

سازمان سما

و اسسهه دانشگاه آزاد اسلامی
دانشگاه سما واحد حاجی آباد



مهندسی نرم افزار

منبع : مهندسی نرم افزار(پرسمن)

دکتر شیر افکن

حمیدرضا رضاپور

مهندسی نرم افزار

فصل ۷: مدل سازی خواسته ها :

جريان ، رفتار، الگوها و برنامه های تحت وب

مدل سازی خواسته ها چیست؟

چه کسی آن را انجام می دهد؟

چرا اهمیت دارد؟

مراحل کار کدام است؟

محصول کار چیست؟

چگونه مطمئن شوم که درست از عهده کار برآمده ام؟

انواع رویکرد برای مدل تحلیل

۱- تحلیل ساخت یافته:

داده ها و فرآیندهایی که این داده ها را تبدیل می کنند، موجودیت های مجزا در نظر گرفته شوند.

اشیاءی داده ای به شیوه ای مدل سازی می شوند که صفات و روابط میان آنها را تعریف کند.

فرآیندهای که اشیای داده ای را دستکاری می کنند، به شیوه ای مدل سازی می شوند که چگونگی تبدیل اشیاءی داده ای را به هنگام جریان یافتن آن ها در سیستم نشان دهند.

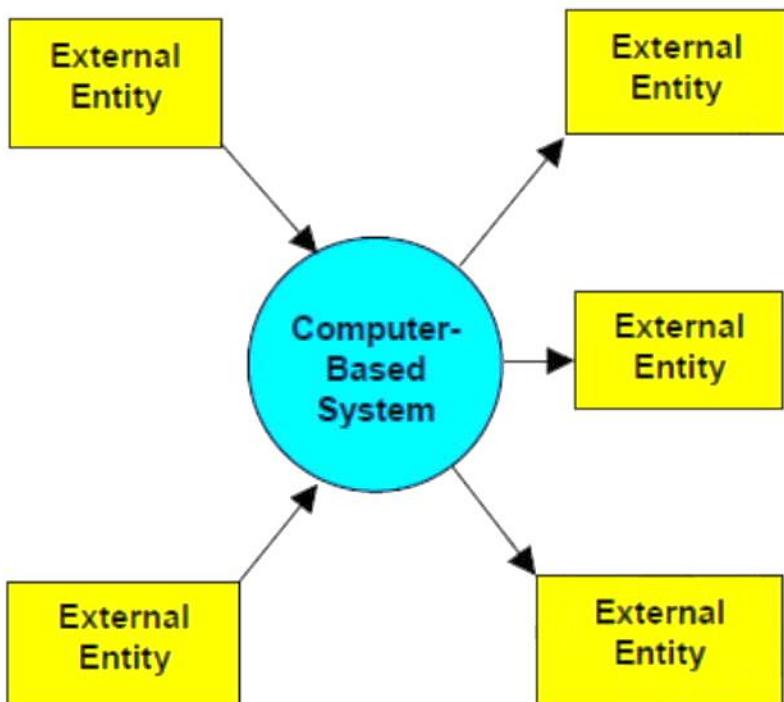
۲- تحلیل شیء گرا:

بر تعریف کلاس ها و شیوه های همکاری آنها با یکدیگر برای برآورده ساختن خواسته های مشتری تأکید دارد.

مدل سازی جریان گرا

data flow diagram

نمایی از سیستم بر اساس ورودی- فرآیند- خروجی به دست می دهد.



General Form of a Level 0 DFD.

یعنی اشیاءی داده ای به درون نرم افزار جریان پیدا می کنند،
توسط عناصر پردازشی تبدیل می شوند و
اشیاءی داده ای حاصل به بیرون نرم افزار جریان پیدا می کنند.
اشیاءی داده ای با پیکان های بر چسب دار نمایش داده می شوند.
تبدیلات با دایره یا حباب نمایش داده می شوند.

DFD به شیوه‌ی سلسله مراتبی نمایش داده می‌شود.
یعنی اولین مدل در جریان داده‌ها (DFD سطح صفر یا نمودار حیطه‌ای) کل سیستم را نمایش می‌دهد.
نمودارهای بعدی جریان داده‌ها، نمودار حیطه‌ای را پالایش می‌کنند و در هر سطح بعدی، جزئیات بیشتری
اضافه می‌شود.

ایجاد مدل جریان داده ها

با نمودار جریان داده ها می توان مدل هایی از دامنه ای اطلاعاتی و دامنه عملیاتی را توسعه داد.

چند دستورالعمل ساده می تواند در به دست آوردن نمودار جریان داده ها کمک کند

۱- نمودار جریان داده ها در سطح صفر باید نرم افزار / سیستم را به عنوان یک حباب منفرد تصویر کند.

۲- ورودی و خروجی اولیه باید به دقت ذکر شود.

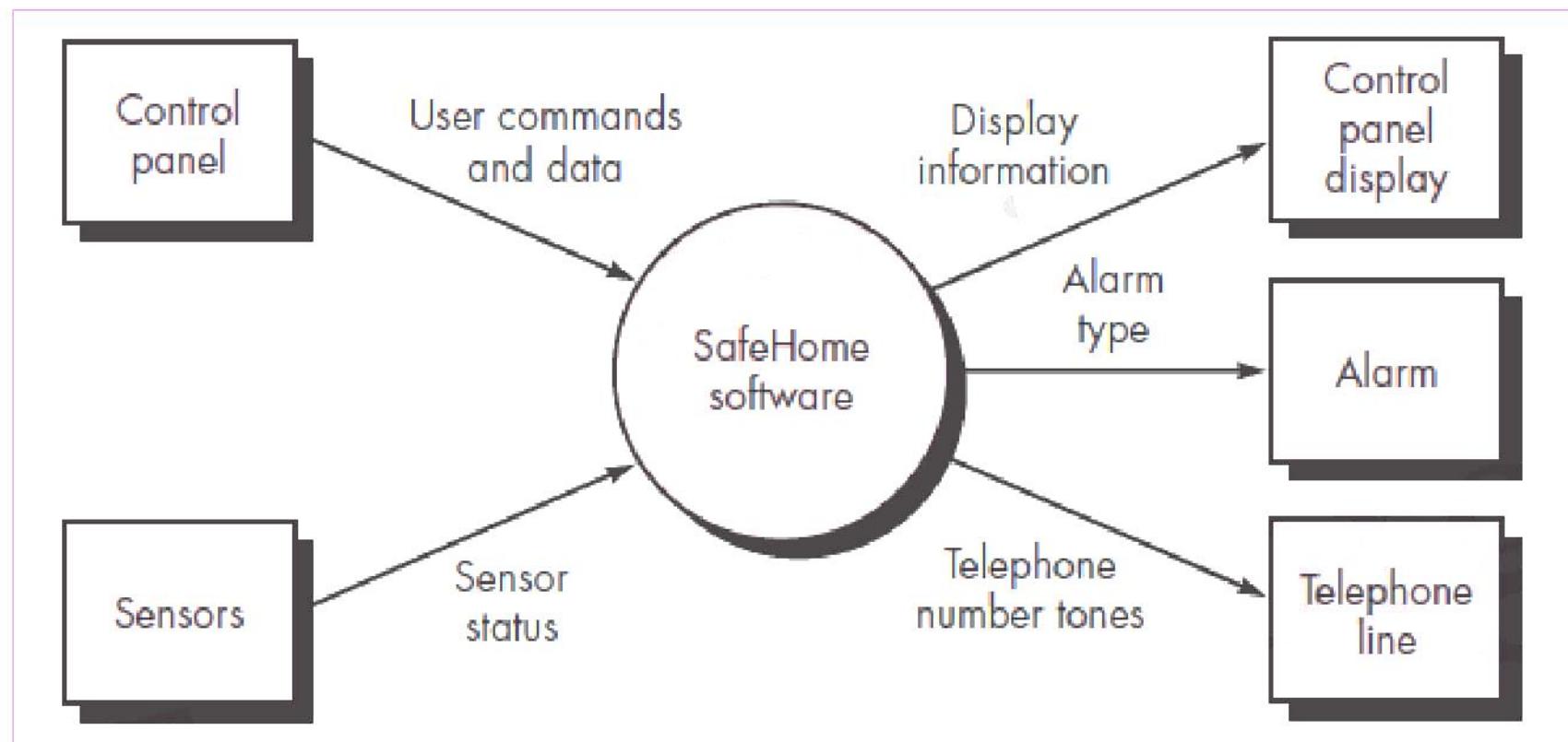
۳- پالایش باید با جداسازی فرآیندهای کاندیدا، اشیاءی داده ای و مخزن های داده ای که قرار است در سطح بعد به نمایش در آیند، آغاز گردد.

۴- همه پیکان ها و حباب ها باید با نام های مناسب نشان گذاری شوند.

۵- پیوستگی جریان اطلاعات باید از سطحی به سطح دیگر حفظ شود.

۶- هر بار تنها یک حباب را باید پالایش کرد.

سطح صفر DFD



چهار گوش ها: اطلاعات مورد نیاز سیستم را تولید و اطلاعات تولید شده توسط سیستم را مصرف می کنند.

پیکان های بر چسب دار : اشیاءی داده ای یا سلسله مراتب هایی از اشیاءی داده ای

تجزیه گرامری

The SafeHome security function enables the homeowner to configure the security system when it is installed, monitors all sensors connected to the security system, and interacts with the homeowner through the Internet, a PC, or a control panel.

During installation, the SafeHome PC is used to program and configure the system. Each sensor is assigned a number and type, a master password is programmed for arming and disarming the system, and telephone number(s) are input for dialing when a sensor event occurs.

When a sensor event is recognized, the software invokes an audible alarm attached to the system. After a delay time that is specified by the homeowner during system configuration activities, the software dials a telephone number of a monitoring service, provides information about the location, reporting the nature of the event that has been detected. The telephone number will be redialed every 20 seconds until telephone connection is obtained.

The homeowner receives security information via a control panel, the PC, or a browser, collectively called an interface. The interface displays prompting messages and system status information on the control panel, the PC, or the browser window. Homeowner interaction takes the following form . . .

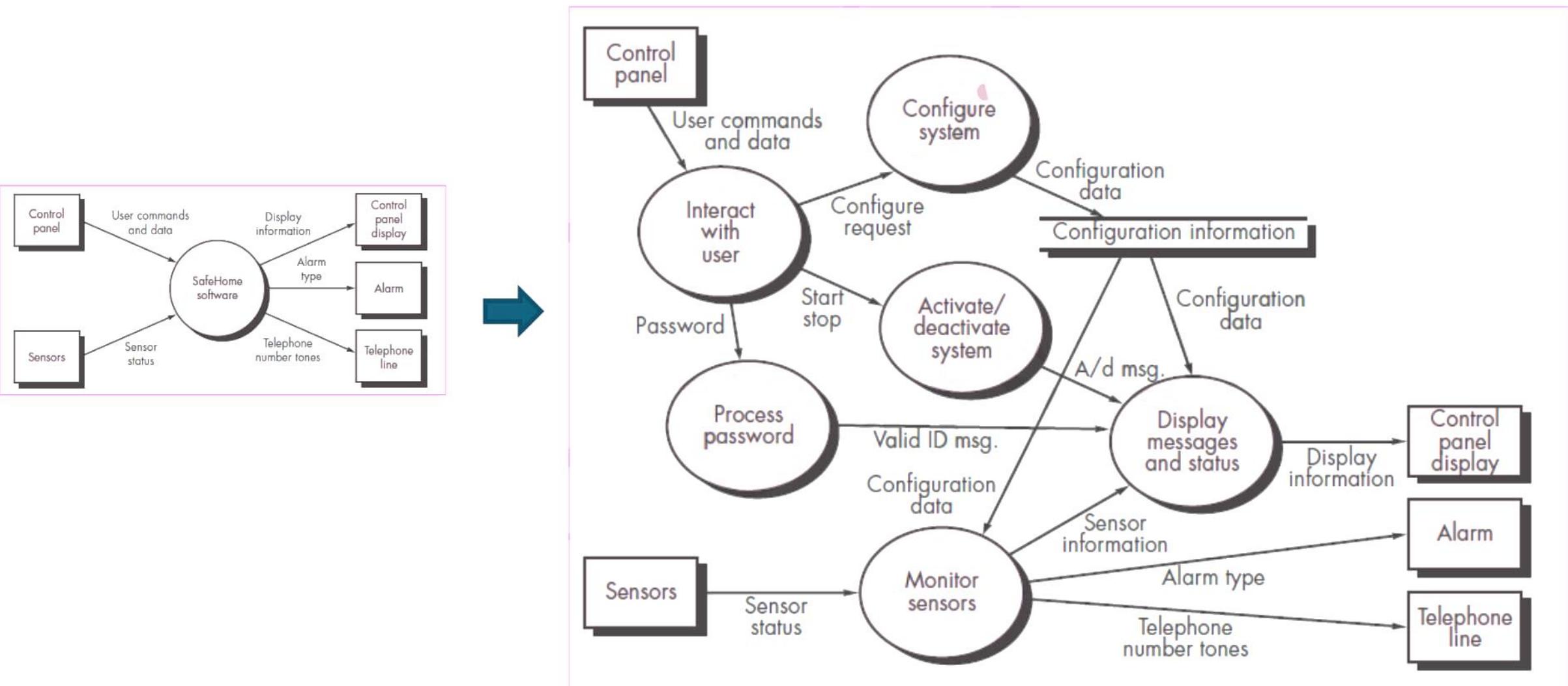
در متن روایی :

- فعل ها (حباب)
- اسم ها یا موجودیت های خارجی (چهار گوش ها)
- اشیاءی داده ای و کنترل (پیکان ها)
- مخزن داده ها (خطوط موازی)

فعل ها و اسم ها را نیز می توان به هم مرتبط کرد.

سطح ۱ DFD

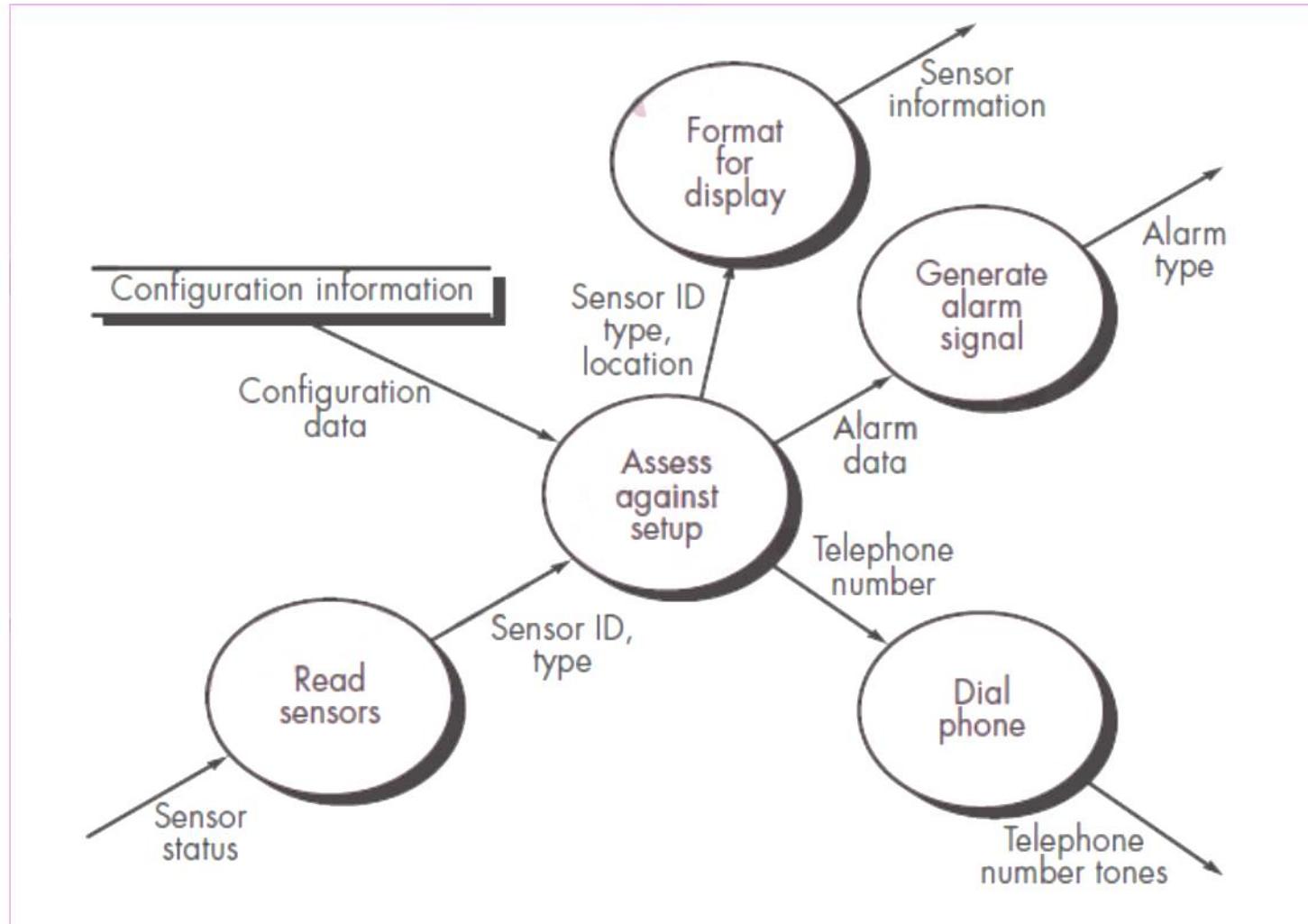
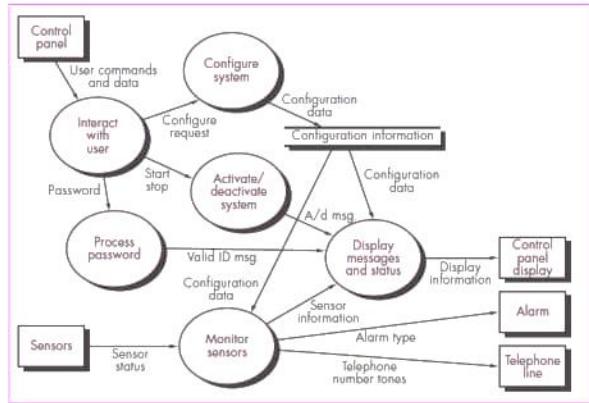
فرآیند به ۶ فرآیند بسط داده شده است که از بررسی تجزیه گرامی به دست آمده اند.
پیوستگی جریان اطلاعات بین سطوح صفر و یک حفظ شده است.



۲ سطح DFD

فرآیندهای DFD سطح یک را باز هم میتوان به سطوح پایین تر پالایش کرد.

برای مثال فرایند monitor sensors را می توان به یک DFD سطح دو پالایش کرد.



پالایش DFD تا کجا ادامه می یابد؟

پالایش DFD ها چندان ادامه می یابد که هر حباب تنها یک عملکرد را نشان دهد. یعنی تا هنگامی که فرایند نشان داده شده توسط حباب، عملی را انجام دهد که به راحتی به عنوان یک مؤلفه ی برنامه قابل پیاده سازی باشد.

ایجاد مدل جریان کنترل

گروه بزرگی از برنامه های کاربردی بیشتر توسط **رویدادها** اداره می شوند تا توسط داده ها،

بیشتر اطلاعات کنترلی تولید می کنند تا گزارش و چیزهایی برای نمایش.

و اطلاعات را با توجه جدی به زمان و کارایی پردازش می کنند.

این گونه برنامه های کاربردی علاوه بر مدل سازی جریان داده ها به مدل سازی جریان کنترل هم نیاز دارند.

یک آیتم کنترلی یا رویدادی به صورت مقدار **بولی** یا فهرستی مجزا از **شرط ها** پیاده سازی می شود.

برای انتخاب رویدادهای بالقوه ی کاندیدا، دستور العمل های زیر پیشنهاد می شود:

- همه حس گرهایی را که نرم افزار می خواند، فهرست کنید.
- همه شرایط وقفه را فهرست کنید.
- همه کلیدهایی را که توسط اپراتور فعال می شوند، فهرست کنید.
- همه شرایط داده ها را فهرست کنید.
- با به کار بردن تجزیه اسمی/ فعلی که در متن روایی پردازش به کار برده شد، همه ی آیتم های کنترلی را به عنوان ورودی ها/ خروجی های ممکن برای تعیین مشخصات کنترلی مرور کنید.
- رفتار سیستم را با شناسایی حالت ها، شناسایی چگونگی رسیدن به هر حالت و تعیین گذارهای میان حالت ها توصیف کنید.
- جا افتادگی های ممکن را کانون توجه قرار دهید.

تعدادی از ایتم های کنترلی و رویدادی نرم افزار SafeHome

events and control items

- **sensor event** (a sensor has been tripped)
- **blink flag** (a signal to blink the display)
- **start/stop switch** (a signal to turn the system on or off)

مشخصات کنترل

The Control Specification(CSPEC)

مشخصات کنترل (CSPEC) رفتار سیستم را به دو شیوه به نمایش می‌گذارد.



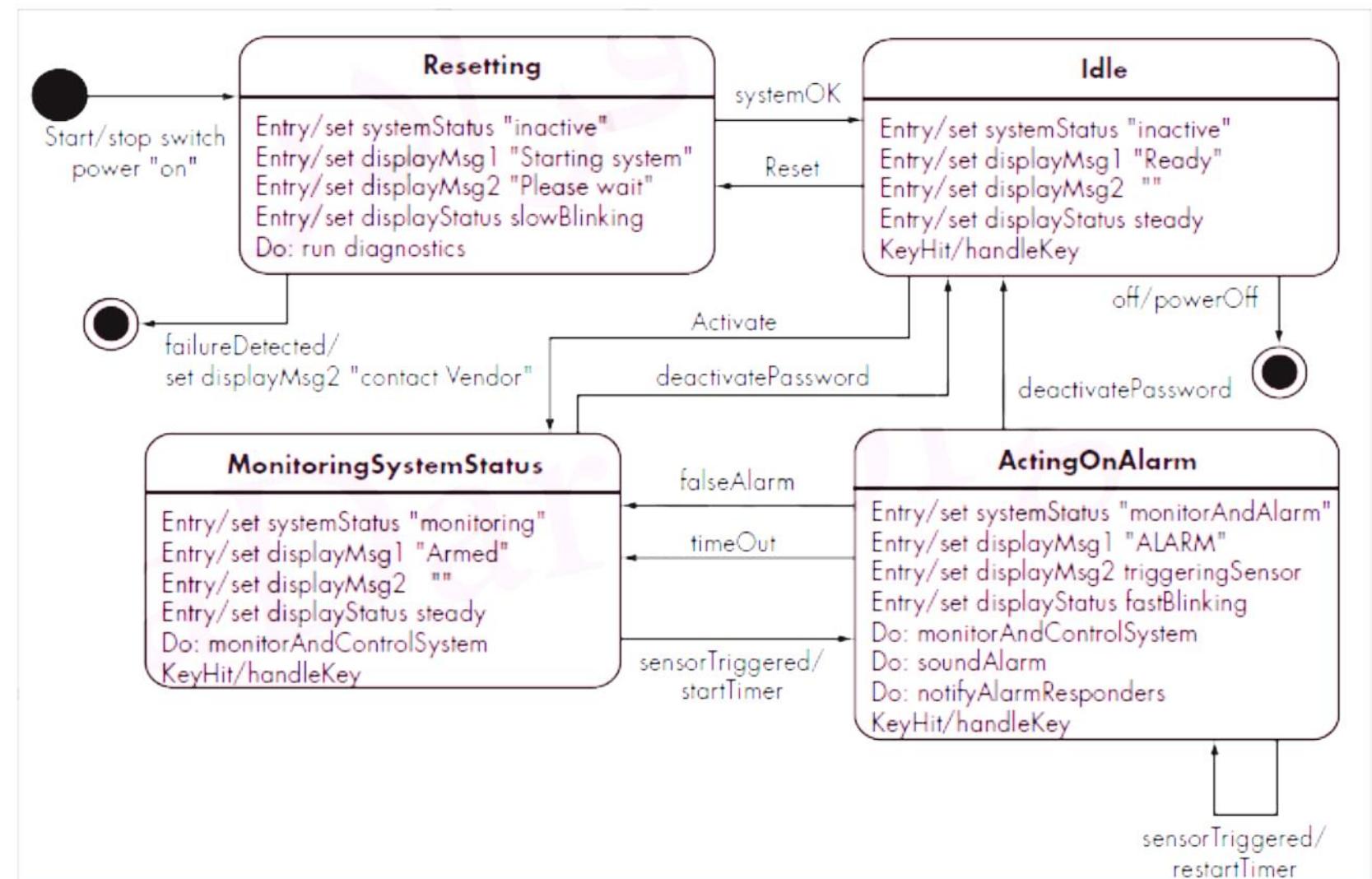
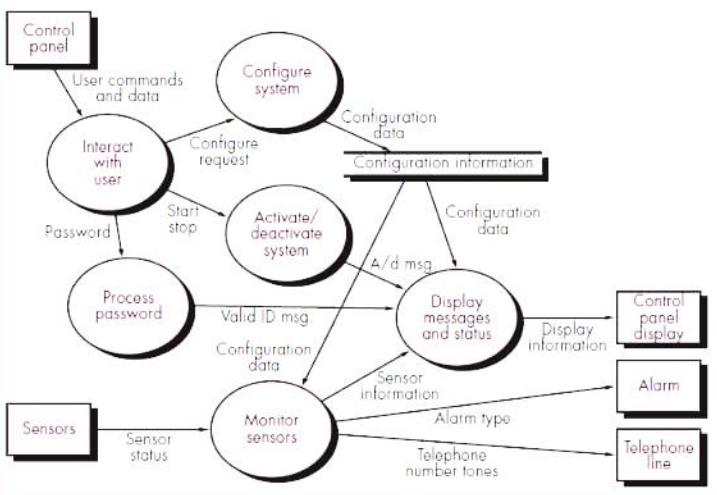
CSPEC حاوی یک نمودار حالت است که رفتار را به صورت ترتیبی مشخص می‌کند.

این مشخصات همچنین حاوی یک جدول فعال سازی برنامه‌ها است.

CSPEC رفتار سیستم را توصیف می‌کند ولی درباره کارکرد داخلی فرایندهایی که در نتیجه این رفتار فعال می‌شوند، هیچ اطلاعاتی نمی‌دهد.

نمودار حالت مقدماتی

preliminary state diagram



جدول فعال سازی فرایندها (PAT)

یک شیوه نسبتاً متفاوت برای نمایش رفتار است. PAT اطلاعات موجود در نمودار حالت را در حیطه‌ی فرایندها و نه حالت‌ها، نشان می‌دهد. این جدول نشان می‌دهد که کدام فرایندها (حباب‌ها) در مدل جریان، هنگامی فراخوانی می‌شود که رویدادی رخ دهد.

input events						
sensor event	0	0	0	0	1	0
blink flag	0	0	1	1	0	0
start stop switch	0	1	0	0	0	0
display action status complete	0	0	0	1	0	0
in-progress	0	0	1	0	0	0
time out	0	0	0	0	0	1
output						
alarm signal	0	0	0	0	1	0
process activation						
monitor and control system	0	1	0	0	1	1
activate/deactivate system	0	1	0	0	0	0
display messages and status	1	0	1	1	1	1
interact with user	1	0	0	1	0	1

مدل سازی جریان داده ها

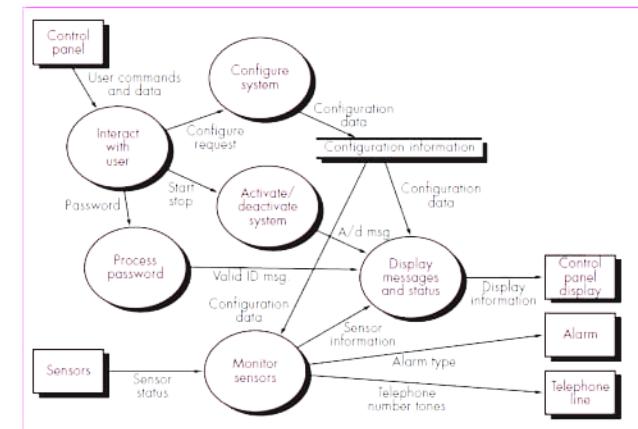
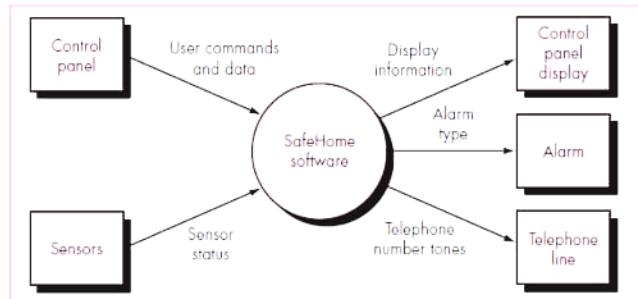


ME

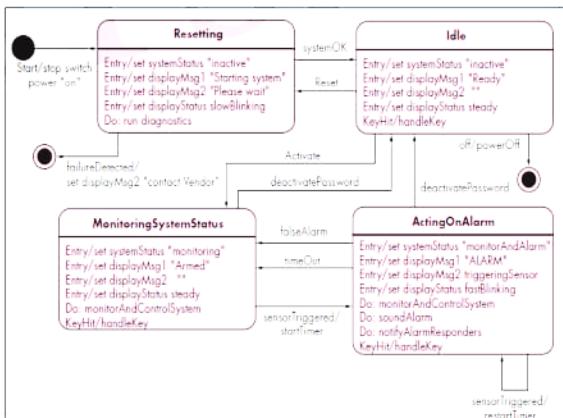
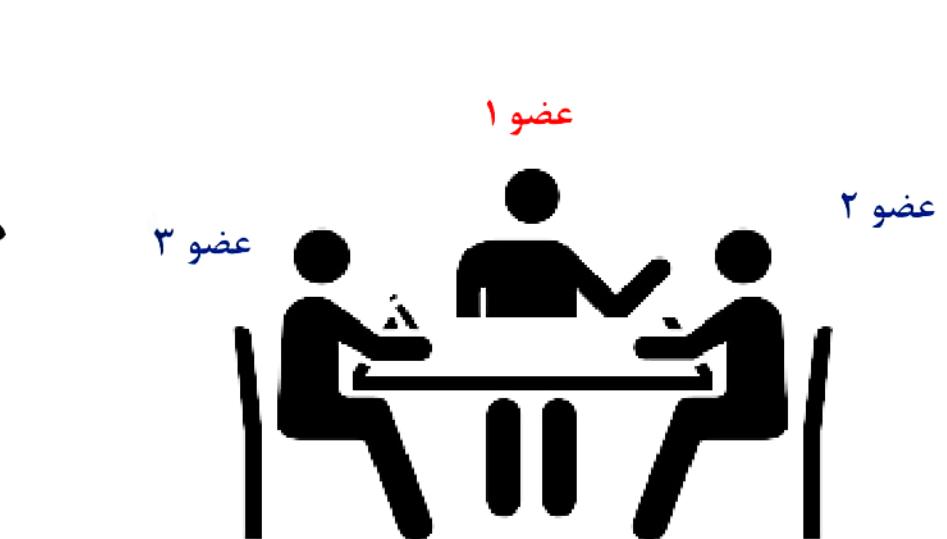
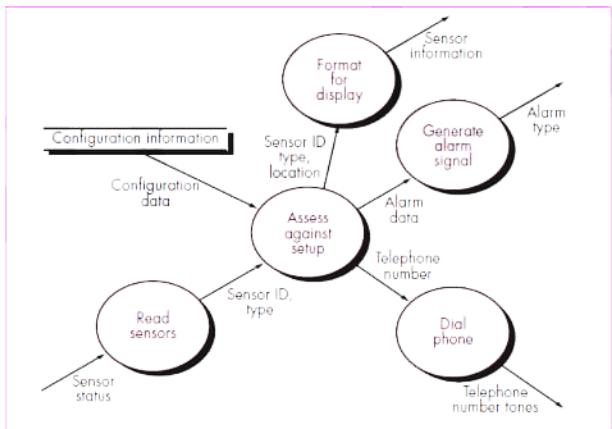
Data Flow Modeling

The scene: Jamie's cubicle, after the last requirements gathering meeting has concluded.

The players: Jamie, Vinod, and Ed—all members of the SafeHome software engineering team.



Input-process-output(IOP)



input events	sensor event	0	0	0	0	1	0
blink flag		0	0	1	1	0	0
start stop switch		0	1	0	0	0	0
display action status complete		0	0	0	1	0	0
in-progress		0	0	1	0	0	0
time out		0	0	0	0	0	1

output	alarm signal	0	0	0	0	1	0
		0	0	0	0	1	0

process activation	monitor and control system	0	1	0	0	1	1
activate/deactivate system		0	1	0	0	0	0
display messages and status		1	0	1	1	1	1
interact with user		1	0	0	1	0	1

مشخصات فرایندها

The Process Specification

مشخصات فرایندها (PSPEC) در توصیف همه ی فرایندهای مدل جریان که در سطح نهایی پالایش ظاهر می شوند، کاربرد دارد.

محتوای تعیین مشخصات فرایندها می تواند شامل متن روایی، توصیفی از زبان طراحی برنامه (PDL) برای الگوریتم فرایند، معادلات ریاضی، جداول، یا نمودارهای فعالیت های UML باشد.

program design language(PDL)

PSPEC ریز مشخصاتی برای هر تبدیل در پایین ترین سطح پالایش در DFD است.

مثالی از کاربرد PSPEC

PSPEC: process password (at control panel).

The *process password* transform performs password validation at the control panel for the *SafeHome* security function.

Process password receives a four-digit password from the *interact with user* function.

The password is first compared to the master password stored within the system.

If the master password matches, `<valid id message = true>` is passed to the *message and status display* function.

If the master password does not match, the four digits are compared to a table of secondary passwords (these may be assigned to house guests and/or workers who require entry to the home when the owner is not present).

If the password matches an entry within the table, `<valid id message = true>` is passed to the *message and status display* function.

If there is no match, `<valid id message = false>` is passed to the *message and status display* function.

ایجاد مدل رفتاری

CREATING A BEHAVIORAL MODEL

این مدل سازی به رفتار پویای سیستم می پردازد.

مدل های رفتاری نشان می دهند که نرم افزار چگونه به رویدادها یا حرکت های خارجی پاسخ می دهد.

برای ایجاد این مدل باید مراحل زیر را اجرا کنید:

۱- ارزیابی همه موارد برای درک کامل تعامل های داخل سیستم

۲- شناسایی رویدادهایی که این تعامل ها را اداره می کنند و درک چگونگی ارتباط این رویدادها با اشیاءی مشخص

۳- ایجاد یک دنباله یا توالی برای هر use case

۴- ساخت یک نمودار حالت برای سیستم

۵- مرور مدل رفتاری برای نشان دادن درستی و سازگاری

شناسایی رویدادها به کمک use case

Identifying Events with the Use Case

Use case دنباله ای از فعالیت ها را نشان می دهد که کنش گر و سیستم را شامل می شوند.
هر گاه که سیستم و کنش گری به تبادل اطلاعات بپردازند یک رویداد رخ می دهد.

The homeowner uses the keypad to key in a four-digit password.

The password is compared with the valid password stored in the system.

If the password is incorrect, the control panel will beep once and reset itself for additional input.

If the password is correct, the control panel awaits further action.

بخش هایی از use case که زیر آن ها خط کشی شده است **رویداد ها** را نشان می دهند.

پس از شناسایی رویدادها باید آنها را به اشیاءی موجود تخصیص دهید.
اشیاء می توانند مسئول ایجاد رویدادها باشند یا رویدادهایی را که در جای دیگر رخ می دهند شناسایی کنند.

The **homeowner** uses the keypad to key in a four-digit password.

The password is compared with the valid password stored in the system.

If the password is incorrect, the **control panel** will beep once and reset itself for additional input.

If the password is correct, the **control panel** awaits further action.

نمایش حالت ها

در حیطه مدل سازی رفتاری، حالت ها را باید از دو نظر مشخص کرد:

- ۱- حالت هر کلاس در زمانی که به وظیفه خود عمل می کند.
- ۲- حالت سیستم از دید ناظر خارجی در زمانی که به وظیفه خود عمل می کند.

حالت یک کلاس هر دو خصوصیت انفعالی **active** و فعال **passive** را به خود می گیرد.

حالت انفعالی صرفاً حالت فعلی همه صفت های یک شیء است.

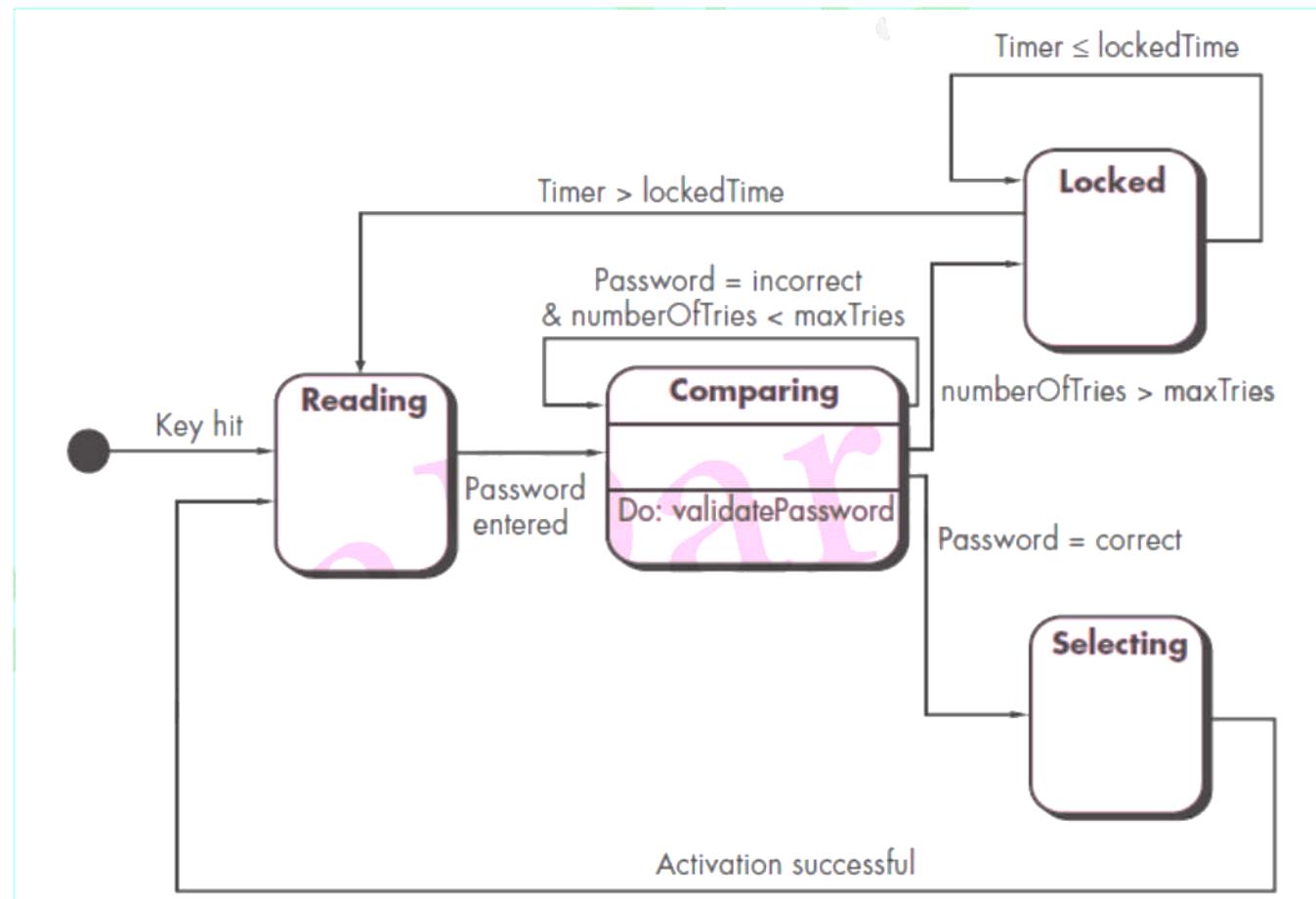
حالت فعال شیء وضعیت فعلی شیء را به هنگام قرار گرفتن در معرض تبدیل یا پردازش مستمر نشان می دهد.

سیستم حالت هایی دارد که رفتار خاص قابل مشاهده از بیرون را به نمایش می گذارد؛ کلاس دارای حالت هایی است که رفتار آن را به هنگام اجرای وظایف به نمایش می گذارد.

نمودار حالت برای کلاس های تحلیل

یک مؤلفه از مدل رفتاری، نمودار حالت UML است که حالت های فعال هر کلاس و رویدادهایی را نشان می دهد که باعث تغییر در این حالت های فعال می شوند.

نمودار حالت برای شیء control panel در عملکرد امنیت SafeHome :



هر کدام از پیکان ها گذاری از یک حالت فعال شیء به حالت فعال دیگر را نشان می دهد.

برچسب های روی هر پیکان رویدادی را نشان می دهند که گذار
را آغاز می کنند (transition).

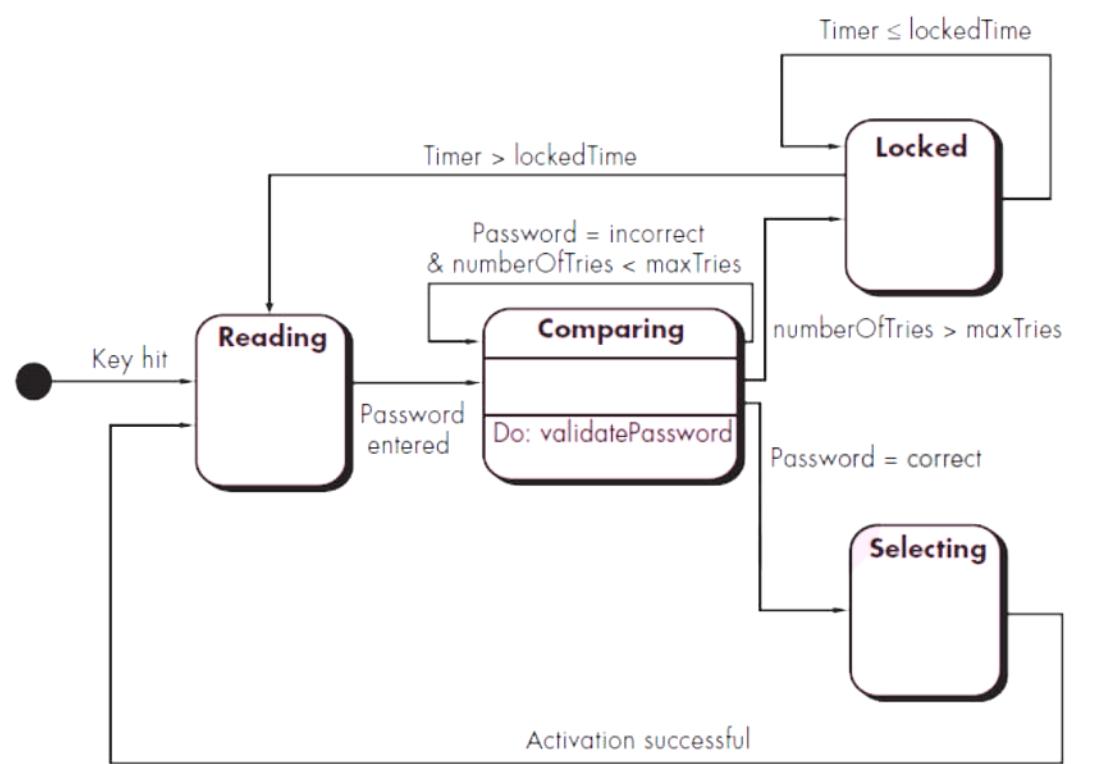
می توانید یک نگهبان و کنش مشخص کنید.

نگهبان یک شرط بولی است که باید برآورده شود تا گذار رخ دهد.

برای مثال نگهبان گذار از حالت reading به حالت
comparing را می توان با بررسی use case تعیین کرد:

If (password input = 4 digits) then compare to stored password

نگهبان مربوط به یک گذار معمولاً به مقدار یک یا چند صفت از شیء بستگی دارد.



نمودارهای ترتیب

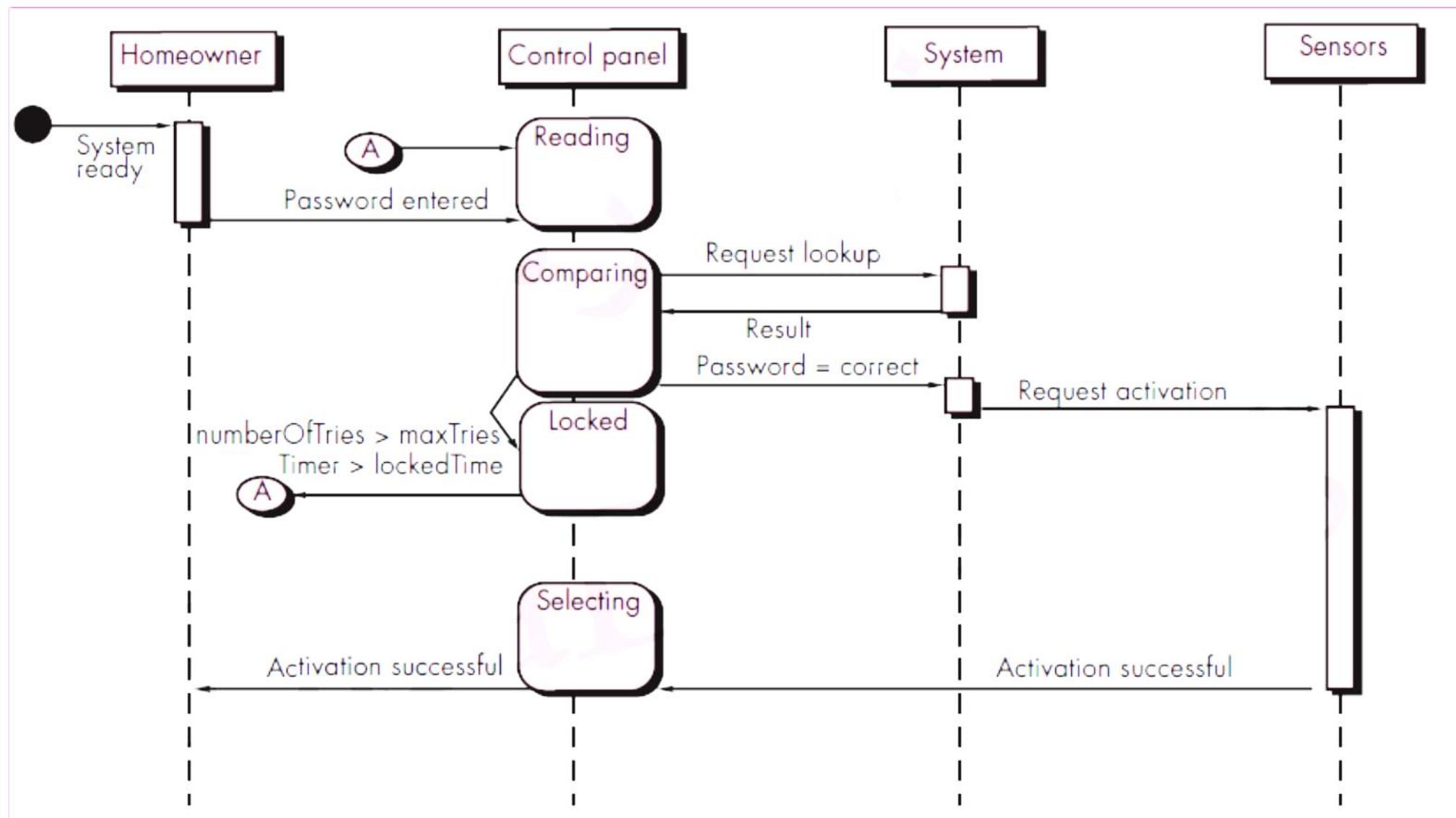
Sequence diagrams

یک روش نمایش رفتار که در UML نمودار ترتیب نامیده می شود، نشان می دهد که رویدادها چگونه باعث گذار یک شیء به شیء دیگر به عنوان تابعی از زمان می شوند.
نسخه خلاصه شده از use case است.

این نمودار کلاس های کلیدی و رویدادهایی را نشان می دهد که باعث جریان یافتن رفتار از کلاسی به کلاس دیگر می شوند.

برخلاف نمودار حالت که رفتار را بدون توجه به کلاس های موجود نشان می دهد، نمودار ترتیب رفتار را با توصیف چگونگی حرکت کلاس ها از حالتی به حالت دیگر به نمایش می گذارد.

بخشی از نمودار ترتیب برای عملکرد امنیت



هر کدام از پیکان ها نشان گر یک رویداد به دست آمده از use case بوده چگونگی کانال زدن رویداد بین اشیاء برای جریان یافتن رفتار را نشان می دهد.

زمان به صورت عمودی (بالا به پایین) سنجیده می شود و مستطیل های باریک عمودی زمان صرف شده در پردازش یک فعالیت را نشان می دهد.

پس از اتمام نمودار ترتیبی، همه رویدادهایی را که باعث گذار میان اشیاءی سیستم می شوند، می توان در مجموعه ای از رویدادهای ورودی و رویدادهای خروجی جمع آوری کرد.

الگوهایی برای مدل سازی خواسته ها

PATTERNS FOR REQUIREMENTS MODELING

الگوهای نرم افزاری، ساز و کارهایی هستند برای آگاهی از دامنه به شیوه ای که بتوان هنگام مواجهه با مسئله ای جدید، آن ها را دوباره به کار برد.

نویسنده ای اصلی یک الگوی تحلیل ، الگو را ایجاد نمی کند، بلکه آن را به موازات اجرای کار مهندسی خواسته ها کشف می کند.

الگوهای تحلیل راهکاری را نشان می دهد که غالبا شامل یک کلاس، یک عملکرد یا رفتار در داخل دامنه کاربردی است.

الگوهای تحلیل در یک مخزن ذخیره می شوند.

کشف الگوهای تحلیل

Discovering Analysis Patterns

مدل خواسته ها از گستره وسیعی از عناصر تشکیل می شود:

مبتنی بر سناریو (Use case)، داده محور (مدل داده ها)، مبتنی بر کلاس، جریان گرا و رفتاری.

Use case اصلی ترین عنصر در توصیف مدل خواسته هاست.

مجموعه ای یکپارچه از Use case می تواند به عنوان مبنا و اساسی برای کشف یک یا چند الگوی تحلیل عمل کند.

الگویی است که مجموعه‌ی کوچکی از Use case یکپارچه را توصیف می‌کند که به همراه یکدیگر، یک کاربرد کلی و پایه‌ای را توصیف می‌کنند.

پایش و کنترل حسگر و محرک در یک سیستم فیزیکی به کمک نرم افزار:

Use case: *Monitor reverse motion*

Description: When the vehicle is placed in *reverse* gear, the control software enables a video feed from a rear-placed video camera to the dashboard display.

The control software superimposes a variety of distance and orientation lines on the dashboard display so that the vehicle operator can maintain orientation as the vehicle moves in reverse.

The control software also monitors a proximity sensor to determine whether an object is inside 10 feet of the rear of the vehicle.

It will automatically break the vehicle if the proximity sensor indicates an object within x feet of the rear of the vehicle, where x is determined based on the speed of the vehicle.

نرم افزار در بسیاری از دامنه های کاربرد متفاوت برای پایش حس گرها و کنترل حرک های فیزیکی ضروری است.

بنابراین یک الگوی تحلیل که خواسته های کلی را برای این توانایی توصیف کند، به طور گسترده قابل استفاده خواهد بود.

مثالی از الگوی خواسته ها

نام الگو : Actuator-Sensor

هدف

انگیزه

قید و بندها

قابلیت استفاده

ساختار

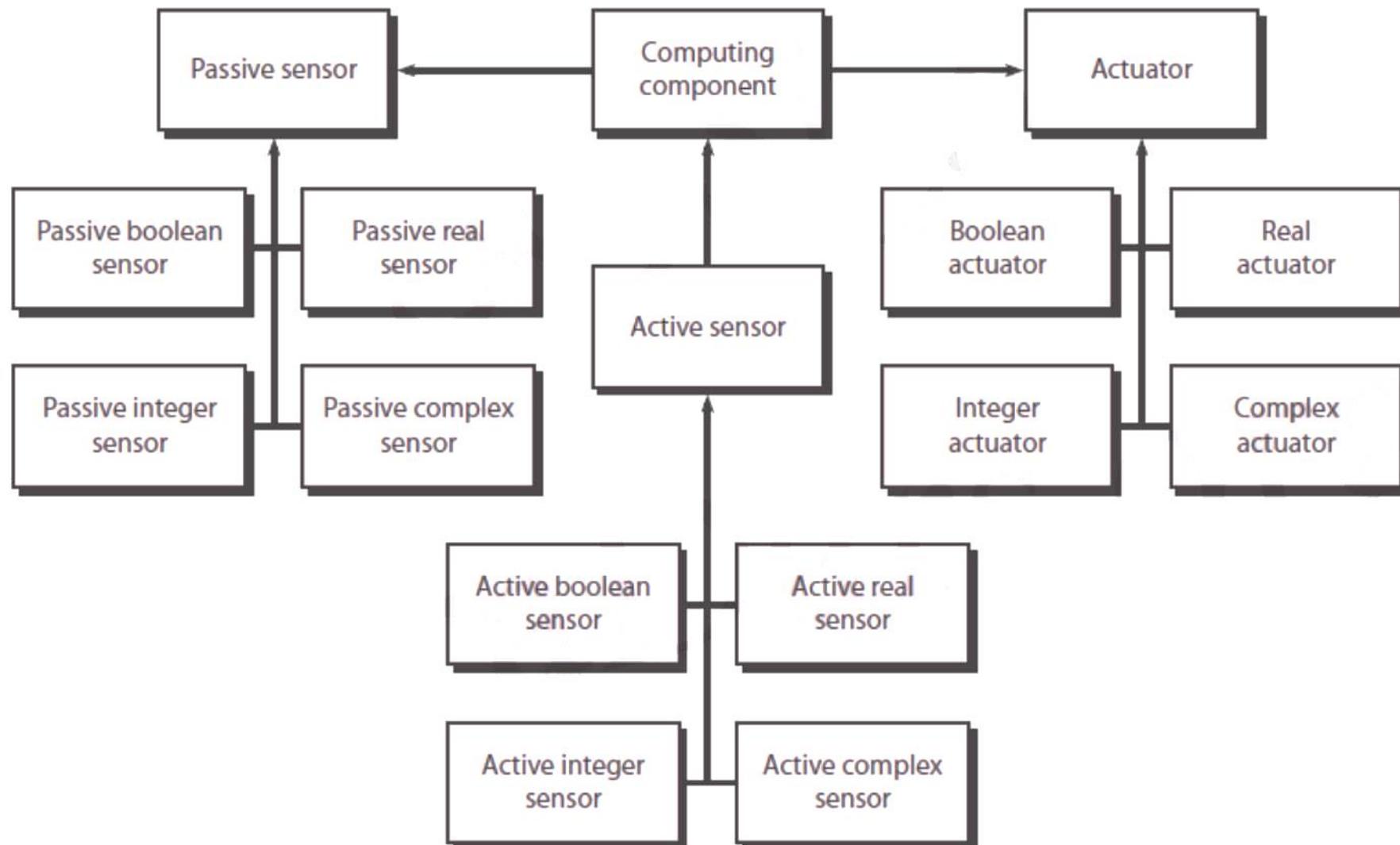
رفتار

مشارکت کنندگان

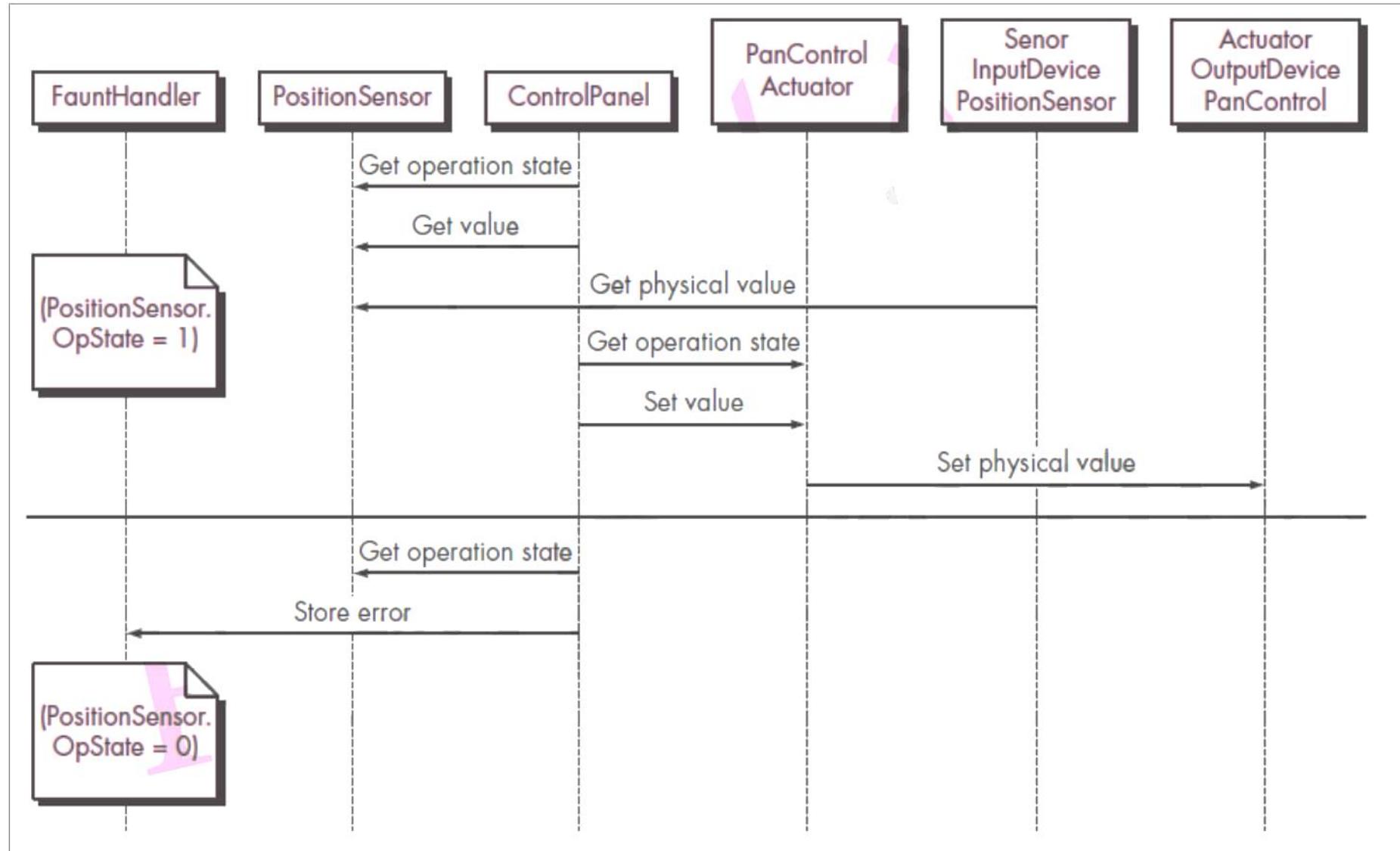
همکاری ها

پیامدها

نمودار ترتیب برای الگوی Actuator-Sensor



نمودار کلاس های UML برای الگوی Actuator-Sensor



پایان فصل ۷