

# **OPTIONS TO INTERNET ENABLE A TRADITIONAL APPLIANCE CONTROL**

**Dale Mayes  
Home Port Engineering LLC  
20004 103rd Court NE  
Bothell, WA 98011  
U.S.A.**

**Telephone: 425.876.1915**

**Fax: 484.970-0941**

**E-Mail: [DMayes@HomePortEngineering.com](mailto:DMayes@HomePortEngineering.com)**

**Web Address: [www.HomePortEngineering.com](http://www.HomePortEngineering.com)**

## ABSTRACT

*AHAM has published their "Connected Home Appliance – Object Modeling" standard. So now we are done, right? ...All the OEMs know exactly how to Internet enable their products and all the products will interoperate seamlessly.*

*A typical Internet enabled system will add at least three new elements to an existing appliance control system: a Network Interface, a Local Host, and a Remote User Interface. There are a variety of options available for each of the new elements, and numerous methods of interconnection.*

*What makes standardization so difficult is that, any of the new elements can be either a new physical device, or virtual functionality added to an existing processor. In other words, the system can be as simple as adding a TCP/IP stack and modem chip to an existing control, or as complex as adding a network interface card, a power line communications modem, a residential gateway, a web pad with RF communications, and a remote web site.*

*This paper/presentation will explore some design alternatives for internet enabling devices, including the pros and cons associated with the potential approaches.*

## BACKGROUND

### AHAM's Smart Appliance Task Force

AHAM's Connected Home Appliances – Object Modeling standard focused on the commonplace behavior that was controllable and observable. Its stated purpose was to promote new appliance services and features enabled through networking. The mission of the Smart Appliance Task Force stated the AHAM abstracted models would not be directly coupled to a product, but that manufacturers would map the AHAM features and functions to their appliances. Figure 1 graphically represents the committee's intent. There were two views discussed during the development of the spec. "OEM X" is utilizing and extending the AHAM models. Their API includes both the AHAM "public" methods, and extended private methods that only they or their partners know. "OEM Y" has implemented the AHAM public API, but also has a separate private implementation. In both cases, the OEM is interpreting the AHAM high-level messages and tailoring them for their product's implementation. Though not shown on the diagram, a subset of OEM X's User Interface, which is limited to the "public" messages, could utilize OEM Y's "public" API.

The mission of the smart appliance task force included creating a set of appliance models that could be mapped to various protocols. The task force purposely avoided the lower layers in the OSI 7 layer model.

### Overview of three New Elements

Internet appliances will generally adhere to the client-server or multi-tier

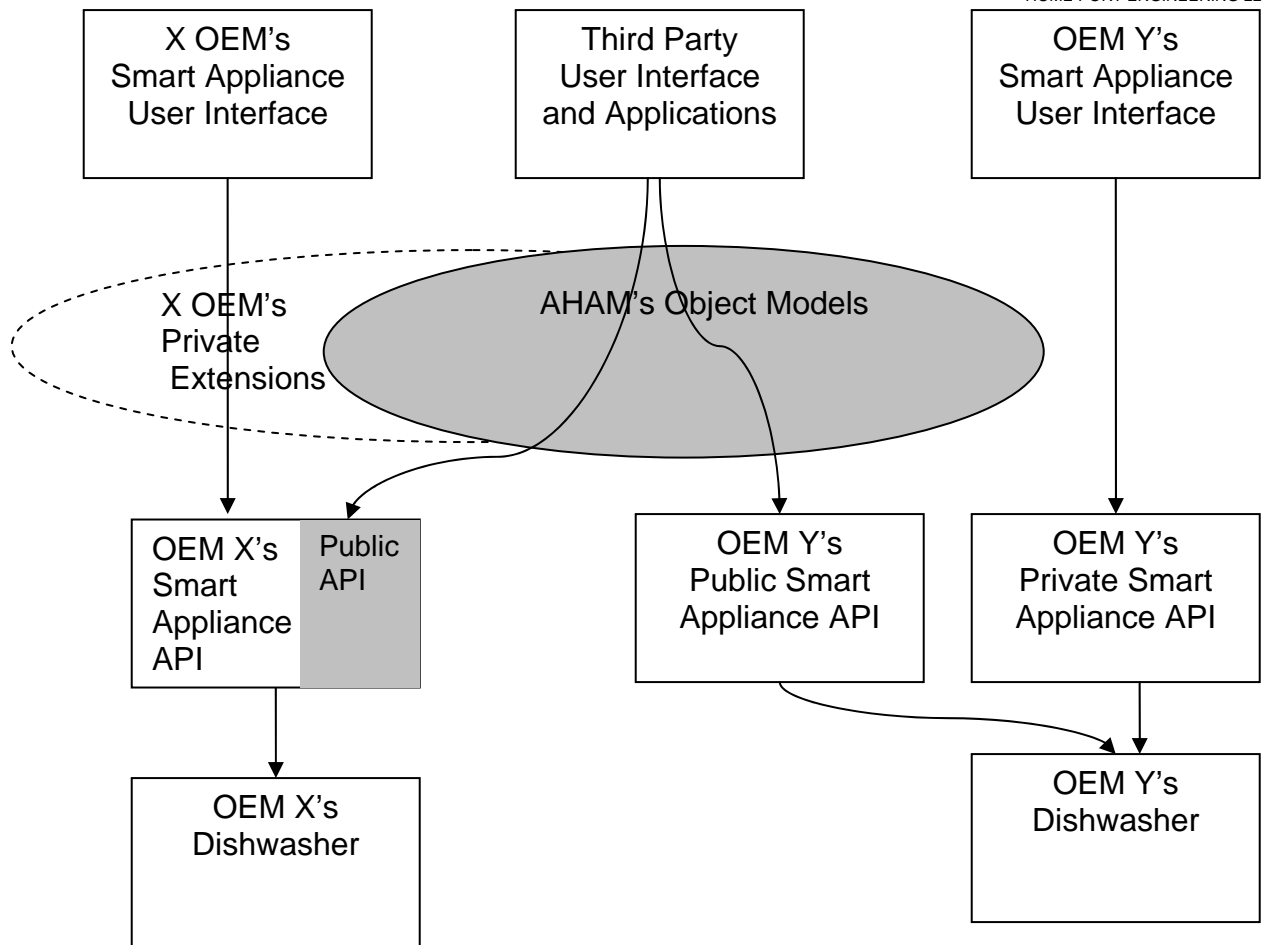


Figure 1 - AHAM's Model

paradigm. The three new elements are a client, server, and a new physical interface at the appliance.

The 'Remote User Interface' represents the user interface device(s) and client side technologies. A smart appliance system can have multiple Remote User Interface targets.

The 'Local Host' provides execution space for the new smart appliance functionality. This element can be integrated into the appliance, reside on a new platform in the home or on a home PC, or can even be out in web space.

The 'Network Interface' is a generic name for the connection from the appliance.

There are numerous options for each of these elements. Even more alternatives are available for the actual implementation.

### EXAMPLE SOLUTIONS

Before we talk about all the possible permutations, let us focus on a couple of possible solutions.

#### Option 1

This example system consists of power line communications for the Network Interface, a residential gateway for the Local Host, and a web tablet for the browser based Remote User

Interface. As Figure 3 shows, the appliances communicate via power line to the residential gateway. There are new Network Interface Cards added to the appliances. The Network Interface Cards handle the power line coupling, and interface with the appliance via serial communication. The gateway serves several purposes. Foremost, it is the Local Host providing an execution platform for the new appliance services. It also operates as a firewall for the home, so it has two Ethernet cards to physically isolate the WAN and LAN. The WAN (Wide Area Network) card is connected to the broadband modem; the other card connects to the home's LAN (Local Area Network). The new appliance services have a web based user interface. In other words, any browser that can access the gateway can exercise the new functionality. The system also shows a portable tablet PC on the refrigerator.

### Options 2

In another example, let's just add a TCP/IP stack and modem chip to an appliance control. In this case, the Network Interface is the phone line. The appliance can either send status messages to a web based Local Host, or be its own Local Host. As its own Local Host, it can send text messages or basic HTML pages. The Remote User Interface in this example is either text or browser based status messages. Because the appliance is not always connected, there can be no control.

### Options 3

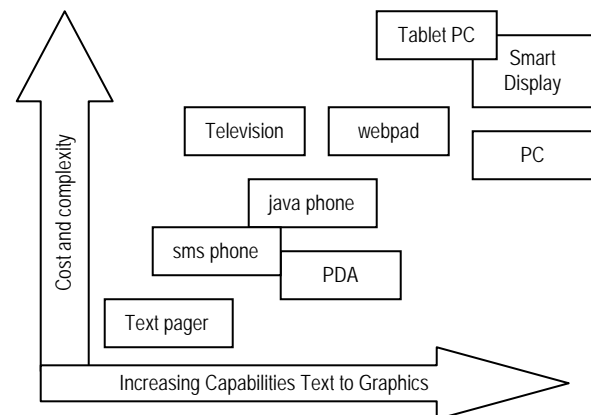
For this final example, assume we have added a 2-way pager into the

appliance. The Network Interface in this case is the pager. The Local Host will be web based. The Remote User Interface in this example remains a browser based PC.

## OPTIONS FOR THE NEW ELEMENTS

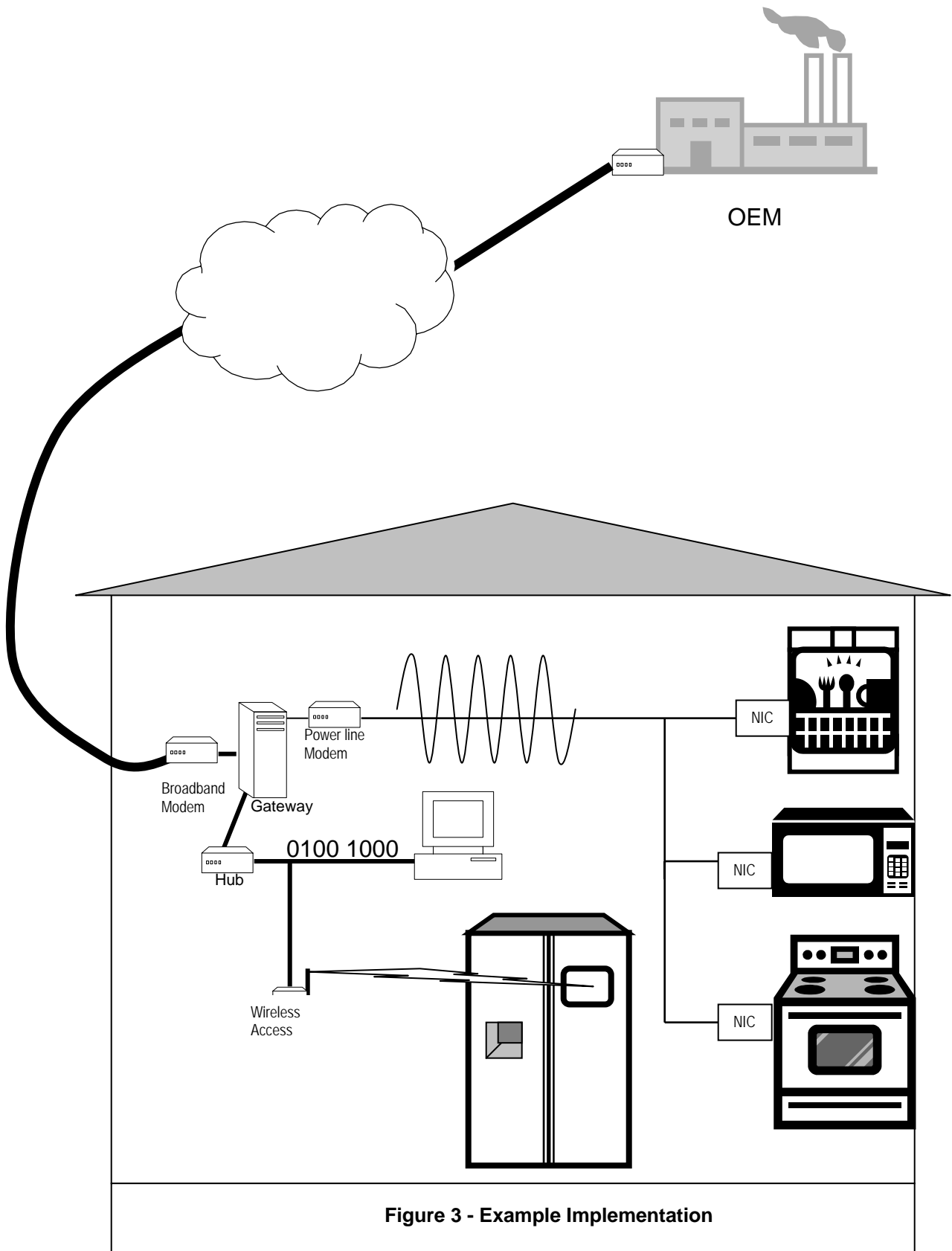
### 'Remote User Interface'

There are two main categories of user interfaces, graphical and text based. As Figure 2 shows, typically text based interfaces are less costly, less complicated, but have fewer capabilities.



**Figure 2 - Remote User Interface Options**

Browser based graphical user interfaces for a PC actually encompasses a variety of approaches. They can be static HTML or dynamic DHTML. The dynamic portion is typically accomplished with JavaScript. Flash is another alternative. Because the client is interpreting the HTML and JavaScript at runtime, the browser (IE, Netscape, Opera...) and its version, can have a



dramatic influence on the look of the final page. If a consistent look is vital across a variety of platforms, Java Applets can remove the browser version dependence because the Java Virtual Machine (JVM) is the rendering agent. An applet is invoked from an HTML page. A simple test for the correct JVM will ensure consistent rendering. If the proper JVM is not present, the user would be prompted to approve its download from Sun's site. Just like other browser plug-ins, once the JVM is installed on the client, it does not need to be downloaded again. It is just automatically used. Applets have a reputation of being slow because all their classes must be loaded into the JVM. However, with accurate applet construction and proper cache management by the JVM, applets are a viable solution. Server-side scripting is another method to achieve dynamic pages. However, many small local hosts do not have this capability.

One of the tenets of web design is, you can never push content to a browser. The browser always pulls information.

There are over a dozen web pads or tablet PCs on the market today. Several vendors have just released models built on Microsoft's Windows XP Tablet PC edition. Perhaps these new systems will have better performance than the previous web pads. Historically, web pads have been limited in their software versions, processor speeds,

and storage. A typical web pad is roughly equivalent to a 4-year-old computer. If using JavaScript on one of these devices, it is vital to understand what Documentation Object Model (DOM) level the browser supports. If using applets on a web pad, run performance tests on the JVM.

PDA's, Handheld PCs, and Java Phones have even more stringent limits than web pads. If using them as a Remote User Interface device, page content should be limited to text and simple graphics. The page sizes should also be tailored for these small-screened devices.

E-mail, Text Paging and Messaging offer some interesting non-GUI based alternatives. Configuring the Local Host with the user's account information is the biggest challenge of these simple devices.

The new Smart Display from Microsoft and ViewSonic is an interesting product. It is a wireless (802.11b) touch screen device that enables remote display and control of a desktop PC system. Think of it as being able to take the capability of your PC's monitor and mouse anywhere around your home, and using it on your lap.

A promising display alternative for the home is the television. With Microsoft's new media terminals, and solutions from the likes of Espial, the television has real potential to be a Remote User Interface solution for smart appliances.

### 'Local Host'

Connecting an appliance to a network can be a relatively trivial portion of Internet enabling a system. Depending on the Local Host selection, the bigger challenge comes with configuring the system and keeping up with environmental changes. For instance, a change of area code could cause a product to lose its ability to connect to a local ISP.

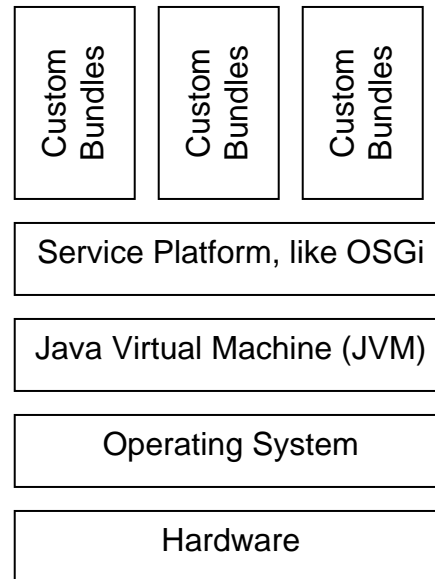
The approach taken for the Local Host is dependent on the Remote User Interface technology and the desired functionality. A small footprint server built into the appliance controller could be sufficient for sending text messages from a single device, but will be too limited to handle multiple appliances, or heavy graphical content. Other factors to consider when evaluating Local Host alternatives include:

- Scalability, adding devices
- Flexibility to handle both new features and other Remote User Interface devices
- Uptime of Local Host
- Systems for deploying software
- Interaction between products
- Skills for installation and setup
- Service needs for the system
- Outside of the home access
- Other non-appliance features

A dedicated stand-alone Local Host platform will provide the greatest flexibility and capability. These types of systems include, Special or Dedicated Purpose Gateways, Open Gateways, PC, Web based server.

A special purpose gateway is designed to accomplish a focused task or work with a limited set of

components. An Open Gateway can



**Figure 4 - Typical Gateway Structure**

execute software from a variety of sources. Both types of gateways should include a Service Platform that enables bundles to interoperate, see Figure 4. The Service Platform also handles the service registry, service discovery (match), and binding functions (attach). Most gateways also have bundle deployment/update capabilities.

A Residential Gateway is a special class of open or special purpose gateway. It typically adds home networking capabilities and a firewall.

The biggest advantage of deploying with a gateway as the Local Host is that it ensures a consistent hardware and software environment. Disadvantages include management of a new system and added hardware costs.

An existing home PC could be used as a Local Host, but it may not

always be on. Also with the variety of hardware, and proliferation of OS versions, no two installations would ever be the same.

As counter intuitive as it sounds, web based servers can also be a Local Host. Only, since they are not local, security and authentication concerns increase.

If an OEM plans to enable more than one appliance model, the local host should not be embedded in a product.

Even if no direct interaction is envisioned between the products, the user is going to have to navigate between all the appliances' user interface pages.

In addition to creating a smart appliance service, designers need to think about how the new smart appliance's icon gets added onto the system's main navigation page. Brute force may work when you just have a few appliances. It may be possible to build all the possible navigation pages for all the combinations of smart product icons, but how will the Local Host know which is the correct navigation page? In addition, what would happen if a user has two smart dishwashers? Did your pre-built navigation pages account for that scenario? A gateway's service platform enables this type of interoperability. For example, a new dishwasher may register with the service platform as an appliance service. Previously, the system's main navigation page service registered with the service platform and asked to be notified of any new appliance services. The service platform notifies the system's main navigation page service about the new dishwasher appliance service, and passes a handle to the

dishwasher's service. The system's main navigation page service attaches through the service platform, receives the dishwasher's icon and entry URL. The system's main navigation page service, then automatically includes the new dishwasher's icon on its page.

### 'Network Interface'

The following list of the potential technologies and providers is rather comprehensive, but is not exhaustive.

- Power line
  - x-10
  - CeBus
  - LonWorks
  - SCP
  - Ethernet
- RF
  - 802.11a
  - 802.11b
  - blue tooth
  - Home-RF-Lite (ZigBee)
- Wired serial
  - LonWorks (FFT)
  - RS232
  - RS485
- Ethernet
- Phone line
  - HPNA
  - modem connection
- 2-way pager
- Wireless modem (Cell phone)

When selecting a transportation medium for connecting the appliances there are several important questions:

- Are no new wires an important requirement?
- Is continuous access needed?
- How much data and how frequently?

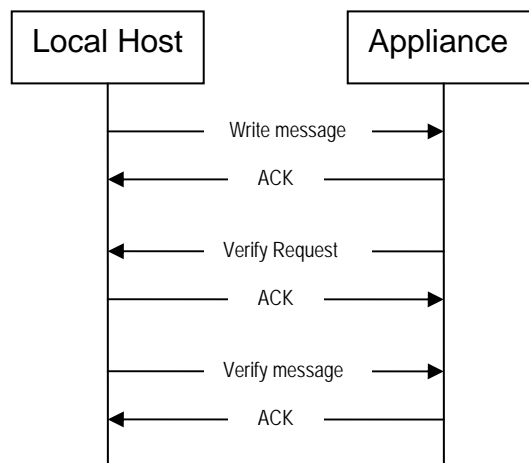
- What is the typical coverage in an average home? What can be done if there is not sufficient coverage?
- How do new devices join a network?
- How does the commissioning process ensure devices are associated with the proper network?
- What is the theoretical throughput of the channel?
- What is a realistic typical throughput of the system?
- How much overhead is associated with each message? In other words, what is the breakdown of message header vs. payload?

Other considerations include the kinds of communication your system will need. Most communication systems can run in a variety of modes, addressed / broadcast, acknowledged, encrypted, or verified. How many buss transactions will be required to complete the type of messaging you need? For example, as shown in Figure 5, a typical addressed, acknowledged, and verified write message will consist of 6 buss transactions to complete the one message.

How exposed is the media? For instance, power line and RF communications do not observe a home's boundaries or property lines. RF signals could be received at curbside, and will definitely travel between neighboring apartments/flats. Power line signals are generally readable from any point below the street transformer. In some countries, this can be over 50 homes.

### INTERFACING WITH AN EXISTING APPLIANCE CONTROL

What type of interface to the appliance control should you build into your system? Obviously, it depends on what features you plan, and the capabilities of the current appliance control. It is relatively easy to execute a 'dumb' on/off controller. There are commercially available remote breakers or switched outlets, but these would be best suited for enabling an electro-mechanical system. Figure 6 represents a simplified control and data flow diagram for a typical appliance. The boxes in the diagram represent the appliance control's modules. The ovals represent the global data structures. The arrowheads provide perspective on data access. An arrow into a data element represents writing data, an arrow from a data element represents a read.



**Figure 5 - Buss Transactions for Verified Acknowledged Write Message**

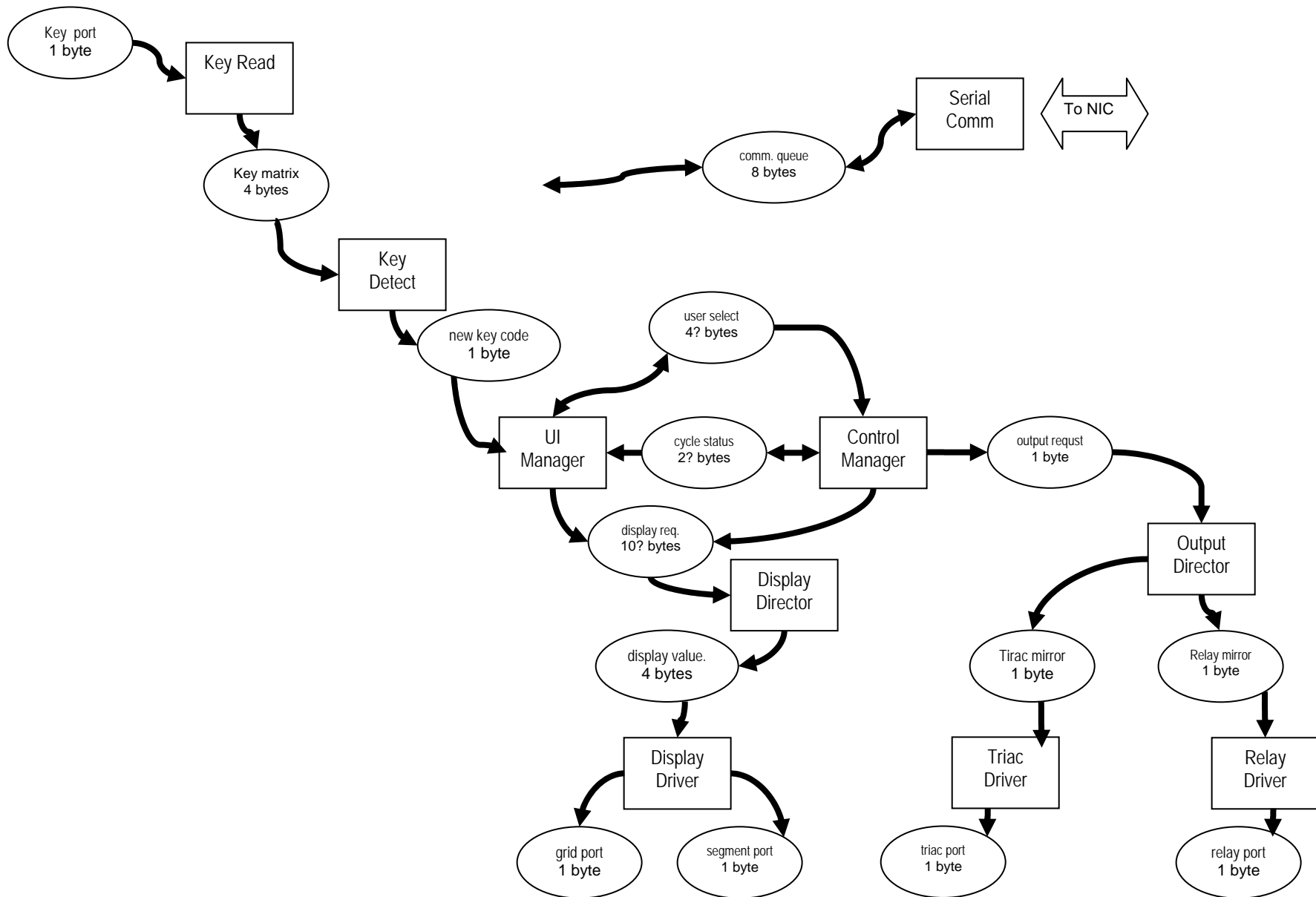


Figure 6 – Simplified Control and Data Flow Diagram for a typical appliance

Where is the logical point(s) to let a remote application interface? Another approach is to build an entirely separate remote control module into the control.

### **TO WHAT BENEFIT?**

There has been a lot of concern expressed about the lack of a killer-ap for internet enabling appliances. Coordinating appliances for meal preparation has huge potential benefits.

Most cooking appliances today have delay start, but their interface is not always intuitive, and often times confusing. A typical family's evening timeline is harried; the kids get home from school around 4. After a quick snack and some playtime, it is off to ballet class for Suzie and soccer practice for Jimmy. While Suzie is pirouetting, Mom takes a step class. Dad pick's Jimmy up from practice on his way home from the office. The whole family arrives home at 6:30, starving. Dinner's main dish takes 35 minutes in the oven. The side dishes require a series of steps in the microwave (23 minutes total). So the family eats dinner about 7:15.

Contrast this to the smart appliance solution; before Mom and the kids leave the house, the main dish is in the oven and the sides are in the microwave. Through the smart appliance web page, the recipes are selected with an 'eat time' set at 6:30. The smart appliance service automatically tweaks the cook time of the main dish to account for the lack of oven pre-heat. It also takes into account the microwave dish's 3-minute rest before serving step. Dinner is ready for the table when the family arrives home.

### **CONCLUSION**

Internet enabling an appliance is not trivial, nor is it impossible. There are many learning curves to climb. With the assistance of an experienced guide, many of the potential pit falls can be avoided.

Also, by designing the correct building blocks into today's controls, your products can be ready to exploit the new data networks and Remote User Interface devices emerging in homes today.

### **ADDITIONAL REFERENCES**

Grand, M. (1998) *Patterns in Java, Volume 1*, New York: John Wiley & Sons, Inc

Powell, T. (2002) *Web Design: The Complete Reference, Second Edition*, New York: McGraw-Hill/Osborne