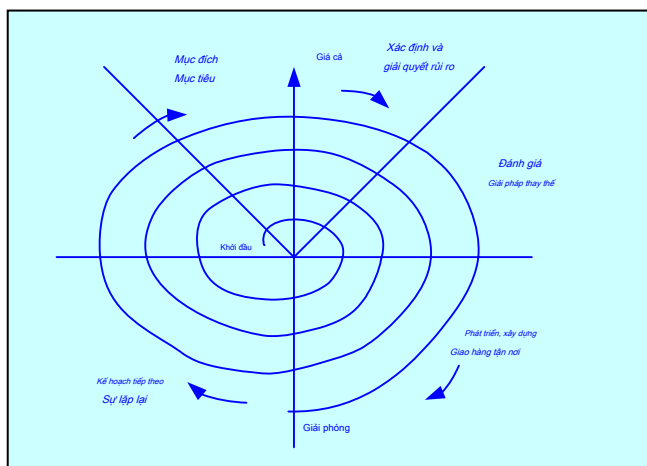


Chúng tôi xác định các giai đoạn chính của vòng đời dự án trên đầu bản vẽ. Những điều này mở rộng từ đặc điểm kỹ thuật đến giao hàng của khách hàng và hỗ trợ chuyển phát sau. Nếu thực hiện theo trình tự ở phía bên trái của bản vẽ, người ta có thể thấy rằng quy trình đặc tả và thiết kế sử dụng mô hình từ trên xuống trong khi việc triển khai và kiểm tra tiến hành từ mô hình từ dưới lên như được phản ánh ở phía bên phải của bản vẽ.

Rõ ràng là mỗi hoạt động phát triển xây dựng một mô hình chi tiết hơn của hệ thống và mỗi bước xác minh sẽ kiểm tra việc triển khai hệ thống hoàn thiện hơn so với các yêu cầu ở giai đoạn đó. Quá trình phát triển kết thúc phần thử nghiệm liên quan đến thiết kế và thiết kế của chu trình phát triển của hệ thống với cả kiểm tra xác nhận và kiểm tra xác nhận so với thông số kỹ thuật ban đầu.

### 12.2.3 Mô hình xoắn ốc

Các *Mô hình xoắn ốc* được đề xuất và phát triển bởi Barry Boehm, *Mô hình xoắn ốc về phát triển và nâng cao phần mềm*, Máy tính, tháng 5 năm 1988. Một phiên bản đơn giản của mô hình đó được trình bày trong hình sau,



Mô hình có cái nhìn định hướng rủi ro về vòng đời phát triển. Mỗi vòng xoắn đều giải quyết những rủi ro chính đã được xác định. Sau khi tất cả các rủi ro đã được giải quyết, mô hình xoắn ốc chấm dứt cũng như mô hình Waterfall và V, khi phát hành một sản phẩm.

Giống như các mô hình trước đó, mô hình Xoắn ốc bắt đầu với đặc điểm kỹ thuật tốt của các yêu cầu. Sau đó, nó hoàn thành lặp đi lặp lại một ít của mỗi giai đoạn. Triết lý của nó là bắt đầu từ quy mô nhỏ, khám phá rủi ro, phát triển kế hoạch đối phó với rủi ro và cam kết một cách tiếp cận cho lần lặp tiếp theo. Chu trình tiếp tục cho đến khi sản phẩm hoàn chỉnh. Mô hình của Boehm chứa nhiều chi tiết hơn mô hình được trình bày ở đây. Trong cả hai trường hợp, mỗi lần lặp lại của vòng xoắn bao gồm sáu bước,

- Xác định mục tiêu, lựa chọn thay thế và ràng buộc Xác định và
- giải quyết rủi ro
- Đánh giá các lựa chọn thay thế
- Phát triển các sản phẩm phân phối-xác minh chúng là chính xác Lập kế
- hoạch cho lần lặp tiếp theo
- Cam kết một cách tiếp cận cho lần lặp lại tiếp theo

Mô hình Xoắn ốc là một cải tiến trên mô hình Waterfall và V vì nó cung cấp nhiều bản dựng cũng như một số cơ hội để đánh giá rủi ro và cho sự tham gia của khách hàng. Tuy nhiên, nó phức tạp, khó quản lý và không giữ cho tất cả các nhà phát triển bận rộn trong tất cả các giai đoạn.

#### 12.2.4 Tạo mẫu nhanh - Tăng dần

Các *Mô hình tạo mẫu nhanh* nhằm cung cấp triển khai nhanh chóng (do đó có tên) các phần cấp cao của cả phần mềm và phần cứng trong giai đoạn đầu của dự án. Phương pháp này cho phép các nhà phát triển xây dựng các phần hoạt động của phần cứng và phần mềm theo từng giai đoạn tăng dần. Mỗi giai đoạn bao gồm thiết kế, mã và kiểm tra đơn vị, kiểm tra tích hợp và phân phối. Ở mỗi giai đoạn trong suốt chu kỳ, người ta kết hợp thêm một chút chức năng dự kiến.

Nguyên mẫu hữu ích cho cả nhà thiết kế và khách hàng. Đối với nhà thiết kế, nó cho phép phát triển sớm các phần chính của chức năng dự kiến của hệ thống. Bằng cách đó, nó giúp thiết lập và xác minh kiến trúc kết cấu cũng như luồng điều khiển thông qua hệ thống. Cách tiếp cận như vậy cho phép người ta xác định sớm các vấn đề lớn (đường cong khúc côn cầu một lần nữa).

Khách hàng được hưởng lợi khi có cơ hội làm việc với một đơn vị chức năng trong chu kỳ phát triển sớm hơn nhiều so với bất kỳ mô hình nào trong ba mô hình trước đó. Khách hàng có thể sử dụng nguyên mẫu trong bối cảnh dự định để cung cấp phản hồi cho các nhà thiết kế về bất kỳ vấn đề nào với thiết kế.

Phản hồi như vậy là một khía cạnh quan trọng của cách tiếp cận vì nó khuyến khích dòng chảy ngược hoặc ngược lại trong quá trình. Nó có thể được sử dụng để tinh chỉnh hoặc thay đổi nguyên mẫu để sửa chữa các vấn đề đã xác định và đảm bảo rằng thiết kế đáp ứng nhu cầu thực sự của khách hàng.

Nguyên mẫu có thể là *tiến hóa* hoặc là *vứt đi*. Nó có lợi thế là có một hệ thống làm việc sớm trong quá trình phát triển. Như đã lưu ý, các vấn đề có thể được xác định sớm hơn và nó cung cấp các biện pháp hữu hình về tiến độ. Tuy nhiên, để có hiệu quả, phương pháp tạo mẫu nhanh đòi hỏi phải lập kế hoạch cẩn thận ở cả cấp quản lý dự án và cấp của nhà thiết kế.

Hãy cẩn thận với cách sử dụng nguyên mẫu,

***Thận trọng,***

Các nguyên mẫu không bao giờ được biến thành sản phẩm cuối cùng.

Bây giờ chúng ta hãy chuyển sang quá trình thiết kế. Thiết kế bắt đầu với thế giới thực. Chúng tôi đang cố gắng giải quyết vấn đề.

Chúng tôi làm như vậy để làm cho cuộc sống của chúng tôi dễ dàng hơn.

### 12.3 Giải quyết vấn đề

Khi chúng tôi bắt đầu thiết kế một sản phẩm mới hoặc phải kết hợp một số tính năng hoặc khả năng mới vào một sản phẩm hiện có, chúng tôi bắt đầu với một tập hợp các yêu cầu thường được nêu dưới dạng văn bản. Mục đích là lập bản đồ các yêu cầu đó - thế giới thực - thông qua một loạt các biến đổi thành giải pháp - thế giới trừu tượng. Trong quá trình thiết kế, chúng tôi chuyển từ thế giới cụ thể, thực tế sang thế giới trừu tượng. Các bước này bao gồm những gì chúng tôi mô tả là thực hành kỹ thuật thiết kế tốt.

#### 12.3.1 Năm bước để thiết kế

Hy vọng rằng chúng ta đã học được từ nhiều năm trước rằng bước đầu tiên để thiết kế không phải là lấy bàn phím hoặc bộ xử lý gần nhất và bắt đầu lấy mã hoặc nối các bộ phận lại với nhau. Với các hệ thống phức tạp ngày nay, việc lập kế hoạch và suy nghĩ trước khi bắt đầu là điều cần thiết cho bất kỳ thiết kế thành công nào. Nếu lấy các yếu tố trung tâm từ mỗi mô hình vòng đời, người ta thấy rằng các nhà thiết kế hệ thống giỏi và các dự án thành công thường tiến hành bằng cách sử dụng tối thiểu năm bước. Các bước này được xác định trong bảng sau.

- Yêu cầu Định nghĩa
- Đặc điểm kỹ thuật của hệ thống
- Thiết kế chức năng
- Thiết kế kiến trúc
- Tạo mẫu

Hình thức của mỗi bước phụ thuộc vào độ phức tạp của sản phẩm cuối cùng. Nếu một người đang làm việc một mình hoặc với một số người khác trong công ty của riêng bạn trong một dự án nhỏ hơn, một tấm bảng trắng ở trung tâm nhà để xe thường có thể đủ. Nếu một người đang dàn xếp một dự án mà

bao gồm các nhà phát triển, nhà sản xuất và các quy định ở một số quốc gia trên thế giới, (ngày nay ngày càng trở nên phổ biến), nhu cầu về hình thức ngày càng tăng. Khi làm việc với từng giai đoạn này của vòng đời sản phẩm, người ta phải nhớ rằng chúng là các hướng dẫn; các phương pháp hay nhất của tập thể. Họ không phải là một danh sách kiểm tra cho một dự án thành công; và chúng không đầy đủ.

Ngày nay, quy trình thiết kế đương đại cũng phải thực thi *IP (sở hữu trí tuệ)*,

viết hoa và sử dụng lại ở mọi giai đoạn thiết kế. Thời của Bob Widler (cha đẻ của op amp) thuyết trình về thiết kế mạch tích hợp trong các quán bar ở Thung lũng Silicon đã qua lâu. Người ta cũng phải xem xét khả năng truy xuất nguồn gốc theo cả chiều thuận và chiều ngược lại. Khả năng truy xuất nguồn gốc nắm bắt các mối quan hệ giữa các yêu cầu và tất cả dữ liệu thiết kế tiếp theo và giúp quản lý các thay đổi yêu cầu.

#### 12.4 Quy trình thiết kế

Khi chúng ta bắt đầu khám phá chu trình phát triển sản phẩm, chúng ta sẽ đi qua từng bước trong số năm bước này.

Thay vì tập trung vào cách một mô hình cụ thể tiếp cận việc giải thích các bước này, chúng tôi sẽ cố gắng xác định các yếu tố thiết yếu của mỗi bước. Cách tiếp cận mà chúng tôi sẽ trình bày là từ trên xuống và lặp đi lặp lại.

Hai bước đầu tiên tập trung vào việc nắm bắt và chính thức hóa hành vi bên ngoài của hệ thống. Ba phần còn lại di chuyển bên trong hệ thống và lặp lại quy trình cho việc triển khai nội bộ dẫn đến hành vi mong muốn và được chỉ định. Như chúng tôi sẽ làm từ bên ngoài, bên trong, chúng tôi sẽ chuyển từ cái chung sang cái cụ thể, nắm bắt và xác định rõ từng khía cạnh của thiết kế.

Một nhiệm vụ chính, khi chúng ta chuyển vào bên trong hệ thống, sẽ là phân rã và tinh chỉnh thiết kế từ một thực thể ngu ngốc mà ai đó cần thành sản phẩm thực hiện nhu cầu đó. Đầu tiên chúng tôi sẽ phân rã (tổ chức) tập hợp các mong muốn của khách hàng thành các khối chức năng, sau đó được ánh xạ thành một kiến trúc. Kiến trúc đó cung cấp tổng hợp các mô-đun phần cứng và phần mềm sẽ tạo nên hệ thống cuối cùng. Bước cuối cùng trong chu trình thiết kế là đưa thiết kế lại với nhau thành một nguyên mẫu và cuối cùng là đưa vào sản xuất.



Bởi vì không có một câu trả lời đúng, bài toán thể hiện một thách thức và một cơ hội để sáng tạo. Một đồng nghiệp từng làm việc trên nhiều thiết kế của một công nghệ đo lường cụ thể đã từng nói, 'mặc dù mỗi thiết kế thực hiện chính xác cùng một chức năng, nhưng mỗi thiết kế cũng đại diện cho một cơ hội để khám phá một cách tiếp cận mới tốt hơn phương pháp cũ.' Người đồng nghiệp đó đã xây dựng sự nghiệp xung quanh việc làm những gì mà mọi người khác nói là không thể làm được... bao gồm một số tên tuổi hàng đầu trong ngành.

Một trong những cách tốt nhất để học cách làm một điều gì đó đơn giản là làm điều đó. Vậy hãy bắt đầu. Khi đi qua từng bước trong quy trình thiết kế mà chúng tôi đã xác định, chúng tôi sẽ thấy cách chúng áp dụng cho thiết kế sau đây. Chúng tôi bắt đầu với một mô tả văn bản.

Là một kỹ sư phát triển cấp cao tại *Thời gian của bạn là tần suất của chúng tôi, Ltd.com*. Bạn vừa hoàn thành một dự án và bây giờ đang sẵn sàng để bắt đầu dự án tiếp theo. Nằm trong kế hoạch ban đầu của dự án đó, bạn và một trong những nhân viên tiếp thị đang đi du lịch khắp đất nước để nói chuyện với những người từ một số công ty kỹ thuật khác nhau. Bạn đang cố gắng xác định những tính năng nào mà khách hàng của bạn muốn thấy trong sản phẩm thế hệ tiếp theo.

Bạn đã đi trên con đường với anh chàng này được vài tuần và rất nóng lòng được về nhà. Tất cả các thành phố bắt đầu giống hệt nhau. Thứ ba, đây phải là Cleveland... hmmm, trông giống như ba thành phố cuối cùng, ồ ồ. Đây là khách hàng cuối cùng cho chuyến đi này. Sáng nay, bạn đang nói chuyện với *High Flying Avionics, Inc*. Họ quan tâm đến một bộ đếm mới có thể được sử dụng trên một số dây chuyền sản xuất thiết bị điện tử hàng không của họ.

Sau vài giờ thảo luận với một trong những nhà quản lý sản xuất, bạn xác định được hầu hết các yêu cầu của họ. Sau cuộc thảo luận của bạn với họ.

Hiện tại, công việc kinh doanh hơi chậm chạp và tiền bạc eo hẹp, vì vậy chúng tôi không có đủ ngân sách để mua nhiều loại nhạc cụ mới khác nhau. Trên thực tế, lý tưởng nhất là chúng tôi muốn có thể sử dụng cùng một công cụ trên một số đường dây của chúng tôi.

Ngày nay, chúng tôi có các kỹ thuật viên của mình chạy hầu hết các bài kiểm tra theo cách thủ công, nhưng trong tương lai, chúng tôi muốn có thể tự động hóa càng nhiều bài kiểm tra này càng tốt. Khi chúng tôi nâng cấp hệ thống của mình, chúng tôi muốn có thể vận hành một số bộ đếm này từ xa từ một PC duy nhất. Dưới đây là một số điều khác mà chúng tôi muốn có thể thực hiện.

Là một phần trong nỗ lực không ngừng của chúng tôi để cải thiện sản xuất và lưu chuyển thông qua các dây chuyền của mình, chúng tôi theo dõi tốc độ các đơn vị đến từng khu vực lắp ráp chính. Để làm được điều đó, chúng tôi cần có thể theo dõi bao nhiêu bộ đàm định vị của chúng tôi đi xuống dây chuyền sản xuất mỗi giờ. Bởi vì chúng tôi hỗ trợ xây dựng số lượng nhỏ các loại bộ đàm khác nhau, tỷ lệ các đơn vị đi qua các điểm giám sát là không cố định. Khi mỗi bộ đàm đến một điểm vào, nó sẽ phá vỡ một chùm tia hồng ngoại. Trên hầu hết các dòng, việc phá vỡ chùm tia tạo ra 1

μ rộng giây, xung âm 5,0 V. Tuy nhiên, chúng tôi có một số dòng cũ hơn mà chúng tôi vẫn phải hỗ trợ. Trên những điều này, xung đang diễn ra tích cực.

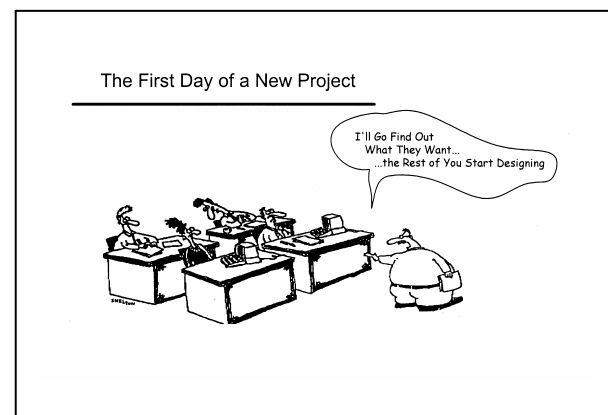
Trên một số dòng mới hơn, chúng tôi phải đo tần số lên đến 150.000 MHz. Chúng tôi cũng có một số bài kiểm tra mà chúng tôi phải đo tần số trong phạm vi  $50\text{KHz} \pm 0,001\text{ KHz}$  và 100 Hz với độ phân giải 0,001 Hz. Trên một dòng khác, chúng tôi có một số công cụ có tín hiệu đầu ra có thời lượng lên đến  $1,0000 \pm 0,0001\text{ms}$  và các loại khác có thời lượng lên đến 9,999 đến 10.000 ms và lên đến  $1.000 \pm 0,001$  giây. Những tín hiệu này không theo chu kỳ. Cuối cùng, chúng ta có một số tín hiệu định kỳ trên cùng các đơn vị đó mà chúng ta phải có thể đo lường với cùng độ chính xác và độ phân giải.

## 12.5 Xác định các Yêu cầu

Sự phát triển của bất kỳ loại hệ thống được hình thành và thiết kế tốt nào đều phải bắt đầu bằng định nghĩa các yêu cầu. Nhu cầu đó tồn tại, không phụ thuộc vào mô hình vòng đời mà người ta chọn để làm việc. Không giống như mọi người

trong bản vẽ kèm theo được diễn giải từ một tác giả không rõ, chúng ta không thể bắt đầu thiết kế cho đến khi chúng ta biết những gì chúng ta phải thiết kế.

Mục tiêu của quá trình xác định yêu cầu là nắm bắt được một mô tả chính thức về hệ thống hoàn chỉnh theo quan điểm của khách hàng, sau đó ghi lại những nhu cầu này dưới dạng định nghĩa bằng văn bản và

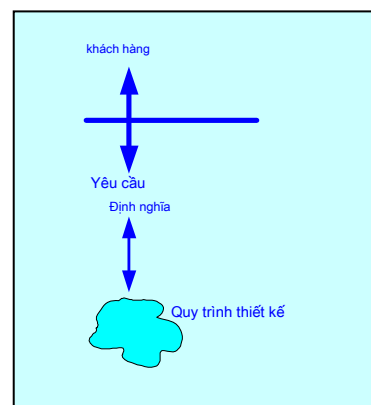


mô tả. Các tài liệu này tạo cơ sở cho các đặc điểm kỹ thuật thiết kế chính thức.

Rất thường xuyên, chúng tôi sử dụng ngôn ngữ tự nhiên của khách hàng và của ngữ cảnh ứng dụng. Chúng tôi làm như vậy vì cách thể hiện chính thức các yêu cầu buộc phải thảo luận sớm và giải quyết nhiều vấn đề phức tạp, liên quan đến nhiều người có chuyên môn trong nhiều lĩnh vực khác nhau; đặc biệt là những người am hiểu về lĩnh vực ứng dụng. Chúng tôi thể hiện vai trò của định nghĩa yêu cầu giữa khách hàng và những người thực hiện thiết kế với

đồ họa đơn giản đi kèm.

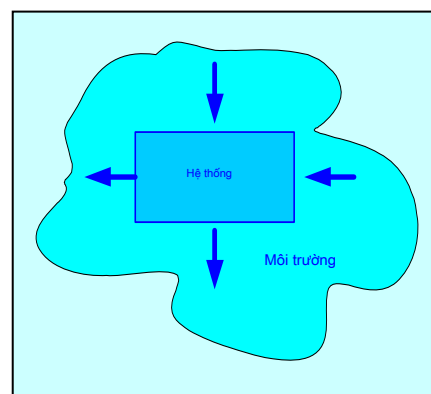
Định nghĩa yêu cầu cung cấp giao diện giữa khách hàng và quy trình kỹ thuật. Nó là bước đầu tiên trong việc chuyển đổi mong muốn của khách hàng thành sản phẩm cuối cùng. Do đó, người ta có thể thấy rằng định nghĩa yêu cầu là một mô tả về một cái gì đó được mong muốn hoặc cần thiết. Nó xác định và nắm bắt một tập hợp các khả năng hoặc hoạt động cần thiết. Khi một người bắt đầu xác định tất cả các yêu cầu, điều quan trọng là phải xem xét *cả hai* hệ thống được thiết kế và môi trường mà nó sẽ hoạt động.

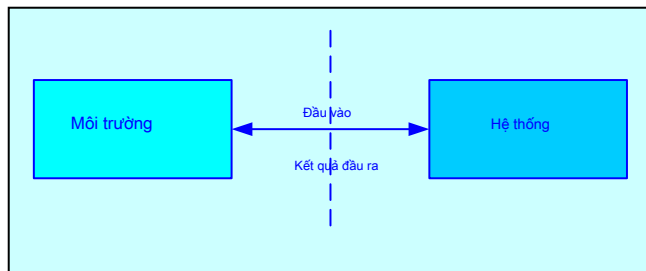


Ở giai đoạn đầu này trong chu kỳ sống của sản phẩm, mục tiêu là nắm bắt và thể hiện những quan điểm thuần túy bên ngoài về môi trường, hệ thống và sự tương tác giữa chúng. Đối với hệ thống, người ta đề cập đến một chế độ xem như là giao diện công cộng của nó. Một người cố gắng xác định *gì* cần phải được thực hiện (và *tốt như thế nào* nó cần được thực hiện) bắt đầu từ nhu cầu và yêu cầu của người dùng.

Chế độ xem đầu tiên của môi trường và của hệ thống có dạng như sau. Rõ ràng là môi trường bao quanh hệ thống. Các đầu vào và đầu ra từ hệ thống có thể đến hoặc đi đến bất kỳ đâu trong môi trường. Khi bắt đầu, người ta không nên đưa ra giả định về mức độ của cả hai.

Bước đầu tiên là tóm tắt và củng cố chế độ xem đó sao cho cả hai đều xuất hiện dưới dạng các hộp đen bên dưới.





Trọng tâm ban đầu phải là thế giới hoặc môi trường (bối cảnh ứng dụng) mà hệ thống sẽ hoạt động. Tiếp theo, phần tiếp theo là mô tả ngày càng chi tiết về vai trò của hệ thống trong môi trường đó và ở mỗi bước sẽ bổ sung và tinh chỉnh các yêu cầu.

Từ quan điểm của môi trường, người ta có thể thấy rằng định nghĩa yêu cầu phải bao gồm đặc điểm kỹ thuật cho môi trường chứa, mô tả / định nghĩa về các đầu vào và đầu ra đến và từ môi trường đó, mô tả hành vi cần thiết của hệ thống và mô tả về cách hệ thống sẽ được sử dụng.

Theo quan điểm của hệ thống, người ta bắt đầu ở mức độ trừu tượng cao với *ở ngoài* lượt xem. Người ta phát triển (các) định nghĩa phù hợp với cấp độ đó. Như đã được thực hiện khi xác định môi trường, thông qua sự tinh chỉnh tiến bộ, người ta chuyển sang mức độ trừu tượng thấp hơn và hiểu và định nghĩa chi tiết hơn.

Ở giai đoạn này trong vòng đời phát triển, khi định nghĩa về các yêu cầu được củng cố và cuối cùng được chính thức hóa thành một đặc điểm kỹ thuật, nên không bị cản trở bởi các kế hoạch thực hiện. Trọng tâm nên tập trung vào hành vi cấp cao của hệ thống. Thông số kỹ thuật đầy đủ, chính xác và nhất quán nội bộ phải có sẵn trước khi người ta có thể bắt đầu thiết kế chính thức. Lý tưởng nhất là nó phải có thể thực thi và do đó có thể hoạt động cùng với một bộ công cụ mô hình hóa. Một đặc tả thực thi như vậy cuối cùng đóng vai trò là cơ sở để xác nhận hệ thống.

Trong khi đặc tả thực thi là một mục tiêu đáng khen ngợi, những khó khăn trong việc đạt được mục tiêu đó có thể xảy ra khi người ta phải bao gồm hỗ trợ cho các ràng buộc phi chức năng, tích hợp các thành phần kế thừa vào một mô hình trừu tượng và có khả năng cần phải kết hợp các ngôn ngữ và ngữ nghĩa miền cụ thể khác nhau.

## 12.6 Xây dựng đặc điểm kỹ thuật yêu cầu

Bây giờ chúng ta hãy xem xét một số điều mà người ta nên nghĩ đến khi bắt đầu xác định và nắm bắt các yêu cầu, sau đó cố gắng xác định chúng trong một đặc tả chính thức. Hình thức, mức độ và hình thức của một thông số kỹ thuật như vậy phụ thuộc vào dự án mà người đó đang làm việc, đối tượng mục tiêu và công ty mà người đó đang làm việc. Hãy nhớ rằng nó là một sản phẩm đang được chuyển giao, không phải là một đồng giấy. Theo nguyên tắc chung, thông số kỹ thuật phải là mức tối thiểu tuyệt đối cần thiết để nắm bắt và xác định rõ ràng tất cả các yêu cầu cần thiết.

Khi nắm bắt các yêu cầu, người ta cố gắng rất cụ thể về các chi tiết theo quan điểm của người dùng. Hãy nhớ rằng người ta đang xác định và chính thức hóa *các yêu cầu*.

Người ta vẫn không thể bắt đầu thiết kế cho đến khi *sự chỉ rõ* đã được hoàn thành và khách hàng đã đồng ý với nó. Hãy nhớ rằng, người ta không nên thảo luận về bộ vi xử lý, bộ nhớ, chip ngoại vi hoặc mô-đun phần mềm tại thời điểm này trong quá trình phát triển.

Khi bắt đầu thiết kế, người ta thường có một số ý tưởng chung, thảo luận bình thường, suy nghĩ, nhưng không có gì chắc chắn.

Người ta có thể sử dụng chúng như một hướng dẫn trong việc chỉ đạo các bước, nhưng người ta không thể thiết kế từ chúng. Tuy nhiên, điều quan trọng là phải cẩn thận, không nên quá phụ thuộc vào những ý tưởng đã định trước. Người ta phải luôn cởi mở với các cách tiếp cận thay thế. Bắt đầu viết mã hoặc vẽ sơ đồ logic tại thời điểm này đang dẫn đến các vấn đề lớn khi dự án tiến hành. Trong tất cả các khả năng, dự án sẽ thất bại.

Đối với môi trường, người ta phải thiết lập một mô tả về tất cả các thực thể có liên quan và các hành vi của tất cả các hoạt động. Người ta phải biết môi trường tương tác với hệ thống như thế nào và các tác động lên môi trường bởi các kết quả đầu ra từ hệ thống. Đối với hệ thống, người ta yêu cầu mô tả tất cả các đầu vào và đầu ra cũng như mô tả đầy đủ về các hành vi chức năng và hoạt động cũng như các ràng buộc công nghệ.

Một cách tự nhiên có thể hỏi tại thời điểm này, làm thế nào người ta có thể có được những thông tin như vậy về (chứ đừng nói đến mô hình) hệ thống và môi trường mà không cần mô tả hoặc biết việc triển khai hệ thống? Nội bộ vốn dĩ vẫn chưa được biết đến vào thời điểm này. Làm thế nào để một người nắm bắt các hành vi mong muốn?

### 12.6.1 Môi trường

Bước đầu tiên hợp lý bắt đầu với việc xác định và mô tả môi trường; thế giới mà hệ thống phải hoạt động. Môi trường là một thế giới thời gian; nó là một tập hợp không đồng nhất của các thực thể thuộc dạng này hay dạng khác. Nó bao gồm tập hợp các thiết bị vật lý mà hệ thống được kết nối với nhau cũng như bất kỳ thuộc tính nào của thế giới vật lý mà hệ thống đang có ý định đo lường hoặc kiểm soát hoặc có thể ảnh hưởng đến hệ thống. Các mục tiêu ban đầu trong việc tìm hiểu môi trường là xác định tất cả các thực thể có liên quan sau đó xác định đặc điểm ảnh hưởng của chúng lên hệ thống và ngược lại. Khi đặc tả yêu cầu đã được hoàn thành, người ta cần có tất cả các thông tin cần thiết về các thực thể đó với đầy đủ chi tiết để hỗ trợ giải quyết vấn đề.

#### 12.6.1.1 Đặc điểm hóa các thực thể bên ngoài

Mỗi thực thể tạo nên môi trường được đặc trưng bởi một tên và một giao diện công khai được trừu tượng hóa. Giao diện đó bao gồm các đầu vào và đầu ra của thực thể cũng như hành vi chức năng của nó. Đặc điểm kỹ thuật của môi trường bên ngoài phải bao gồm những điều sau đây cho mỗi thực thể,

- *Tên và mô tả của thực thể*

Tên phải gợi ý về những gì thực thể đang hoặc hiện. Mô tả phải trình bày bản chất của thực thể. Nó có phải là dữ liệu, một sự kiện, một biến trạng thái, một thông báo, v.v. không?

Ví dụ, một thực thể có thể là thứ cần được kiểm soát, chẳng hạn như bánh lái trên máy bay hoặc luồng không khí rõ ràng phải được tính đến trong một hệ thống kiểm soát như vậy.

- *Trách nhiệm - Hoạt động*

Những hoạt động hoặc hành động mà môi trường dự kiến sẽ thực hiện. Hệ thống thủy lực di chuyển bánh lái là một phần của môi trường. Hành động hoặc trách nhiệm của nó là di chuyển bánh lái theo tín hiệu đến từ hệ thống đang được thiết kế.

- *Các mối quan hệ*

Mối quan hệ giữa đơn vị và các trách nhiệm hoặc hoạt động của đơn vị là gì? Mối quan hệ đó là nhân quả hay phản ứng? Đó là một nhà sản xuất hay một người tiêu dùng?

- *An toàn và độ tin cậy*

Các vấn đề về an toàn và độ tin cậy phải được đưa vào sớm trong quá trình đặc tả. Đối với môi trường, ở giai đoạn yêu cầu, trọng tâm chủ yếu là an toàn. Mục tiêu là xác định tất cả các vấn đề và mối nguy hiểm quan trọng về an toàn để chúng có thể được giải quyết chi tiết trong đặc tả thiết kế hệ thống. Người ta cũng nên xác định bất kỳ cơ quan quản lý nào dưới sự bảo trợ của hệ thống sẽ hoạt động.

## 12.6.2 Hệ thống

Tiếp theo, tiêu điểm chuyển sang quan điểm của hệ thống. Các câu hỏi tương tự đặt ra cho môi trường hiện được hỏi về hệ thống. Cũng như việc mô tả đặc tính của môi trường, các mục tiêu ban đầu là xác định tất cả các khía cạnh của giao diện công cộng của hệ thống, sau đó mô tả các ảnh hưởng của chúng đến môi trường và ngược lại.

### 12.6.2.1 Đặc điểm hóa hệ thống

Đặc tính của hệ thống bắt đầu bằng việc xác định các đầu vào và đầu ra.

- *Đầu vào và đầu ra của hệ thống*

Hệ thống tương tác với thế giới thực thông qua các thực thể được mô tả và xác định trong đặc điểm môi trường. Đầu vào của hệ thống là đầu ra từ các thực thể môi trường và đầu ra từ hệ thống là đầu vào cho các thực thể môi trường. Có thể dễ dàng nhận thấy rằng hệ thống I / O đã được đặc trưng hóa trong đặc tả thực thể môi trường

Đối với mỗi biến I / O như vậy, thông tin sau đã có sẵn,

Tên của tín hiệu

Việc sử dụng tín hiệu làm đầu vào hoặc đầu ra

Bản chất của tín hiệu như một sự kiện, dữ liệu, biến trạng thái, v.v.

Làm việc với đặc tả môi trường, người ta có thể viết cấu trúc, miền hiệu lực và các đặc tính vật lý của mỗi tín hiệu. Đối với những điều này, người ta có thể thêm bất kỳ ràng buộc kỹ thuật hoặc công nghệ nào được xác định.

- *Trách nhiệm - Hoạt động*

Như đã được thực hiện với đặc điểm kỹ thuật của môi trường, bây giờ tập trung chuyển sang chức năng mà hệ thống dự định thực hiện. Trước khi được thiết kế, hệ thống sẽ xuất hiện dưới dạng một hộp đen. Nó chỉ có thể được nhìn từ quan điểm bên ngoài. Một phần về hành vi chức năng hiện được bao gồm trong đặc điểm kỹ thuật.

Mô tả chức năng xác định hành vi bên ngoài của hệ thống. Nó đưa ra đặc điểm về ảnh hưởng của các kết quả đầu ra của hệ thống đối với các thực thể môi trường và phản ứng dự kiến của hệ thống đối với đầu vào từ các thực thể môi trường. Nó trình bày chi tiết cách hệ thống được sử dụng và được sử dụng bởi người dùng. Một đặc tả như vậy tương đương với việc phát triển một mô hình của hệ thống.

Mô tả chức năng có thể được nắm bắt theo nhiều cách khác nhau. Một cách tiếp cận hiệu quả là sử dụng các công cụ UML đã thảo luận trước đó. Người ta có thể xây dựng một khung nhìn như vậy thông qua ca sử dụng và sơ đồ lớp. Một góc nhìn khác có thể đạt được thông qua biểu đồ trạng thái cấp cao và biểu đồ hoạt động; dữ liệu và sơ đồ luồng điều khiển thường được sử dụng trong phương pháp luận thiết kế có cấu trúc đưa ra một phần ba.

Khi người ta xây dựng các sơ đồ này và đặc điểm kỹ thuật, cần phải cẩn thận để đảm bảo rằng các trạng thái được chỉ định (và cuối cùng được mô hình hóa) phù hợp với ứng dụng. Người ta phải chắc chắn rằng các hành động được ghi lại trong đặc điểm kỹ thuật phản ánh chính xác hành vi mong muốn (bên ngoài) của hệ thống như nhận thức và dự định của khách hàng. Trong đặc điểm kỹ thuật, người ta phải đảm bảo rằng các điều kiện hoặc ràng buộc đối với hành vi của nó chỉ là một hàm của các yếu tố đầu vào đi vào hệ thống, các trạng thái được chỉ định, các sự kiện bên trong và khoảng thời gian thích hợp (tương đối hoặc tuyệt đối).

- *An toàn và độ tin cậy*

Trong việc xây dựng các yêu cầu về an toàn và độ tin cậy cho hệ thống, trọng tâm là các mục tiêu cấp cao của mỗi mục tiêu và vào chiến lược để đạt được các mục tiêu đó.



### Các cân nhắc về an toàn cần giải quyết

Các hướng dẫn, quy tắc hoặc quy định về an toàn của các cơ quan quản lý được xác định trong phần môi trường của đặc điểm kỹ thuật.

### Về độ tin cậy, người ta có thể chỉ định

Các mục tiêu về thời gian hoạt động của hệ thống

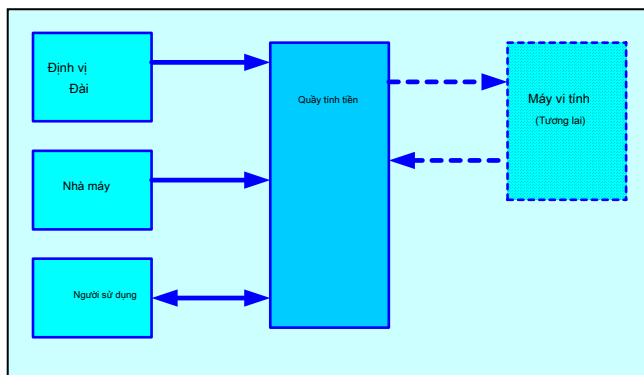
Các rủi ro tiềm ẩn, thất bại và phương thức thất bại Chiến

lược quản lý thất bại

### Thí dụ

Bắt đầu từ báo cáo chuyển đi từ *High Flying Avionics, Inc.*, trong đó đã thảo luận về nhu cầu của họ đối với một bộ đếm mới, hãy cùng nhau đặt các đặc điểm kỹ thuật yêu cầu.

Bước đầu tiên trong quá trình suy nghĩ, người ta trích xuất và tóm tắt các thông tin cần thiết từ báo cáo chuyển đi. Bằng cách đó, người ta có thể bắt đầu tập trung vào những gì cần được bao gồm trong đặc tả yêu cầu. Từ các cuộc thảo luận với khách hàng, một bản phác thảo cấp cao về hệ thống và môi trường nắm bắt được những phần thiết yếu của vấn đề. Bước tiếp theo là bắt đầu chính thức hóa mô hình của hệ thống và môi trường.



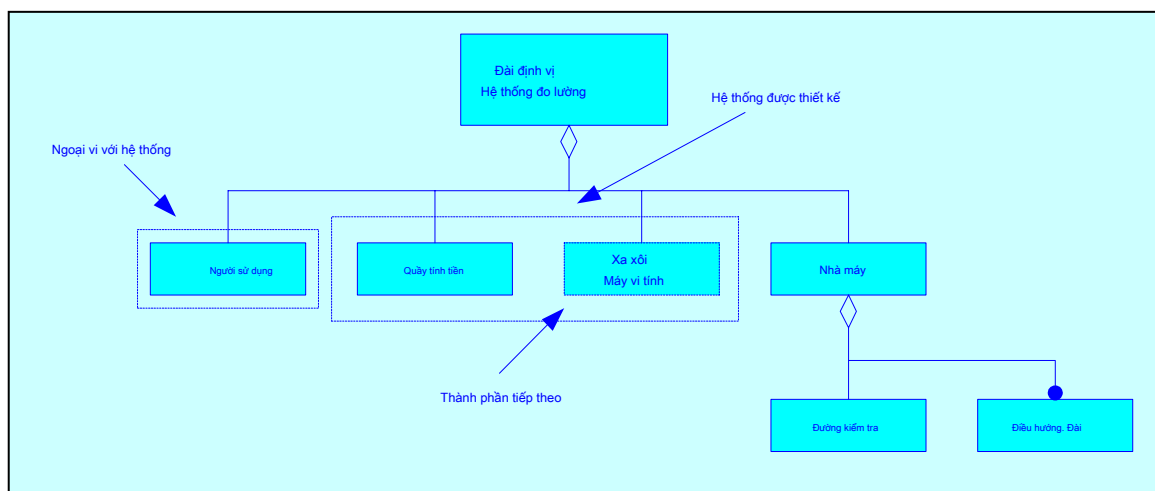
Trong cấu hình ban đầu, môi trường chứa,

- Một bộ radio điều hướng sẽ được thử nghiệm
- Người dùng đang thực hiện thử nghiệm
- Nhà máy

Tín hiệu truyền từ đài điều hướng đến bộ đếm, nhưng không phải ngược lại. Nhà máy cũng có đầu vào cho quầy; chúng bao gồm hệ thống điện và

môi trường xung quanh trong nhà máy. Tương tác của người dùng là hai chiều. Người dùng phải chọn và cấu hình phép đo sẽ thực hiện và sau đó xem kết quả sau khi quá trình đo hoàn tất. Đối với máy tính, việc trao đổi tín hiệu với bộ đếm cũng xảy ra tương tự theo cả hai hướng.

Trong mô hình đang phát triển, nhà máy có thể được xem như một tập hợp các dây chuyền thử nghiệm và các bộ đàm cần thử nghiệm. Sau đó, máy tính từ xa sẽ được thêm vào. Hệ thống được thiết kế, tức là bộ đếm, tương tác với cả ba thực thể. Sự tương tác như vậy được phản ánh trong hình sau.



Bây giờ chuyển sang cấp độ chi tiết tiếp theo,

## Môi trường

- Khách hàng đã nói rằng quây phải hoạt động trong môi trường nhà máy trên bất kỳ dây chuyền sản xuất nào. Dựa trên sự hiểu biết như vậy, người ta có thể đưa ra các giả định nhất định về nhiệt độ, công suất và ánh sáng xung quanh.
- Khoảng thời gian và tần số trên radio điều hướng và các sự kiện từ thiết bị giám sát dây chuyền sản xuất phải được đo
- Khoảng thời gian có thể là chu kỳ hoặc không theo chu kỳ nhưng không thể là cả hai.
- Cực tính của tín hiệu sự kiện được đếm có thể là tích cực hoặc tiêu cực.
- Hiển thị dữ liệu và thông báo cho chế độ và phạm vi là kết quả đầu ra duy nhất được mong đợi từ bộ đếm.

- Giả thiết được đưa ra là các tín hiệu cần đo là độc lập với nhau.
- Trong tương lai, các lệnh sẽ được gửi từ máy tính đến bộ đếm để chỉ đạo hoạt động của nó. Dữ liệu sẽ được gửi từ bộ đếm vào máy tính.

#### Quầy tính tiền

- Bộ đếm phải có khả năng đo khoảng thời gian, tần số và đếm các sự kiện.
- Các tần số là cố định nhưng, trải dài trong một phạm vi giá trị.
- Các khoảng thời gian trải dài một loạt các giá trị và có thể là định kỳ hoặc theo chu kỳ nhưng không thể là cả hai.
- Bộ đếm sẽ hỗ trợ khả năng cho người dùng tự chọn chế độ và phạm vi đo cho tất cả các tín hiệu đầu vào.
- Bộ đếm sẽ tiếp tục thực hiện và hiển thị thuộc tính đã chọn của tín hiệu cho đến khi nguồn điện vào hệ thống bị tắt hoặc cho đến khi người dùng thực hiện lựa chọn khác.
- Bộ đếm sẽ chỉ đo một tín hiệu tại một thời điểm. Một sự kiện có thể được mô
- hình hóa như một tín hiệu thời gian không theo chu kỳ.
- Thiết kế sẽ đủ linh hoạt để cho phép bao gồm khả năng gửi lệnh từ máy tính đến bộ đếm trong tương lai để chỉ đạo hoạt động của nó.
- Phản ứng của bộ đếm đối với các lệnh từ xa sẽ giống như phản ứng của nó đối với các lựa chọn của bảng điều khiển phía trước, ngoại trừ dữ liệu đo được sẽ được gửi từ bộ đếm đến máy tính cũng như màn hình bảng điều khiển phía trước.

Bước tiếp theo là chính thức hóa, trong một đặc điểm kỹ thuật, những gì đã biết về hệ thống sẽ được thiết kế. Tài liệu, S *Yêu cầu của system Đặc điểm kỹ thuật*, mở đầu bằng một bản tóm tắt của thiết kế.

### Đặc điểm kỹ thuật yêu cầu đối với bộ đếm kỹ thuật số

#### Sự mô tả hệ thống

Đặc điểm kỹ thuật này mô tả và xác định các yêu cầu cơ bản đối với bộ đếm kỹ thuật số. Bộ đếm có thể đo tần suất, khoảng thời gian, khoảng thời gian và các sự kiện. Hệ thống hỗ trợ ba phạm vi đo cho mỗi tín hiệu và hai cho các sự kiện. Bộ đếm sẽ được vận hành bằng tay với khả năng hỗ trợ hoạt động từ xa trong tương lai. Bộ đếm phải có chi phí thấp và linh hoạt để nó có thể được sử dụng trong nhiều ứng dụng.

#### Đặc điểm kỹ thuật của môi trường bên ngoài

Bộ đếm hoạt động trong môi trường công nghiệp với nhiệt độ và môi trường ánh sáng cấp thương mại.

Thiết bị sẽ hỗ trợ hoạt động của đường dây và pin.

#### Đặc điểm kỹ thuật đầu vào và đầu ra của hệ thống

##### Đầu vào hệ thống

Hệ thống sẽ có thể đo các tín hiệu sau

Tần suất trong ba phạm vi:

- Phạm vi cao lên đến 150.000 MHz
- Phạm vi trung bình lên đến 50.000 KHz
- Phạm vi thấp lên đến 100.000 Hz

Khoảng thời gian trong ba phạm vi

- Độ phân giải cao lên đến 1,0000 ms
- Độ phân giải trung bình lên đến 10.000 ms
- Độ phân giải thấp lên đến 1.000 giây Khoảng

thời gian trong ba phạm vi

- Độ phân giải cao lên đến 1,0000 ms
- Độ phân giải trung bình lên đến 10.00 ms
- Độ phân giải thấp Sự kiện lên đến 1.000 giây -

lên đến 99 sự kiện trong 1 phút

Tất cả các đầu vào tín hiệu sẽ được

- Dữ liệu kỹ thuật số
- Điện áp 0,0 đến 4,5 VDC

#### Kết quả đầu ra của hệ thống

Hệ thống phải đo và hiển thị các tín hiệu sau bằng màn hình 6 chữ số

Tần suất trong ba phạm vi:

- Phạm vi cao lên đến  $200.000 \pm 0,001$  MHz.
- Phạm vi trung bình lên đến  $200.000 \pm 0,001$  KHz.
- Phạm vi thấp lên đến  $200.000 \pm 0,001$  Hz

Khoảng thời gian trong ba phạm vi

- Độ phân giải cao lên đến  $2.000 \pm 0.0001$  ms
- Độ phân giải trung bình lên đến  $20,00 \pm 0,01$ ms
- Độ phân giải thấp lên đến  $2.000 \pm 0,001$  giây

Khoảng thời gian trong ba phạm vi

- Độ phân giải cao lên đến  $2,0000 \pm 0,0001$  ms
- Độ phân giải trung bình lên đến  $20,00 \pm 0,01$  mili giây
- Độ phân giải thấp lên đến  $2.000 \pm 0,001$  giây

Sự kiện trong hai phạm vi

- Nhanh lên 200 sự kiện trong 1 phút 2000
- Làm chậm đến sự kiện trong 1 giờ

#### Giao diện người dùng

Người dùng sẽ có thể chọn các điều sau đây bằng cách sử dụng các nút và công tắc trên bảng điều khiển phía trước của thiết bị

Chế độ:

Tần suất, Khoảng thời gian, Khoảng thời gian, Phạm vi sự kiện

Tần suất, Khoảng thời gian, Khoảng thời gian - Sự kiện Cao, Trung bình, Thấp

- Nhanh, Chậm

Kích hoạt cạnh

Tần suất, Khoảng thời gian và Sự kiện

Khoảng thời gian cạnh tăng

hoặc giảm

Tăng tới cạnh tăng Giảm

xuống cạnh giảm Tăng tới

cạnh giảm Giảm xuống

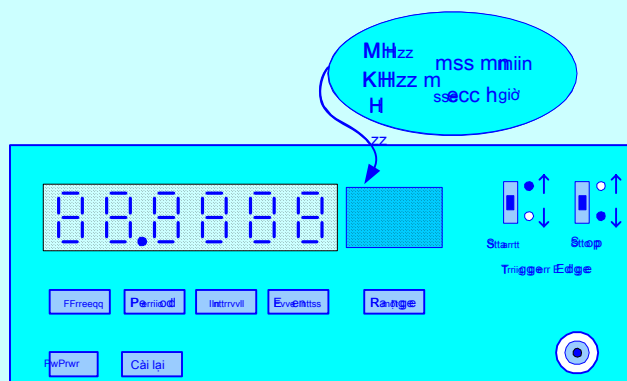
cạnh tăng

Cài lại

Bật / tắt nguồn

Kết quả đo phải được trình bày trên màn hình 6 chữ số; các số không ở đầu sẽ bị triệt tiêu. Màn hình phải có thể đọc được dưới ánh sáng mặt trời trực tiếp và từ mọi góc độ.

Bảng điều khiển phía trước sẽ xuất hiện như sau,



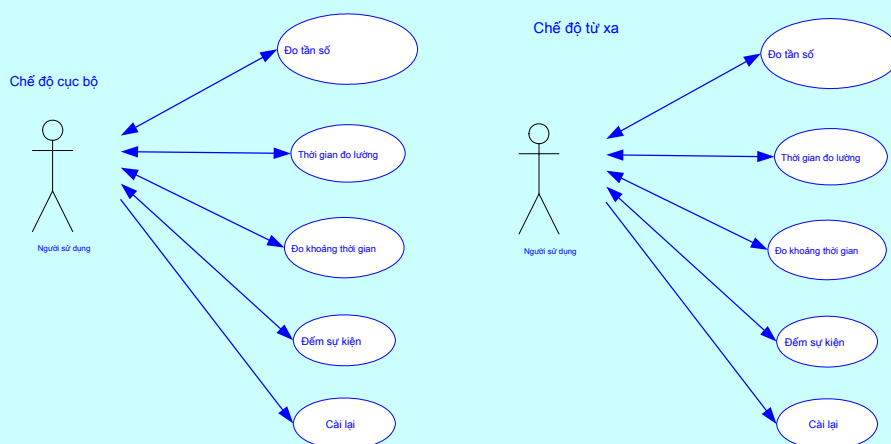
### Trường hợp sử dụng

Các trường hợp sử dụng cho bộ đếm được cho trong hai sơ đồ sau. Đầu tiên cho biết hoạt động thủ công thông qua bảng điều khiển phía trước và thứ hai thông qua kết nối từ xa với máy tính.

Tùy chọn từ xa sẽ không được bao gồm trong mô hình ban đầu, nhưng sẽ được kết hợp trong bản phát hành sau. Thời điểm phát hành sẽ được xác định.

Việc thực thi chức năng đo đã chọn sẽ không phụ thuộc vào cách (cục bộ hoặc từ xa) chức năng đó được chọn.

Khi BẬT nguồn, chế độ mặc định là *Đo tần số*. Tất cả các phạm vi sẽ mặc định là giá trị cao nhất của chúng.



**Đo tần số** Bộ đếm sẽ liên tục đo và hiển thị tần số của tín hiệu đầu vào trên phạm vi hiện đang được chọn miễn là *Tần số* chế độ được chọn.

Nếu tần số của tín hiệu đầu vào vượt quá giá trị cho phép tối đa trên phạm vi đã chọn, màn hình sẽ hiển thị giá trị đọc đầy đủ và sẽ nhấp nháy.

Nếu tần số của tín hiệu đầu vào thấp hơn giá trị cho phép tối thiểu trên phạm vi đã chọn, màn hình sẽ hiển thị số đọc bằng không.

Nếu tín hiệu đầu vào trở về giá trị trong giới hạn của dải, giá trị của tần số sẽ được hiển thị.

Phạm vi có thể được thay đổi bất cứ lúc nào bằng cách nhấn vào *chọn phạm vi* nút bấm. Việc sử dụng có thể chọn để đo tần số bắt đầu trên cạnh dương hoặc âm của tín hiệu bằng cách nhấn *bắt đầu kích hoạt cạnh* nút bấm.

**Thời gian đo lường** Bộ đếm sẽ liên tục đo và hiển thị chu kỳ của tín hiệu đầu vào trên phạm vi hiện được chọn miễn là *Giai đoạn = Stage* chế độ được chọn.

Nếu khoảng thời gian của tín hiệu đầu vào vượt quá giá trị cho phép tối đa trên phạm vi đã chọn, màn hình sẽ hiển thị giá trị đọc đầy đủ và sẽ nhấp nháy. Nếu khoảng thời gian của tín hiệu đầu vào thấp hơn giá trị cho phép tối thiểu trên phạm vi đã chọn, màn hình sẽ hiển thị số đọc bằng không.

Nếu tín hiệu đầu vào trở về một giá trị trong giới hạn của phạm vi, giá trị của khoảng thời gian sẽ được hiển thị.

Phạm vi có thể được thay đổi bất cứ lúc nào bằng cách nhấn vào *chọn phạm vi* nút bấm.

Việc sử dụng có thể chọn để đo khoảng thời gian bắt đầu trên cạnh dương hoặc âm của tín hiệu bằng cách nhấn *bắt đầu kích hoạt cạnh* nút bấm.

**Đo khoảng thời gian** Bộ đếm sẽ liên tục đo và hiển thị thời lượng của phần đã chọn của tín hiệu đầu vào trên phạm vi hiện được chọn miễn là *Khoảng thời gian* chế độ được chọn.

Nếu thời lượng của phần đã chọn của tín hiệu đầu vào vượt quá giá trị tối đa cho phép trên phạm vi đã chọn, màn hình sẽ hiển thị giá trị đọc đầy đủ và sẽ nhấp nháy.

Nếu thời lượng của phần đã chọn của tín hiệu đầu vào thấp hơn giá trị cho phép tối thiểu trên phạm vi đã chọn, màn hình sẽ hiển thị bằng không.

Nếu tín hiệu đầu vào trở về một giá trị trong giới hạn của phạm vi, giá trị của khoảng thời gian của phần đã chọn của tín hiệu đầu vào sẽ được hiển thị.

Phạm vi có thể được thay đổi bất cứ lúc nào bằng cách nhấn vào *chọn phạm vi* nút bấm.

Người dùng có thể chọn bắt đầu đo khoảng thời gian trên cạnh dương hoặc âm của tín hiệu bằng cách nhấn *bắt đầu kích hoạt cạnh* nút bấm.

Người dùng có thể chọn kết thúc khoảng thời gian đo trên cạnh dương hoặc âm của tín hiệu bằng cách nhấn *dừng kích hoạt cạnh* nút bấm.

Lưu ý rằng khoảng thời gian của tín hiệu từ cạnh dương sang cạnh dương hoặc từ cạnh âm đến cạnh âm giống với chu kỳ của tín hiệu.



**Sự kiện** Bộ đếm sẽ liên tục đếm và hiển thị số lần xuất hiện của tín hiệu đầu vào trên phạm vi đang được chọn. Số lượng tích lũy sẽ được đặt lại về 0 khi kết thúc khoảng thời gian đếm đã chọn.

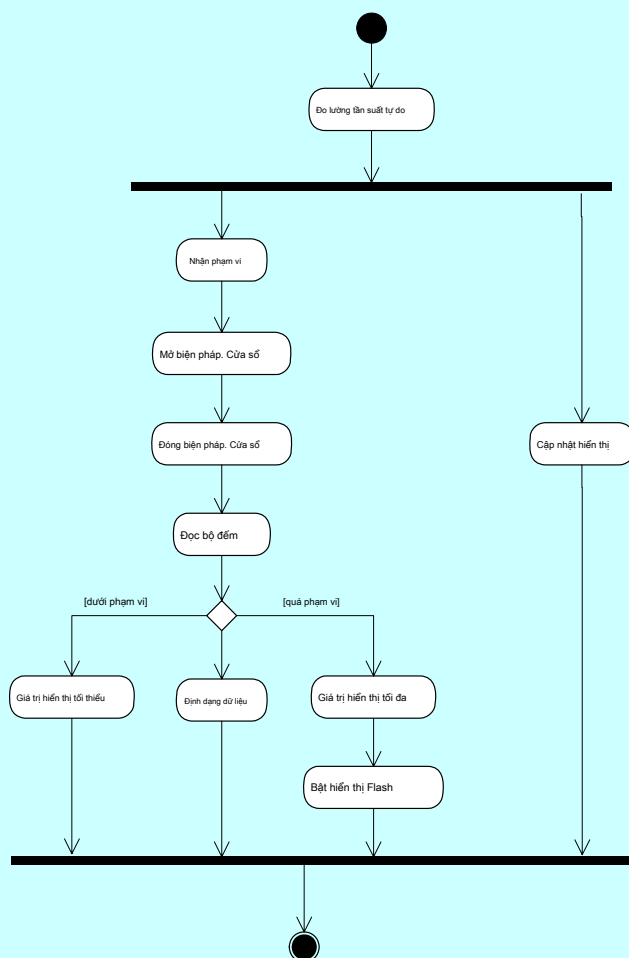
Phạm vi có thể được thay đổi bất cứ lúc nào bằng cách nhấn vào *chọn phạm vi* nút bấm.

Người dùng có thể chọn tăng số đếm trên cạnh dương hoặc âm của tín hiệu đầu vào bằng cách nhấn *bắt đầu kích hoạt cạnh* nút bấm.

Nếu số lượng tích lũy vượt quá giá trị tối đa cho phép trên phạm vi đã chọn, màn hình sẽ hiển thị giá trị đọc đầy đủ và sẽ nhấp nháy.

#### Đặc điểm kỹ thuật chức năng hệ thống

Hệ thống được thiết kế để thực hiện 4 loại phép đo kỹ thuật số khác nhau trong miền thời gian và tần số bao gồm tần số, khoảng thời gian, khoảng thời gian và sự kiện. Các hoạt động liên quan đến *Đo tần số* chế độ được hiển thị trong sơ đồ sau.



Các phép đo thời gian và tần số sẽ được thực hiện để cung cấp ba phạm vi độ phân giải có thể lựa chọn của người dùng, dải tần số cao / tín hiệu thời lượng ngắn hơn, một giây cho tín hiệu tần số trung bình / thời lượng tầm trung và một phần ba cho tín hiệu tần số thấp / thời lượng dài hơn. Khả năng đo lường sự kiện sẽ hỗ trợ hai khoảng thời gian đếm có thể lựa chọn, ngắn hơn và dài hơn.

Đối với các phép đo tần suất, chu kỳ và sự kiện, người dùng sẽ có thể chọn bộ kích hoạt cạnh dương hoặc âm. Đối với các phép đo khoảng thời gian, người dùng sẽ có thể chọn cực tính của tín hiệu khởi động và dừng một cách độc lập.

#### Thông số kỹ thuật hoạt động

Hệ thống sẽ hoạt động trong môi trường thương mại / công nghiệp tiêu chuẩn

Phạm vi nhiệt độ 0-85C

Độ ẩm lên đến 90% RH không ngưng tụ

Nguồn 120-240 VAC 50 Hz, 60 Hz, 400 Hz, 15 VDC

Hệ thống sẽ hoạt động tối thiểu 8 giờ khi pin đã được sạc đầy. Cơ sở thời gian của hệ thống phải đáp ứng các thông số kỹ thuật sau

Nhiệt độ ổn định 0-50 C

$<6 \times 10^{-6}$

Tỷ lệ lão hóa

90 ngày  $<3 \times 10^{-8}$

6 tháng  $<6 \times 10^{-7}$

1 năm  $<25 \times 10^{-6}$

#### Đặc điểm kỹ thuật về độ tin cậy và an toàn

Bộ đếm phải tuân theo các tiêu chuẩn thích hợp

An toàn: UL-3111-1, IEC-1010, CSA 1010.1 EMC:

CISPR-11, IEC 801-2, -3, -4, EN50082-1 MTBF: Tối thiểu

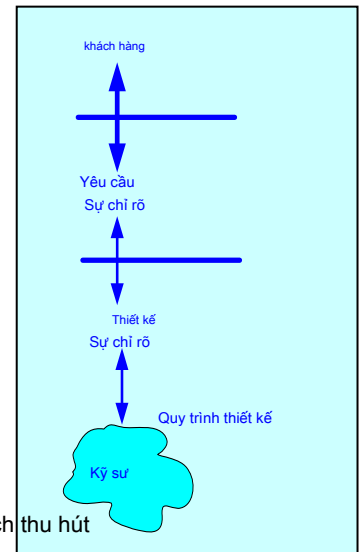
10.000 giờ

## 12.7 Đặc điểm kỹ thuật thiết kế hệ thống

Các *Đặc điểm thiết kế hệ thống* dựa trên *Đặc tả Yêu cầu Hệ thống*. Nó chỉ định *làm sao* của thiết kế không phải *gì*. Đặc điểm kỹ thuật được viết bằng ngôn ngữ của nhà thiết kế và theo quan điểm của nhà thiết kế. Nó đóng vai trò là cầu nối giữa khách hàng và nhà thiết kế như chúng ta thấy trong đồ họa đang phát triển của mình.

Các *Yêu cầu kỹ thuật* cung cấp một cái nhìn từ bên ngoài của hệ thống nhìn vào trong; các *Đặc điểm kỹ thuật thiết kế* cung cấp một cái nhìn từ bên trong nhìn ra ngoài. Cũng lưu ý rằng *Đặc điểm kỹ thuật thiết kế* có hai chủ,

- Nó phải chỉ định giao diện công cộng của hệ thống từ bên trong hệ thống.
- Nó phải chỉ định *làm sao* các yêu cầu được xác định cho và bởi giao diện công cộng phải được đáp ứng bởi các chức năng nội bộ của hệ thống.



Chúng tôi đã thấy rằng *Yêu cầu kỹ thuật* được viết bằng những thuật ngữ ít trang trọng hơn với mục đích thu hút cái nhìn của khách hàng về sản phẩm. Các *Đặc điểm kỹ thuật thiết kế* phải chính thức hóa các yêu cầu đó bằng ngôn ngữ chính xác, rõ ràng. Tạm thời gạt những thay đổi không thể tránh khỏi xảy ra trong thời gian tồn tại của bất kỳ dự án nào, đặc điểm kỹ thuật thiết kế phải đủ rõ ràng, mạnh mẽ và hoàn chỉnh để một nhóm kỹ sư có thể phát triển sản phẩm

mà không bao giờ nói chuyện với tác giả của thông số kỹ thuật.

### 12.7.1 Hệ thống

Là một phần của việc chính thức hóa và định lượng

yêu cầu của hệ thống, người ta phải gắn các con số, dung sai và ràng buộc cụ thể cho tất cả các tín hiệu đầu vào và đầu ra của hệ thống. Tất cả các mối quan hệ thời gian phải được xác định. Các hành vi hoạt động và chức năng của hệ thống được mô tả chi tiết.

#### *Ghi chú thiết kế,*

Một bài kiểm tra tốt về khả năng tồn tại của một đặc điểm kỹ thuật thiết kế là câu hỏi, 'Nếu tôi gửi thông tin này cho đồng nghiệp của tôi (người đang làm việc cho một trong những nhà thầu phụ của chúng tôi), anh ấy hoặc cô ấy có hiểu điều này không?' Nếu câu trả lời là không, thông số kỹ thuật phải được tái khám.

### 12.7.1.1 Định lượng hệ thống

Việc lượng hóa các đặc tính của hệ thống bắt đầu với các đầu vào và đầu ra, dựa trên các yêu cầu đã định. Các chi tiết kỹ thuật cần thiết được thêm vào để cho phép kỹ sư thực hiện chính xác và trung thực thiết kế thực tế.

- *Đầu vào và đầu ra của hệ thống*

Đối với mỗi biến I / O, những điều sau được chỉ định

#### Tên của tín hiệu

Việc sử dụng tín hiệu làm đầu vào hoặc đầu ra

Bản chất của tín hiệu như một sự kiện, dữ liệu, biến trạng thái, v.v.

Bắt đầu với đặc điểm kỹ thuật yêu cầu, chúng tôi cung cấp các mô tả chi tiết khi cần thiết và kết hợp bất kỳ ràng buộc kỹ thuật hoặc công nghệ bổ sung nào có thể cần thiết.

Đặc điểm kỹ thuật đầy đủ của tín hiệu bao gồm giá trị danh nghĩa, phạm vi, dung sai mức, thời gian, dung sai thời gian.

Mối quan hệ qua lại với các tín hiệu khác bao gồm bất kỳ ràng buộc nào đối với các mối quan hệ đó.

- *Trách nhiệm - Hoạt động*

Đặc điểm kỹ thuật hoạt động và chức năng

Các đặc tả chức năng và hoạt động sẽ định lượng hành vi động của hệ thống hiện đã được xây dựng. Đặc tả yêu cầu chức năng xác định các chức năng chính mà hệ thống phải thực hiện từ chế độ xem cấp cao. Đặc tả hoạt động cố gắng nắm bắt các chi tiết cụ thể về cách các chức năng đó hoạt động trong bối cảnh của môi trường hoạt động.

Cách thức mà một chức năng cụ thể phải hoạt động, các điều kiện áp đặt cho hoạt động và phạm vi của hoạt động đó hiện đã được nắm bắt. Đặc điểm kỹ thuật phải xem xét các con số cụ thể - độ chính xác và dung sai.

Tất cả các biến trong đặc điểm kỹ thuật chức năng, tất cả các điều kiện hoạt động và tất cả các chế độ hoạt động bình thường và bất thường phải được định lượng. Các đặc điểm kỹ thuật

có thể bao gồm kiến thức về miền cụ thể mà khách hàng đã biết về độc quyền hoặc theo kinh nghiệm. Những kiến thức như vậy có thể rất quan trọng đối với thiết kế.

Khi nêu các yêu cầu thiết kế cụ thể cho hệ thống, người ta có thể sử dụng bảng, phương trình hoặc thuật toán, ngôn ngữ thiết kế chính thức hoặc mã giả, sơ đồ luồng hoặc sơ đồ UML chi tiết như biểu đồ trạng thái, biểu đồ trình tự và mốc thời gian. Sơ đồ, mã hoặc danh sách các bộ phận ngoại trừ trong một số trường hợp hạn chế không được bao gồm.

### Thông số kỹ thuật công nghệ (và khác)

#### Phần công nghệ bao gồm tất cả các thông số kỹ thuật chi tiết và cụ thể

liên quan đến thiết kế của phần cứng và phần mềm hệ thống. Có thể dễ dàng xác định sáu lĩnh vực cần được xem xét.

##### 1. Ràng buộc về địa lý

Các ứng dụng phân tán có thể trải dài trong một phòng đơn, mở rộng để bao gồm một nhà máy hoàn chỉnh hoặc bao gồm một số quốc gia. Do đó, người ta phải giải quyết cả các hạng mục kỹ thuật như cấu trúc liên kết kết nối, phương pháp truyền thông, hạn chế sử dụng, và ô nhiễm môi trường cũng như các vấn đề phi kỹ thuật như chi phí liên quan đến phương tiện vật lý và việc lắp đặt nó.

##### 2. Đặc tính hóa và các ràng buộc đối với tín hiệu giao diện

Giả định được đặt ra rằng các tín hiệu giữa hệ thống và thế giới bên ngoài là tín hiệu điện, quang học hoặc không dây hoặc chúng có thể được chuyển đổi thành hoặc từ dạng như vậy. Đặc tính vật lý cần thiết của mỗi loại rõ ràng sẽ phụ thuộc vào loại tín hiệu. Đó là, một tín hiệu điện được chỉ định khác với một tín hiệu quang.

Vì nhiều tín hiệu giao diện có thể được điều khiển bởi môi trường bên ngoài, có khả năng chúng nằm ngoài tầm kiểm soát của nhà thiết kế. Do đó, điều quan trọng là phải thu được càng nhiều thông tin về chúng càng tốt.

##### 3. Yêu cầu giao diện người dùng

Nếu hệ thống giao tiếp với các thiết bị thế giới bên ngoài như thiết bị y tế hoặc thiết bị đo đạc, thì thông tin được trình bày như thế nào và liệu có

là bất kỳ giao thức liên quan và có liên quan nào phải được xem xét. Cũng có thể có các tiêu chuẩn chi phối cách trình bày thông tin đó.

Cân nhắc rủi ro đáng kể sẽ phát sinh nếu mỗi nhà cung cấp thiết bị điện tử hàng không trình bày thông tin chuyến bay quan trọng và các biện pháp kiểm soát cho phi công máy bay theo một cách khác. Thảm họa sắp xảy ra tại Three Mile Island, một phần là do sự nhầm lẫn do quá nhiều thông tin gây ra.

#### 4. Những ràng buộc về thời gian

Hệ thống có thể phải thực hiện dưới các ràng buộc thời gian thực cứng hoặc mềm. Những ràng buộc như vậy có thể chỉ định độ trễ đối với các tín hiệu bắt nguồn từ các thực thể bên ngoài, phản hồi đối với đầu ra hệ thống của các thực thể bên ngoài và / hoặc độ trễ hệ thống nội bộ.

#### 5. Cân nhắc về cơ sở hạ tầng điện

Phải có đặc điểm kỹ thuật và đặc tính điện của bất kỳ cơ sở hạ tầng điện nào. Bao gồm trong phần này của thông số kỹ thuật là tiêu thụ điện năng, nguồn cung cấp điện cần thiết, dung sai và công suất của các nguồn cung cấp đó, khả năng chịu đựng nguồn điện bị suy giảm và sơ đồ quản lý nguồn điện.

- *An toàn và độ tin cậy*

Khi xây dựng các yêu cầu thiết kế về độ an toàn và độ tin cậy của hệ thống, trọng tâm chuyển sang các mục tiêu chi tiết của từng mục tiêu và chiến lược để đạt được các mục tiêu đó.

Các cân nhắc về an toàn cần giải quyết

Hiểu và xác định rõ bất kỳ vấn đề môi trường và an toàn nào. Đặc điểm kỹ thuật về độ tin cậy nên bao gồm

Yêu cầu đối với kiểm tra chẩn đoán, bảo trì từ xa, nâng cấp từ xa và chi tiết của chúng.

Số cụ thể cho MTTF và MTBF của bất kỳ mạch tự kiểm tra nào được tích hợp sẵn. Phải có con số cụ thể cho MTTF và MTBF của chính hệ thống. Hiệu suất của hệ thống bị hỏng một phần hoặc toàn bộ phải được xem xét.

Bây giờ chúng ta hãy tập hợp mọi thứ lại với nhau,

#### *Ví dụ tiếp tục...*

Bây giờ chúng tôi sẽ tiếp tục với sự phát triển của quây. Các *Đặc điểm kỹ thuật thiết kế*

sẽ theo sau, nhưng mở rộng, những gì đã được nắm bắt trong *Yêu cầu kỹ thuật*.

Bây giờ trọng tâm sẽ là cung cấp các con số, phạm vi và dung sai cụ thể cho các tín hiệu trong hệ thống.

Một lần nữa, chúng ta sẽ cùng nhau đưa ra bất kỳ suy nghĩ nào về môi trường và hệ thống trước khi viết thông số kỹ thuật.

#### Môi trường

Các thông số kỹ thuật liên quan đến môi trường đã được thảo luận trước đó. Không có thay đổi nào ở đây.

#### Quây tính tiền

- Khi chỉ định thiết bị đo lường và kích thích, thông số kỹ thuật của thiết bị đó thường tốt hơn 10 lần (một bậc của cường độ) so với các tín hiệu phải đo hoặc tạo ra.

Biên độ đó được cung cấp khi chỉ định phạm vi và dung sai trên khả năng đo lường của bộ đếm.

- Thông số kỹ thuật về đếm sự kiện dựa trên mức độ chi tiết của thời gian của khoảng thời gian mà các sự kiện được đếm.
- Các giá trị được hiển thị tại các ranh giới đo lường hiện đã được xác định.

Bước tiếp theo là cung cấp bất kỳ chi tiết bổ sung nào có thể cần thiết và định lượng đầy đủ các thông số kỹ thuật của bộ đếm.

### Đặc điểm kỹ thuật thiết kế cho một bộ đếm kỹ thuật số

#### Sự mô tả hệ thống

Đặc điểm kỹ thuật này mô tả và xác định các yêu cầu thiết kế chi tiết cho bộ đếm kỹ thuật số. Bộ đếm có thể đo tần suất, khoảng thời gian, khoảng thời gian và các sự kiện. Hệ thống hỗ trợ ba phạm vi đo cho mỗi tín hiệu và hai cho các sự kiện. Bộ đếm sẽ được vận hành bằng tay với khả năng hỗ trợ hoạt động từ xa trong tương lai. Bộ đếm phải có chi phí thấp và linh hoạt để nó có thể được sử dụng trong nhiều ứng dụng.

#### Đặc điểm kỹ thuật của môi trường bên ngoài

Bộ đếm hoạt động trong môi trường công nghiệp với nhiệt độ và môi trường ánh sáng cấp thương mại.

Thiết bị sẽ hỗ trợ hoạt động của đường dây và pin. Chi tiết cụ thể được bao gồm trong Thông số kỹ thuật hoạt động.

#### Đặc điểm kỹ thuật đầu vào và đầu ra của hệ thống

##### Đầu vào hệ thống

Hệ thống sẽ có thể đo các tín hiệu sau

Tần suất trong ba phạm vi:

- Phạm vi cao lên đến 150.000 MHz
- Phạm vi trung bình lên đến 50.000 KHz
- Phạm vi thấp lên đến 100.000 Hz

Khoảng thời gian trong ba phạm vi

- Độ phân giải cao lên đến 1,0000 ms
- Độ phân giải trung bình lên đến 10.000 ms
- Độ phân giải thấp lên đến 1.000 giây Khoảng

thời gian trong ba phạm vi

- Độ phân giải cao lên đến 1,0000 ms
- Độ phân giải trung bình lên đến 10.00 ms
- Độ phân giải thấp lên đến 1.000 giây

Sự kiện

- Sự kiện tới 99 mỗi phút
- Mức tín hiệu 0-4.0 V  $\pm$  0,5 V
- Thời gian chuyển tiếp 10 ns  $\leq$  T tăng lên, T ngà  $\leq$  50 ns



### Độ nhạy điện áp

- 50 mV RMS sang  $\pm$  Tín hiệu xoay chiều 5.0 V + tín hiệu dc

Tất cả các đầu vào tín hiệu sẽ được

- Dữ liệu kỹ thuật số
- Dải điện áp 0,0 đến 4,5 VDC

#### Kết quả đầu ra của hệ thống

Hệ thống phải đo và hiển thị các tín hiệu sau bằng màn hình 6 chữ số

Tần suất trong ba phạm vi:

- **Tầm cao**

Đo lường:	0 - 200 $\pm$ 0,0001 MHz
Trưng bày:	200.000 MHz.
- **Phạm vi trung bình lên đến** 200.000 KHz.
 

Đo lường:	0 - 200 $\pm$ 0,0001 KHz
Trưng bày:	0 - 200.000 KHz.
- **Phạm vi thấp lên đến** 200.000 Hz
 

Đo lường:	0 - 200 $\pm$ 0,0001 Hz
Trưng bày:	0 - 200.000 Hz.

Khoảng thời gian trong ba phạm vi

- **Độ phân giải cao lên đến** 2,0000 ms
 

Đo lường:	0 - 2,00000 $\pm$ 0,00001 ms
Trưng bày:	0 - 2.0000 $\pm$ 0.0001 ms
- **Độ phân giải trung bình lên đến** 20,00 ms
 

Đo lường:	0 - 20.0000 $\pm$ 0.0001 ms
Trưng bày:	0 - 20.000 $\pm$ 0,001 ms
- **Độ phân giải thấp lên đến** 2.000 giây
 

Đo lường:	0 - 2.0000 $\pm$ 0.0001 giây
Trưng bày:	0 - 2.000 $\pm$ 0,001 giây

## Khoảng thời gian trong ba phạm vi

- Độ phân giải cao lên đến 2,0000 ms
 

Đo lường:	$0 - 2,00000 \pm 0,00001 \text{ ms}$
Trưng bày:	$0 - 2.0000 \pm 0.0001 \text{ ms}$
- Độ phân giải trung bình lên đến 20,00 ms
 

Đo lường:	$0 - 20.0000 \pm 0.0001 \text{ ms}$
Trưng bày:	$0 - 20.000 \pm 0,001 \text{ ms}$
- Độ phân giải thấp lên đến 2.000 giây
 

Đo lường:	$0 - 2.0000 \pm 0.0001 \text{ giây}$
Trưng bày:	$0 - 2.000 \pm 0,001 \text{ giây}$

## Sự kiện trong hai phạm vi

- Nhanh chóng lên đến 200 sự kiện trong 1 phút
 

Đo lường:	$0 - 200 \pm 1 \text{ sự kiện}$
Trưng bày:	$0 - 200 \pm 1 \text{ sự kiện}$
- Làm chậm tới 2000 sự kiện trong 1 giờ
 

Đo lường:	$0-2000 \pm 1 \text{ sự kiện}$
Trưng bày:	$0-2000 \pm 1 \text{ sự kiện}$

## Giao diện người dùng

Người sử dụng có thể chọn các điều sau đây bằng cách sử dụng các nút và công tắc trên bảng điều khiển phía trước của thiết bị.

Các nút ấn chọn chế độ được lồng vào nhau để đảm bảo rằng chỉ có thể chọn một chế độ tại một thời điểm.

Chế độ:

Tần suất, Khoảng thời gian, Khoảng thời gian, Phạm vi sự kiện

Tần suất, Khoảng thời gian, Khoảng thời gian - Sự kiện Cao, Trung bình, Thấp

- Nhanh, Chậm

## Kích hoạt cạnh

## Tần suất và Chu kỳ

## Cạnh tăng hoặc giảm

## Khoảng thời gian

Tăng tới cạnh tăng Giảm  
xuống cạnh giảm Tăng tới  
cạnh giảm Giảm xuống  
cạnh tăng

## Cài lại

Nút đặt lại sẽ xóa màn hình về tất cả các số 0 và đặt lại chuỗi thời gian / đếm nội bộ.

Bộ đếm sẽ được đặt trong *tần số* chế độ với *phạm vi* đặt thành KHz và *cạnh kích hoạt* đặt thành *trở lại*.

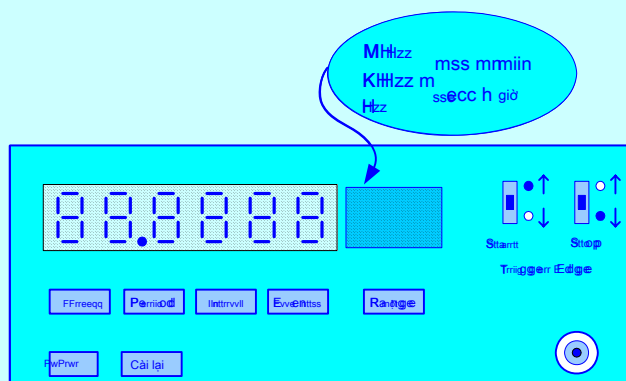
## Bật / tắt nguồn

Kết quả đo phải được hiển thị trên màn hình LED 6 chữ số; các số không ở đầu sẽ bị triệt tiêu.

Dấu thập phân sẽ di chuyển để phản ánh giá trị thích hợp cho phạm vi đã chọn khi nhấn nút nhấn phạm vi được nhấn.

Màn hình phải có thể đọc được dưới ánh sáng mặt trời trực tiếp và từ mọi góc độ.

Bảng điều khiển phía trước sẽ xuất hiện như sau,



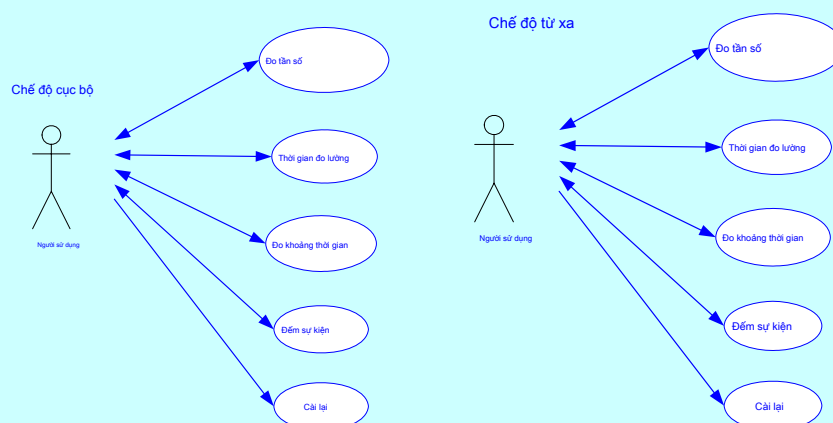
### Trường hợp sử dụng

Các trường hợp sử dụng cho bộ đếm được cho trong hai sơ đồ sau. Thao tác đầu tiên được chỉ định bằng tay thông qua bảng điều khiển phía trước và thao tác thứ hai thông qua kết nối từ xa với máy tính.

Tùy chọn từ xa sẽ không được bao gồm trong mô hình ban đầu, nhưng sẽ được kết hợp trong bản phát hành sau. Thời điểm phát hành sẽ được xác định.

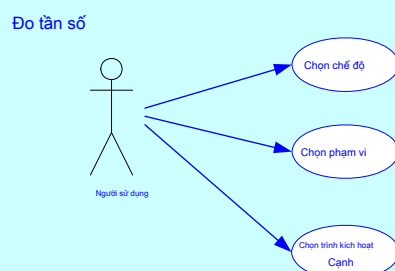
Việc thực thi chức năng đo đã chọn sẽ không phụ thuộc vào cách (cục bộ hoặc từ xa) chức năng đó được chọn.

Khi BẬT nguồn, chế độ mặc định là Tần số đo. Tất cả các phạm vi sẽ mặc định là giá trị cao nhất của chúng.



**Đo tần số** Bộ đếm sẽ liên tục đo và hiển thị tần số của tín hiệu đầu vào trên phạm vi hiện đang được chọn miễn là

*Tần số* chế độ được chọn. Các trường hợp sử dụng sau được xác định cho *đo tần số* chế độ.



Nếu tần số của tín hiệu đầu vào vượt quá giá trị tối đa cho phép trên phạm vi đã chọn, màn hình sẽ nhấp nháy và hiển thị một trong các giá trị sau dựa trên phạm vi đã chọn,

- 200.000 MHz
- 200.000 KHz.
- 200.000 Hz

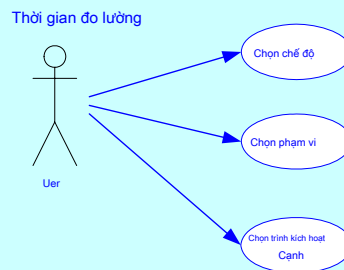
Nếu tần số của tín hiệu đầu vào thấp hơn giá trị cho phép tối thiểu trên phạm vi đã chọn, màn hình sẽ hiển thị số đọc bằng không.

Nếu tín hiệu đầu vào trở về giá trị trong giới hạn của dải, giá trị của tần số sẽ được hiển thị.

Phạm vi có thể được thay đổi bất cứ lúc nào bằng cách nhấn vào *chọn phạm vi* nút bấm.

Việc sử dụng có thể chọn để đo tần số bắt đầu trên cạnh dương hoặc âm của tín hiệu bằng cách nhấn *bắt đầu kích hoạt cạnh* nút bấm.

**Thời gian đo lường** Bộ đếm sẽ liên tục đo và hiển thị chu kỳ của tín hiệu đầu vào trên phạm vi hiện được chọn miễn là *Giai đoạn = Stage* chế độ được chọn. Các trường hợp sử dụng sau được xác định cho *đo thời gian chế độ*.



Nếu khoảng thời gian của tín hiệu đầu vào vượt quá giá trị cho phép tối đa trên phạm vi đã chọn, màn hình sẽ nhấp nháy và sẽ hiển thị một trong các giá trị sau dựa trên phạm vi đã chọn,

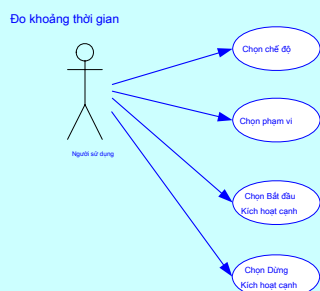
- 2,0000 mili giây
- 20.000 mili giây
- 2.000 giây

Nếu khoảng thời gian của tín hiệu đầu vào thấp hơn giá trị cho phép tối thiểu trên phạm vi đã chọn, màn hình sẽ hiển thị số đọc bằng không.

Nếu tín hiệu đầu vào trở về một giá trị trong giới hạn của phạm vi, giá trị của khoảng thời gian sẽ được hiển thị.

Phạm vi có thể được thay đổi bất cứ lúc nào bằng cách nhấn vào *chọn phạm vi* nút bấm. Việc sử dụng có thể chọn để đo khoảng thời gian bắt đầu trên cạnh dương hoặc âm của tín hiệu bằng cách nhấn *bắt đầu kích hoạt cạnh* nút bấm.

**Đo khoảng thời gian** Bộ đếm sẽ liên tục đo và hiển thị thời lượng của phần đã chọn của tín hiệu đầu vào trên phạm vi hiện được chọn miễn là *Khoảng thời gian* chế độ được chọn. Các trường hợp sử dụng sau được xác định cho *đo khoảng thời gian* chế độ.



Nếu thời lượng của phần đã chọn của tín hiệu đầu vào vượt quá giá trị tối đa cho phép trên phạm vi đã chọn, màn hình sẽ nhấp nháy và sẽ hiển thị một trong các giá trị sau dựa trên phạm vi đã chọn,

- 2,0000 mili giây
- 20.000 mili giây
- 2.000 giây

Nếu thời lượng của phần đã chọn của tín hiệu đầu vào thấp hơn giá trị cho phép tối thiểu trên phạm vi đã chọn, màn hình sẽ hiển thị bằng không.

Nếu tín hiệu đầu vào trở về một giá trị trong giới hạn của phạm vi, giá trị của khoảng thời gian của phần đã chọn của tín hiệu đầu vào sẽ được hiển thị.

Phạm vi có thể được thay đổi bất cứ lúc nào bằng cách nhấn vào *chọn phạm vi* nút bấm.

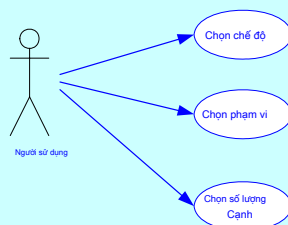
Người dùng có thể chọn bắt đầu đo khoảng thời gian trên cạnh dương hoặc âm của tín hiệu bằng cách nhấn *bắt đầu kích hoạt cạnh* nút bấm.

Người dùng có thể chọn kết thúc khoảng thời gian đo trên cạnh dương hoặc âm của tín hiệu bằng cách nhấn *dừng kích hoạt cạnh* nút bấm.

Lưu ý rằng khoảng thời gian của tín hiệu từ cạnh dương sang cạnh dương hoặc từ cạnh âm đến cạnh âm giống với chu kỳ của tín hiệu.

**Sự kiện** Bộ đếm sẽ liên tục đếm và hiển thị số lần xuất hiện của tín hiệu đầu vào trên phạm vi đang được chọn. Số lượng tích lũy sẽ được đặt lại về 0 khi kết thúc khoảng thời gian đếm đã chọn. Các trường hợp sử dụng sau được xác định cho *đếm sự kiện* chế độ.

Đếm sự kiện



Nếu số lượng tích lũy vượt quá giá trị cho phép tối đa trên phạm vi đã chọn, màn hình sẽ nhấp nháy và sẽ hiển thị một trong các giá trị sau dựa trên phạm vi đã chọn,

- 200 phút
- 2000 giờ

Phạm vi có thể được thay đổi bất cứ lúc nào bằng cách nhấn vào *chọn phạm vi* nút bấm.

Người dùng có thể chọn tăng số đếm trên cạnh dương hoặc âm của tín hiệu đầu vào bằng cách nhấn *bắt đầu kích hoạt cạnh* nút bấm.

#### Đặc điểm kỹ thuật chức năng hệ thống

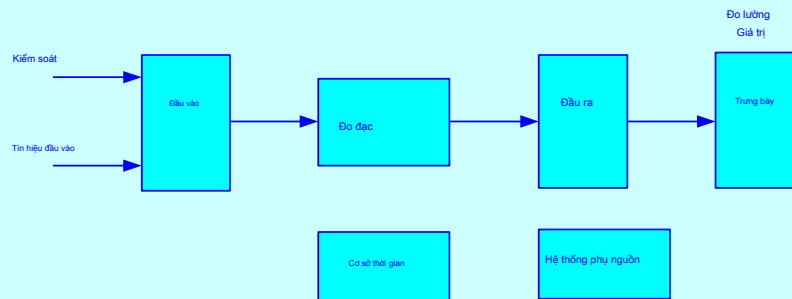
Hệ thống được thiết kế để thực hiện 4 loại phép đo kỹ thuật số khác nhau bao gồm tần số, khoảng thời gian, khoảng thời gian và các sự kiện.

Các phép đo thời gian và tần số sẽ được thực hiện để cung cấp ba phạm vi độ phân giải có thể lựa chọn của người dùng, dải tần số cao / tín hiệu thời lượng ngắn hơn, một giây cho tín hiệu tần số trung bình / thời lượng tầm trung và một phần ba cho tín hiệu tần số thấp / thời lượng dài hơn. Khả năng đo lường sự kiện sẽ hỗ trợ hai khoảng thời gian đếm có thể lựa chọn, ngắn hơn và dài hơn.

Đối với các phép đo tần suất, chu kỳ và sự kiện, người dùng sẽ có thể chọn bộ kích hoạt cạnh dương hoặc âm. Đối với các phép đo khoảng thời gian, người dùng sẽ có thể chọn cực tính của tín hiệu khởi động và dừng một cách độc lập.

Hệ thống sẽ được thiết kế để không loại trừ việc kết hợp tùy chọn truy cập từ xa trong tương lai.

Hệ thống bao gồm 6 khối chính như được cho trong sơ đồ khối sau.



**Hệ thống con đầu vào** - hệ thống con đầu vào phải cung cấp khả năng cho người dùng lựa chọn bất kỳ chức năng đo, phạm vi và cực tính kích hoạt nào. Hệ thống con cũng chọn và định tuyến tín hiệu đầu vào đến phần thích hợp của hệ thống con đo lường.

**Hệ thống con đầu ra** - hệ thống con đầu ra thực hiện phạm vi, lựa chọn cạnh, thông tin điều khiển và định dạng dữ liệu để trình bày thích hợp trên màn hình bảng điều khiển phía trước.

**Cơ sở thời gian** - hệ thống con cơ sở thời gian là một chuỗi phân chia và vòng lặp bị khóa pha được điều khiển từ bộ dao động tinh thể 100 MHz. Hệ thống con này sẽ cung cấp hai giai đoạn đồng hồ để điều khiển kiểm soát nội bộ và logic quyết định. Mỗi giai đoạn sẽ là  $200.0000 \pm 0,0001$  MHz.

Cơ sở thời gian cũng sẽ cung cấp các tần số sau được sử dụng để xác định các cửa sổ đo lường cho các sự kiện và phép đo tần số và cung cấp các tần số đếm cho các phép đo khoảng thời gian và khoảng thời gian.

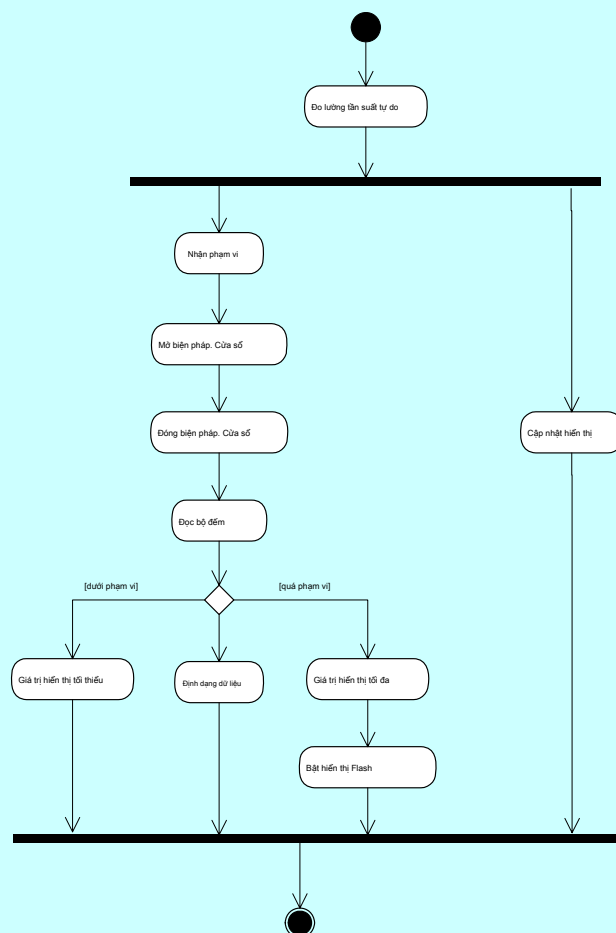
- Tần số -  $200.0000 \pm 0.0001$  MHz Giai đoạn -
- $100.0000 \pm 0.0001$  MHz
- Khoảng thời gian -  $100,00 \pm 0,0001$  MHz Sự kiện -
- $10,00 \pm 0,01$  Hz

**Hệ thống con đo lường** - hệ thống con đo lường cung cấp logic và điều khiển để thực hiện các phép đo thời gian và tần số.

- Phép đo tần số sẽ được thực hiện bằng cách mở một cửa sổ cho  $1,00 \pm 0,01$  giây. Trong thời gian cửa sổ mở, hệ thống con đo lường sẽ chuyển tần số đầu vào chưa biết vào bộ đếm BCD 7 giai đoạn. Khi cửa sổ đóng lại, bộ đếm sẽ chứa giá trị của tần số chưa biết.



Các hoạt động cần thiết để thực hiện phép đo tần số như được cho trong sơ đồ sau.



- Các phép đo khoảng thời gian và khoảng thời gian sẽ được thực hiện bằng cách mở một cửa sổ trên cạnh tín hiệu được chỉ định. Trong khi cửa sổ đang mở, tần số  $100.0000 \pm 0.0001$  MHz sẽ được đưa vào bộ đếm BCD 7 tầng. Khi cửa sổ đóng lại, bộ đếm sẽ chứa các giá trị của khoảng thời gian không xác định.
- Bộ đếm sẽ chứa số lượng sự kiện đã xảy ra trong khoảng thời gian đo.

Phép đo sự kiện sẽ được thực hiện bằng cách mở cửa sổ trong  $1,00 \pm 0,01$  giây đối với chế độ nhanh và  $3600,0 \pm 0,1$  giây đối với chế độ chậm. Trong thời gian cửa sổ mở, hệ thống con đo lường sẽ chuyển đầu vào chưa xác định vào bộ đếm BCD 4 giai đoạn. Khi cửa sổ đóng lại, bộ đếm sẽ chứa thước đo số lượng sự kiện đã xảy ra trong khoảng thời gian.

**Hệ thống cung cấp điện** - hệ thống con cung cấp điện sẽ cung cấp các điện áp sau ở các mức dòng điện được chỉ định cho logic bên trong.

+ 5,0 ± 0,01 VDC @ 10 A

+ 15,0 ± 0,01 VDC @ 500 mA

- 15,0 ± 0,01 VDC @ 500 mA

Khi bật nguồn, sẽ có một tín hiệu đặt lại âm tính. Tín hiệu đó phải duy trì ở trạng thái thấp trong thời gian tối thiểu là 10 ms và phải có khả năng chìm đến 1A.

**Trung bày** - màn hình thiết bị phải hiển thị kết quả của phép đo đã chọn trên màn hình LED màu đỏ 6 chữ số, 7 đoạn. Bố cục của các tính năng và chức năng chính được đưa ra trong sơ đồ trước đó.

#### Thông số kỹ thuật hoạt động

Hệ thống sẽ hoạt động trong môi trường thương mại / công nghiệp tiêu chuẩn

Phạm vi nhiệt độ 0-85C

Độ ẩm lên đến 90% RH Nguồn điện không ngưng tụ Lựa

chọn điện áp đường dây tự động

- 100 - 120 VAC ± 10% 50, 60, 400 Hz ± 10%
- 220 - 240 VAC ± 10% 50, 60 Hz ± 10%

Hệ thống sẽ hoạt động trong tối thiểu 8 giờ khi được sạc đầy pin

Trọng lượng tịnh / kích thước 2,75 kg, H: 90 mm x W: 200mm x D: 300 mm Cơ sở thời gian

của hệ thống phải đáp ứng các thông số kỹ thuật sau

Nhiệt độ ổn định 0-50 C

<6 x 10<sup>-6</sup>

Tỷ lệ lão hóa

90 ngày	<3 x 10 <sup>-8</sup>
6 tháng	<6 x 10 <sup>-7</sup>
1 năm	<25 x 10 <sup>-6</sup>

#### Đặc điểm kỹ thuật về độ tin cậy và an toàn

Bộ đếm phải tuân theo các tiêu chuẩn thích hợp

An toàn: UL-3111-1, IEC-1010, CSA 1010.1 EMC:

CISPR-11, IEC 801-2, -3, -4, EN50082-1 MTBF: Tối thiểu

10.000 giờ

### 12.8 Thông số kỹ thuật hệ thống so với Yêu cầu hệ thống

Kiểm tra các bước khác nhau đã được vạch ra cho đến thời điểm này, dường như có rất nhiều sự trùng lặp. Có vẻ như *Đặc điểm thiết kế hệ thống* và *Đặc điểm kỹ thuật yêu cầu hệ thống* chỉ là các tên khác nhau cho cùng một thứ. Họ không; các yêu cầu và thông số kỹ thuật về cơ bản là các loại mô tả khác nhau.

***Yêu cầu*** - Đưa ra một mô tả về một cái gì đó muốn hoặc cần thiết. Chúng là một tập hợp các thuộc tính cần thiết.

Nói chung các yêu cầu đến từ bộ phận tiếp thị hoặc bán hàng và chúng đại diện cho nhu cầu của khách hàng. Định nghĩa và đặc tả yêu cầu không liên quan đến tổ chức bên trong của hệ thống. Nó nhằm mô tả *what* một hệ thống phải làm và *how* nó phải làm điều đó, không phải *how* nó thực hiện nó. Các *Đặc điểm thiết kế hệ thống* được tạo ra bởi kỹ thuật như một câu trả lời và mô tả cách thực hiện các yêu cầu. Sau đó, hai nhóm thương lượng và lặp lại cho đến khi các yêu cầu và thông số kỹ thuật nhất quán.

***Sự chi rõ*** - Là một mô tả của một số thực thể có hoặc thực hiện các thuộc tính đó.

Đặc tả hệ thống là một phương tiện để chuyển mô tả nhu cầu sang một cấu trúc và mô hình chính thức hơn.

Có vẻ như mọi phần của thiết kế đều cần một thông số kỹ thuật khác. Chắc chắn, điều này là đúng. Các thông số kỹ thuật có thể tồn tại ở nhiều cấp độ khác nhau khi thiết kế được tinh chỉnh và trau chuốt. Những thứ khác nhau phải được định lượng và ở các mức độ chi tiết khác nhau trong các giai đoạn phát triển sản phẩm khác nhau. Các *Đặc điểm thiết kế hệ thống*

có thể yêu cầu kênh truyền thông liên hệ thống truyền dữ liệu với tốc độ 10.000 byte / giây với tốc độ lỗi bit cụ thể. Chi tiết *Phần cứng* và *Phần mềm*

*Thông số kỹ thuật* thiết lập các yêu cầu và ràng buộc đối với các thành phần tương ứng của chúng để có thể đáp ứng các thông số kỹ thuật đó.

Đặc tả là một mô tả chính xác về hệ thống đáp ứng các yêu cầu đã nêu. Lý tưởng nhất là một tài liệu đặc tả

Hoàn thành

Thích hợp

Dễ hiểu

Có thể theo dõi các yêu cầu Rõ

ràng

Có thể sửa đổi

Có thể viết

Thông số kỹ thuật phải được thể hiện bằng một ngôn ngữ hoặc ký hiệu trang trọng nhất có thể và có thể đọc được. Lý tưởng nhất là nó cũng phải chạy được. A *Đặc điểm kỹ thuật của hệ thống* nên tập trung chính xác vào chính hệ thống. Nó phải cung cấp một mô tả đầy đủ về các đặc điểm có thể nhìn thấy bên ngoài của nó, nghĩa là, giao diện công khai của nó. Khả năng hiển thị bên ngoài tách biệt rõ ràng những khía cạnh có thể nhìn thấy về mặt chức năng đối với môi trường mà hệ thống hoạt động với những khía cạnh của hệ thống phản ánh cấu trúc bên trong của nó.

## 12.9 Phân vùng và phân rã hệ thống

Tại thời điểm này trong chu trình thiết kế, tất cả các yêu cầu hệ thống đã được xác định, nắm bắt và chính thức hóa thành *Đặc tả thiết kế hệ thống*. Bước tiếp theo là di chuyển vào bên trong hệ thống và bắt đầu quá trình xác định và thiết kế chức năng làm phát sinh hành vi bên ngoài.

Trong tất cả các cuộc thảo luận trước đây, tính mô-đun và tính đóng gói đã được nhấn mạnh nhiều lần. Trước tiên, chúng ta sẽ xem xét lý do tại sao cách tiếp cận như vậy được khuyến nghị và sau đó là những gì nên được coi là quá trình phân rã và cuối cùng là phân vùng hệ thống thành các mô-đun phần cứng và phần mềm được tiến hành.

### 12.9.1 Suy nghĩ ban đầu

Vì vậy, cho câu hỏi đầu tiên, 'Tại sao chúng tôi làm điều này?' Tái sử dụng là một trong những lý do quan trọng. Với mỗi thiết kế mới, người ta nên luôn nhìn vào dự án trước đó cũng như dự án tiếp theo. Gì

có thể được sử dụng từ dự án cuối cùng để đẩy nhanh sự phát triển của dự án này không? Làm cách nào để thiết kế hiện tại có thể được triển khai để hỗ trợ một tính năng trong tương lai? Có những phần nào của thiết kế này có thể được sử dụng trong các dự án trong tương lai?

Thứ hai, nhiều trình biên dịch tạo mã đối tượng trong các phân đoạn, một mã cho mỗi mô-đun. Các hành động như vậy có thể đặt ra các hạn chế về kích thước đối với các mô-đun riêng lẻ. Việc xây dựng mô-đun kém có thể ảnh hưởng đáng kể đến việc truy cập bộ nhớ, tăng số lần bỏ lỡ bộ nhớ cache, thúc đẩy sự cố và làm giảm đáng kể hiệu suất.

Thứ ba, thông thường, việc phân công công việc được thực hiện trên cơ sở từng mô-đun. Các ranh giới mô-đun phải được xác định để giảm thiểu giao diện giữa các phần khác nhau của hệ thống. Thực tiễn như vậy cũng đơn giản hóa quá trình ký hợp đồng phụ một số công việc. Các vấn đề bảo mật cũng đóng một vai trò khi xem xét hợp đồng phụ. Cho dù làm việc cho một công ty đồ chơi hay trong một dự án nhạy cảm của chính phủ, người ta cần phải xem xét những thông tin nào để cung cấp cho các nhà cung cấp bên ngoài. Bằng cách phân tách hệ thống đúng cách, các phần có thể được thuê ngoài và các phần mà quyền kiểm soát sẽ được giữ lại có thể được xác định dễ dàng hơn.

Thứ tư, các mô-đun nên được đóng gói với mục tiêu ổn định các giao diện mô-đun trong thời gian đầu của thiết kế.

Thứ năm, phân chia hệ thống thành các mô-đun được kết hợp lỏng lẻo được xác định rõ ràng giúp đảm bảo một thiết kế an toàn và mạnh mẽ. Cách tiếp cận như vậy giúp ngăn chặn sự cố trong một bộ phận của hệ thống lây lan sang và ảnh hưởng đến bộ phận khác.

Cần thấy rõ tầm quan trọng của việc phân vùng một thiết kế mới; bước tiếp theo là kiểm tra quy trình để làm như vậy. Quá trình bắt đầu với hệ thống cấp cao nhất sau đó *cải tiến dần dần* mô hình đó thành các phần nhỏ hơn và dễ quản lý hơn để có thể dễ dàng thiết kế và xây dựng hơn.

Ban đầu, trọng tâm là chế độ xem chức năng của hệ thống hơn là các phần cứng và phần mềm cụ thể. Điều quan trọng là phải hiểu và nắm bắt hành vi ở cấp độ cao trước tiên. Bước tiếp theo sẽ là ánh xạ các chức năng đó, chức năng đó, vào phần cứng và phần mềm khi cần thiết để thỏa mãn các ràng buộc được xác định trong giai đoạn đầu của thiết kế. Phân vùng là quan trọng trong giai đoạn đầu của quá trình phát triển

của hệ thống để hỗ trợ tấn công sự phức tạp của một hệ thống lớn và sau đó là hướng dẫn để đạt được một kiến trúc vật lý tốt.

Khi chúng ta bắt đầu suy nghĩ về việc tổ chức hệ thống thành tập hợp các phần sẽ thực hiện các yêu cầu của khách hàng, chúng ta nên nhìn vấn đề từ cả góc độ cấp cao và từ một cái nhìn chi tiết hơn. Điều quan trọng cần nhớ là phát triển một phân vùng không phải là một quá trình một lần; nó không cần thiết phải hoàn hảo trong lần đầu tiên. Quá trình phân vùng có thể sẽ cần được thực hiện nhiều lần trước khi đạt được sự phân rã thỏa đáng và khả thi.

Trước khi bắt đầu phân vùng hệ thống, có một số suy nghĩ chung.

1. Hãy nhớ rằng với mọi quy tắc hoặc hướng dẫn, luôn phải có chỗ cho những trường hợp ngoại lệ.

2. Mỗi mô-đun nên giải quyết một phần được xác định rõ của vấn đề.

Việc kết hợp chức năng giữa các mô-đun làm cho tất cả các khía cạnh của quá trình phát triển và hỗ trợ trở nên khó khăn hơn nhiều. Bằng cách đó, người ta có thể dễ dàng tạo ra phần cứng và mã spaghetti của mì. Những thay đổi trong tương lai đối với các mô-đun như vậy sẽ rất khó thực hiện và dễ dẫn đến các tác dụng phụ không mong muốn và các phần không liên quan của hệ thống đột nhiên không hoạt động.

Mặc dù mong muốn có các mô-đun được xác định rõ ràng, với các giao diện đơn giản, giải quyết các phần của vấn đề được gói gọn một cách độc đáo, nhưng đôi khi trong các ứng dụng nhúng, người ta không thể làm như vậy vì hạn chế về hiệu suất hoặc kinh tế.

3. Hệ thống nên được phân vùng để chức năng dự kiến của mỗi mô-đun dễ hiểu.

Nếu các bên khác có thể hiểu được thiết kế, thì họ sẽ có thể duy trì và gia hạn nó khi cần thiết trong suốt thời gian sử dụng của sản phẩm. Hãy nhớ rằng, hơn một nửa số kỹ sư tham gia vào thiết kế hệ thống nhúng không thực hiện các thiết kế mới; họ duy trì và cải tiến các thiết kế hiện có.

Trong quá trình phát triển, các thiết kế dễ hiểu sẽ dẫn đến ít bất ngờ hơn khi thiết kế sắp hoàn thành. Tất cả các bên quan tâm sẽ có thể theo dõi

thiết kế và nhận xét khi quá trình được giải quyết. Một thiết kế quá phức tạp sẽ nhanh chóng làm nản lòng những lời chỉ trích sớm. Mọi người sẽ không dành thời gian để tìm hiểu hệ thống phải làm gì. Thật không may, sự chấp nhận sớm như vậy thường bị thay thế bởi sự từ chối sau đó và những nỗ lực thiết kế lại tiềm năng lớn. Mặc dù điều quan trọng là phải tự hào về công việc của mình, nhưng một người nên luôn tìm kiếm những ý kiến mang tính xây dựng của người khác.

4. Phân vùng nên được thực hiện để các kết nối giữa các mô-đun chỉ được giới thiệu vì các kết nối giữa các phần của vấn đề.

Đừng đặt một phần chức năng vào một mô-đun chỉ vì không có nơi nào khác cho nó.

5. Việc phân vùng phải đảm bảo rằng các kết nối giữa các mô-đun càng độc lập càng tốt.
6. Một lần nữa, hãy giữ những thứ như nhau với nhau. Thực hành như vậy giúp giảm thiểu sai sót. Việc phân vùng cũng được thực hiện để giúp đáp ứng các mục tiêu kinh tế của thiết kế.

Khi nào hình thành các phân vùng, quá trình này phải được xem xét từ một số quan điểm.

Chỉ xem xét một quan điểm duy nhất hoặc bỏ qua bất kỳ quan điểm nào có thể có những ảnh hưởng lâu dài đáng kể. Vào cuối ngày, hệ thống có thể không đáp ứng được mong đợi của khách hàng cũng như các thông số kỹ thuật về hiệu suất.

Khi quá trình phân hủy diễn ra, thiết kế trước tiên nên được xem xét từ một *chức năng* quan điểm. Kết quả từ các bước phân rã là một mô hình chức năng và có thể được sử dụng để xác định kiến trúc hệ thống. Trong số nhiều điều cần được xem xét, hai điều nên xuất hiện sớm trong quá trình này là *khớp nối* và *sự gắn kết* của các mô-đun mà hệ thống đang được phân rã. Mục tiêu là phát triển *các mô-đun liên kết lỏng lẻo, có tính gắn kết cao*. Hãy xem những điều này có nghĩa là gì.

### 12.9.2 Khớp nối

*Khớp nối* là một heuristic cung cấp một ước tính về mức độ phụ thuộc lẫn nhau của các mô-đun. Các mô-đun được kết hợp chặt chẽ thường sẽ sử dụng dữ liệu được chia sẻ hoặc trao đổi thông tin kiểm soát. Khi sự phụ thuộc lẫn nhau của mô-đun tăng lên, thì sự phức tạp của việc quản lý các mô-đun đó cũng như khó khăn hơn,

- Gỡ lỗi thiết kế trong quá trình phát triển
- Khắc phục sự cố hệ thống trong trường hợp có lỗi tại hiện trường, Bảo trì
- các mô-đun và hệ thống, và
- Sửa đổi thiết kế để thêm các tính năng hoặc khả năng.

Mục tiêu chính là làm cho các mô-đun của hệ thống càng độc lập càng tốt - mục tiêu là giảm thiểu hoặc giảm thiểu sự ghép nối.

#### **Thiết kế Heuristic**

Khớp nối càng thấp, công việc đã được thực hiện tốt hơn trong quá trình phân vùng.

Dưới đây là một số điều cần suy nghĩ trong giai đoạn đầu của thiết kế để giúp giảm bớt sự ghép nối:

#### 1. Loại bỏ tất cả các tương tác không cần thiết giữa các mô-đun.

Nếu một phần cụ thể của chức năng hoặc thông số được chia sẻ không phải là một phần mà nhiệm vụ dự kiến của hai mô-đun, thì hãy loại bỏ nó.

#### 2. Giảm thiểu lượng tương tác thiết yếu giữa các mô-đun.

Trong khi điều này nghe giống như điểm trước đó, nó không phải. Nếu một phân tích ban đầu cho thấy rằng cần phải có một số tương tác với mô-đun khác, thì cần phải nỗ lực để giảm bớt sự phức tạp của giao diện bắt buộc đó. Mục đích là giữ cho mọi thứ đơn giản.

Một số cách để giúp giảm bớt sự phức tạp bao gồm:

##### a. Giảm số lượng kết nối giữa các mô-đun và do đó giảm số lượng phần dữ liệu phải lưu chuyển giữa các mô-đun.

##### b. Cố gắng đi theo con đường trực tiếp nhất đến một tín hiệu hoặc một phần dữ liệu nếu thích hợp.

Có một số trường hợp mà cách triển khai tốt nhất là sử dụng proxy làm giao diện cho một tín hiệu hoặc một phần dữ liệu. Tuy nhiên, nói chung, tốt nhất là giảm số lượng mô-đun liên quan.

##### c. Nói chung, tránh sử dụng các biến toàn cục được chia sẻ. Một phương pháp tốt hơn là truyền dữ liệu vào một mô-đun thông qua danh sách tham số hoặc giao diện gọi của nó.

Tuy nhiên, với các ứng dụng nhúng, có những lúc việc chia sẻ như vậy rất quan trọng đối với các hạn chế về thời gian đáp ứng.



- d. Tránh kết nối phức tạp giữa hoặc giữa các mô-đun. Nguyên tắc cơ bản của mọi thiết kế là giữ cho mọi thứ đơn giản.
- e. Không đặt các giá trị mã cứng vào danh sách tham số của mô-đun hoặc giao diện gọi trừ khi thực sự cần thiết. Chúng tôi phải làm như vậy trong trường hợp một mô-đun giao diện hoặc cổng phải ở một vị trí địa chỉ cụ thể; đừng biến điều này thành một thông lệ chung.

### 3. Nói lòng sự tương tác cần thiết giữa các mô-đun, nếu có thể.

Trừ khi môi trường yêu cầu mức độ phối hợp cao giữa một số mô-đun để hoàn thành nhiệm vụ hoặc để đảm bảo giao tiếp không có lỗi, chỉ cần chuyển cho mô-đun thông tin cần thiết để hoàn thành công việc. Sau đó, hãy đợi một dấu hiệu cho thấy nhiệm vụ đã hoàn thành. Thực hiện một số phần khác của nhiệm vụ.

#### 12.9.3 Sự gắn kết

Một ý tưởng liên quan đến *khớp nối* Là *sự gắn kết*. Khái niệm khớp nối giải quyết sự phân vùng của một hệ thống; địa chỉ liên kết mang các mảnh lại với nhau. Độ liên kết là thước đo sức mạnh của mối liên quan chức năng của các phần tử trong một mô-đun. Mục tiêu là tạo ra các mô-đun mạnh mẽ, có tính gắn kết cao mà các phần tử của nó có liên quan thực sự và chặt chẽ với nhau. Ngược lại, các phần tử không được liên quan chặt chẽ đến các phần tử trong mô-đun khác. Chúng tôi muốn *tối đa hóa* sự gắn kết và *giảm thiểu* khớp nối.

Việc sử dụng sự gắn kết như một thước đo độ tin cậy và chất lượng đã có từ giữa những năm 1960. Một số năm sàng lọc và tích hợp ý tưởng của nhiều người nghiên cứu các thiết kế và cách tiếp cận thiết kế khác nhau đã khiến Constantine và Yordon (Y và C 1979 Page-Jones) hình thành thang đo liên kết dựa trên số liệu để bảo trì.

Hãy xem xét một số loại liên kết khác nhau.

**Sự gắn kết chức năng** - Mô-đun thực hiện một tác vụ duy nhất và tất cả các phần tử bao gồm đóng góp vào việc thực hiện một tác vụ đó.

**Liên kết tuần tự** - Mô-đun thực hiện một nhiệm vụ như một tập hợp các thủ tục tuần tự. Dữ liệu đầu ra của mỗi quy trình trở thành dữ liệu đầu vào cho quy trình tiếp theo. Tất cả các phần tử bao gồm liên quan đến một trong các quy trình đó.

**Sự gắn kết giao tiếp** - Mô-đun thực hiện một tác vụ có một số thủ tục hoạt động trên cùng một tập dữ liệu đầu vào, chẳng hạn như tác vụ xử lý ảnh.

**Sự gắn kết về thủ tục** - Mô-đun thực hiện một số thủ tục có thể có hoặc không liên quan đến một hoạt động chung. Kiểm soát, thay vì dữ liệu, chuyển từ quy trình này sang quy trình tiếp theo.

**Sự gắn kết thời gian** - Mô-đun thực hiện một số thủ tục hoặc hoạt động không liên quan được sắp xếp theo thứ tự trong thời gian.

**Liên kết logic** - Mô-đun thực hiện một số thủ tục là những phương pháp có thể thay thế để hoàn thành một nhiệm vụ. Một tập hợp con trong số các lựa chọn thay thế đó được người dùng bên ngoài lựa chọn để thực thi nhiệm vụ.

**Sự liên kết ngẫu nhiên** - Mô-đun tổng hợp một số thủ tục không liên quan. Sự gắn kết như vậy - hoặc thiếu chúng không nên được sử dụng.

Chúng tôi so sánh các loại liên kết và khớp nối khác nhau từ một số quan điểm khác nhau trong phần sau. Xếp hạng là Xuất sắc / Dễ = 5... Kém / Khó = 1

<i>Sự gắn kết</i>	<i>Khớp nối</i>	<i>Dễ dàng Sửa đổi</i>	<i>Dễ dàng Hiểu biết</i>	<i>Dễ dàng Bảo trì</i>
Chức năng	5	5	5	5
Tuần tự	4	4	4	3-4
Giao tiếp	3	3	3	3
Thủ tục	2-3	2-3	2-3	2
Thời gian	1	3	3	2
Hợp lý	1	1	2	1
Tình cờ	1	1	1	1

Bản

Liên kết và Khớp nối trong Thiết kế Hệ thống

Các phân tích liên kết và khớp nối cung cấp một bộ số liệu tốt để bắt đầu đánh giá các khía cạnh kiến trúc cấp cao của một thiết kế. Tuy nhiên, hãy nhớ rằng cả hai đều là hướng dẫn. Công việc để đảm bảo rằng thiết kế chắc chắn và nó đã được kiểm tra kỹ lưỡng vẫn cần được thực hiện.

Có rất nhiều thiết kế tốt yêu cầu các mô-đun được kết hợp chặt chẽ. Điện thoại di động CDMA là một ví dụ điển hình cho điều này. Người ta có thể có các thiết kế đa xử lý được kết hợp chặt chẽ cũng như các thiết kế dựa trên thông điệp được truyền đi, ngụ ý không phải là một thiết kế là đúng hay sai, hay tốt hơn thiết kế khác, mà chỉ là cách nó được thực hiện để đáp ứng các yêu cầu.

#### 12.9.4 Cân nhắc thêm

Với các hệ thống ngày nay, *không gian* quan điểm thường là chủ yếu. Đây là một cái nhìn bên ngoài của hệ thống và nó tạo ra một kiến trúc chức năng phân tán. Với quan điểm như vậy, hiệu suất và chi phí truyền thông được tính đến. Kết hợp chặt chẽ với quan điểm không gian là quan điểm của *nguồn* sự phân bố; một lần nữa, một cái nhìn bên ngoài. Những nỗ lực như vậy dẫn đến một "kiến trúc tài nguyên". Một lần nữa, hiệu suất, chi phí và độ tin cậy là những yếu tố phải được xem xét.

Cuối cùng, người ta phải xem xét *phần cứng* và *phần mềm*. Phân rã trở thành một quá trình thiết kế dẫn đến một kiến trúc phần cứng như đã được thảo luận trước đó. Bây giờ hiệu suất phải được xem xét. Với tư cách là các nhà phát triển nhúng, chúng tôi đang đóng một vai trò trực tiếp trong việc thiết kế và lựa chọn nền tảng phần cứng cũng như môi trường phần mềm. Việc cân bằng một cách thông minh trong hai lĩnh vực này có thể giúp chúng ta đi một chặng đường dài hướng tới việc phát triển một hệ thống an toàn, mạnh mẽ và chất lượng cao / hiệu suất cao.

#### 12.10 Thiết kế chức năng

Mục đích ở giai đoạn này của thiết kế là tìm ra một thiết bị bên trong thích hợp *chức năng* kiến trúc cho hệ thống. Chúng tôi đang bắt đầu xây dựng *làm sao* các yêu cầu đã được xác định có thể được thực hiện. Trọng tâm hiện tại là phân tích vấn đề. Thông qua phân tích như vậy, sự hiểu biết hơi lỏng lẻo về thiết kế có thể được chuyển thành một mô tả chính xác. Kết quả của một quá trình như vậy là một văn bản hoặc hình ảnh chi tiết

mô tả của hệ thống. Kết quả cuối cùng là một định nghĩa chức năng nhất quán hoàn chỉnh về các nhiệm vụ được yêu cầu.

Để thiết lập sự đánh giá cao về mô hình chức năng của một hệ thống, hãy xem xét một chiếc máy bay. Nếu một máy bay là hệ thống được thiết kế, thì mô hình chức năng cấp cao nhất có thể không được bao gồm nhiều hơn 3 chức năng chính: *cất cánh*, *bay*, và *đất đai*. Với quan điểm như vậy, chúng tôi không đưa ra tuyên bố nào về các vấn đề như cấu trúc hỗ trợ cho máy bay (bánh xe, bầu trời, cầu phao), hệ thống đẩy (máy bay phản lực, tên lửa, cánh quạt) hoặc phương pháp nâng (cánh - máy bay thông thường hoặc lượn - máy bay trực thăng). Đầu tiên, những điều này không quan trọng. Những quyết định như vậy có thể được hoãn lại cho đến sau này. Ưu điểm của cách tiếp cận như vậy là tính linh hoạt sớm - thời gian để khám phá trước khi bắt đầu hạn chế hệ thống. Một mô tả chức năng chỉ đơn giản là chính thức hóa hành vi dự kiến của thiết kế.

Mô tả chức năng phải được viết để những người am hiểu về lĩnh vực ứng dụng và những người sẽ phát triển phần cứng và phần mềm hiểu được. Đặc điểm kỹ thuật cũng phải sao cho nó có thể được nhiều bên quan tâm và đa dạng xem xét và kiểm tra so với thực tế. Nếu nó quá phức tạp để đọc và hiểu, sẽ không ai đọc nó. Khi dự án hoàn thành được giao, đã quá muộn để phát hiện ra rằng quan điểm của khách hàng và quan điểm của nhà phát triển về thực tế là hoàn toàn khác nhau.

Việc phân rã chức năng đầu tiên được thực hiện dựa trên việc tìm kiếm các biến và sự kiện bên trong thiết yếu trong hệ thống. Sau đó, quá trình thiết kế bao gồm các cải tiến hoặc phân tách liên tiếp cho từng chức năng (sử dụng chính xác cùng một quy trình) cho đến khi thu được các chức năng cơ bản hoặc chức năng cơ bản. Sự phân hủy như vậy tạo thành một *mô hình chức năng*

của hệ thống. Mô hình được thể hiện, bằng tập hợp các chức năng như vậy, phải đủ để xác minh chất lượng thiết kế và để đánh giá hành vi và hiệu suất của hệ thống.

Trong quá trình mô hình hóa và xác minh, các hoạt động của hệ thống và các yêu cầu hiệu suất liên quan có thể được phân bổ cho các chức năng bên trong và xác định mối quan hệ giữa các chức năng đó. Quá trình như vậy cũng cho phép người ta ước tính hiệu suất dự kiến của hệ thống.

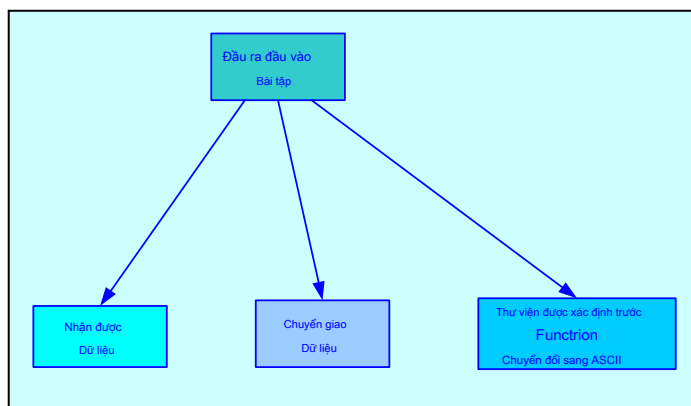
Đối với đặc tả yêu cầu, lý tưởng nhất, mô hình chức năng phải có thể thực thi được để cho phép xác minh đối với đặc điểm kỹ thuật. Có các công cụ

ngày nay điều đó sẽ cho phép chúng tôi làm điều này. Một trong những công cụ như vậy là mô hình Verilog hành vi. UML cũng đang bắt đầu biến các mô hình thực thi trở thành hiện thực.

Mô hình chức năng khác với đặc điểm kỹ thuật và cũng với kiến trúc vật lý sẽ được phát triển tiếp theo. Đặc điểm kỹ thuật mô tả *bên ngoài*

hành vi của hệ thống; mô hình chức năng nhằm mục tiêu *nội bộ* hành vi đó sẽ dẫn đến ngoại cảnh. Mô hình kiến trúc giải quyết các thành phần phần cứng và phần mềm vật lý mà các chức năng được ánh xạ trên đó.

Trong hình sau đây, là sự phân rã mức đầu tiên của một tác vụ đầu vào / đầu ra đơn giản. Hệ thống phải nhận dữ liệu từ và truyền dữ liệu ra thế giới bên ngoài. Liên kết với nhiệm vụ, là một chuyển đổi mã sang ASCII.



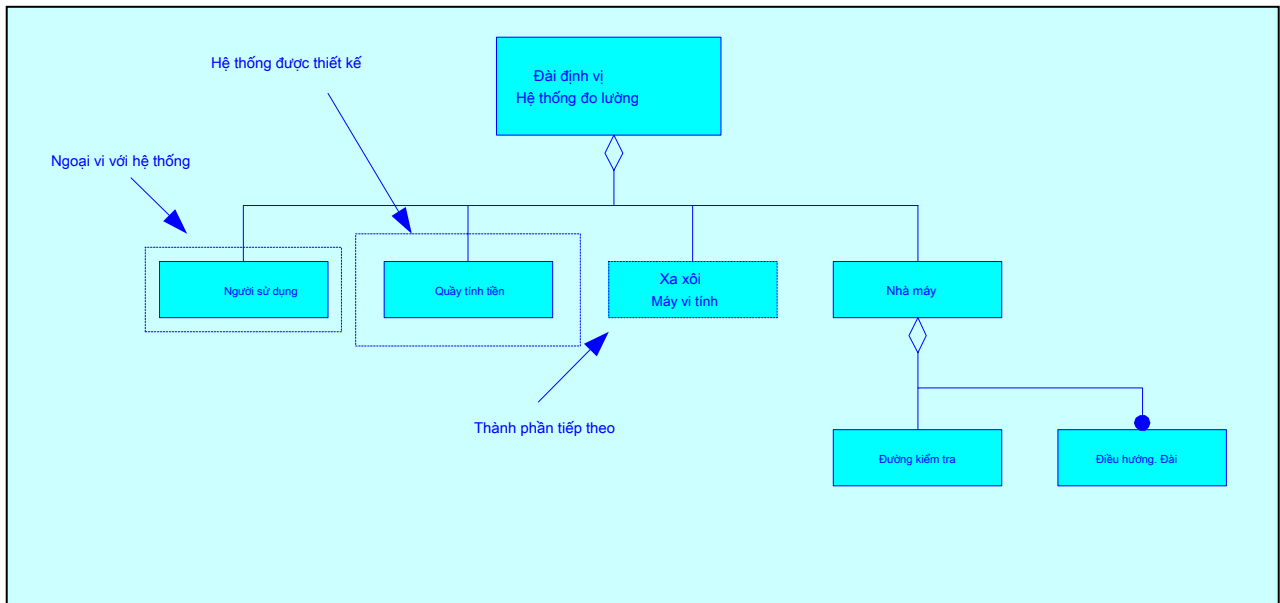
Mỗi chức năng này có thể được phân rã thêm khi cần thiết. Nếu được yêu cầu, các chức năng cấp hai cũng có thể được tinh chỉnh liên tiếp để cung cấp mức độ chi tiết cần thiết để hiểu và thực hiện thiết kế.

Bước tiếp theo trong phân tích là xác định các thông điệp truyền giữa người dùng hoặc các đối tượng bên ngoài đang hoạt động khác và hệ thống cũng như các tín hiệu nội bộ truyền giữa các khối chức năng chính. Chúng tôi xác định cách người dùng sẽ tương tác với hệ thống để khiến hệ thống thực hiện những gì dự định.

Bây giờ chúng ta hãy áp dụng hiểu biết của chúng tôi về phân vùng vào thiết kế chức năng của hệ thống quầy. Đầu tiên và quan trọng nhất... chúng ta phải tiếp tục trì hoãn ý tưởng làm việc với các cấu trúc dữ liệu cụ thể, bit, byte, bộ vi xử lý hoặc logic mảng trong một thời gian nữa. Mặc dù quan trọng sau này trong quá trình, nhưng hiện tại, chúng hạn chế việc khám phá và có thể làm sai lệch sự phân rã chức năng của hệ thống

*Ví dụ tiếp tục...*

Sơ đồ đầu tiên trình bày tổng hợp các đối tượng trong hệ thống. Sự tổng hợp đó bao gồm cả môi trường và bộ đếm đang được thiết kế.



Mô hình của hệ thống đo lường được thể hiện dưới dạng tập hợp của

- Người dùng,
- Nhà máy,
- Máy tính từ xa trong tương lai, Bộ
- đếm

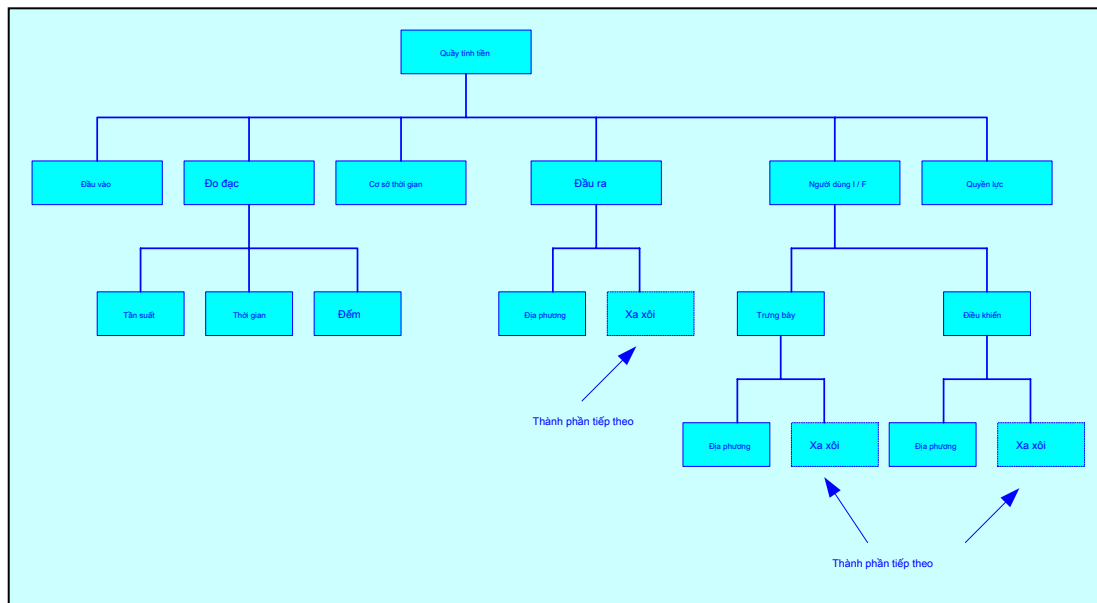
Nhà máy là tập hợp các dây chuyền thử nghiệm và số lượng bộ đàm dẫn đường phải được thử nghiệm. Lưu ý rằng chúng tôi đang sử dụng thuật ngữ lỏng lẻo hơn *tập hợp* hơn là *thành phần* đây.

Đặc điểm kỹ thuật thiết kế cung cấp một sơ đồ khối cấp cao của hệ thống. Đối với vấn đề này, một sơ đồ như vậy cung cấp một điểm khởi đầu tốt cho quá trình phân rã thứ bậc ban đầu của hệ thống. Sơ đồ sau đây trình bày chi tiết về thành phần bộ đếm và đưa ra một phân tích có thể có cho hệ thống đó.

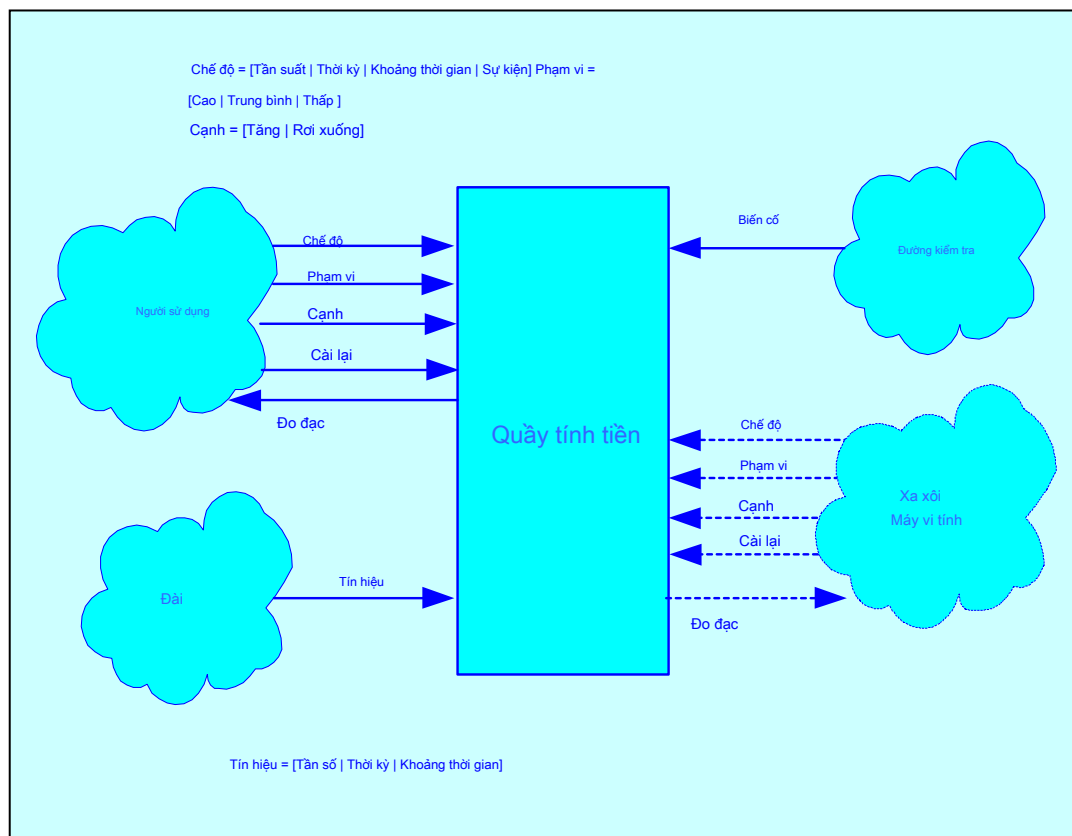
Giao diện với thế giới bên ngoài được tách biệt thành hai khối chức năng. Đầu tiên được liên kết với *trình bày thông tin* cho người dùng. Thứ hai được tính với *đưa thông tin vào* từ người dùng và các tác vụ khác cần thiết để hỗ trợ

đo đạc. Cả hai khối chức năng được phân rã thêm thành các hoạt động cục bộ so với các hoạt động từ xa.

Lựa chọn như vậy được thực hiện trong trường hợp đầu tiên bởi vì màn hình được coi là một chức năng đầu ra và điều khiển là một chức năng đầu vào. Trong trường hợp thứ hai, hai bộ chức năng khác nhau và ngữ pháp khác nhau để diễn đạt các lệnh của người dùng được dự đoán. Các hoạt động của bảng điều khiển phía trước có xu hướng khá thẳng về phía trước; các hoạt động từ xa có thể được tham gia nhiều hơn một chút. Chắc chắn, đây không phải là những lựa chọn duy nhất.

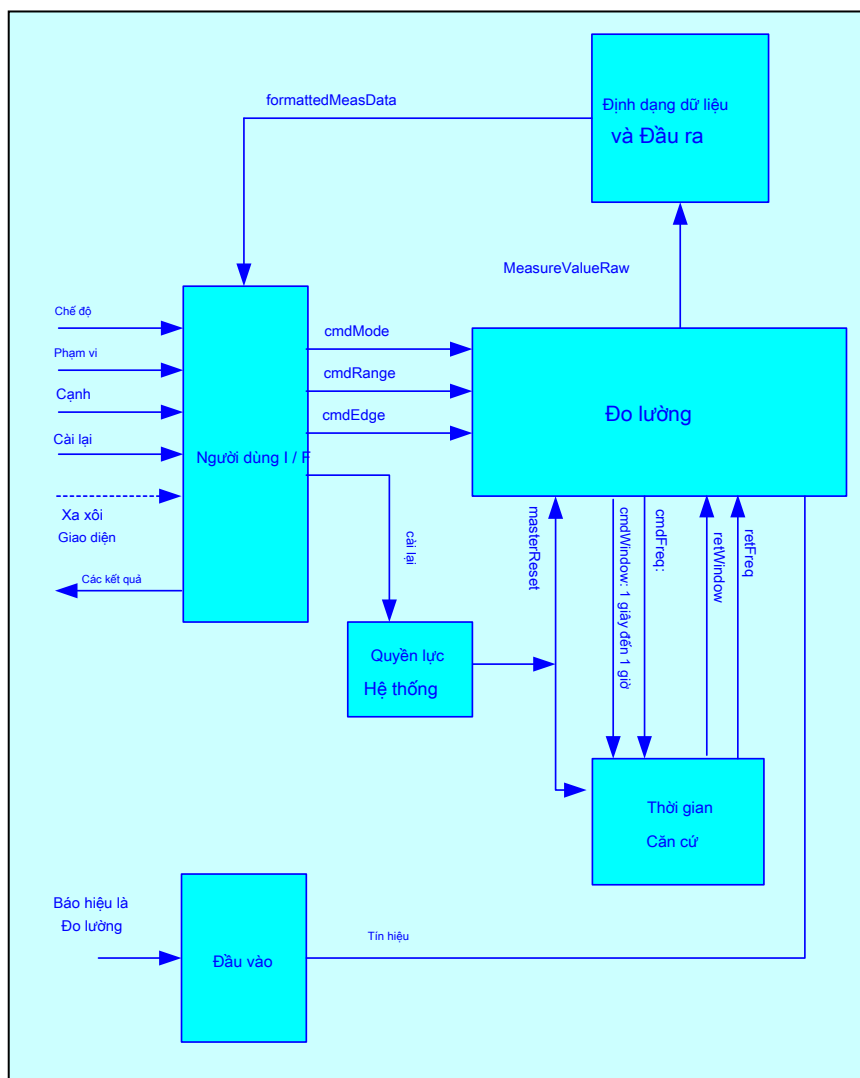


Hình vẽ tiếp theo chụp giao diện giữa quầy và môi trường xung quanh.





Hình vẽ tiếp theo thể hiện sự phân vùng chức năng và luồng tín hiệu giữa các khối chức năng chính.



Tiếp theo, kiến trúc hệ thống được xây dựng và các chức năng sau đó được ánh xạ vào các khối phần cứng và phần mềm bao gồm hệ thống đó.

### 12.11 Thiết kế kiến trúc

Khi thực hiện một thiết kế kiến trúc, mục tiêu là chọn giải pháp thích hợp nhất cho vấn đề ban đầu dựa trên việc khám phá nhiều kiến trúc khác nhau và lựa chọn phân vùng và phân bổ chức năng phần cứng / phần mềm phù hợp nhất.

Chế độ xem của một phân vùng hiện thay đổi để phản ánh sự hiểu biết chi tiết hơn về hệ thống và liên quan đến *lập bản đồ* hoặc phân bổ từng mô-đun chức năng vào (các) khối phần cứng hoặc phần mềm vật lý thích hợp. Một ánh xạ như vậy hoàn toàn mô tả việc triển khai phần cứng của hệ thống.

Như đã lưu ý trước đó, người ta phải luôn cố gắng mở rộng phạm vi của thiết kế kiến trúc để không loại trừ những cải tiến có thể có trong tương lai. Chắc chắn, điều này liên quan đến việc cân bằng giữa tính tổng quát và tính thực tiễn cũng như đồng thời đáp ứng các yêu cầu cụ thể khác. Tuy nhiên, kế hoạch nên dành cho một hệ thống phát triển trong suốt thời gian tồn tại của nó; nếu điều này được thực hiện tốt, các tiện ích bổ sung, thứ không thể tránh khỏi trong các hệ thống ngày nay, sẽ dễ dàng hơn nhiều.

Mục tiêu chính của hoạt động thiết kế kiến trúc là phân bổ hoặc ánh xạ các phần khác nhau của chức năng hệ thống tới các khối phần cứng và phần mềm thích hợp. Công việc dựa trên cấu trúc chức năng chi tiết. Các yêu cầu về hiệu suất được phân tích và cuối cùng là các hạn chế được áp đặt bởi các công nghệ có sẵn cũng như những hạn chế phát sinh từ các thông số kỹ thuật phần cứng và phần mềm được xem xét.

Các ràng buộc quan trọng phải được xem xét bao gồm các mục như

Sự phân bổ địa lý,

Giao diện vật lý và người dùng,

Thông số kỹ thuật hiệu suất hệ thống,

Ràng buộc về thời gian và yêu cầu về độ tin cậy, Công suất tiêu thụ,

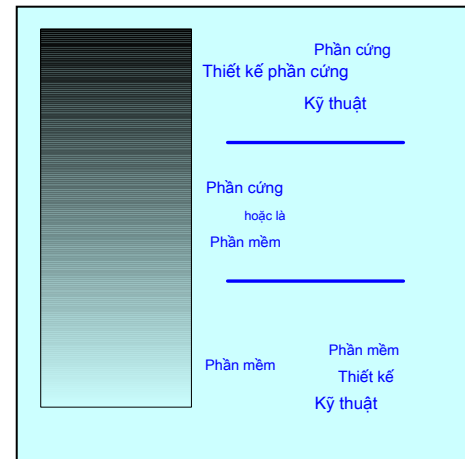
Các thành phần và chi phí kế thừa.

Những ràng buộc như vậy là yếu tố mạnh mẽ để quyết định phần nào của hệ thống nên được thực hiện trong phần mềm và phần nào nên được thực hiện trong phần cứng.

Việc phân bổ hợp lý các phần chức năng nói chung là hiển nhiên đối với một phần quan trọng của hệ thống. Đối với những người đó, có thể nói dễ dàng, 'phần này phải là phần cứng hoặc phần này phải là phần mềm.' Nguồn điện, màn hình, thông tin liên lạc

cổng và gói chứa hệ thống nhất thiết phải là phần cứng. Hệ điều hành và các trình điều khiển liên quan, nó thường được đồng ý, nhất thiết phải là phần mềm.

Tình huống được thể hiện bằng đồ thị như trong sơ đồ đi kèm. Có một vùng màu xám giữa phần cứng và phần mềm mà phương pháp triển khai không được xác định chính xác. Trong việc lựa chọn các thành phần tạo nên lĩnh vực này, người ta sẽ đưa ra các quyết định về kỹ thuật hoặc



đánh đổi tốc độ, chi phí, kích thước, trọng lượng, cũng như nhiều yếu tố khác.

Các *lập bản đồ* trên một kiến trúc như vậy hoàn toàn xác định việc triển khai phần cứng của hệ thống. Phần cứng của hệ thống được chỉ định bởi một kiến trúc vật lý có thể bao gồm một hoặc nhiều bộ vi xử lý, thiết bị logic phức tạp hoặc logic mảng hoặc / và các mạch tích hợp tùy chỉnh. Điều quan trọng cần lưu ý là với các hệ thống ngày nay, các bộ vi xử lý và vi điều khiển này có thể đảm nhận nhiều tính cách khác nhau: (CISC) tập lệnh phức tạp, tập lệnh giảm (RISC) hoặc lõi xử lý tín hiệu kỹ thuật số (DSP).

Đối với hầu hết các ứng dụng, một phần đáng kể của phần mềm có thể dễ dàng tách rời khỏi phần cứng và do đó cho phép nó phát triển đồng thời. Phần còn lại, nằm trong ranh giới, khó phân vùng hơn và nằm dưới cái được gọi là *đồng thiết kế*.

### 12.11.1 Đặc điểm kỹ thuật và thiết kế phần cứng và phần mềm

Đặc tả hệ thống cung cấp một lượng hóa chi tiết các đầu vào, đầu ra và hành vi chức năng của hệ thống dựa trên các yêu cầu ban đầu của chúng tôi. Việc phân rã chức năng tương tự như các bước được thực hiện trong việc xác định các yêu cầu. Khi kiến trúc của thiết kế bắt đầu hình thành, mục tiêu là xác định càng đầy đủ càng tốt các thông số kỹ thuật cho từng thành phần vật lý trong hệ thống và các giao diện giữa chúng. Đặc điểm kỹ thuật của phần cứng cho toàn bộ hệ thống là

quyết định bằng cách xác định kiến trúc phần cứng và tất cả các thuộc tính của nó. Đặc điểm kỹ thuật của phần mềm có được bằng cách xác định việc triển khai phần mềm hoặc sơ đồ khối (sử dụng bất kỳ phương pháp nào trong số nhiều phương pháp) cho mỗi thành phần phần mềm của kiến trúc.

Mỗi tập con chức năng được triển khai trong phần mềm được mô tả bằng một đặc tả phần mềm chi tiết thể hiện mức độ ưu tiên của từng tác vụ và mối quan hệ phụ thuộc không gian (khớp nối dữ liệu) và thời gian giữa các tác vụ. Biểu đồ UML bao gồm biểu đồ trạng thái chi tiết, biểu đồ thời gian, biểu đồ tuần tự, biểu đồ hoạt động và biểu đồ cộng tác đều có thể rất hữu ích ở giai đoạn này trong thiết kế.

Việc triển khai phần mềm có thể sử dụng hoặc không sử dụng nhân thời gian thực. Với nhân thời gian thực không có giá trị, thời gian phát triển được giảm xuống, nhưng không giảm chi phí xuất xưởng hoặc thông số kỹ thuật hiệu suất dựa trên thời gian. Đối với các hệ thống không sử dụng nhân thời gian thực (chiếm 80% đến 90% các hệ thống vừa và nhỏ), người ta có thể đạt được sự tối ưu hóa thiết kế tốt hơn khi giải quyết các hạn chế thời gian thực cứng tốc độ cao. Trong những trường hợp như vậy, giải pháp đang được điều chỉnh thủ công cho vấn đề cụ thể hơn là điều chỉnh một giải pháp có mục đích chung cho một trường hợp cụ thể.

Đối với thiết kế phần mềm, những điều sau đây phải được phân tích và quyết định:

- Có sử dụng nhân thời gian thực hay không.
- Có thể kết hợp một số chức năng để giảm số lượng tác vụ phần mềm không? Nếu vậy, làm thế nào?
- Mức độ ưu tiên cho mỗi nhiệm vụ.
- Một kỹ thuật thực hiện cho mỗi mối quan hệ giữa các nhiệm vụ.

Khi thích hợp, một nhân thời gian thực hoặc các dịch vụ của hệ điều hành có thể được sử dụng. Nói chung, mục tiêu chính là giảm độ phức tạp của bộ phận tổ chức để giảm quy mô và độ phức tạp của phần mềm và kết quả là thời gian phát triển, kiểm tra và gỡ lỗi.

Trong những trường hợp như vậy, một lựa chọn thường xuyên là *Xếp hạng Lập kế hoạch Monotonic* chính sách (điều này sẽ được thảo luận ngay sau đây... các nhiệm vụ được thực hiện thường xuyên hơn được chỉ định mức độ ưu tiên cao hơn). Các chức năng vĩnh viễn (những chức năng chạy liên tục mà không có sự kiện kích hoạt)

và một số hàm tuần hoàn không có ràng buộc về thời gian thường được thực hiện trong một tác vụ nền.

Để thực hiện các mối quan hệ giữa các nhiệm vụ, nên sử dụng các lệnh gọi thủ tục càng nhiều càng tốt, do đó đơn giản hóa phần tổ chức và giảm chi phí giữa các nhiệm vụ. Việc triển khai như vậy chỉ có thể thực hiện được giữa các chức năng với mức độ ưu tiên tương đối ngày càng tăng. Các tác vụ được kích hoạt bởi các sự kiện phần cứng được gọi thông qua hệ thống thăm dò hoặc ngắt của bộ xử lý.

Đối với mỗi phần con cụ thể của hệ thống trong đó phân vùng không rõ ràng, một thông số kỹ thuật chi tiết được viết; phân vùng phần cứng / phần mềm cuối cùng được xác định thông qua một quá trình tinh chỉnh liên tiếp như đã được thực hiện trong quá trình phân rã trước đó. Mỗi phân vùng phần cứng / phần mềm cũng phải bao gồm các giao diện phần cứng và trình điều khiển phần mềm để hỗ trợ bất kỳ giao tiếp giữa các thành phần.

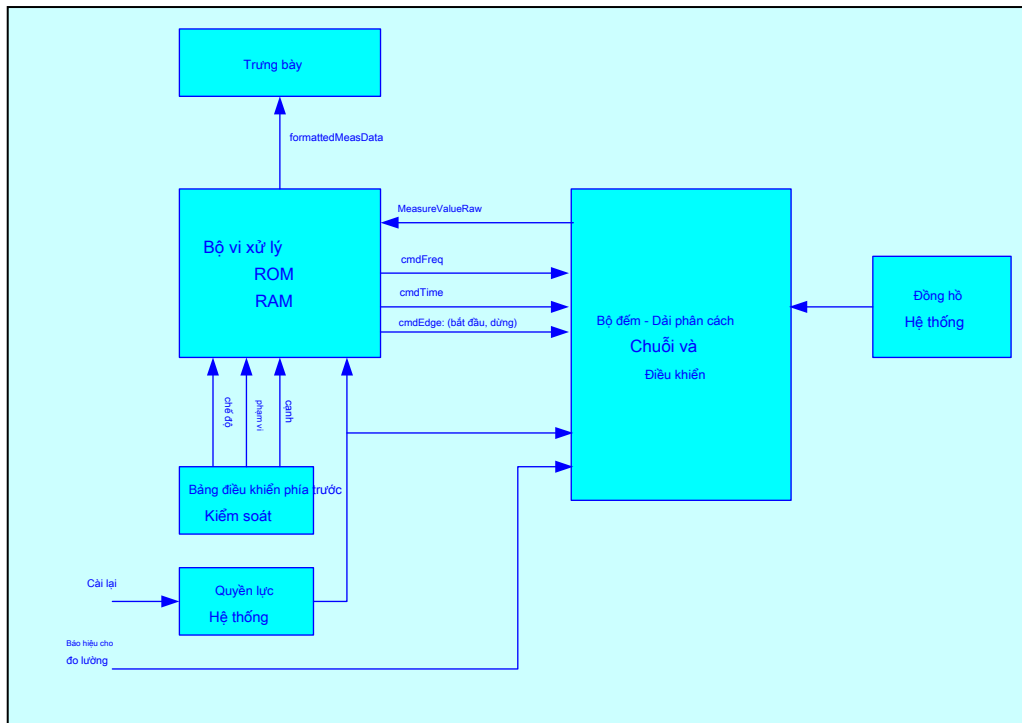
Kết quả là một ánh xạ hoàn chỉnh của tất cả các chức năng còn lại và các quan hệ chức năng vào kiến trúc phần cứng.

Trong số các tiêu chí quan trọng mà chúng tôi cố gắng tối ưu hóa là:

- Chi phí thực hiện (hoặc nhà máy), thời gian
- và chi phí phát triển,
- Các hạn chế về hiệu suất và độ tin cậy, Điện năng
- tiêu thụ,
- Kích thước

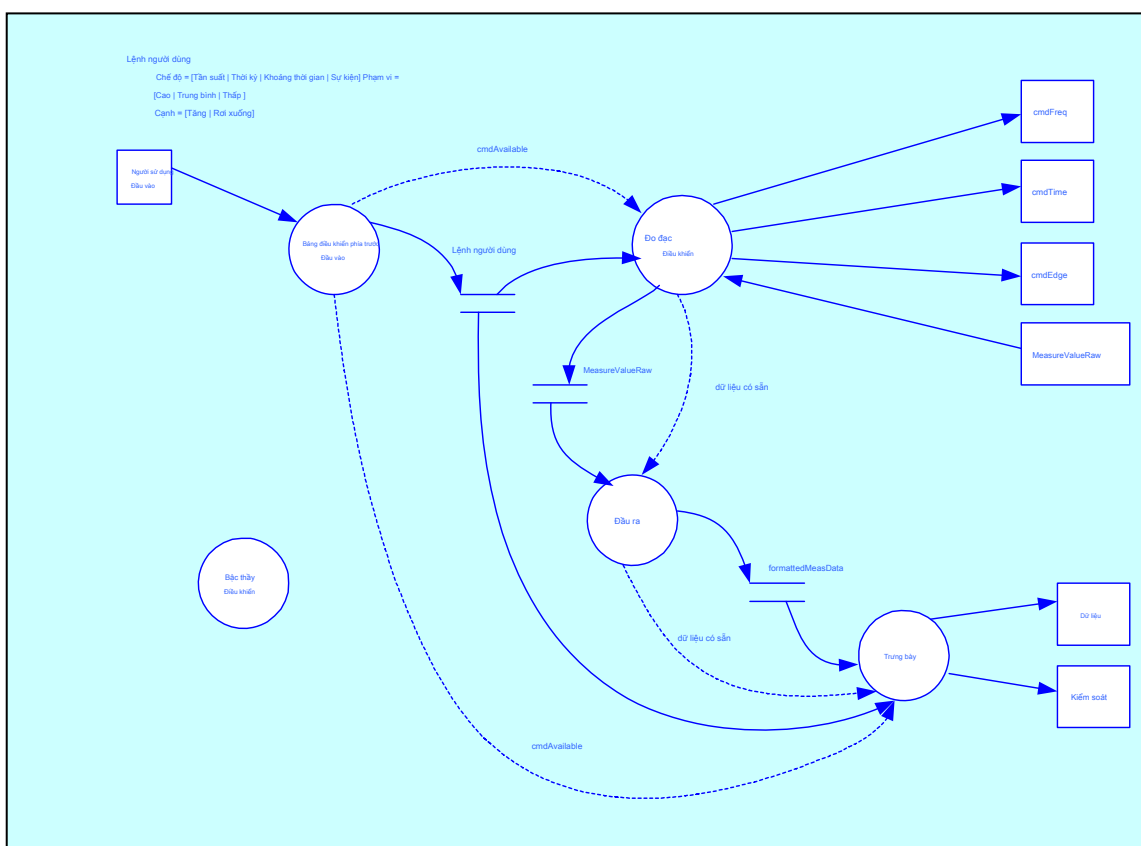
*Ví dụ tiếp tục...*

Bước tiếp theo trong việc phát triển bộ đếm bắt đầu bằng việc xây dựng kiến trúc phần cứng; kiến trúc phần mềm tuân theo. Sau đó, chúng tôi ánh xạ từng chức năng đã xác định trước đó vào kiến trúc. Sơ đồ sau đây trình bày các thành phần phần cứng.



Trong thiết kế, bộ vi xử lý, màn hình, bảng điều khiển phía trước và hệ thống điện rõ ràng là phần cứng. Về lý thuyết, hệ thống đồng hồ cũng như chuỗi phân chia bộ đếm và điều khiển liên quan có thể được triển khai trong phần mềm. Tuy nhiên, tần số mà bộ đếm dự định hoạt động (200 MHz) sẽ làm sai lệch quyết định đối với giải pháp phần cứng.

Sơ đồ sau xác định các tác vụ phần mềm chính, dữ liệu được chia sẻ và I / O trong sơ đồ luồng dữ liệu và điều khiển. Các *nhiệm vụ bảng điều khiển phía trước* liên tục kiểm tra (trực tiếp bằng cách thăm dò hoặc gián tiếp bằng cách ngắt) trạng thái của bảng điều khiển phía trước để người dùng nhập liệu. Một thay đổi trong đầu vào được ghi lại và chuyển đến *nhiệm vụ hiển thị* (sẽ cập nhật màn hình cho phù hợp) và *nhiệm vụ đo lường*. Các *nhiệm vụ đo lường* đưa ra các lệnh thích hợp cho bên ngoài *kiểm soát chuỗi bộ chia khối*. Vào cuối mỗi phép đo, dữ liệu thô được đọc từ bộ chia bộ đếm và được chuyển đến *nhiệm vụ đầu ra*.



Các *nhiệm vụ đầu ra* định dạng dữ liệu đúng cách và gửi dữ liệu đó đến *nhiệm vụ hiển thị* để hiển thị trên bảng điều khiển phía trước. Các *kiểm soát tổng thể nhiệm vụ* quản lý lịch trình của tất cả các nhiệm vụ và thực hiện mọi công việc dọn phòng cần thiết hoặc các nhiệm vụ khác khi cần thiết.

## 12.12 Mô hình chức năng so với mô hình kiến trúc

Một câu hỏi hay mà người ta có thể đặt ra ở giai đoạn này là 'Tại sao cần thiết kế mô hình chức năng và mô hình kiến trúc?' Chúng tôi bắt đầu bằng cách xem xét bất kỳ phần cứng, phần mềm hệ thống, sự kết hợp, điều đó không quan trọng. Rõ ràng là tổ chức bên trong của một hệ thống dựa trên một tập hợp các thành phần và sự kết nối giữa chúng với nhau. Một mô hình thích hợp phải bao gồm cả hai yếu tố *chức năng* cấp độ và ở *kiến trúc* trình độ để có thể đại diện và đánh giá hệ thống phần cứng / phần mềm.

### 12.12.1 Mô hình chức năng

Mô hình chức năng mô tả một hệ thống thông qua một *tập hợp các yếu tố chức năng tương tác*.

Thiết kế tiến hành ở mức độ cao mà không có sự thiên vị ban đầu đối với bất kỳ triển khai cụ thể nào. Chúng ta có quyền tự do khám phá và sáng tạo. Hành vi của một phần tử chức năng được mô tả tốt nhất bằng mô hình phân cấp và đồ họa. Các mô-đun chức năng sẽ tương tác bằng cách sử dụng một trong ba kiểu quan hệ sau:

- *Các biến chia sẻ* quan hệ - Định nghĩa trao đổi dữ liệu mà không có phụ thuộc thời gian.
- *Các đồng bộ hóa* quan hệ - Trong đó xác định phụ thuộc thời gian.
- *Các chuyển tin nhắn* bằng cổng - Hàm ý mối quan hệ giữa người sản xuất / người tiêu dùng.

Chúng ta sẽ thảo luận về từng điều này khi chúng ta nghiên cứu các quá trình và giao tiếp giữa các quá trình. Tất cả chúng đều rất quan trọng trong việc thiết kế và phát triển các hệ thống nhúng ngày nay.

### 12.12.2 Mô hình kiến trúc

Mô hình kiến trúc mô tả *vật lý* kiến trúc của hệ thống dựa trên các thành phần thực như bộ vi xử lý, logic phân mảng, bộ xử lý mục đích đặc biệt, các thành phần tương tự và kỹ thuật số, và nhiều kết nối giữa chúng.

### 12.12.3 Sự cần thiết của cả hai mô hình

Hai quan điểm này, khi được xem xét riêng biệt, không đủ để mô tả hoàn toàn thiết kế của các hệ thống đương đại. Nó là cần thiết để thêm *lập bản đồ* giữa

*chức năng* quan điểm và *kiến trúc* một. Một ánh xạ như vậy xác định một phân vùng (chức năng) và việc phân bổ các thành phần chức năng cho các phần tử phần cứng. Đây còn được gọi là *cấu hình kiến trúc*.

Các *mô hình chức năng*, nằm giữa mô hình đặc tả và mô hình kiến trúc, phù hợp để đại diện cho tổ chức bên trong của một hệ thống. Nó giải thích tất cả các chức năng cần thiết và các khớp nối giữa chúng - được thể hiện theo quan điểm của vấn đề ban đầu. Sử dụng một sơ đồ như vậy dẫn đến một giải pháp độc lập về công nghệ. Đặc biệt với loại mô hình này, tất cả hoặc một phần của mô tả có thể được thực hiện bằng phần mềm hoặc phần cứng.



Mô hình chức năng là cơ sở cho sự phân vùng thô của hệ thống. Việc phân vùng như vậy tự nhiên dẫn đến việc lựa chọn các chức năng để triển khai trong phần cứng hoặc phần mềm. Các *cấu trúc kiến trúc* được chi tiết hơn và thường tuân theo mô hình chức năng; kiến trúc cũng có thể được áp đặt trước.

### 12.13 Tạo mẫu

Giai đoạn nguyên mẫu dẫn đến một nguyên mẫu hệ thống hoạt động. Triển khai nguyên mẫu bao gồm:

- Thiết kế chi tiết
- Gỡ lỗi
- Thẩm định
- Thử nghiệm

Tạo mẫu tự nhiên là một quá trình từ dưới lên vì nó bao gồm việc lắp ráp các bộ phận riêng lẻ và bổ sung ngày càng nhiều các chức năng trừu tượng. Mỗi cấp độ của việc thực hiện phải được xác nhận. Nghĩa là, nó phải được kiểm tra sự phù hợp với các thông số kỹ thuật ở cấp độ tương ứng trong thiết kế từ trên xuống.

Việc triển khai phần cứng và phần mềm có thể được phát triển đồng thời và có sự tham gia của các chuyên gia trong cả hai lĩnh vực, hy vọng sẽ giảm tổng thời gian triển khai. Thường thì điều này không xảy ra trong thực tế. Thông thường, phần mềm dẫn đầu phần cứng. Không-không-it, một giải pháp hoàn chỉnh có thể được tạo ra và / hoặc tổng hợp cho cả phần cứng (ở dạng ASIC và lõi tiêu chuẩn, v.v.) và phần mềm (ở dạng giao diện phần cứng / phần mềm). Nguyên mẫu kết quả sau đó có thể được xác minh.

#### 12.13.1 Thực hiện

Các hoạt động trong bước này phụ thuộc nhiều vào công nghệ được sử dụng. Hãy nhớ rằng, nguyên mẫu là công cụ để hiểu và xác nhận thiết kế hệ thống. Đó là một bằng chứng về khái niệm. Một lời cảnh báo, đừng vội vàng phân tích hoặc thiết kế để đi đến nguyên mẫu. Ngoài ra, đừng ngại vứt bỏ nguyên mẫu. Đối với các dự án nhỏ, đôi khi cố gắng chuyển đổi nguyên mẫu thành sản phẩm cuối cùng có tác dụng. Đối với các dự án lớn, nó thường là một khái niệm bằng chứng mà hầu như không bao giờ có thể được di chuyển.

Hãy nhớ rằng, những người vội vàng thiết kế và viết mã vì có rất nhiều thử nghiệm phải làm sẽ dành nhiều đêm dài để mọi thứ hoạt động hiệu quả và thậm chí là những đêm dài hơn với những khách hàng không hài lòng. Vì một số lý do, khách hàng dường như không có khiếu hài hước khi sự thất bại của một sản phẩm mà họ đã mua chỉ khiến họ mất vài triệu đô la. Nếu bạn đang bán hàng cho một thị trường chung, *Công ty của bạn* vừa mất vài triệu chi phí R&D mà bạn vẫn chưa có sản phẩm để đưa ra thị trường. Vì vậy, bây giờ, nó thậm chí còn tồi tệ hơn, bởi vì bạn đã bỏ lỡ cơ hội bán hàng với một sản phẩm mà bạn không thể bán được vì nó được hình thành kém hoặc nó vẫn chưa sẵn sàng.

### 12.13.2 Phân tích thiết kế hệ thống

Chúng tôi đã nghiên cứu quy trình thiết kế hệ thống trong khi chuyển từ yêu cầu sang thiết kế. Bây giờ thiết kế cấp độ đầu tiên đã có, nó phải được phân tích một cách nghiêm túc. Bước này cung cấp một số kiểm tra quan trọng về thiết kế, trước hết, nó xác minh rằng giải pháp đáp ứng các yêu cầu và thông số kỹ thuật ban đầu. Ở giai đoạn này trong quy trình thiết kế, cũng có thể cần phải đánh đổi các khía cạnh kiến trúc và chức năng khác nhau của thiết kế. Sự đánh đổi đó phải được thực hiện theo các tiêu chí được xác định trong đặc điểm kỹ thuật ban đầu.

Bước đầu tiên đòi hỏi một phân tích tính của hệ thống. Ở giai đoạn này, kiến trúc của hệ thống được kiểm tra. Quan tâm ngay lập tức là *không phải* hệ thống sẽ hoạt động như thế nào tại thời điểm chạy. Các mục tiêu chính là có một hệ thống dễ hiểu, dễ xây dựng, kiểm tra và bảo trì. Tất cả thường xuyên từ các nhà thiết kế mới (và, thật không may, một số người nên biết rõ hơn), tôi đã nghe thấy cụm từ, "... nhưng nó hoạt động !!!" Đối với những hệ thống mà chúng tôi tự hào mang tên mình, việc đưa nó hoạt động chỉ là một phần rất nhỏ của công việc. Hơn nữa, thật dễ dàng để tạo một phiên bản duy nhất của bất kỳ hệ thống nào hoạt động. Tạo ra một hoặc mười triệu thiết kế giống nhau trong quá trình sản xuất mà cuối cùng sẽ hoạt động an toàn và đáng tin cậy là một thách thức lớn hơn nhiều.

Mục tiêu chính trong bất kỳ thiết kế nào là hoàn thành công việc. Chúng tôi muốn thiết kế phải thật đáng tin cậy và được ghi chép đầy đủ đến nỗi mọi sửa đổi và mở rộng trong tương lai đều không cần nỗ lực. Tất nhiên, điều cần lưu ý là người ta cũng phải biết độ tin cậy đủ là bao nhiêu và khi nào thì dừng việc ghi lại tài liệu. Được ghi chép đầy đủ có nghĩa là vừa đủ để mọi người có thể dễ dàng hiểu thiết kế, nhưng không quá nhiều để nó trở thành thiết kế có thể phân phối chính.

### 12.13.2.1 Phân tích tĩnh

Phân tích tĩnh cần xem xét 3 lĩnh vực:

#### 1. Khớp nối

Chúng tôi đã kiểm tra khía cạnh này của một thiết kế. Việc ghép nối có liên quan đến số lượng và độ phức tạp của các mối quan hệ tồn tại giữa các mô-đun hệ thống khác nhau. Nó cũng đưa ra một thước đo để đo lường các tác động của một sự thay đổi. Mục tiêu là khớp nối lỏng lẻo.

#### 2. Tính kết dính

Một vấn đề khác đáng được nhấn mạnh lại, tính gắn kết là thước đo tính đồng nhất về chức năng của các phần tử tạo nên các mô-đun. Điều này áp dụng cho cả các thành phần và các quan hệ. Người ta phải xem xét cả quan điểm bên ngoài và bên trong. Sự gắn kết bên ngoài bắt đầu với việc đặt tên thích hợp và ý nghĩa cho các phần tử. Bên trong, cấu trúc và mối quan hệ giữa các thành phần được phân tích. Ví dụ, kết hợp thông qua dữ liệu được chia sẻ có tính gắn kết hơn so với thông điệp. Thông điệp ngụ ý một sự phụ thuộc theo thời gian.

#### 3. Sự phức tạp

Hai loại phức tạp được xác định: *chức năng* và *hành vi*.

Độ phức tạp chức năng được đặc trưng bởi:

- Số lượng các chức năng bên trong và các thành phần quan hệ

Mục đích là để giữ cho chúng nhỏ. Nói chung, khi số lượng chức năng và quan hệ giảm đi thì độ phức tạp của thiết kế cũng giảm theo. Lưu ý, điều này không có nghĩa là hy sinh sự rõ ràng.

- Kết nối giữa các phần tử bao gồm mỗi mô-đun.

Các cuộc thảo luận trước đó về khớp nối cũng được áp dụng ở đây. Giữ mọi thứ đơn giản.

Sự phức tạp về hành vi được đặc trưng bởi

- Số lượng đầu vào và đầu ra  
Một lần nữa, mục tiêu là một con số nhỏ hơn.
- Độ dài và dễ đọc và hiểu mô tả của mô-đun.

Nếu yêu cầu một số đoạn văn hoặc một trang văn bản (bằng phông chữ dưới 6 điểm) để mô tả chức năng của một trong các mô-đun, thì mô-đun đó có thể quá phức tạp. Để đơn giản hóa các mô tả như vậy, hãy sử dụng bảng, phương trình logic hoặc mã giả.

- Điều khiển luồng thông qua mô-đun và số lượng và cấu trúc của các biến trạng thái.

Có một luồng điều khiển chính duy nhất thông qua mô-đun và giữ cho số lượng trạng thái nhỏ.

#### 12.13.2.2 Phân tích động

Mục tiêu khi thực hiện phân tích động trên hệ thống là xác định cách nó sẽ hoạt động trong bối cảnh gần đúng với môi trường làm việc cuối cùng. Phân tích động xem xét những điều sau:

- Xác minh hành vi

Mục tiêu là đảm bảo rằng hành vi của hệ thống, trong môi trường hoạt động của nó, đáp ứng *hoạt động* sự chỉ rõ. Đó là, nó có thực hiện các chức năng mà nó đã dự định thực hiện không? Việc xác minh này bao gồm hành vi ở ranh giới của các chức năng đó. Để có thể làm được như vậy, tất nhiên, ngay từ đầu chúng ta cần có một đặc điểm kỹ thuật tốt.

- Phân tích hiệu suất

Phân tích Hiệu suất đảm bảo rằng hệ thống, trong môi trường hoạt động của nó, đáp ứng *hiệu suất* sự chỉ rõ. Trọng tâm là các giá trị cụ thể cho đầu vào và đầu ra. Chúng ta sẽ nói về điều này trong chương sau.

- Phân tích đánh đổi

Phân tích đánh đổi là cần thiết để xác định giải pháp tối ưu cho các hạn chế và mục tiêu đã cho. Việc phân tích như vậy, chỉ dựa trên một số tiêu chí nhỏ về hiệu suất, có thể ảnh hưởng đến sự thành công hay thất bại cuối cùng của sản phẩm.

## 12.14 Các cân nhắc khác

Hai hoạt động bổ sung bổ sung và đồng thời cần được xem xét trong thế giới kinh doanh ngày nay là vốn hóa và tái sử dụng và các yêu cầu và quản lý truy xuất nguồn gốc. Hãy xem xét ngắn gọn về cách viết hoa. Tái sử dụng thiết kế là một trong những chủ đề trung tâm trong văn bản này.

### 12.14.1 Viết hoa và tái sử dụng

#### viết hoa

Viết hoa và tái sử dụng là những hoạt động cần thiết cho quá trình thiết kế đương đại. Khai thác hợp lý và hiệu quả tài sản trí tuệ (SHTT) là rất quan trọng hiện nay. Tài sản trí tuệ là các thiết kế, thường được cấp bằng sáng chế, có thể được bán cho một bên khác để phát triển và bán như (một phần) sản phẩm của họ. Ví dụ, công ty MIPS thiết kế kiến trúc máy tính. Họ không thực sự tự thực hiện bất kỳ công việc nào, thiết kế là sản phẩm của họ.

#### Tái sử dụng

Bất kỳ xem xét nào về việc tái sử dụng thành phần là một hoạt động được thực hiện trong giai đoạn thiết kế chức năng và kiến trúc của dự án. Lưu ý, đôi khi người ta cũng có thể xem xét những điều này trong quá trình tạo mẫu.

Một trong những mục đích chính là giúp các nhà thiết kế rút ngắn vòng đời phát triển. Việc sử dụng lại thành phần được tạo điều kiện thuận lợi theo hai cách: *hiện tại* và *Tương lai*. Hiện tại, việc sử dụng lại được hỗ trợ bằng cách xác định một tập hợp các thành phần chức năng hoặc kiến trúc bên ngoài (hiện có) có thể đáp ứng một số phần của chức năng mong muốn. Việc tái sử dụng trong tương lai được hỗ trợ bằng cách xác định các thành phần trong hệ thống theo thiết kế sẽ có thể tái sử dụng trong các dự án hoặc sản phẩm khác.

Để được sử dụng lại, một thành phần cần phải được

- Được xác định rõ ràng
- Được mô-đun hóa thích hợp
- Tuân theo một số tiêu chuẩn trao đổi

Một mô-đun được thiết kế tốt, được suy nghĩ kỹ lưỡng sẽ dễ dàng thích nghi với một tình huống mới hơn nhiều so với một mô-đun được ai đó ghép lại với nhau cho một số mục đích đặc biệt và hầu như không hoạt động. Điều này cũng đúng đối với một phần của hệ thống được mô-đun hóa tốt.

Nếu, trong giai đoạn thiết kế, một người đưa ra quyết định xem xét các mô-đun có thể được sử dụng lại, thì cơ hội sử dụng lại đó sẽ tăng lên rất nhiều. Cuối cùng, nếu mục tiêu là mô-đun có khả năng ứng dụng rộng rãi hơn so với địa điểm địa phương, thì các thiết kế phải phù hợp với các tiêu chuẩn quốc gia và quốc tế hiện có. Với thị trường quốc tế ngày nay đang phát triển hàng ngày, chúng tôi phải thiết kế theo các tiêu chuẩn như vậy.

Một lần nữa, sự xâm nhập và đánh đổi trong thế giới thực là một phần của quá trình. Trong khi thiết kế để có thể tái sử dụng hoặc phần đầu cho một thiết kế mô-đun, có những yếu tố khác cần được xem xét. Giả sử không có đủ dung lượng ROM cho mã nếu nó được thiết kế hoàn toàn theo mô-đun, thì mã có thể được sửa đổi để rất cụ thể cho ứng dụng. Mặc dù được biết rằng quyết định này có thể tạo ra vấn đề trong tương lai, nhưng chúng tôi sẽ kết thúc với một sản phẩm không cạnh tranh nếu ngân sách vượt quá.

### 12.14.2 Yêu cầu Truy xuất nguồn gốc và Quản lý

#### Quản lý yêu cầu

Truy xuất nguồn gốc yêu cầu đề cập đến khả năng tuân theo vòng đời của một yêu cầu (từ đặc điểm kỹ thuật ban đầu) theo cả chiều thuận và chiều ngược lại thông qua toàn bộ quá trình thiết kế và thiết kế. Cần phải rõ ràng rằng khả năng xác định nguồn gốc là mối quan hệ một-nhiều giữa một yêu cầu và các thành phần mà nó liên quan hoặc theo dõi (hoặc thực hiện nó). Một bản ghi chính xác và đầy đủ về khả năng xác định nguồn gốc giữa các yêu cầu và các thành phần hệ thống cung cấp một số thông tin quan trọng trong suốt vòng đời của sản phẩm. Trong số này bao gồm,

- Phương tiện để người quản lý dự án (có thể là) khách hàng giám sát tiến độ phát triển.
- Một đường dẫn có thể được sử dụng trong quá trình xác minh và xác nhận sản phẩm so với thông số kỹ thuật ban đầu. Biết được một yêu cầu cụ thể đã được thực hiện ở đâu và như thế nào tạo điều kiện xác nhận rằng yêu cầu đã được thực hiện một cách trung thực.
- Phương tiện xác định mô-đun phần cứng hoặc phần mềm nào bị ảnh hưởng nếu một yêu cầu thay đổi.

### Quản lý yêu cầu

#### Địa chỉ quản lý yêu cầu

Yêu cầu sửa đổi

Những thay đổi

Cải tiến

Đỉnh chính

Trong quá trình thiết kế, khó tránh khỏi những thay đổi như vậy vì nhiều lý do. Do đó, một quy trình rõ ràng tạo điều kiện thuận lợi cho một cách thức phù hợp với những sửa đổi đó phải được sử dụng trong toàn bộ quá trình thiết kế.

### 12.15 Lưu trữ dự án

Khi sản phẩm cuối cùng đã được đưa vào sản xuất, vẫn còn một số công việc phải hoàn thành. Trong quá trình phát triển, một lượng lớn thông tin thiết kế quan trọng đã được tạo ra. Hầu hết thông tin đó phải được giữ lại vì nhiều lý do. Nếu sản phẩm tuân theo vòng đời điển hình, các lỗi phải được sửa sẽ được phát hiện khi khách hàng sử dụng sản phẩm, sẽ có các bản sửa đổi trong tương lai, các tính năng mới sẽ được mong đợi và bổ sung, sản phẩm thế hệ tiếp theo sẽ xây dựng dựa trên hiện tại để đặt tên cho một vài. Câu hỏi hiển nhiên là cái gì phải được lưu lại?

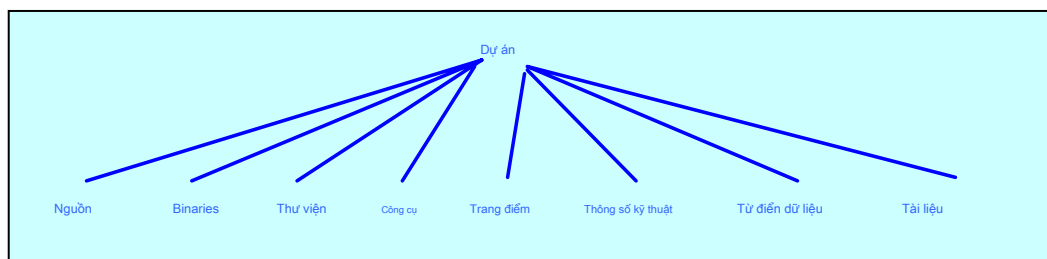
Vấn đề xử lý những gì cần lưu trữ không khác gì đối đầu với thiết kế ban đầu. Điều đó nói rằng, chúng tôi sử dụng cùng một cách tiếp cận và bắt đầu từ đầu. Dự án điển hình sẽ có nhiều người đóng góp. Một danh sách cơ bản có thể bao gồm,

- Kế hoạch sản phẩm
- Thiết kế và phát triển
- Kiểm tra
- Chế tạo
- Tiếp thị
- Bán hàng

Mỗi nhóm sẽ có thông tin, kiến thức, tài liệu, công cụ sẽ

quan trọng trong tương lai. Hãy tập trung vào tập hợp con kỹ thuật trong số này: thiết kế và phát triển, thử nghiệm và sản xuất.

Trong các nghiên cứu trước đây về thiết kế an toàn và mạnh mẽ, chúng tôi đã xác định cấu trúc thư mục dự án phần mềm điển hình. Sơ đồ đó được trình bày ở đây để tham khảo.



Mỗi nhóm tham gia vào quá trình phát triển nên có một thư mục giống nhau ghi lại phần của họ trong dự án. Các thư mục dự án và tất cả nội dung của chúng là một trong những mục chính phải được lưu trữ. Đây là những điều hiển nhiên.

Bây giờ, càng ít rõ ràng hơn. Ngày nay, phần mềm, chương trình cơ sở, và các công cụ phần mềm rất cần thiết cho việc thiết kế và phát triển bất kỳ hệ thống nhúng nào. Nếu mã nguồn bị mất hoặc khả năng xây dựng lại từ các nguồn bị mất, bất kỳ công việc nào trong tương lai của dự án sẽ bị suy giảm nghiêm trọng. Ngày nay, mã nguồn không còn chỉ có nghĩa là danh sách C, C++, Java, hoặc trình hợp dịch dưới dạng điện tử hoặc trên phương tiện từ tính.

Trong những năm trước, phần cứng thường được hỗ trợ bởi tài liệu bản cứng vẽ tay. Nếu một bản vẽ bị mất, nó có thể được các nhà thiết kế lành nghề phục hồi bằng cách thiết kế ngược phần hiện có. Ngày nay, một tập hợp phong phú gồm CAD (thiết kế có sự hỗ trợ của máy tính), CAM (sản xuất có sự hỗ trợ của máy tính) và các công cụ tổng hợp và mô hình hóa IC và FPGA đã thay thế các phương pháp cũ.



Tất cả những công cụ đó đều chạy trên máy tính. Tất cả các công cụ đó đều được nhà cung cấp của họ sửa đổi hoặc cập nhật thường xuyên. Tất cả những công cụ đó cũng có vòng đời sản phẩm và cuối cùng sẽ không còn được hỗ trợ nữa. Nếu các công cụ được lưu trữ sẽ không thực thi trên máy tính chạy hệ điều hành ngày nay, chúng sẽ ít được sử dụng. Ngày nay, ngoài việc lưu trữ sản phẩm cuối cùng, việc lưu trữ môi trường phát triển hoàn chỉnh - máy tính, ổ cứng, hệ điều hành, v.v. - cũng rất đáng được xem xét. Tài liệu cho các công cụ cũng nên được bao gồm trong danh sách đó.

Một bước thiết yếu khi kho lưu trữ được thu thập đã được thiết lập là tiến hành những gì được gọi là

*xây dựng trình nử*. Một bản dựng trình nử bắt đầu với một môi trường hoặc bối cảnh hoàn toàn mới. Tiếp theo, các công cụ lưu trữ được cài đặt và thiết lập. Các công cụ được chạy, nếu thích hợp, và được kiểm tra để xem liệu các thành phần được chỉ định của sản phẩm có thể được tạo lại hay không. Nếu quá trình không thành công, các thành phần bị thiếu phải được xác định, thêm vào kho lưu trữ và quá trình lặp lại.

Thông thường, trong suốt quá trình phát triển sản phẩm, chúng tôi xây dựng các công cụ đơn giản có mục đích đặc biệt để giúp quản lý các nhiệm vụ phát triển và xây dựng hệ thống, sau đó quên rằng chúng là một phần thiết yếu của quá trình xây dựng hoặc tổng hợp khi tạo kho lưu trữ. Việc xây dựng trình tiết nhanh chóng tiết lộ khi những công cụ đó bị thiếu.

Ngày nay, việc đầu tư tài chính vào tất cả các khía cạnh của sự phát triển dự án là rất quan trọng. Giữ lại và bảo vệ khoản đầu tư đó để sử dụng trong tương lai là một bước kết thúc quan trọng đối với chu kỳ.

## 12.16 Tóm tắt

Trong chương này, chúng tôi đã giới thiệu và nghiên cứu các giai đoạn chính của quá trình phát triển hệ thống nhúng. Các khía cạnh chi tiết hơn của quá trình đó được đề cập cùng với việc nghiên cứu thiết kế và kiểm tra các yếu tố phần cứng và phần mềm cụ thể của hệ thống.

Chúng tôi thấy rằng thiết kế là quá trình chuyển các yêu cầu của khách hàng thành hệ thống làm việc. Chúng tôi đã học được sự phức tạp của các hệ thống đương đại hiện nay đòi hỏi một cách tiếp cận chính thức hơn và các phương pháp chính thức hơn. Sau một đặc điểm kỹ thuật chính thức, chúng tôi xem xét các cách phân vùng hệ thống như một bước trong việc phát triển một thiết kế chức năng. Chúng tôi hình thành một mô hình chức năng sau đó phát triển và tinh chỉnh mô hình. Cuối cùng, chúng tôi lập bản đồ chức năng của chúng tôi

mô hình về cấu trúc kiến trúc. Trong khi đó, chúng tôi kết thúc với một nguyên mẫu đang hoạt động, phân tích thiết kế hệ thống cả trong và sau khi phát triển.

Chúng tôi đã xem xét một số cân nhắc quan trọng khác trong vòng đời thiết kế. Chúng bao gồm tài sản trí tuệ, tái sử dụng thành phần / mô-đun, quản lý yêu cầu và quy trình lưu trữ.