NERES

PROGRESS REPORT

Nonintrusive Appliance Load Data Acquisition Method

George W. Hart

September 1984

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ENERGY LABORATORY
NORTHEAST RESIDENTIAL EXPERIMENT STATION
711 VIRGINIA ROAD
CONCORD, MASSACHUSETTS 01742

PROGRESS REPORT

Nonintrusive Appliance Load Data Acquisition Method

George W. Hart

September 1984

MIT Energy Laboratory
Northeast Residential Experiment Station
711 Virginia Road
Concord, Massachusetts 01742

Prepared for Electric Power Research Institute
Under Contract No. RP 2568-2

LEGAL NOTICE

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (CONTINUED)

TABLES

1.0 INTRODUCTION

This report presents the progress to date by MIT on the Non-intrusive Appliance Load Data Acquisition Method (EPRI Project # RP-2568-2). The goal of this project is to develop the algorithms necessary for the successful development of a nonintrusive residential load-monitoring device. This device is to be a low-cost stand-alone microprocessor-based instrument capable of determining the energy usage and operating characteristics of the individual major appliances in a residence, based only on the voltage and current measurements available at the utility kWh meter socket. After the device is installed in a residence, it will learn the nature of the appliances therein and gather statistics for load-research purposes on the individual appliances.

The most crucial part of the development of an algorithm capable of breaking down the total residential load into its individual appliance components is the determination of appliance signatures. Appliance signatures are defined to be characteristics of individual appliances which can be observed in the aggregate load. For the success of this project, it is necessary that suitable signatures be determined which can resolve different appliances as finely as possible, and which are consistent and stable under different appliance operating conditions. The signatures must also be of such a nature that they can be effectively measured, or calculated from measurements in real time by low-cost sensors and microprocessors. Signatures, whatever they may ultimately consist of, can be organized into a vector space in which the components are separably measurable independent parameters of the appliance operation (e.g., real power, reactive power, harmonic content). The vectorial nature of signatures is explored further in Section 3.1.

Given a space of signatures, the yet-to-be-developed algorithm will observe the aggregate voltage and current to track the operation of individual appliances and gather statistics (such as kWh demand vs. time of day) to be transmitted via telephone or stored on physical media for load researchers. To do this it must first learn what appliances are in the house, identifying them by signature. Then it can proceed to recognize successive occasions of the same appliance, tabulating the statistics of its operation. Eventually it should be able to identify the appliances, to the maximum extent possible, by their common names (e.g., "Refrigerator").

The major difficulties of the project are to develop an appropriate signature-space and an associated algorithm that is sophisticated enough to accomplish these goals with an accuracy high enough to make it worthwhile. Additional tasks of the project involve measuring the accuracy of the developed algorithms and determining the level of detail needed by load researchers. We expect, and results to date confirm, that the appliances which draw large amounts of power will be the easiest to identify consistently from a point external to the residence. Fortunately, these tend to be the larger energy users and of greatest interest to utilities.

Section 2 of this report breaks down the overall problem into seven sequential subproblems and reports on our general progress with each of them. In Section 3 the overall problem is sliced along other planes, into a set of detailed subproblems, each of which spans several of the subproblems of Section 2. Section 4 presents a simple working algorithm which can track individual appliances, given their signatures, but can not learn new ones. In Section 5, a tentative method for representing appliances and the state of the entire residence is developed; this work is still in the exploratory stages. The question of marginal value to load researchers of data on increasingly smaller appliances is discussed in Section 6.

2.0 BREAKDOWN OF PROBLEM

The overall problem is broken down into seven subparts which can be viewed as a sequence which leads to a complete device with all the desired properties. Of course the subparts are not solved in sequence, but simultaneously, after many iterations through the sequence. This breakdown is not identical to that described in the project work plan, but includes all the same tasks, and in retrospect is a more logical organization. In the following sections, the seven problems are described, and work completed or in progress by MIT is presented.

2.1 Understanding Appliances

The first of the six subproblems is for the researchers to attain an understanding of the nature of the enormous range of appliances available for residential use today. The different types of loads, circuits, and control mechanisms available produce different types of signatures as they operate. For example, refrigerators tend to cycle on and off three or four times per hour, many electric range burners switch on and off automatically within a period of 5-30 seconds, and hand-mixers with governor speed controls switch many times per second. Many motors draw a sharp current spike for several cycles, while others draw an increased current for seconds or longer as the shaft accelerates. Different types of electronic power-supply designs draw different characteristic harmonic currents. Such variations can be put to use in a signature space for distinguishing these appliances. The main goal of this subproblem is not a formal classification, but merely to sensitize our intuitions to the range of information available at a residence's electric service entrance which can be incorporated in putative signature spaces.

In addition to increasing our general knowledge of appliances from the consumer and engineering standpoints, we have made measurements of appliances with four sets of instruments to see what we could observe of their operation. The four types of measurements are described in the following four sections. In addition, we intend to consult with design engineers at a major appliance manufacturer to see if there are any insights they can offer us that might lead to signatures which promise to discriminate appliances more effectively.

2.1.1 Oscilloscope Tracings

Oscilloscope tracings of current and voltage at the wall plug of an appliance offer the finest detailed view of the operating characteristics over short time periods. These measurements were taken using a specially designed instrument into which the appliance is plugged (see Fig. 2-1). The current is measured by way of a shunt in the neutral, connected to a differential amplifier, and the voltage is measured by way of a voltage divider. These outputs are input to a storage oscilloscope, which chops between the two waveforms so that synchronous voltage and current waveforms can be recorded. In addition, a relay in the line is connected through a delay circuit so that at the push of a button the oscilloscope sweep begins, and then two (60-Hz) cycles later the relay closes and the appliance turns on. In this way, starting transients can be observed consistently. The device can also be used with the relay closed to observe "steady-state" operation of the appliance.

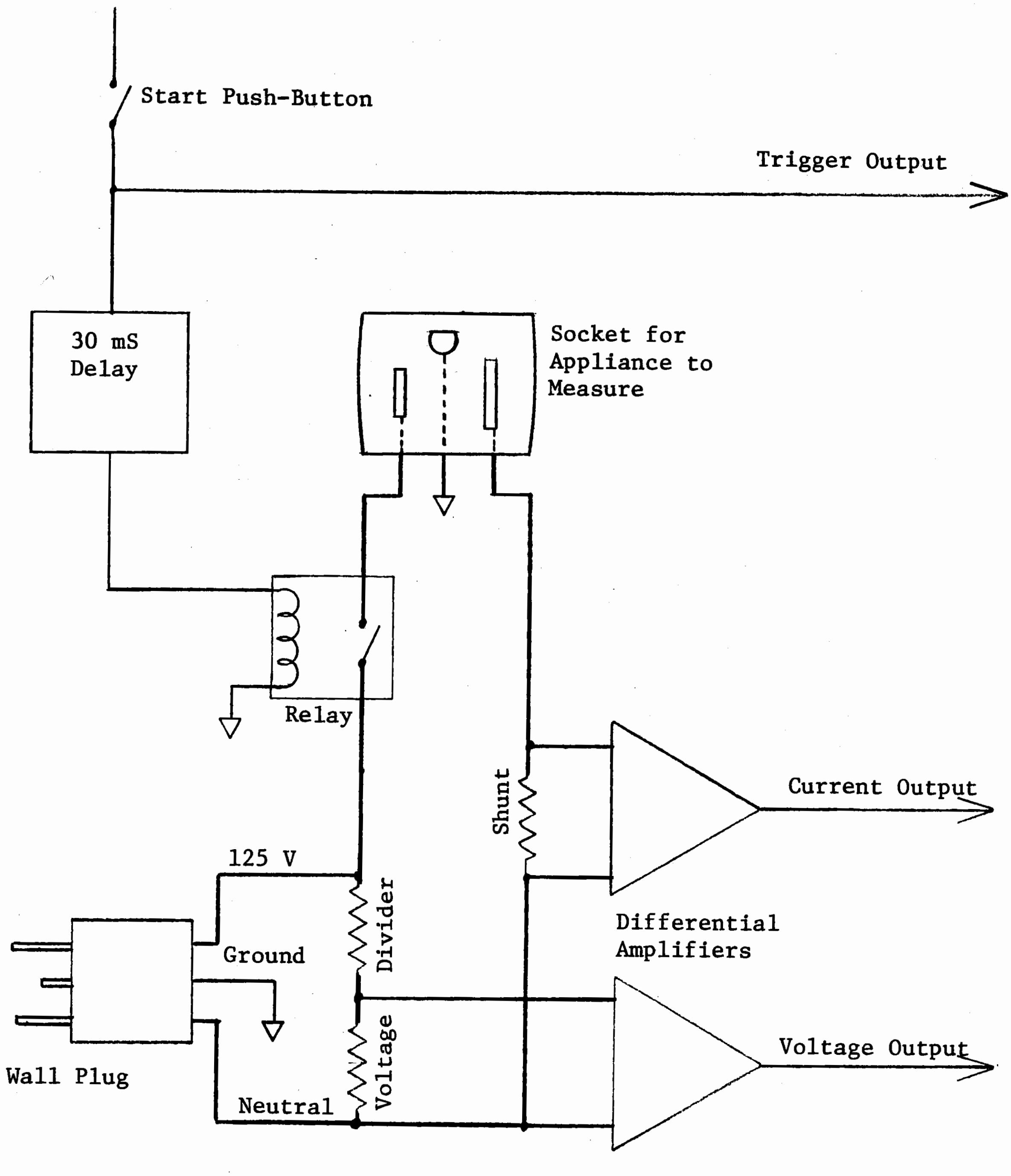
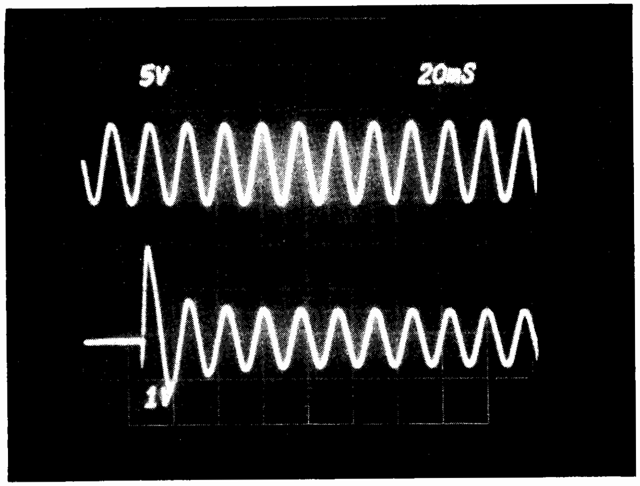


Figure 2-1
Appliance Waveform Instrument

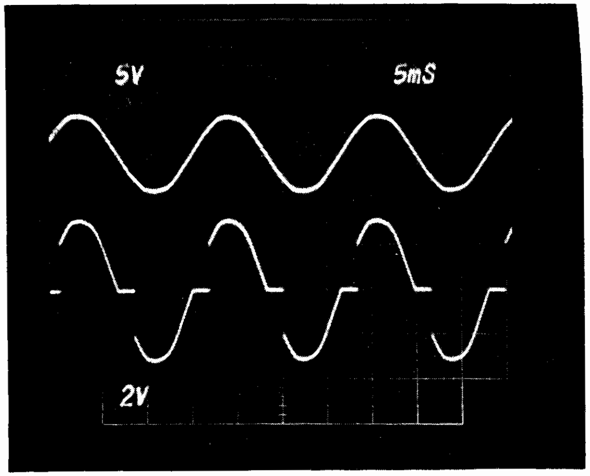
This technique, applied to a household inventory of appliances, revealed a wide range of steady-state waveforms and starting characteristics. Figure 2-2 shows a sampling of waveforms collected in this manner. In each case the voltage waveform is at the top and the current waveform below. The time scale is provided by observing the 60-Hz voltage oscillations. The current vertical scale varies from appliance to appliance.

From these data it was determined that harmonic content of the current waveform was potentially very useful for distinguishing the smaller appliances--those under 200 W or so--which displayed a wide range of waveform shapes. In larger appliances, motors were observed to draw a fairly consistent triangular-shaped current waveform, and resistive appliances predictably showed sinusoidal waveforms. (These observations were later verified and quantified by looking at spectral analysis plots of digitized waveforms collected with the AC Monitor, as described in Section 2.1.3 below.) From this and other data discussed below, it was decided that current harmonics add little information to that obtainable by simpler means when distinguishing large appliances, but that they could become quite useful if we intend to recognize appliances under 200 W. Section 3.1.2 presents a fuller discussion of harmonics as a signature component.

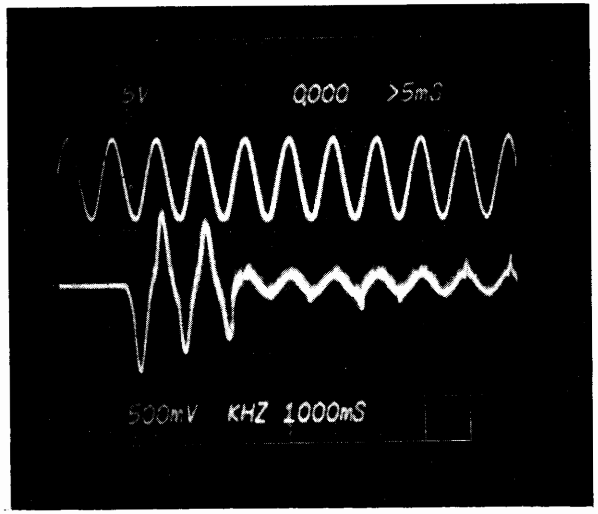
Observation of the starting currents drawn by different appliances showed three different types of behavior. We have loosely termed these <u>circuit-transients</u>, <u>mechanical-transients</u>, and <u>switched-transients</u>, and respectively describe them in the following three paragraphs. Section 3.1.3 considers transients in further detail.



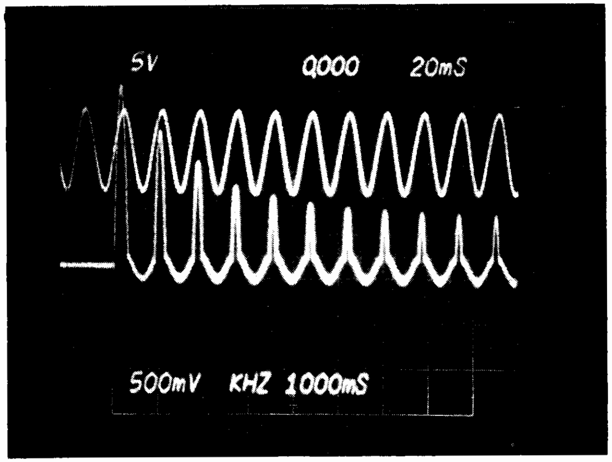
Incandescent Bulb



Light Dimmer

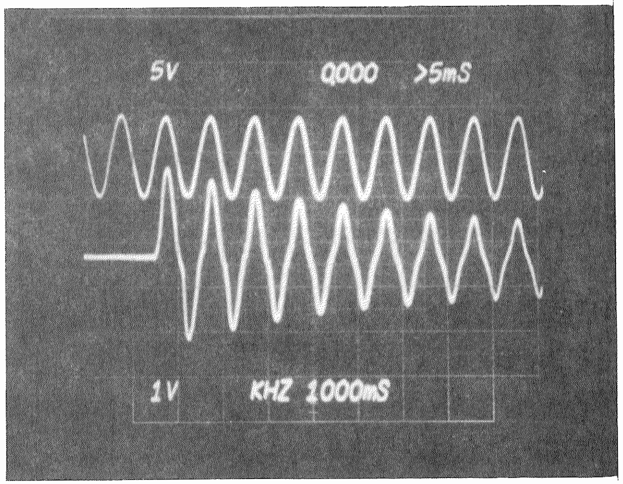


Hand Mixer

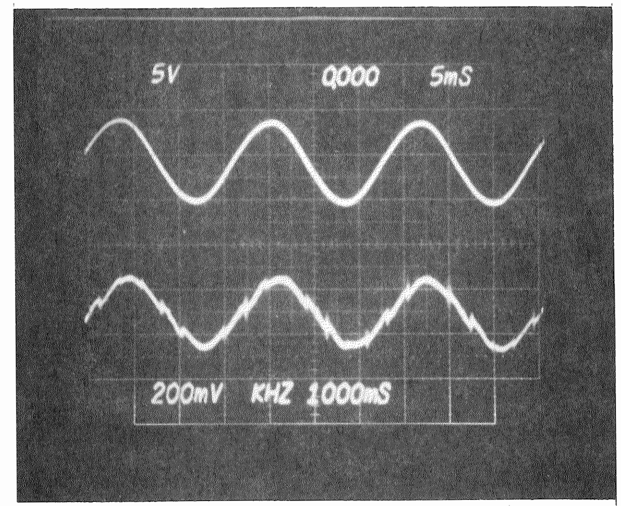


Black & White TV

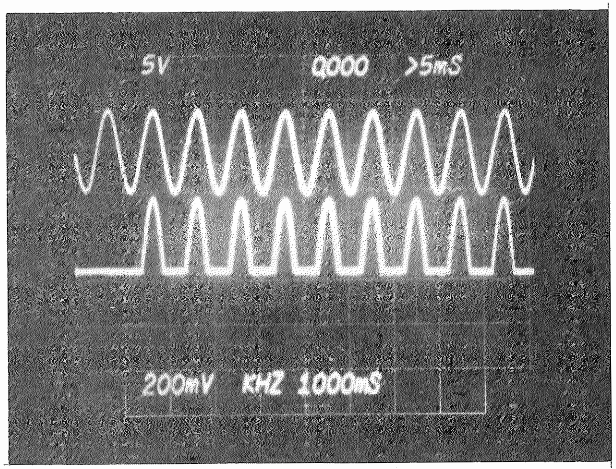
Fig. 2-2. Waveform variety in small appliances.



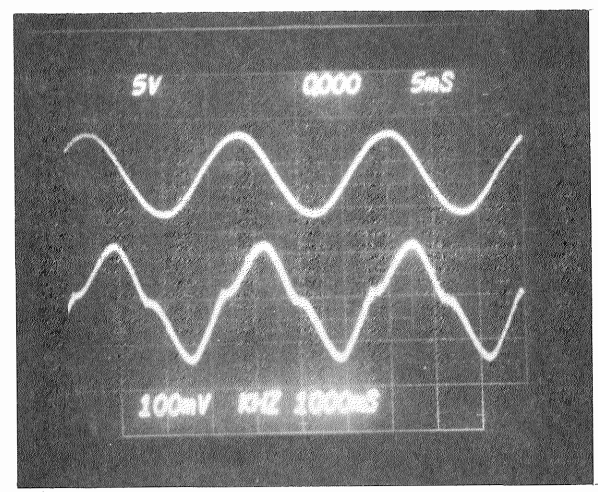
Blender



Sewing Machine



Crock Pot (low setting)



Fluorescent Lamp

Fig. 2-2. Waveform variety in small appliances (continued).

In some small appliances (e.g., small motors or fluorescent lights) a current transient of short duration, lasting at most one or two voltage cycles, was observed. See Fig. 2-3 for a sample of such a circuit-transient. The size of these transients was observed to vary markedly from event to event when the appliance was repeatedly turned on. These presumably are true transients in the circuit-theory sense of the term, and vary according to phase of the voltage at the time the appliance was switched on. The residual flux in electromagnetic devices should also affect this type of transient.

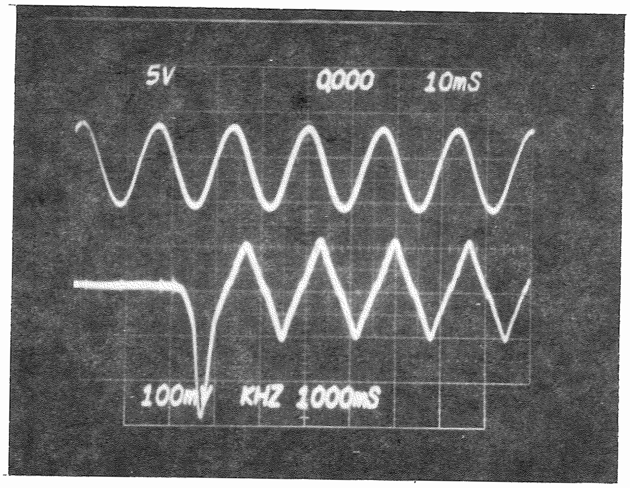


Fig. 2-3. Example circuit transient (fluorescent lamp).

The mechanical-transient type of starting current was observed to last up to several seconds (see Fig. 2-4). This was observed on large motors (e.g., fans, vacuum cleaner) and was quite consistent from measurement to measurement as the appliance was repeatedly turned on. These are presumably transients in the load, the electrical consequence of the mechanical transient of the load getting up to speed. They display an envelope with the character of an exponential decay which approaches a steady running level. As would be expected, the magnitude is reduced if the load is already in motion. This is the case when a fan which is switched on shortly after it was switched off, so that the blades are still moving.

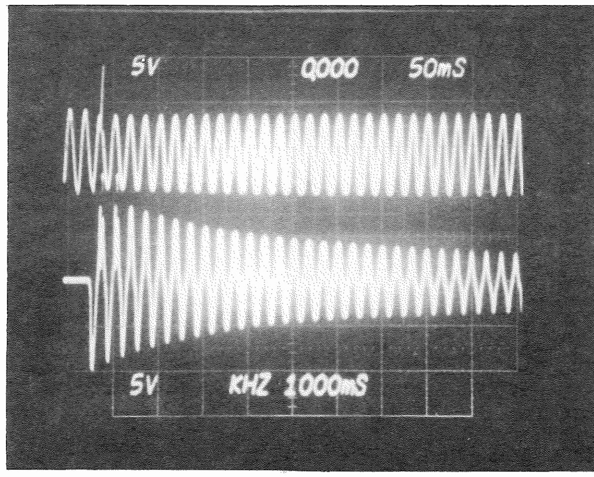


Fig. 2-4. Example mechanical transient (vacuum cleaner).

The <u>switched-transient</u> type of starting current is also observed on large motors (refrigerators) (see Fig. 2-5). In this case the current waveform is of a fairly constant amplitude for a period, and it then drops down in a stepwise manner to a smaller amplitude for continuous operation. This effect is presumably the result of an auxiliary starting coil in the motor which is designed to provide additional starting torque before turning off with a thermal switch.

Starting transients (of all three types) remain a potential signature component. In the case of the switched-transients, the amplitude and duration of the starting current, if constant, could be a useful characteristic. Mechanical transients also seem potentially useful. If the load is always switched on from a state of rest, the amplitude and/or time-constant of the exponential envelope could form signature components. Even if the amplitude varies with different starting conditions, the time-constant may remain fairly consistent, but this has not been verified experimentally. However, the variability and short duration of circuit-transients make them less useful and potentially confusing.

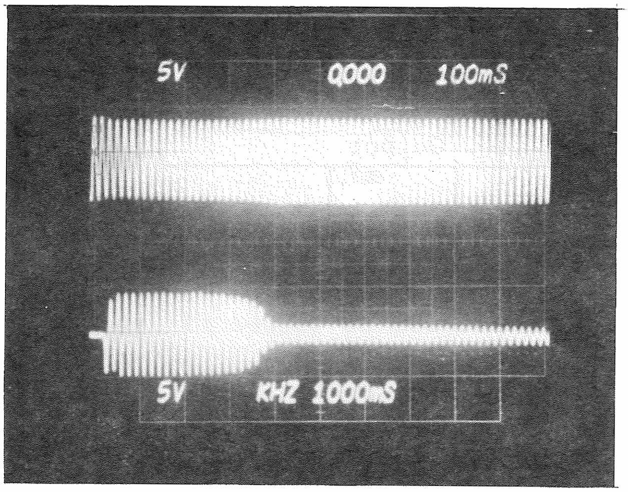


Fig. 2-5. Example switched transient (refrigerator).

2.1.2 Power Transducer Output

For many purposes, the complete waveforms discussed above present too much information. If only the RMS power (real and/or reactive) of an appliance is of interest, then the details of the waveform within each cycle can be ignored. We have taken measurements similar to the above, while looking only at the RMS power drawn by the appliance, as measured with a power transducer (model W106 by American Aerospace Controls). Again, a special instrument was made into which appliances are plugged (see Fig. 2-6). The output of the instrument is connected to a chart recorder so that tracings can be collected.

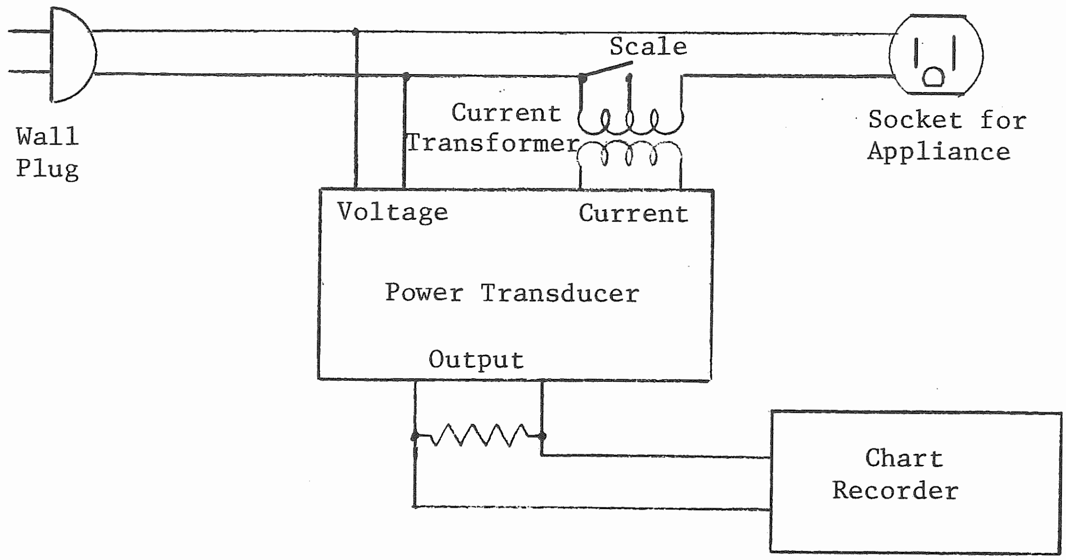


Fig. 2-6. Power transducer instrument.

This apparatus is useful for looking at the long-term behavior of an appliance--for example, the cycling on and off of an automatically controlled device such as the quartz heater shown in Fig. 2-7. Details of the starting currents described above are lost, however. Figure 2-8 compares power- transducer output and oscilloscope tracings for the mechanical-transient of a table fan starting. The responses of these two instruments are considerably different. In the case of the power transducer, its analog filter, which performs the RMS function, introduces a significant distortion over short time periods. Because of this, we conclude that this kind of power transducer would not be a suitable "front-end" sensor for the final load-monitoring device if starting transients of any kind are to be used as a signature component. We have not investigated whether or not other types of power transducers are commercially available with dynamic properties better suited for our purposes, because the Digital AC Monitor described in the next section does have the required response time as well as many other desirable properties.

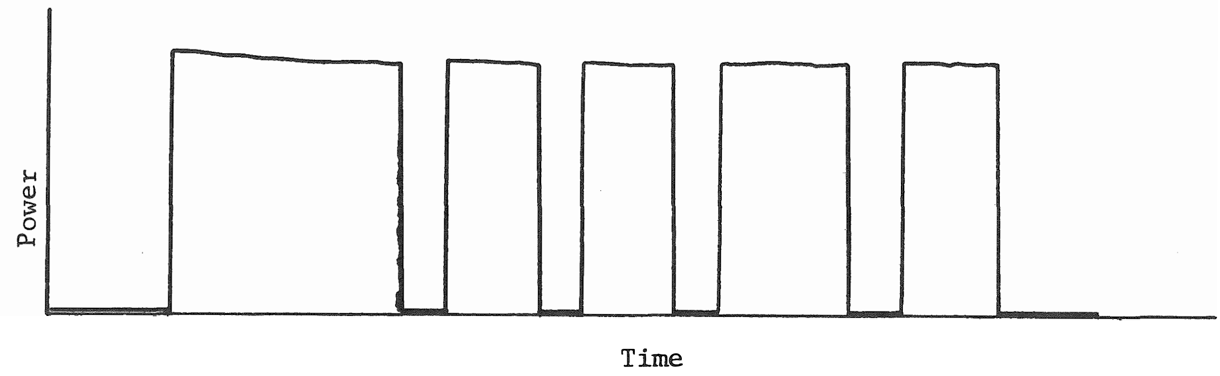


Fig. 2-7. Power transducer output for quartz heater.

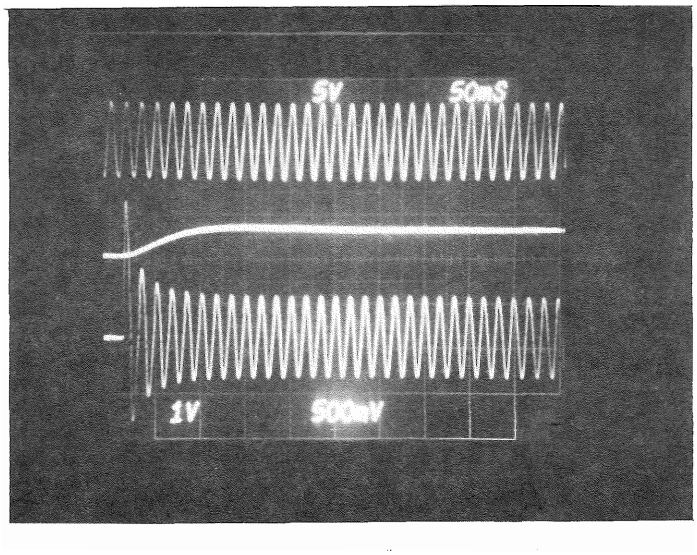


Fig. 2-8. Comparison of power transducer and waveform.

2.1.3 Individual Power and Admittance Measurements (Digital AC Monitor)

In order to measure the RMS power drawn by an appliance, this information has to be extracted from the information available in the complete waveform. Although a conventional power transducer performs this function, as discussed in the previous section, it does not respond quickly enough to step changes in the power. The third instrument which we have used for measuring appliance characteristics is the Digital AC Monitor. This is a general-purpose instrument capable of measuring power and many other properties of ac circuits. It was designed and built at the MIT Energy Laboratory, Northeast Residential Experiment Station, for Department of Energy purposes. Further documentation is included in the Appendix. For the purposes of this section, it is only necessary to know the essentials of its operation.

The AC Monitor is a microprocessor-based instrument which samples current and voltage waveforms, each 64 times per 60-Hz cycle. It then works numerically on the digitized waveforms to calculate a variety of quantities, including real and reactive RMS power, voltage, current, impedance, admittance and total harmonic distortion. The calculated quantities are output digitally, over an RS-232 line, to a terminal or computer programmed to receive the data. The digitized waveforms can also be transmitted for plotting or analysis. The instrument can timeshare between as many as eight circuits if desired. It can be programmed to calculate and output either "instantaneous" values (based on a single 60-Hz cycle) or average values (averaging "instantaneous" values over a specified time period).

Using the waveform instrument of Fig. 2-1, connected to a Digital AC Monitor communicating with a computer terminal, individual appliance measurements were taken and plotted. One of our concerns was to see how power, current and impedance of typical appliances vary as a function of line voltage. Section 3.1.1 discusses the problem this introduces. Line-voltage variations were introduced by connecting the waveform instrument to the output of a Variac. The experimental setup is shown in Fig. 2-9. An example of such testing is given in Fig. 2-10 which shows the measured power, current and admittance of a table fan as a function of line voltage over a 20 V operating range. The solid line indicates the real part of each complex quantity and the dashed line indicates the imaginary component.

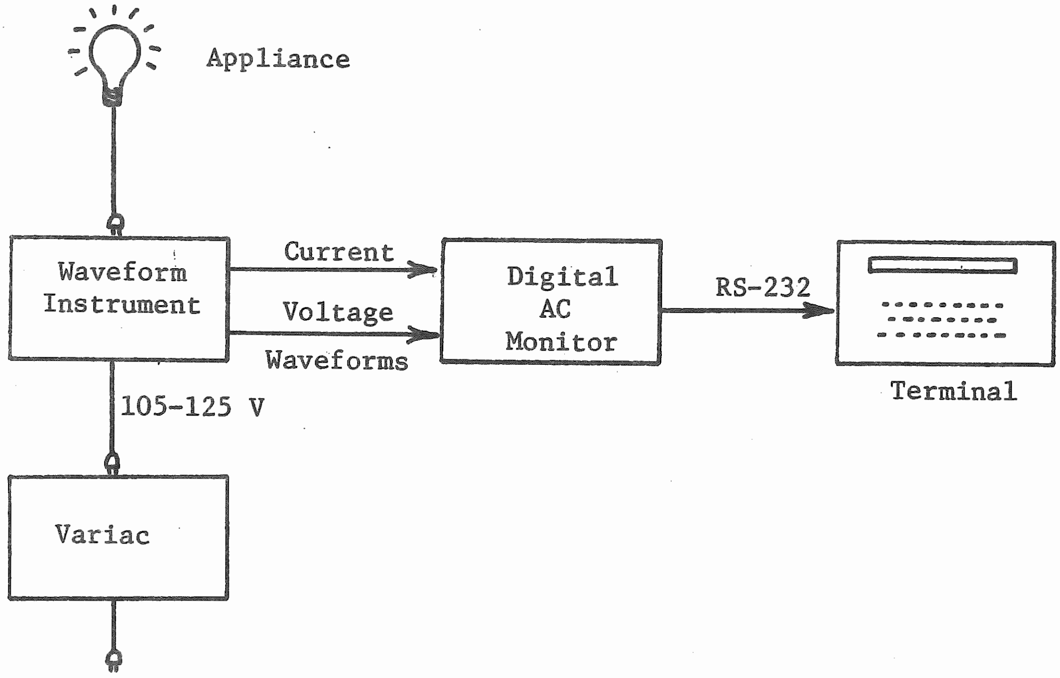


Fig. 2-9. Measuring effect of line voltage change.

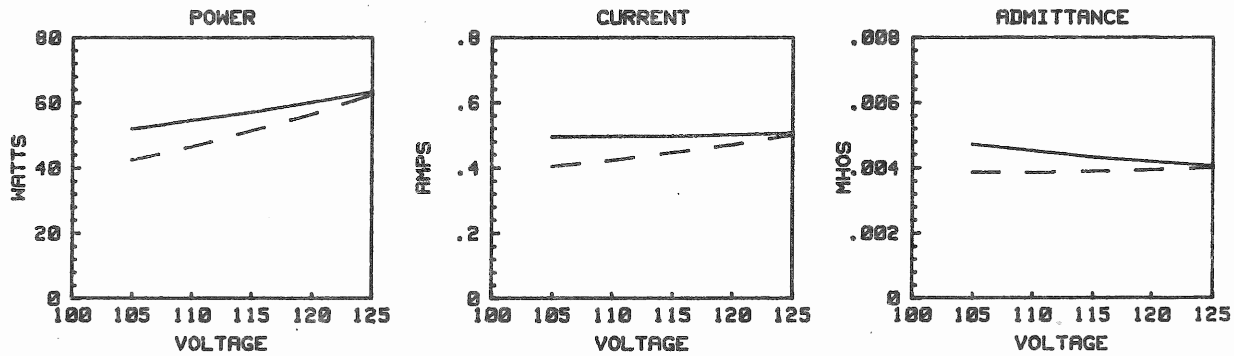


Fig. 2-10. Effect of line voltage on power, current and admittance of table fan.

From data of this nature, it was determined that of power, current and admittance, admittance remains the most constant as a function of line voltage. Accordingly, admittance is to be used as a signature component. This subject is considered in further detail in Section 3.1.1.

2.1.4 In Situ Admittance Measurements

The majority of our measurements of appliance operation have been of appliances in normal use, in place, at the residence of the author. The measurements discussed in this section are all taken by the Digital AC Monitor, monitoring the two out-of-phase 120 V legs on the utility side of the house distribution panel. Figure 2-11 shows the instrumentation arrangement. Current transformers were placed around the two legs so that a small shunt and operational amplifier provides the ± 5 V signal needed for the AC Monitor. Line-voltage signals are provided by voltage dividers which simply plug into wall sockets on the two legs. The AC Monitor communicates by RS-232 at 9600 baud to an HP9845B desktop computer programmed to collect and format the data.

Data and software are transferred to and from the MIT computing resources via tape cartridge. A great deal of effort has gone into the development of software for analysis of this data. Data that has been collected in this manner and which is presented in Section 3 of this report falls into the following categories:

- Real and/or reactive power on one or both legs of the house as a function of time (with one-second resolution).
- Admittance of one or both legs as a function of time.
- DC currents as a function of time.
- Line voltage as a function of time.
- Voltage or current waveforms.
- Spectral analysis of current (Discrete Fourier Transform of current waveforms).

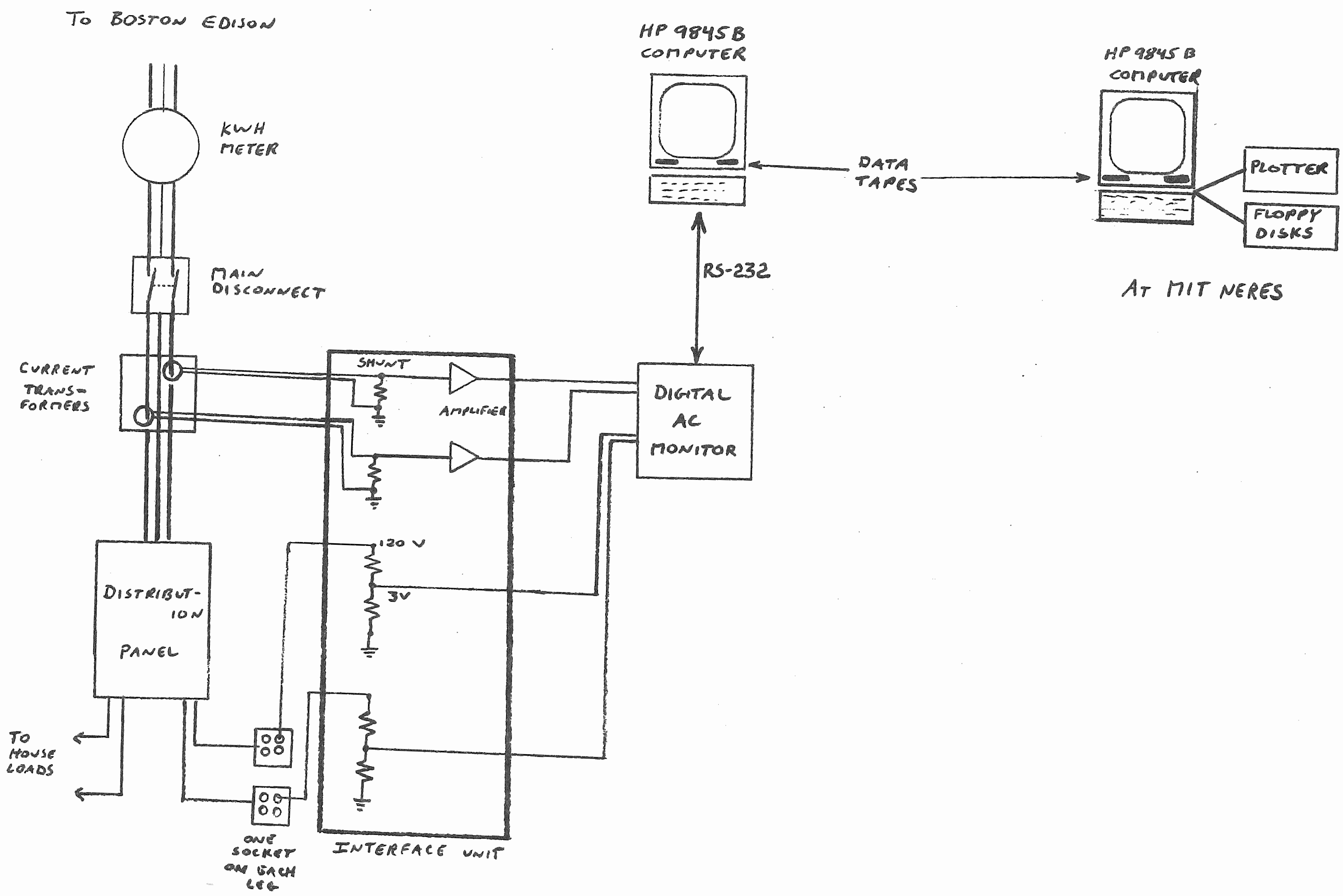


Fig. 2-11. Ad hoc residential data system.

- Scatter-plots of whole-house load transitions (real and reactive power or admittance) resulting from the switching on and off of individual appliances.

This data is used to support a wide variety of conclusions which are taken up in detail in subsequent sections.

2.2 Representing Appliances

After the first subproblem is complete, gaining an understanding of the nature and variety of residential appliances, the second subproblem can be approached. This subproblem is to decide upon a system for modeling appliances. Each appliance model must be capable of representing the information needed to keep track of the appliance's cycling on and off, and to identify the appliance by its common name (e.g., "refrigerator"). It must contain all the relevant signature information and whatever other information is necessary for the application of the signatures. The appliance representation is to be learned (created and modified) as the algorithm determines the appliance inventory. It can then be accessed and applied as the algorithm follows the behavior of the individual appliances.

It is not the models of individual appliances so much as an appliance representation system (ARS) that has to be determined in this subproblem. The ARS is in effect a "language" for describing particular appliances. We will specify this language and the algorithms for its use. The algorithms will then apply it to each particular appliance. The major difficulty in specifying the ARS is to make it powerful enough to describe a wide range of appliances usefully, yet not so complex that it can not be automatically applied by reasonable algorithms.

The abstractness of this introduction may be clarified by Sections 2.2.1 and 2.2.2, which describe examples of ARSs which display two extreme properties. The first example ARS allows appliances to be represented by a region of the signature vector-space. This is fairly straightforward to apply by automatic algorithms, and Section 4 uses this ARS fairly successfully on a limited range of appliances. One shortcoming of this Signature Space Region (SSR) ARS is that it can not represent many types of complex appliances (e.g., dishwashers and washing machines) which contain more than one electrical component and which can not be described as being either simply ON or OFF, but rather have a set of possible states that they can be in at any given time. Other shortcomings of this simple representation are discussed in Sections 3 and 4.

The second ARS presented below allows for a much wider range of appliances. It allows appliances to be described as arbitrary finite state machines. In principle at least, this can describe any conceivable appliance. The difficulty with this ARS is that it is not likely that algorithms can be developed to learn the appliance inventory automatically, because the unconstrained nature of the ARS allows for too many possibilites, many of which would never be instantiated in a real-world appliance.

The probable compromise between these two extremes will allow appliances to be represented by a tightly constrained set of finite state machines, with only a finite set of allowable topologies. An example of such a system is given at the end of Section 5. The methodology by which we hope to arrive at a suitable ARS is to converge on it from these two directions, with the examples of the two following sections as starting points. By considering particular appliances for which the Signature Space Region ARS is inadequate, we can determine the minimal ways in which it must be expanded. To do this we must consider particular models for particular appliances, but this is not the goal in itself. The goal is to arrive at the system which can minimally subsume the particular models.

Approaching from the other direction, we must consider the learning and recognizing algorithms of Sections 2.3 and 2.4 to verify that the ARS can be applied to the residential measurements. If an ARS were too complex to apply, it would have to be simplified for the success of the overall project. The cost of this reduction would be a class of appliances which could not be automatically recognized. This aspect of our research is still at a very exploratory stage and is discussed in Section 5.

2.2.1 Signature Space Regions

The Signature Space Region (SSR) Appliance Representation System (ARS) allows each appliance to be modeled by a region of the signature vector-space. It can be specified in two parts: the individual signature components and the geometry of allowable regions. For example, Fig. 2-12 shows a rectangular region of a two-dimensional signature space which might be used to represent a 100 W incandescent light bulb. The components of the signature space are the independent real and reactive power measurements (on one leg of the house). The region is a rectangle that is 10 W wide and 5 VAR high, centered on the point (100 W, 0 VAR). (For reasons to be discussed later, admittance, though less familiar, is preferable to power for our purposes.)

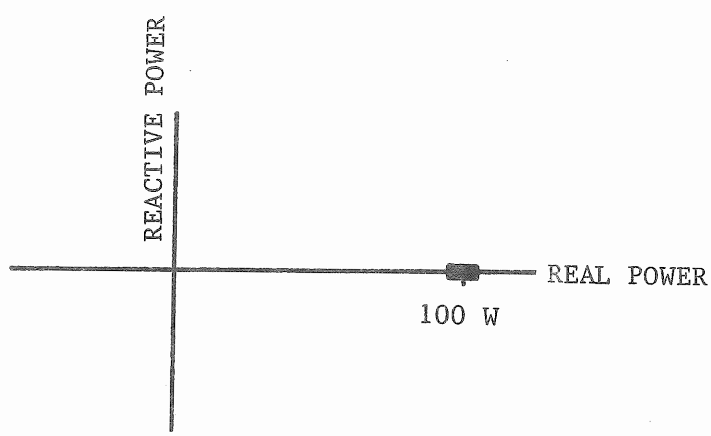


Fig. 2-12. Rectangular region-of-space representation for 100 W incandescent light bulb.

To apply this representation to a monitored residence, the algorithm would monitor the real and reactive power consumed by the house, and if it increased by 100 W (with no change in reactive power), the algorithm would determine that the light bulb turned on. Changes which were not exactly 100 W yet still in the specified region would count as this same bulb. If the power changes by an amount which is the negative of a vector in this region, the algorithm would determine that the bulb turned off. The size of the region allows for noise and other errors in the power measurements.

To learn this representation from monitored residence data, the algorithm could perform a cluster analysis of changes in real and reactive power, note a large number of changes clustered at approximately 100 W, and draw a rectangle to include a large fraction of them. It might then decide that the appliance is a light bulb from time-of-day statistics and the amount of real and reactive power drawn.

Note that this simple picture is merely a sketch. It would have to be expanded considerably in a functional algorithm. Consider for example the complications that arise from the existence of other 100 W appliances, from changes in power due to line-voltage fluctuations, or from the fact that the filament may be part of a three-way bulb in which first one, then the other, then both filaments are switched on. These and other problems are discussed in Sections 3 and 4.

The allowable geometry for these SSRs is worth careful consideration. The optimal set of allowable shapes is probably not the set of rectangles. Although rectangles allow a computationally simple determination of whether or not a given vector is included, they do not have the correct statistical properties with respect to measurement error. If the dimensions of the region are determined primarily by noise in the measurement process, and if the noise has a Gaussian distribution, then ellipsoids are more likely to capture a given percentage of the distribution while minimizing the accidental capture of other appliances. (Note: we are expanding the definition of the term <u>ellipsoid</u> here to refer to generalized ellipsoids in the multi-dimensional signature space, not restricted to only the three-dimensional case.) It is yet to be determined whether to allow ellipsoids with arbitrary axis orientations, or only ellipsoids in which the axes are parallel to the signature space axes. Figures 2-13 and 2-14 display these two possibilities. Requiring parallel axes as in Fig. 2-13 assumes that the measurement errors in the separate components are independent. If the errors are correlated, then the arbitrary ellipsoid regions of Fig. 2-14 would be desirable. This would require some additional storage and computation time, but could increase the overall accuracy of the device. Initial investigation of real and reactive power measurements as described above in section 2.1.4 shows little correlation, but the signature space components must be specified, and a wider selection of appliances needs to be examined before the geometry can be specified.

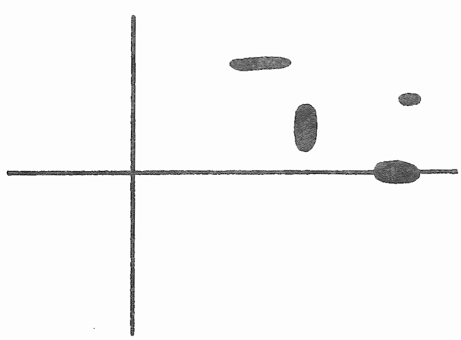


Fig. 2-13. Ellipsoids with axes parallel to signature axes.

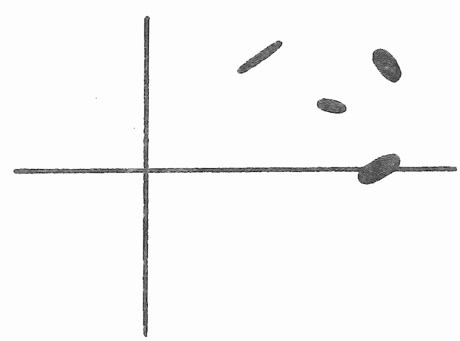


Fig. 2-14. Arbitrary ellipsoids.

2.2.2 Finite State Machines

The finite state machine (FSM) appliance representation system (ARS) allows for far more complexity in appliance specifications than the SSR ARS. The FSM ARS is effectively a language in which one can describe a set of (mutually exclusive) states that the appliance can be in, and the signatures that are observed when the appliance changes from one state to another. For the purposes of this report, a FSM is defined to be a diagram, such as Fig. 2-15. The diagram consists of a set of circles (the states) and interconnecting arrows (the transitions). The circles are each labeled with the name of the state. These state names are for identification purposes only and are not essential to the FSM. The transitions are labeled with the signature vector that is observed when the machine changes from the state at the tail of the arrow to the state at the tip of the arrow. Two such diagrams which are topologically equivalent (including identical transition signatures) are considered to be the same FSM.

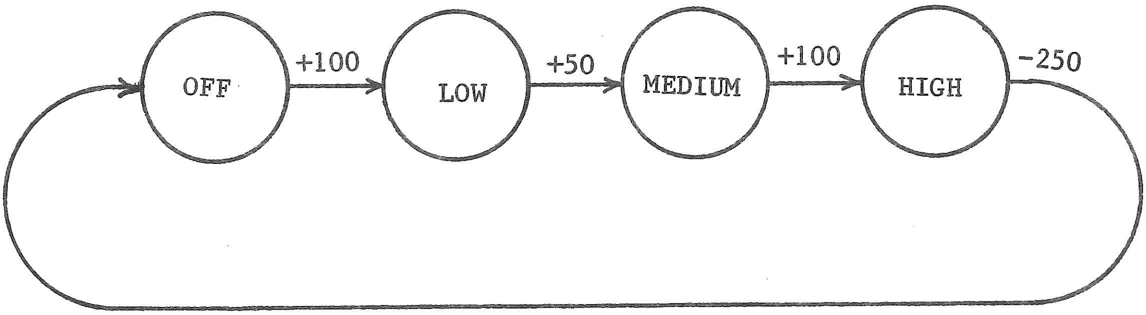


Fig. 2-15. FSM for 100-150-250 W three-way lamp.

The example depicted in Fig. 2-15 represents a "three-way" lamp. There are four states and four transitions. The four states, OFF, LOW, MEDIUM, and HI, can be traversed in a cyclic fashion in that sequence only. The transitions are labeled assuming that the bulb is of the 100-150-250-W variety. For clarity, only a single real-power value is indicated for each transition; in practice, this would be only one component of a higher dimensional signature vector. Note that the transitions indicate only the change that is observed when the lamp changes state, not the total power consumed by the appliance. For example, when the bulb is switched from LOW to MEDIUM, the 100 W filament turns off and the 150 W filament turns on. What is observed, however, is a net increase of 50 W, so this is the signature associated with that transition.

A more complex FSM is used to describe an appliance which can change from any of four states to any other, rather than only in the cyclic fashion of the three-way lamp. Consider for example a three-speed fan controlled by push-buttons, so that it can change between states arbitrarily. It would still have four states, but there would be more transitions than in the cyclically constrained lamp. Figure 2-16 is the FSM for such a fan, assuming the power levels are 100, 200 and 300 W. Again, only the real-power component of the signature vectors is used in labeling the transitions. Reactive power and other signature components, although required in the complete FSM, are omitted for clarity.

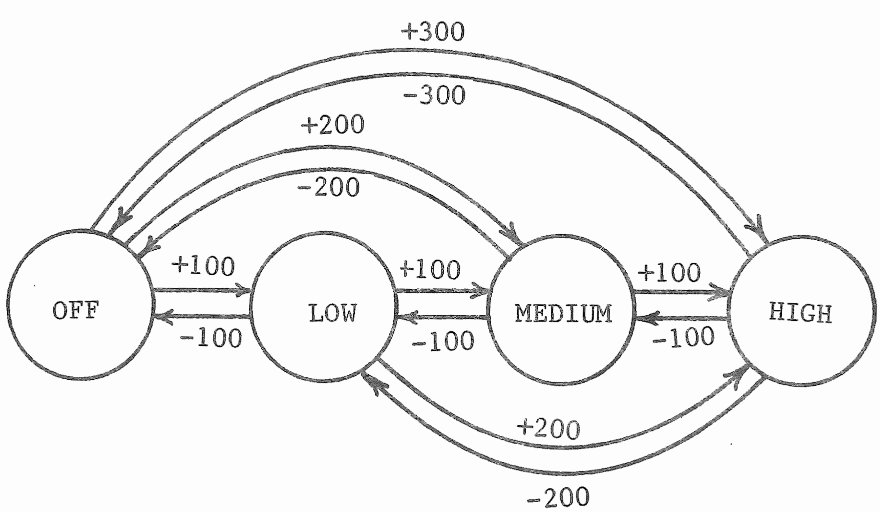


Fig. 2-16. FSM for 100-200-300 W push-button controlled fan.

Although the transitions of these FSMs are labeled with signature vectors, it is more likely that SSRs are the appropriate labels. The same options exist here as with the SSR ARS: the geometry of the region could be constrained to be rectangles, parallel-axis ellipsoids, or arbitrary ellipsoids, but which of these we have not yet determined. The dimensions of the regions could vary from appliance to appliance and from transition to transition within an appliance.

The SSR ARS of Section 2.2.1 can be viewed as a special case of the FSM ARS. In the SSR ARS, only two states are considered, OFF and ON, and only two transitions, which must be the negatives of each other. Thus the FSM ARS reduces to the SSR ARS if FSMs are required to fit the template shown in Fig. 2-17. In this figure, X represents a SSR and -X its negative. Constraining FSMs to only those of this shape is an extreme restriction of the power of the FSM model, and accordingly, many appliances would not be properly represented. However, the full FSM ARS as presented in this section is insufficiently constrained to be useful. The goal of the appliance representation subproblem is to find an appropriate set of constraints, as discussed in Section 5.

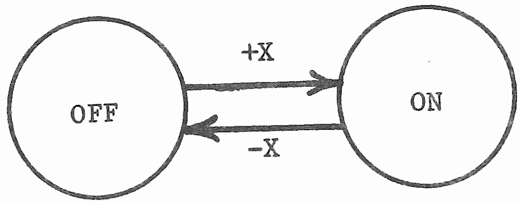


Fig. 2-17. FSM equivalent to signature space region.

2.3 Recognition Problem

The third major subproblem is to develop an algorithm which, if given the representations of the appliances in the residence, can determine individual appliance activity. It must be capable of recognizing the individual appliance transitions from the measurements available of the entire residence. Although presented separately here, this subproblem and the learning problem described in Section 2.4 are actually deeply intertwined. The learning problem is to develop algorithms which can learn the appliance representations for the individual appliances; the recognition problem is to develop algorithms which use those representations to track state changes. The set of appliance representations is the data structure through which the two algorithms communicate.

We begin by introducing some terminology while sketching the computational scenario that seems likeliest at this time. If developed successfully, the physical load data-acquisition device will be installed in the line between the utility's pole transformer and the residence's distribution panel. For convenience of installation, this will probably be at the kilowatt-hour meter socket. Sensors at this point will provide a data stream of <u>primary residence variables</u> as a function of time. The exact nature of these variables depends on the sensors selected (see Section 3.2) and might be voltage and current, for example. From this data the <u>residence transitions</u> will be calculated. The residence transitions are a stream of signature vectors as calculated for the entire residence. For example, it is likely that this will be a vector which includes conductance and susceptance on each of the two legs as four of its components. Many of the residence transitions will simply be appliance signatures. Some residence transitions will be the "sum" of two or more appliance signatures, created when more than one appliance changes state at the same time. Other residence transitions might be the result of noise or other errors. It is the job of the recognition algorithm to determine which residence transitions fall into which of these classes. When the residence transition is determined either to be an appliance signature or able to be broken down into several simultaneous appliance signatures, the recognition algorithm must update a residence state vector which keeps track of the current state of each appliance in the residence.

The residence as a whole is modeled as an n-tuple of (constrained) finite state machines, the residence model, containing one entry for each appliance that has been identified. The residence state vector is an n-tuple of FSM states containing one entry for each appliance FSM in the residence model. The nth entry of the residence state vector is the state of the nth appliance of the residence model. If the residence state vector can be determined accurately as a function of time, then any conceivable appliance usage statistics can be calculated straightforwardly as a small addition to the program.

The recognition problem deserves immediate attention for three reasons. First, it is far easier than the learning problem. If the recognition problem can not be readily approached, then the entire project surely will not succeed. We would want to find this out as soon as possible. Second, it provides a fallback position in the event that the learning problem proves to be intractible. If an entirely automatic algorithm is too difficult, we might want to lower our goals to a recognizing device only. This would involve some human intervention in the house to be monitored--perhaps running each appliance one at a time while identifying it by name to the device--so a simple algorithm could be taught the different appliance models. The device could continue automatically after that, using the recognizing algorithm and gathering statistics. Although this would involve some intrusion into the residence, it would not involve hardware installation or removal internal to the house. The third and most important reason for addressing the recognition problem at this point is that it provides a check on our progress in the first two subproblems. By having a computer try to identify appliance activity from a specified residence model, we can see how successful we have been in our understanding and modeling of appliances and in our selection of signatures.

We have created and operated a simple, yet reasonably successful, recognition program. The algorithm uses the SSR ARS to model seventeen appliances. Over periods of several days, the algorithm correctly recognizes over 90% of the major (over 200 W) residence transitions of an entire house under typical conditions. Section 4 describes this program and its accuracy in further detail. A similar program which uses the FSM ARS would be only slightly more complex. Although the program of Section 4 is valuable as a demonstration, it should not be taken as a model of the final recognition algorithm. We expect that the final algorithm will be considerably more sophisticated and quite different in structure. The recognition and learning processes will be integrated in a way that bears little resemblance to the algorithm presented here.

2.4 Learning Problem

The fourth major subproblem is to develop an algorithm capable of building the residence model from the observed residence transitions. This algorithm can be viewed as a subroutine which is called by the recognition program every time a transition occurs which can not be interpreted in terms of the residence model and residence state vector. It would ascertain why the transition was uninterpretable and update the residence model and state vector accordingly.

The learning problem is by far the most difficult portion of the entire project. So far we have not approached this problem except to note that a statistical cluster-analysis technique, as briefly mentioned above in Section 2.2.1, can be used to identify SSRs which contain large numbers of transitions. To go beyond this and create the states and transition linkages of the appropriate FSMs is a difficult challenge, but one which we believe is practical, given a sufficiently constrained set of FSMs in the ARS.

Because we have not seriously approached this problem, there is little to report here except to sketch some approaches. Two general strategies for the learning problem present themselves: (1) to build up from "two-state" appliance models, or (2) to break down a complex residence model. A combination of these two approaches might also be viable. These strategies are described in the following two paragraphs.

In the "build-up" approach, the algorithm would begin by assuming that all appliances are of the two-state ON/OFF type shown in Fig. 2-17. The cluster-analysis technique would indicate which transitions occur frequently and consistently enough to warrant inclusion. With this as a beginning, the algorithm would begin to correct the residence model, improving or combining FSMs as necessary until it converged upon a satisfactory residence model. We stress that the details of the method by which this is to be accomplished are not yet clear to us.

In the "break-down" approach, the algorithm would begin by building a single FSM that models the entire residence. This FSM would then be separated into independent appliance models by a mathematical factoring process that is sketched in Section 5. The single FSM which represents an entire house would be very complex. Consider, for example, that if there were sixteen independent two-state appliances the FSM would have the structure of a sixteen-dimensional hypercube which has 65536 nodes and over a million transitions. Although this approach seems more difficult at first glance, it is too soon to eliminate it from consideration. (Note for example that the real and reactive power four-vector for the entire house is an available clue as to whether or not two observed states belong to the same residence FSM state.)

2.5 Identification Problem

The identification problem is to determine the common names of the household appliances from the learned representations and the observed usage statistics. Given the enormous range of available appliances, it is not clear how well this task can be automated. We expect that it will involve a multidimensional feature space which will include signature components and observed usage components. For example, refrigerators as a class might be represented as a region of the feature space where the greatest operating power level is between 100 and 500 W, the power factor is inductive, one leg rather than two is used, the duty cycle is between ten minutes and an hour, there is little correlation with time of day, and slight correlation with temperature. These would seem likely parameters but the exact set of features to incorporate in the feature space remains to be determined. It is not clear at this point exactly how the range of topologies allowed by a FSM ARS would be incorporated in such a feature space. Perhaps this will involve the design of an "expert system" rather than a simple feature lookup.

After specifying the feature space, the difficult part of this task will be to obtain enough data from particular appliances to feel secure that the existing range of any particular appliance class has been covered. An algorithm to check the observed characteristics of an appliance against a list of appliance class regions is straightforward.

The probable locus of the identification algorithm is _not_ in the appliance load data-acquisition device. The scenario we expect is that the individual appliance-recognition devices will transfer their residence model and appliance usage statistics by telephone line or transportable media to a central computer facility at regular intervals. A mainframe computer at that facility is the logical place to run the identification algorithm for three reasons. First, the algorithm requires a large database of appliance classes which need not be repeated in every appliance load data-acquistion device. Second, the database will evolve over time because the full range of existing appliances will only be encountered slowly and because new appliances are constantly introduced into the marketplace. Consistency would be promoted by maintaining the database at a single central facility. Finally, we expect that the identification process can not be fully automated, so it would be best to attempt it at a location expect that it will involve a multidimensional feature space which will include signature components and observed usage components. For example, refrigerators as a class might be represented as a region of the feature space where the greatest operating power level is between 100 and 500 W, the power factor is inductive, one leg rather than two is used, the duty cycle is between ten minutes and an hour, there is little correlation with time of day, and slight correlation with temperature. These would seem likely parameters but the exact set of features to incorporate in the feature space remains to be determined. It is not clear at this point exactly how the range of topologies allowed by a FSM ARS would be incorporated in such a feature space. Perhaps this will involve the design of an "expert system" rather than a simple feature lookup.

After specifying the feature space, the difficult part of this task will be to obtain enough data from particular appliances to feel secure that the existing range of any particular appliance class has been covered. An algorithm to check the observed characteristics of an appliance against a list of appliance class regions is straightforward.

The probable locus of the identification algorithm is <u>not</u> in the appliance load data-acquisition device. The scenario we expect is that the individual appliance-recognition devices will transfer their residence model and appliance usage statistics by telephone line or transportable media to a central computer facility at regular intervals. A mainframe computer at that facility is the logical place to run the identification algorithm for three reasons. First, the algorithm requires a large database of appliance classes which need not be repeated in every appliance load data-acquistion device. Second, the database will evolve over time because the full range of existing appliances will only be encountered slowly and because new appliances are constantly introduced into the marketplace. Consistency would be promoted by maintaining the database at a single central facility. Finally, we expect that the identification process can not be fully automated, so it would be best to attempt it at a location where human experts can intervene as required.

2.6 Testing the Algorithm

The testing of any proposed algorithms can take place in three ways. The first is that the algorithm can be tested in real time in an operating residence in the way that the recognition algorithm of Section 4 has been tested in the home of the author. This allows a direct comparison between the output of the program and what the author knows to be the case in the house. It should be very useful for "shaking down" the algorithm and finding major difficulties that it can not handle. For more refined testing and optimization of parameters, a more controlled test procedure is necessary.

The second and third tests involve a controlled test procedure involving either real or simulated data streams. The method which uses real data requires that monitored house data be collected during a period in which the occupants keep a log of appliance usage. The log could be generated automatically if we use a house with either the Electric ARM or New England Electric System MATREC System already installed. We would feed the measured residence transition data repeatedly to the algorithm, "tweaking" its parameters until optimal correspondence with the log is attained. To use simulated data, real data would be collected, and individual appliance events would be extracted from it manually. These events would then be fed to a synthetic load-data generator which would put together appliance combinations according to the tester's command. The simulated method has the advantage that appliance combinations and coincidences which did not occur in any observed data stream can be created and tested. The real-data method has the advantage that appliance combinations and coincidences which the testers did not think of are likely to occur. We expect that all three types of algorithm testing will be used.

Testing should be performed in two stages. First the recognition, learning, and identification algorithms should be tested separately, then the combined algorithm should be tested as a unit. The separate testing is easier and more likely to point to those parts of the algorithms which need improvement. Because of the deep intertwining of the recognition and learning processes, joint testing is necessary to attain confidence in the overall method.

Testing of the recognition algorithm, given a correct residence model, is straightforward. Statistics of the percentage of transitions correctly identified, or the percentage of energy use correctly apportioned to the individual appliances, can easily be tabulated. (This information is given in Section 4 for the simple recognition procedure presented there.) Quantitative evaluation of the learning algorithm is more difficult because its output is a list of FSMs.

An important point to note about the learning algorithm is that it can not learn everything about the residence all at once. It will have to update the residence model continually over time as it observes new appliances and new aspects of appliances that are already partially modeled. As a consequence, one test of success for the learning algorithm is that it be able to converge upon the correct residence model if it is given the residence transition stream for a sufficiently long period. To be self-correcting, we would like to require further that it should be able to converge upon the correct residence model from any incorrect residence model which it may have erroneously created. This is a difficult test for us to administer and a difficult test for the algorithm to pass. The most difficult part is verifying that this can be done in a reasonable time period. It is often possible to demonstrate that an algorithm will eventually converge upon a satisfactory solution, yet not know how long that will take. Given the intended application of the algorithm, it would be unsatisfactory if the algorithm took more than one or two months of data to converge upon the residence model. (An interesting possibility to consider is that the convergence might be hastened by an initial assumption about the residence model before any actual data is examined.)

If the algorithm could learn a reasonable residence model in less than one or two months, we would consider it successful. This raises a serious problem in the task of algorithm verification. In order to show that the learning aspects of the overall algorithm are operational, a very long time-stream of real residence transitions may be required. This may be difficult to manage with the storage facilities available on the HP9845B desktop scientific computers which we have so far been using. If a problem arises in this area, there are two possible solutions. The first is that we could transfer the algorithm to a mainframe computer for testing purposes. The second possibility is that we could settle for testing the learning aspects of the algorithm exclusively with simulated data, created and then discarded, as the algorithm requires it. Our hope is that this long a time period not be necessary except for appliances that are used only seasonally.

When an algorithm is developed, we will begin testing it by the first procedure described, in situ testing in real time, to the maximum extent possible. The detailed design of the load-data simulator and further testing procedures will wait until after the signature vector components have been determined. We expect that a transition percentage system will be used in the final test procedure as a test of success. We will allow the overall algorithm to operate on a real or simulated data stream, giving it time to build a residence model. After the residence model has been determined (to within a specified tolerance), the time required to learn it will be noted and the algorithm will proceed. We will keep a record of its action for each <u>major transition</u>, where a major transition is defined to be one in which the real power components sum to over 200 W. We will compare the algorithm's assessment of every major transition with the logged appliance activity (if real data is used) or with the command to the load-data simulator (if simulated data is used). A numerical accuracy score will be determined by taking the percentage of major transitions which are identified correctly and subtracting the percentage which are identified incorrectly. Transitions which the algorithm gives up on and ignores will count as zero. Our goal is that the accuracy of the overall algorithm as computed in this way will be over 90%.

The 200-W threshold for major transitions has not been selected for any principled reason. It is merely our assessment of the region below which the algorithm is likely to perform poorly. Our intention is to keep separate statistics on transitions binned by power level (e.g., 0-100, 100-200, 200-300 W, etc.). From this data we can not only calculate the major transition percentage score as defined above, but also see in detail where the algorithm is failing.

2.7 Implementation

The final implementation of the appliance load data acquisition algorithm into a physical device is not part of this project. We do need to keep it in mind, however, as the goal towards which the project is aiming. Accordingly, it is necessary that we consider the computational needs of the microprocessor-based realization of the final device.

Without developed algorithms in hand, this can of course only be an estimate. The only benchmark we have to extrapolate from is the simple recognition program detailed in Section 4. This program currently runs in real-time on an HP9845B desktop computer. The computer CPU contains two HP proprietary eight-bit microprocessors. The program is written in an intrepreted BASIC, and fits, along with all necessary data structures, in less than half of the 187 K bytes of memory available.

If we assume that the algorithm in the physical device will be written in assembly language or compiled by an efficient compiler, and if we assume that the target hardware will contain a 32-bit microprocessor such as the MC68000, then we can conservatively estimate at least a factor of ten improvement in speed, and realistically much more. Benchmark execution-time data provided by Hewlett Packard suggests a factor between ten and one hundred. We further assume that the ultimate learning and recognition algorithms will require no more than ten times as much processing as the recognition algorithms of Section 4. A factor of ten increase for learning is conservative because the device will spend far more time recognizing than learning. The learning algorithm only needs to be called when an unrecognized transition is detected. We conclude then that a state-of-the-art microprocessor such as the MC68000 should be sufficient to serve as the CPU in the final implementation of the appliance load data-acquisition device.

If necessary, the processing could be split between two micro-processors. One could act as a front-end to monitor the sensors and calculate the residence transitions. It would transfer the signature to the main processor whenever a residence transition was detected. The main processor would run the recognition and learning algorithms. This is analogous to the division between the 8085-based Digital AC Monitor and the HP9845 described in Section 2.1.4.

3.0 DETAILED PROBLEMS

There are many detailed problems to be addressed in the development of the different aspects of the appliance load data acquisition algorithm. This section describes a number of these problems and offers solutions that seem appropriate at this point in our research. These detailed problems each play a role in several of the subproblems described above in Section 2. The first six subsections are concerned primarily with characteristics of signatures. The remaining subsections are concerned primarily with appliance representations.

3.1 Nature of Signature

The selection of the signature space components is the single most crucial aspect of the entire development effort. If the selected features do not separate similar appliances, then they are insufficient for our purposes. If they are too difficult to compute, then the resulting algorithm will not be able to operate in real time. If the selected features occur inconsistently, then an appliance may appear to be two or more separate appliances. Our plan is to approach this selection warily. For our first pass through the subproblems of Section 2, we will use only admittance four-vectors as signatures. After making some initial progress towards a learning algorithm, we will return to reassess the signature selection problem.

Table 3-1 presents a taxonomy of the signature components we have considered. The primary dichotomy of the breakdown between steady-state and transient characteristics leads to two orthogonal sub-spaces of the signature vector space which play an important role in certain algorithms. Components from the steady-state half of the taxonomy generate what we call the steady-state subspace of the signature vector space. These components relate to characteristics which are relevant to the entire period during which an appliance is ON (or in any given particular state) and therefore can be used in ways in which the transient components, which only relate to transition periods, can not.

Table 3-1 Signature Component Taxonomy

Signature	Steady-state	60 Hz	Admittance Current Power
		Harmonic	"
		DC	"
		Subharmonic	"
	Transient	Type	Circuit Mechanical Switched
		Amplitude	Peak Integrated DC
		Temporal	Duration Time Constant

The steady-state components of the taxonomy can be further classified according to frequency, relative to the 60-Hz voltage fundamental. The top group of the table--power, current and admittance at the fundamental--includes the most consistent characteristics we have observed. They contain roughly the same information but differ successively by a factor of the line voltage. The choice between the three is considered in Section 3.1.1. Admittance has been selected for our first pass through the algorithm-development process because it is most consistent with line-voltage variations under most circumstances. The same three options exist at higher frequencies, at lower frequencies and at the dc level. Harmonics are discussed in Section 3.1.2. Subharmonics are not applicable directly because they are misinterpreted by the method presented in Section 3.3 for detecting transitions. This effect, in the case of the subharmonics of washing machine agitation, is discussed in Section 3.4. Steady-state dc currents have only been observed for very small appliances such as the crock pot of Fig. 2-2, set on LOW. (We presume this is the effect of a diode placed in series with the same heating element used for the HIGH setting.) Steady-state harmonic and dc characteristics remain potential signature components for a second pass through the development process, but they only seem to add new information over fundamental characteristics if appliances under 200 W are considered.

The transient half of Table 3-1 is broken down according to type, characteristics dealing with the amplitude, and characteristics dealing with the duration. Section 3.1.3 discusses these properties of the transients. Because of the general nonrepeatability of transients, because they are less useful than steady-state properties for some algorithms, and because they lack the linearity discussed in the next paragraph, our initial transient subspace will be null. We may reconsider transients at a later time if the steady-state subspace proves insufficient for the overall algorithm.

Signature components, to be useful, must obey certain linearity laws. The primary constraint is that the signature of a given appliance transition must be independent of the state of the remaining appliances. For example, if a 375 W increase in power consumption occurs when the refrigerator turns ON, then we would like that always to be 375 W, whether everything else in the house happened to be ON or OFF. Fortunately, this is the case as long as the primary sensors are linear. The second linearity constraint we would like to see is to have the residence signature observed when two appliances change state simultaneously be the sum of the two separate signatures observed if the appliances were to change states at different times. This is only partially the case. The signature components in the steady-state subspace behave linearly in this way. The transient components might happen to, but in general do not. This makes the steady-state components especially valuable for the program of Section 4 when breaking down simultaneous transitions.

Note that if only steady-state components are to be used, then the FSM ARS can be reorganized somewhat to take advantage of the fact that each state is associated with an operating level. For example, each state can be associated with the power which the appliance draws while in the state. The transition signatures (power-level changes) then become redundant as they can be calculated by subtracting the connecting state power levels.

Any (invertible) linear transformation of the signature space will result in a space equally suitable for the purposes of the learning and recognition algorithms. For example, if real and reactive power are used as two of the components, and a sensor problem results in misscaled and phase-shifted measurements, the algorithm is not adversely affected. This is because the transition signatures are merely moved about in the space; this affects the learning and recognizing algorithms equally. On the other hand, other pairs of measurements which convey the same information, such as volt-amperage and power factor, even if measured with complete accuracy, are unsuitable for the algorithm because they lack linearity. It is worth mentioning in this context that each of the steady-state signatures listed in Table 3-1 can be viewed either as a single complex number (phasor) or as a pair of real components. The choice between a vector space over the real numbers or one with half as many components over the complex number field is immaterial from a mathematical point of view. We select the option with pairs of real numbers because it is closer to the computational representation.

One final note is that because of the lack of linearity in the transient subspace it may be preferable, from a certain point of view, to think of the entire signature space as the cross product of the steady-state vector space with a transient feature space. Only the steady-state subspace displays vectoriality. The transient subspace is merely a collection of eclectic characteristics. Transient type, for example, can only take on the three values presented in Section 2.1.1.

3.1.1 Power, Current or Admittance: Line-Voltage Effects

For use as a signature, an appliance transition can be characterized by the accompanying change in either power, current or admittance (more than one would be redundant). All three measures share the required linearity properties. They differ, successively, by a factor V, the utility line voltage. If V were constant, they would contain exactly the same information, and any of the three could be selected. Given the vicissitudes of V, however, the three must be examined to see which is the most consistent signature. We wish the signature to remain constant if the line voltgae varies from 115 to 125 V. Three separate arguments lead to admittance as the correct choice.

From the theoretical point of view, appliances can be modeled as two-terminal linear passive networks, i.e., "RLC" circuits. Such a circuit is completely characterized by a single complex constant, admittance, independent of the applied voltage. (The real and imaginary components of the admittance are the conductance and susceptance.) In contrast to the constancy of admittance, current will vary in proportion to V, and power will vary in proportion to V squared. This argues ineluctably for admittance if we can accept the RLC modeling. The complication, of course, is that no real-world device behaves exactly like an RLC circuit. The conductance of resistive devices, for example, changes with temperature; motor admittance varies in more complex ways, as a function of load and voltage. But given that many appliances can be approximately modeled by RLC circuits and that the deviations from the model are not consistent, admittance is certainly the appropriate choice.

A second argument for admittance is based on measurements of individual appliances as described in Section 2.1.3. To see how well the RLC model fits actual appliances, we measured the admittance of a selection of appliances in a laboratory environment in which we controlled the line voltage. Figures 3-1 through 3-3 show three observed behaviors. The electric coffee pot of Fig. 3-1 displays a nearly constant admittance. The water serves to stabilize the heating-element temperature and, as a consequence, the resistance of the element is nearly constant. The light bulb of Fig. 3-2, although still resistive, is not temperature stabilized. Higher voltages cause higher power levels, which cause higher temperatures, which cause higher filament resistances. Therefore, the conductance decreases with increased voltage. The current increases less than linearly and the power increases less than quadratically. The refrigerator motor of Fig. 3-3 shows a more complex behavior. Apparently some aspect of the design was optimized for 110 V which causes the minimum in the power at that point. The data for these and other appliances lead to the conclusion that admittance, although not constant, is generally more stable than current and power.

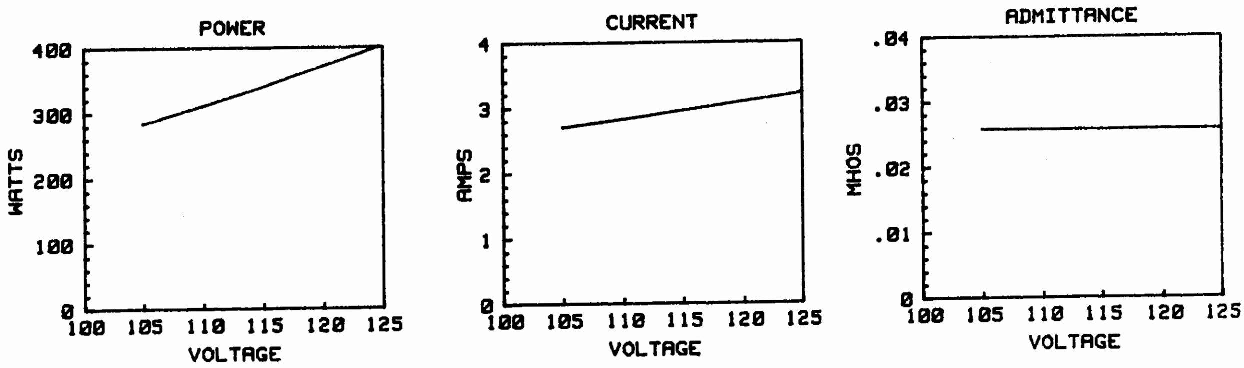


Fig. 3-1. Effect of voltage change on coffee pot.

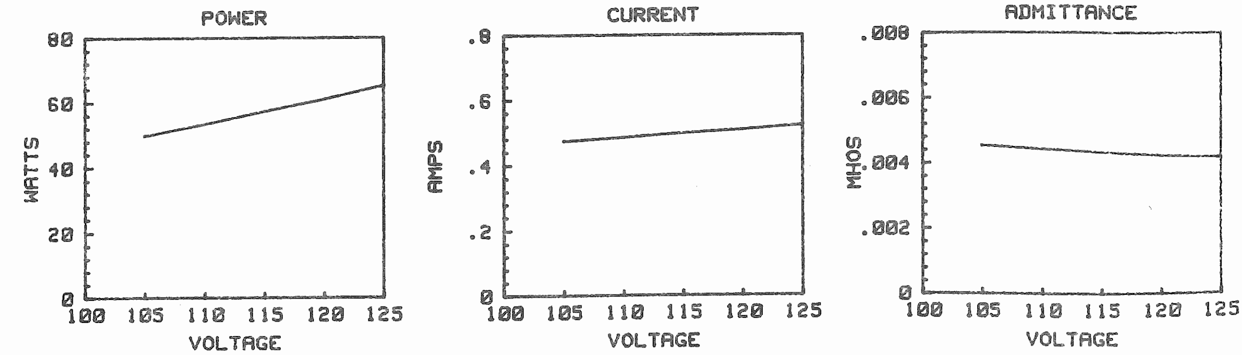


Fig. 3-2. Effect of voltage change on 60 W light bulb.

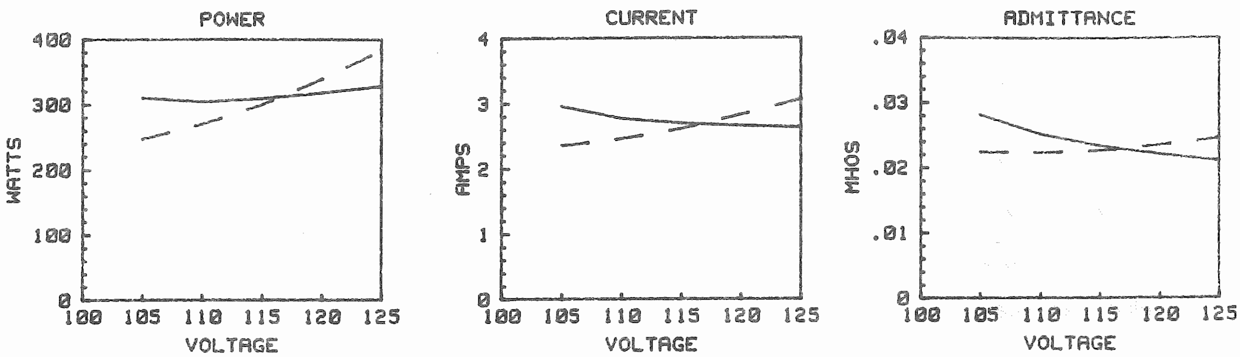


Fig. 3-3. Effect of voltage change on refrigerator.

As a check on the two previous arguments, and in order to determine the value of admittance for actual utility line-voltage variations, we have collected data in the field, as described in Section 2.1.4. The voltage at the residence is usually held fairly stiffly at about 125 to 126 V. Figure 3-4 shows the voltage on one leg for a typical 24-hour period. With such a consistent voltage, admittance, current and power are of equal utility. Several short periods were observed, however, during which the line voltage varied widely and unsteadily from this behavior. (These periods began after lightning storms and observed flickers. We presume that a storm-triggered fault caused the area feeders to be reconfigured, moving the residence temporarily further from the local substation.) Advantage was taken of these intervals to collect data which contrast power and admittance measurements.

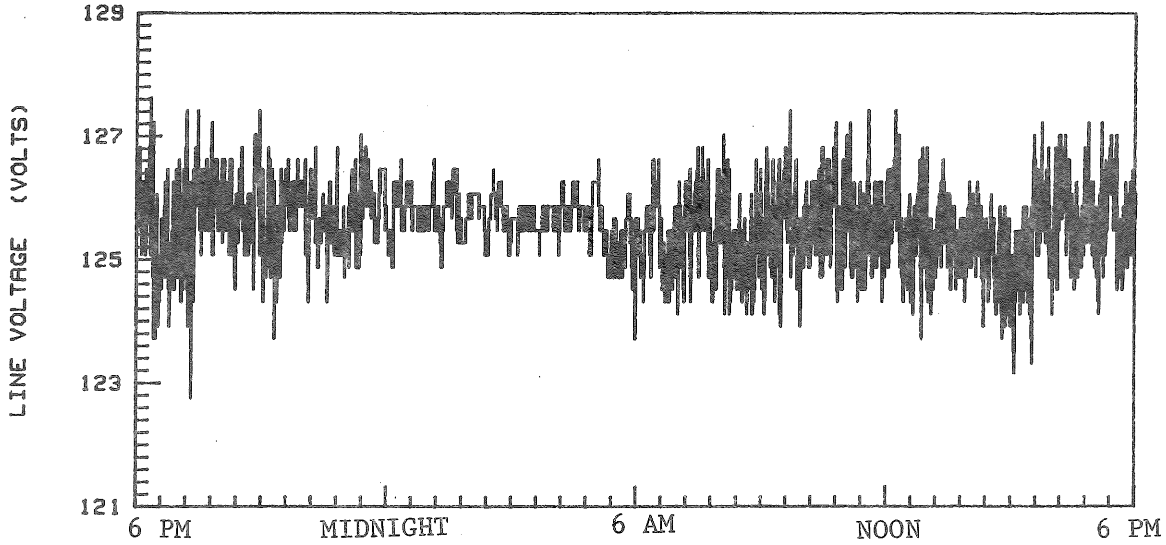


Fig. 3-4. Line voltage variation over 24 hours.

Figure 3-5 is a scatter plot of the real and reactive power signatures associated with the refrigerator turning ON and OFF over several days, including days displaying a wide range of voltages. The asterisks indicate ON transitions, and the circles mark the OFF transitions. The ON transitions cluster with a higher real power than the OFF transitions because the power usage of the refrigerator decreases slightly with time during each cycle, as can be seen in Fig. 3-6. Part of the wide scatter visible in Fig. 3-5 is due to line-voltage variation. A scatter plot of admittance for the same transition events shows reduced scatter. Instead of plotting admittance per se, we plot normalized power at 125 V, the average line voltage for this residence. Figure 3-7 indicates the admittance distribution by plotting the normalized power, which is calculated as

$$P_{norm} = 125^2\ Y = (125/V)^2\ P$$

where Y is admittance, V is the line voltage and P is the measured complex power. Normalized power is used because it is strictly proportional to admittance, yet yields familiar units. (A 100 W light bulb displays a normalized power of about 100 W rather than an admittance of 0.0064 Mho.)

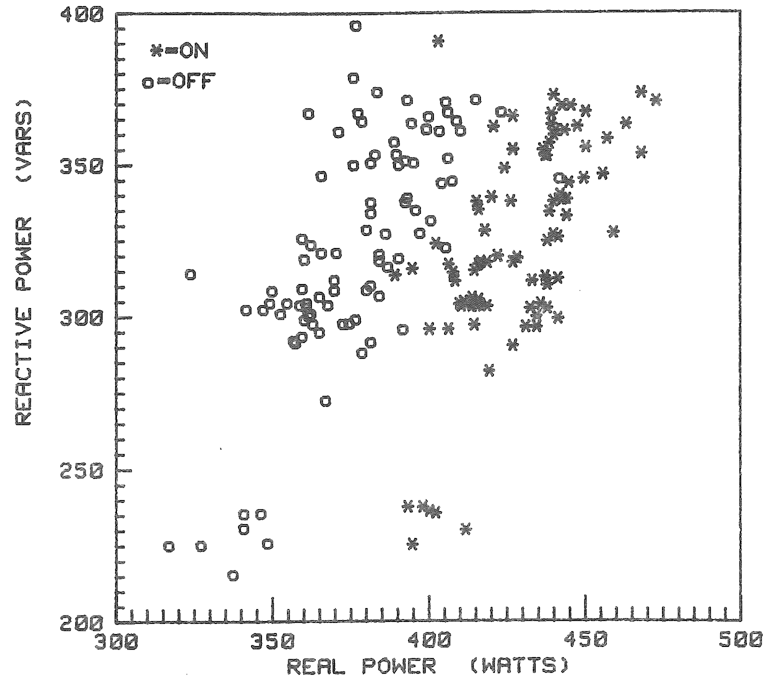


Fig. 3-5. Refrigerator transitions as measured.

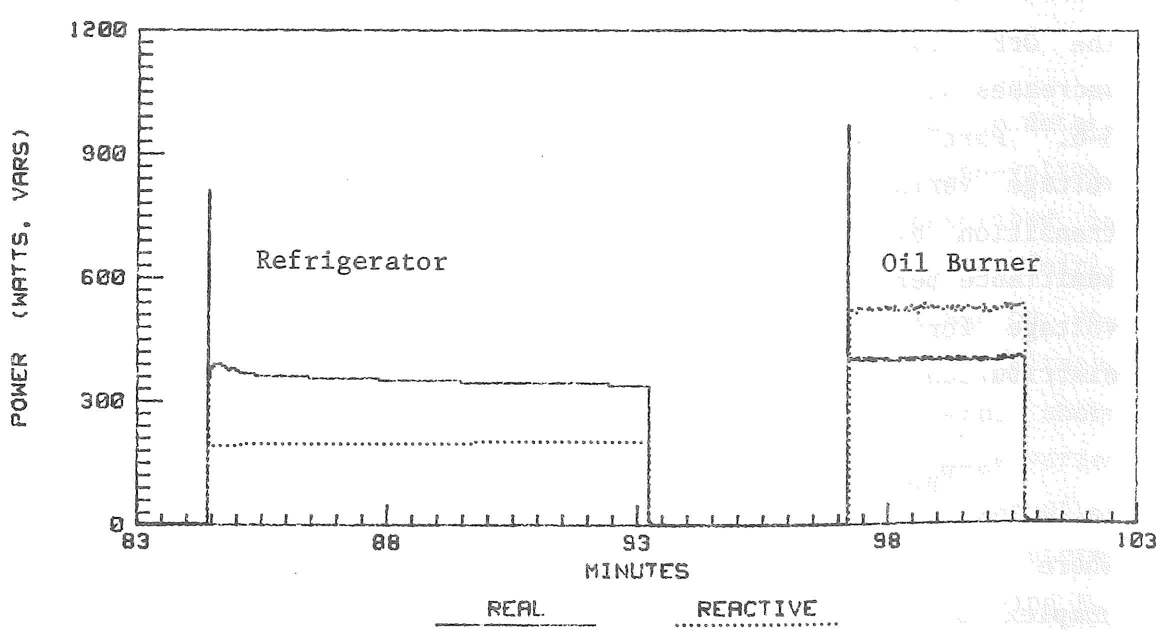


Fig. 3-6. Refrigerator and oil burner as function of time.

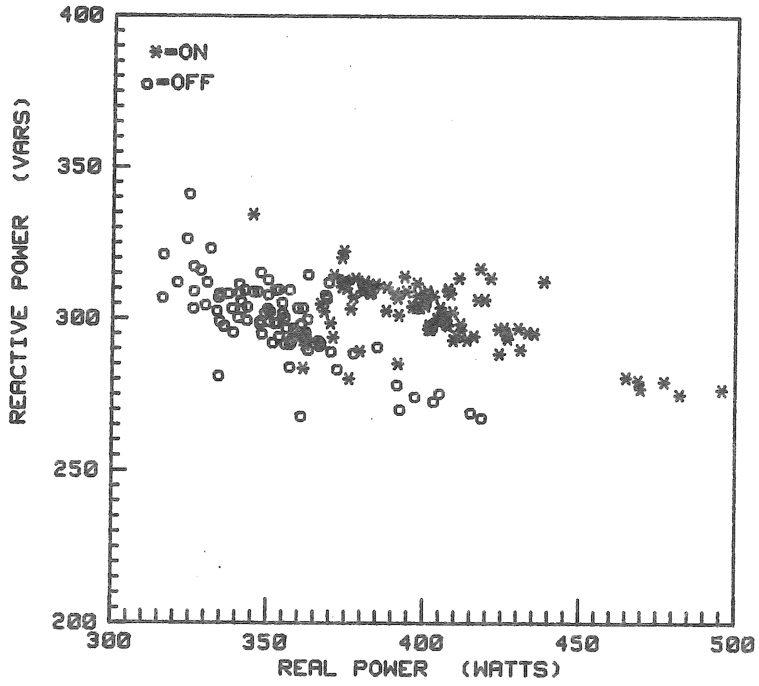


Fig. 3-7. Refrigerator transitions normalized to 125 V.

Comparison of Figs. 3-5 and 3-7 reveals, as predicted, that admittance clusters more tightly than power. This improvement is along the imaginary axis only, however. The real power shows a wide variation from transition to transition for reasons unrelated to line voltage variations. Temperature and pressure conditions in the compressor at start-up are the most likely explanation for the wide scatter in the real part of the refrigerator transition admittance. Figures 3-8 and 3-9 provide a similar comparison for furnace transitions over the same time period. Again, admittance is the more tightly clustered signature. The real part of the signature shows a greater clustering improvement than for the refrigerator. Much of the remaining variation may be attributable to the weakness of the RLC model in the case of motors.

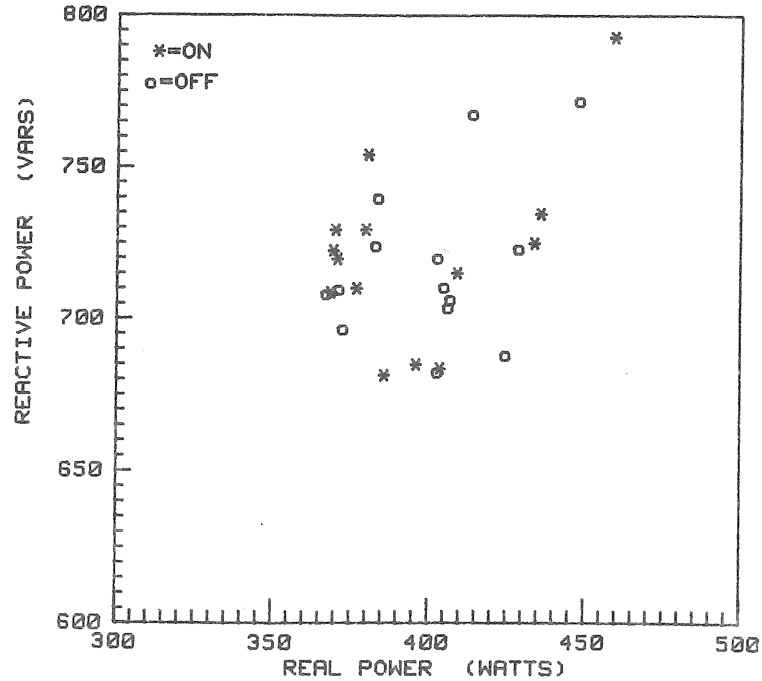


Fig. 3-8. Oil burner transitions as measured.

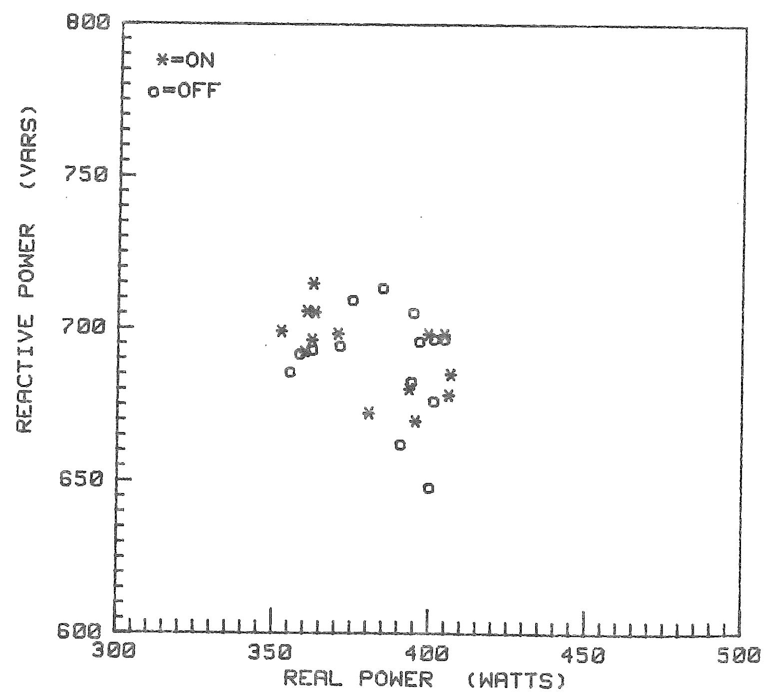


Fig. 3-9. Oil burner transitions normalized to 125 V.

We conclude that admittance (at the fundamental) is superior to current or power as a signature component because of its more consistent clustering property. Tight clustering of signatures is important to reduce the possibility of different appliances having overlapping representations, and to allow more appliances to be distinguished in the same signature space. The observed clusters are still of fairly sizable dimensions, especially considering that the RLC model predicts that SSRs will be points of no dimension. Part of the remaining scatter is a consequence of the transition-detecting algorithm used here, which is described in Section 3.3. Suggestions discussed there should improve the clustering.

3.1.2 Harmonic Currents

Harmonic currents at each multiple of the fundamental generate an infinite sequence of independent signatures which can, in principle, be used to distinguish appliances. The third and/or fifth harmonics, being relatively sizable in motors, are likely candidates. Figure 3-10 shows the digitized current waveform measured for an entire residence, as described in Section 2.1.4, when the only sizable appliance turned on was a 1400 W vacuum cleaner. (The lack of smoothness of the curve is a consequence of digitization in the AC Monitor and not significant.) Figure 3-11 shows the corresponding amplitude spectrum, as calculated by a discrete Fourier transform. The fundamental amplitude is almost 16 A; the third harmonic is 3 A; the fifth harmonic is less than 1 A. Because only one or two harmonics would be used in practice, a simple integration rather than a complete DFT would be used to compute them.

Although the magnitudes of the harmonics are plotted in Fig.3-11, they would not be used as signatures because they lack the linearity properties discussed in Section 3.1. The magnitude of a harmonic resulting from two appliances is not the sum of the individual magnitudes unless they happened to have the same phase. Instead, the complex amplitude phasor, being linear regardless of phase, would be used.

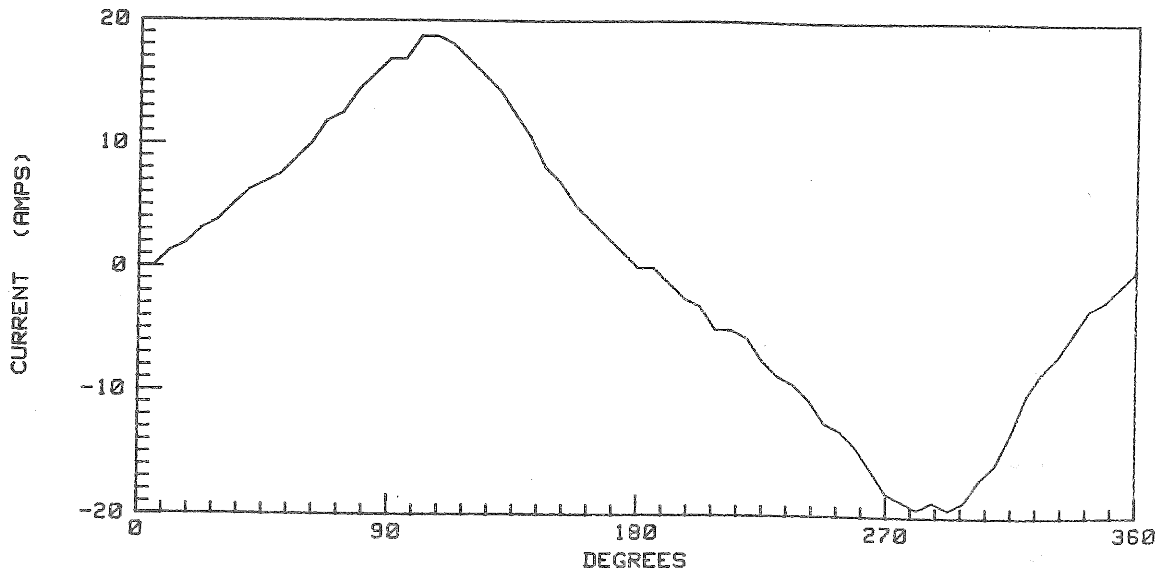


Fig. 3-10. Vacuum cleaner current waveform.

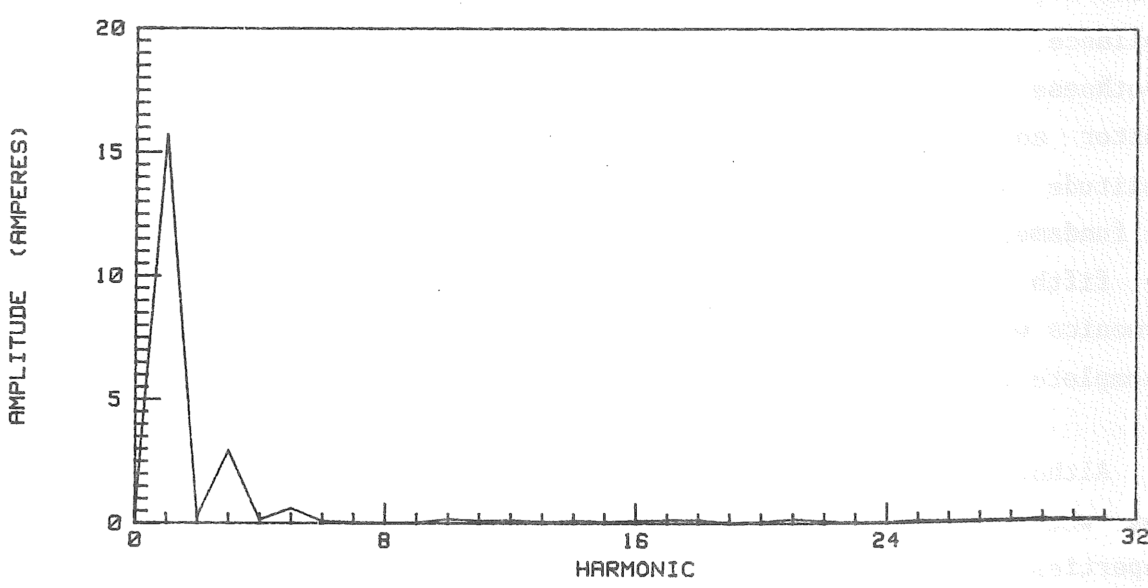


Fig. 3-11. Harmonic current content of vacuum cleaner.

Complex harmonic amplitude therefore has many suitable signature properties. It suffers from two drawbacks, however. The first is that for large appliances harmonics add little information that is not available from the admittance at the fundamental. Motors can be distinguished from resistive appliances by examining the susceptance. The third harmonic does not need to be consulted. Smaller appliances, as pointed out in Section 2.1.1, may be distinguished by their harmonics. Our first pass through the algorithm-development process will not focus on smaller appliances, however.

The second drawback of harmonic signatures concerns their stability and the problem of normalizing them to line-voltage effects. It was shown above that power or current at the 60 Hz fundamental can be usefully normalized by converting to admittance. It is not clear how well this can be done for harmonic current or power. The harmonic currents measurable at the service entrance are attributable not only to motors or other appliances which generate them, but also to resistive appliances which simply "pass through" any line-voltage harmonics. In addition, the harmonics generated in motors can be expected to vary with line voltage, voltage harmonics, and load. These effects may or may not cause serious problems. For now we will suspend our consideration of harmonic signatures.

3.1.3 Transient Currents

Signatures based on properties of starting transients hold the promise that they might distinguish between two appliances which appear identical in the steady-state components of the signature space. The signature component taxonomy presented in Section 3.1 lists three types of transient properties which we have considered. The classification by type into circuit, mechanical or switched transients was presented in Section 2.1.1. The peak value of the transient can be fairly consistent for some appliances, as Fig. 3-12 demonstrates. Here, the real and reactive power drawn by an oil burner (furnace) is plotted as it was manually turned ON and OFF five times. The real-power spike, over the one-second averages plotted here, ranges from 600 to 750 W. The reactive power varies from 0 up to 150 VAR. Other appliances show less consistency. Part of the variation may be due to the short duration of the spike causing it to be missed or obscured by the one-second averaging process. A hardware spike detector (rather than the software method used here) might provide more consistent information. To the extent that a spike represents the energy required to accelerate the motor shaft, it is possible that integrating the area under the power spike might provide a more consistent measure than the peak value.

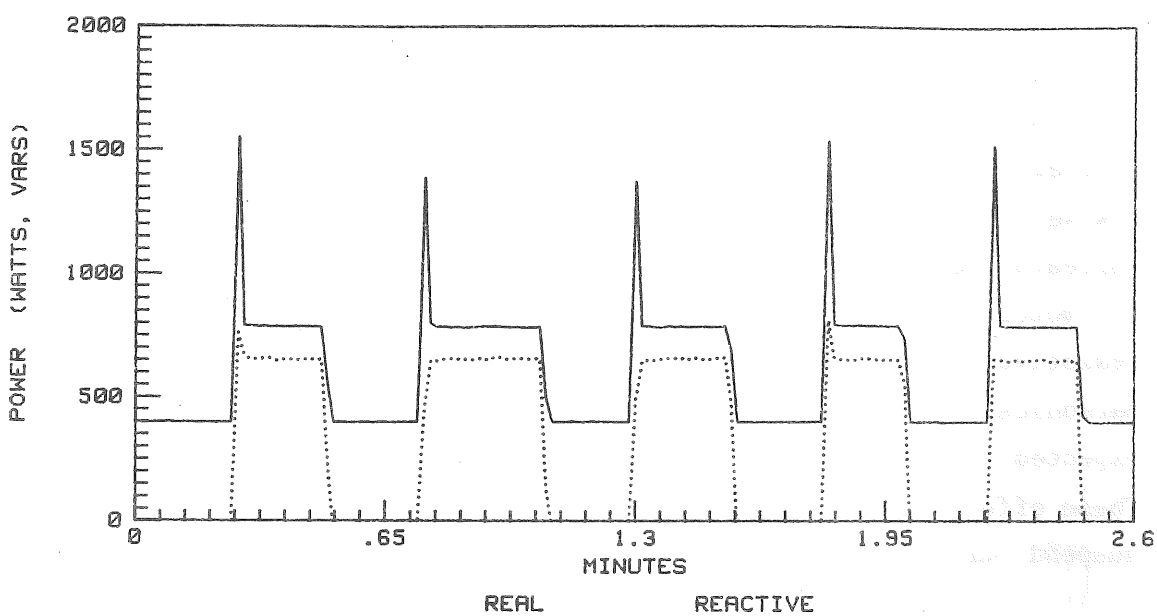


Fig. 3-12. Oil burner starting spikes.

Another property of the transient that is worth examining is the dc current associated with motor starts. Figure 3-13 shows the dc current measured during the same five furnace starts shown in Fig.3-12 (The zero offset is a miscalibration effect that can be ignored.) The dc transient results from the residual magnetism remaining after the previous turn-off, and it varies in sign and in magnitude each time the motor is turned on.

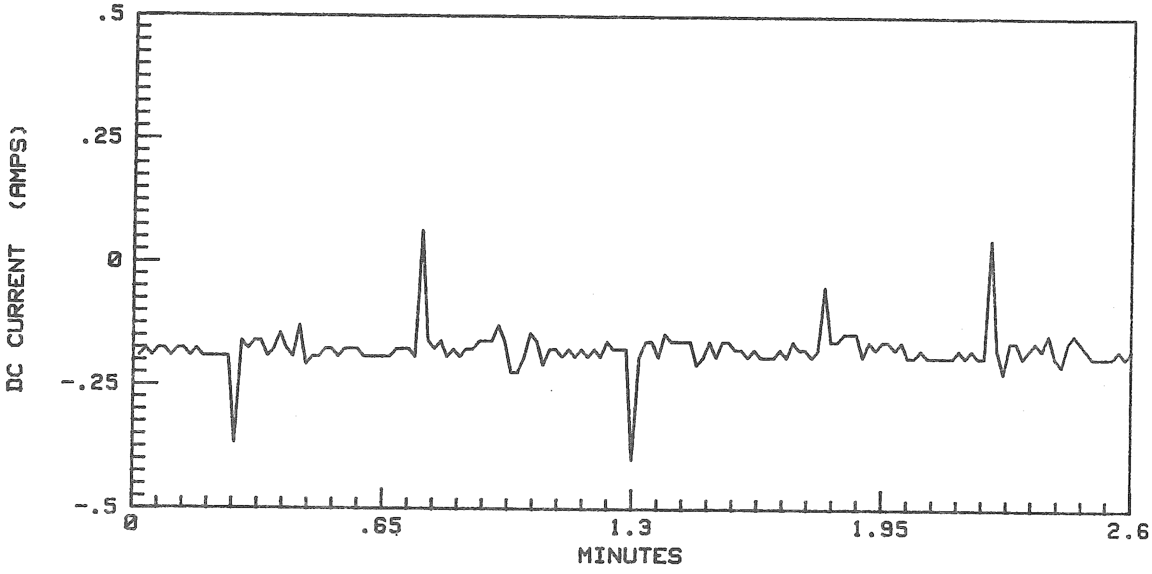


Fig. 3-13. DC current during oil burner starting spikes.

Two other properties of transients which may be useful are the duration or, in mechanical transients, the time-constant.

Although transient signatures hold promise as important discriminators of otherwise similar appliances, they suffer from two weaknesses. The first is that they are inconsistent and therefore potentially confusing to any clustering algorithm. The second is that they give information only about the turn-on event, not the turn-off event. If two appliances are both ON which are distinguished only by transient properties, the transient can be used to determine the order in which they turned ON, but not the order in which they turn OFF. We are therefore concentrating on the steady-state signatures, but allow for the possibility that it may be valuable to return to transient properties at a later time.

3.2 Sensors

In this section we review the properties which are desirable in the front-end sensors which can provide the primary residence data to the final device.

Linearity: The sensors should be linear in the sense described in Section 3.1. If not linear, they should be nonlinear in a well understood way which can be corrected in the software to obtain linear signatures. For example, if the sensor measured the log of power, the software could exponentiate to obtain the power.

Response Time: The sensor must have a rapid response to step changes in the input. If a large resistive appliance turns on, causing a step change in residence power, the sensor should reach a steady state very quickly after the power step. Otherwise a subsequent transition could become blended in with the first. A fast response time would allow two nearly simultaneous signatures to remain distinct. The necessary response time is discussed in Section 3.5.

Repeatability and Precision: In order that SSRs for similar appliances not overlap, they should be as small as possible. A sensor which possesses a high degree of repeatability is necessary so that noise does not dominate the measurements. Noise would cause the SSRs to expand beyond their actual nature.

Low Cost: In order to satisfy the goal that the final device be produced at low cost, the sensors can not be unduly expensive.

The digital-sampling technique of the Digital AC Monitor, described in Section 2.1.3 and the appendix has all of these properties. Furthermore, it is sufficiently versatile to compute admittance, harmonic amplitudes, or virtually any other describable signature from the digitized waveforms. We therefore expect that the most effective sensors will sample current and voltage waveforms and process them numerically.

3.3 Edge Detection and Spikes

The detection of residence transitions is a crucial first step in the overall algorithm. Time periods in which the residence changes from one steady state to another must be identified and the change quantified. The method used so far to collect data (for Sections 3 and 4) is quite simple and can be improved.

This method requires the specification of a step threshold. The threshold used here is 7 W for real power components and 7 VAR for reactive power components. The threshold that would be used in the final algorithm will depend on the repeatability and precision of the sensor device. Each of the separate steady-state signature components is sampled repeatedly. The sampling rate will be specified based upon a number of factors including processing capacity and sensor time constant. The periods used in this paper are one second and one half second. Each sample is compared to the sample at the beginning of the current steady-state period. If the change in each of the components is less than the the corresponding step threshold, then the residence is considered to be in the same steady state. If any of the components change by an amount exceeding the step threshold, then the residence is considered to be in a state of change. Sampling continues during the period of change (which could be just two samples) until no components change by more than the step threshold from one sample to the next. At this point the residence is considered to be in a new steady state. The differences in each of the signature components from the previous state to the new state are appended together to form the steady-state portion of the residence transition. The transient components of the transition can then be computed individually according to their nature. Duration, for example, is determined by counting the length of time the signature took to change to the new steady state.

This edge-detection algorithm passes completely over transients in power or other components and waits for steady state to be attained. If spike amplitude was desired as a signature component, it would be detected with a special variable which is programmed to seek out the maximum during the state of change.

Because this algorithm is only computing the difference from one sample to another, it is susceptible to effects of noise. A future algorithm will consider the average value of each component for a period just after and just before the transition with the expectation that this will result in tighter clustering. We expect that averaging out the noise in this manner will tighten clusters such as those of Figs. 3-7 and 3-9.

A second problem with this algorithm is that it does not recognize the gradual convergence of exponentially decaying transients. If a transient gradually approaches a steady-state level, this edge-detection algorithm perceives a steady state as soon as the sample-to-sample change is less than the step threshold. This could be at a power level considerably different from the eventual steady-state power if the transient continues to decay at a slow rate for many samples. A possible solution to this problem is to analyze the transient type and, if it shows exponential decay, estimate or wait for the steady-state value. It is not clear that this is required, however. Given an ARS which allows turn-on signatures to differ from turn-off signatures, the actual transition which would have been measured at the end of the transient is not crucial. All that matters is that the simple algorithm consistently finds the power level at which the change is less than the step threshold. This signature would be learned and used for recognition.

3.4 Transitions and Noise

The nature of the nonintrusive algorithm requires that signatures be changes in residence variables rather than levels of the variables. The total power usage of the residence does not contain much information as to exactly which appliances are ON or OFF. Only changes in the power level can be used to determine this. For this reason the first operation the algorithm performs on residence power levels is edge detection to find changes in power. For the most part, step changes are sought out because they are characteristic of an appliance transition. There are two exceptions to this which we have observed. In one case a ramp rather than a step is the signature. In the second, a sequence of steps is present during the extended time period in which an appliance is in a certain state, rather than just the instants at which the appliance changes state. These two effects interact with the problem of signature noise.

Figure 3-14 shows power usage of a dishwasher during a single wash cycle. No other appliances were ON in the house during this time interval. Several components of the dishwasher can be recognized from their signatures. The one that is of interest here is the ramp, repeated six times, in which the power increases by about 250 W over almost two minutes. Note that in the fourth instance, the ramp is severed by the turn-off of the heating element. Although this slow ramp is a signature highly characteristic of the dishwasher, it is not detected by a transition detector because there is no sudden change in power. Instead the edge detector will see a sequence of small positive steps. Ramps as signatures could be included in the overall algorithm, but they introduce certain complications. One complication, visible in this figure, is that of their extended duration, which lends itself to overlapped transitions that can be complex to break down. We are not sure how common ramps are because this is the only appliance we have observed which displays consistent ramps. Because of this and the added complication they introduce, our first pass through the algorithm development process will not include ramps.

The opposite case from the dishwasher, which changes state without a step transition, is the top-loading washing machine, which remains in the same state during a long sequence of step transitions. Figure 3-15 shows the power usage during a washing-machine cycle. The oscillation of the agitator causes the power usage to jiggle constantly during agitation. This appears as small steps increasing and decreasing the power level during the entire time the appliance is in certain states. Compared to power usage at the 60-Hz fundamental, this jiggle is a subharmonic. A similar jiggle can be seen in the furnace power trace of Fig. 3-6. Because these small steps are a distinctive steady-state signature for certain states of certain appliances, it may be worthwhile to incorporate them in the algorithm by keeping a noise-level statistic between major transitions. Changes in the noise level can then be used as a signature, but not immediately since some time is required to determine the noise accurately. This same statistic or a similar one could be used to detect the presence of ramps, which appear as a sequence of small transitions with the same sign. The noise level can also be used as a dynamic measure of expected accuracy for the next observed transition, perhaps even to the point of having the algorithm "give up" and not risk the possible errors associated with learning new appliances during periods of high noise.

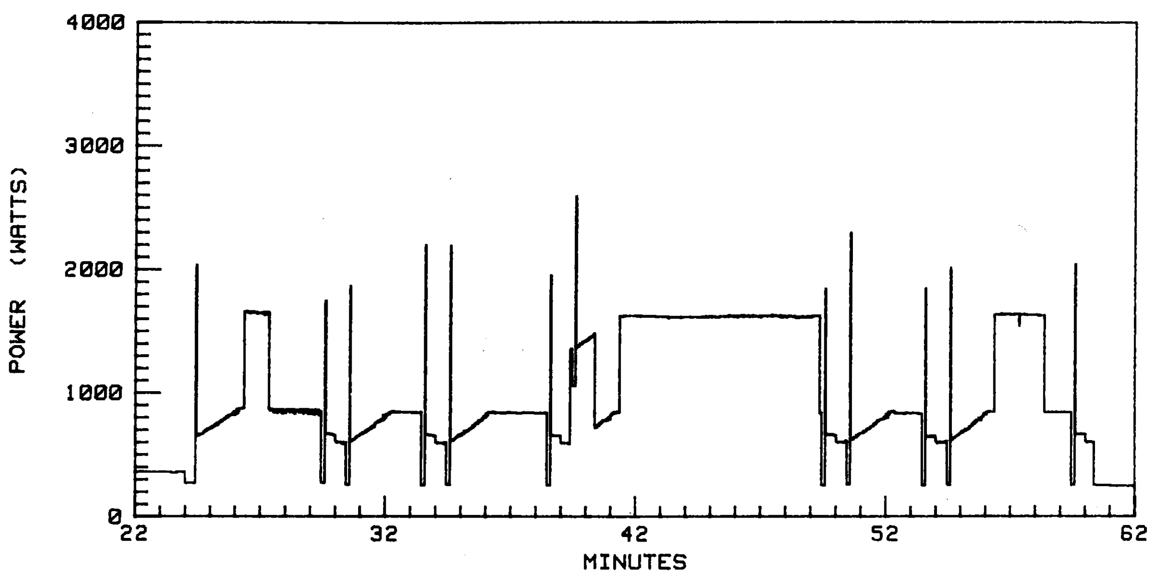


Fig. 3-14. Dishwasher power usage.

Because of these three uses, we expect that the final algorithm will keep track of the noise level. This will probably involve setting a transition-size threshold for each signature component. (This threshold would be larger than the step threshold of Section 3.3.) Changes which are above the threshold would be used directly by the algorithm. Changes below the threshold would be incorporated, perhaps with simple average and RMS statistics, to calculate a noise level for each interval between the larger transitions.

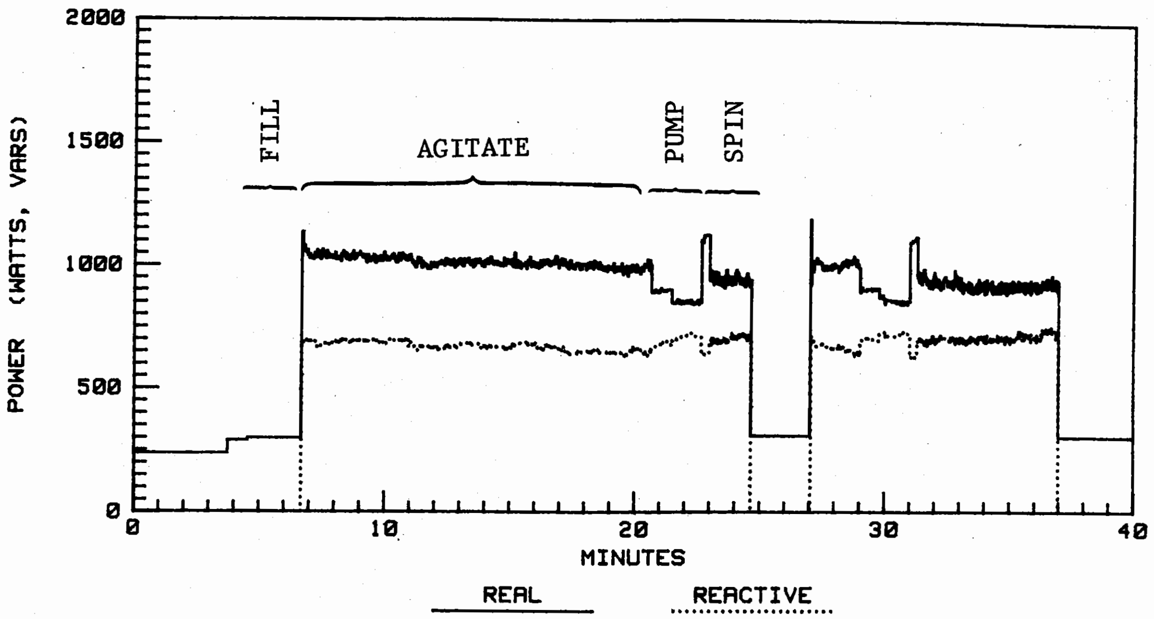


Fig. 3-15. Washing machine power usage.

3.5 Simultaneous Transitions

A certain fraction of all residence transitions will be the combined effect of two or more appliance transitions occurring simultaneously. The precise moment of appliance transitions need not be identical for this to happen. It is sufficient that one transition start before the transient of a previous transition dies out. As discussed in Section 3.1, the steady-state portions of the separate appliance transitions add linearly, but the effect on the transient components is unpredictable. The steady-state components can therefore be used to break down the separate appliance transitions. When a residence transition is observed which does not agree with any previously learned appliance, several actions can be taken. One is to try to break down the transition as the sum of two known appliances. This strategy is used effectively in the recognition program of Section 4 which simply performs an exhaustive search of all pairs of appliances. That search is reduced to a small fraction of its possible size by only considering the turn-on or turn-off transition for each appliance which is compatible with its current state. For example, if the last recognized transition of the refrigerator was a turn-on transition, the search does not consider sums of pairs of transitions in which one element is another refrigerator turn-on.

The exhaustive search approach, though effective for a small number of appliances, would become computationally expensive if hundreds of appliances were modeled. The number of pairs to consider would be quite large, though not necessarily prohibitive. Fortunately, additional information is available which can reduce or eliminate the search. The two simultaneous appliance transitions are likely to have corresponding unmatched transitions in the residence transition stream, as shown in Fig. 3-16. For example, if two appliances happen to turn ON at the same time, they will probably turn OFF at different times. A sophisticated recognition algorithm can hold on to the unrecognized double turn-on transition until it detects the two separate turn-off events. At that point it may note that the last recognized transition for each of the appliances was also a turn-off and that the sum of these three unexpected transitions is zero. These conditions together resolve the simultaneous transition.

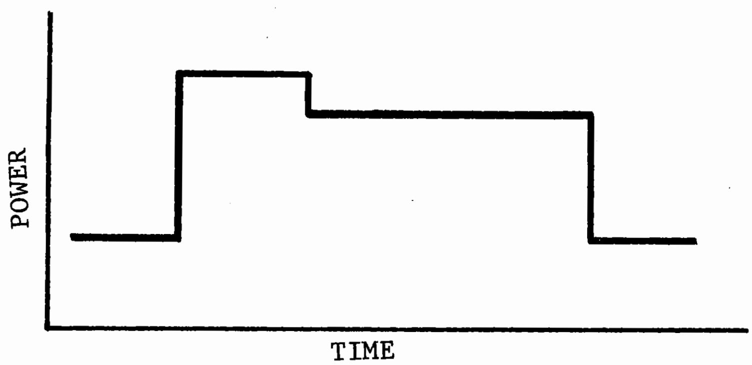


Fig. 3-16. Transitions due to two appliances turning on simultaneously.

Statistics of the frequency of each pair of appliances changing simultaneously would be kept for use by the learning algorithm. If a FSM ARS is used, the two separate appliance models could be fused together into a more complex FSM if their simultaneous transition statistics were higher than normal.

Note that the situation shown in Fig. 3-16 is ambiguous in three ways. Any one of the three transitions might be broken down into the sum of the negatives of the other two. The particular breakdown that is appropriate in any situation depends on which two of the three are recognized independently as known appliances. More complex breakdowns in which three or more appliances change state at once, and several simultaneous events must be broken down together, are resolvable in principle. A sequence of unrecognized or unmatched transitions which sum to zero indicates this situation. However the expense of breaking down these complex situations and their relative unlikelyhood make it implausible to attempt this.

One strategy for reducing the likelihood of simultaneous transitions is to increase the sampling rate of the edge detector. If two appliance transitions happen closely separated in time, there may be a short period of steady state between them. An edge detector with a rapid sample rate could detect the steady state in cases where an edge detector with a slower sampling rate would combine the two transitions into a single residence transition.

There are two limits on the sampling rate. One is simply the computational burden it imposes on the overall algorithm. This is relatively slight because at each sample all that is required is a check of the change in signature compared with the step threshold. If the rate is increased, most of the additional samples result in no additional transitions for the higher leveled aspects of the algorithm to analyze. Those additional transitions which are created by the separate detection of two separate appliance transitions when otherwise only one would have been detected save a great deal of work for the higher leveled algorithms because it is no longer necessary to separate out the individual transitions.

The upper bound for sampling is sixty times per second, because RMS power is not defined on periods less than a cycle. The real limit on the sampling rate is the rate at which the sensor signals can be expected to change. If power usage of an appliance showed a true step change at transitions, then very rapid sampling would be justified. Measurements show however that most appliances take 0.1 to 1 second to reach steady state, and fans and refrigerators may take up to 10 seconds to stabilize. A sampling rate between two and ten samples per second therefore seems reasonable.

One interaction between sampling rate and spikes should be mentioned in this context. The edge detector described above, with a fast enough sampling rate, can catch the maximum of a spike as a moment of zero derivative and report a transition to steady state. If this happened consistently, it would pose no problem to a FSM ARS which could model the period of spike as a state. This state would be exited immediately by the transition from the end of the spike down to steady-state level. A problem arises, however, when the top of the spike is detected inconsistently. For any given sample rate there will be some spike duration which is sometimes detected as a steady state and sometimes not, depending on the relative timing between the spike and the measurments. This effect has been observed for a fan by the program of Section 4. A branching FSM such as that of Fig. 3-17 could be used to represent such an appliance. If the edge detector caught the top of the spike as steady state, the state SPIKE would be entered, then exited. Otherwise, the ON state would be entered directly. Other approaches to this problem are of course possible, but this method seems the most natural if FSM modeling is available.

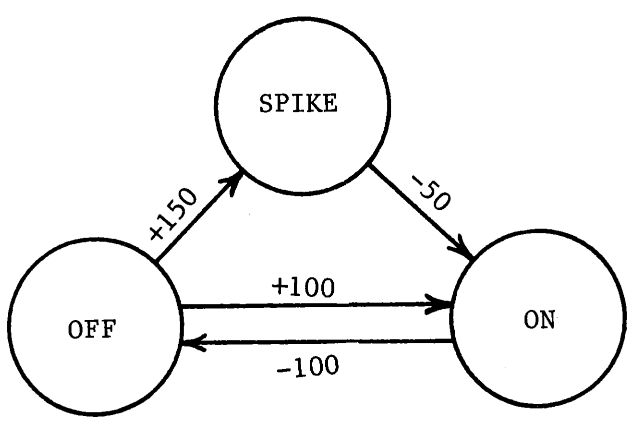


Fig. 3-17. FSM for appliance with inconsistent spike.

3.6 Continuously Variable Appliances

There is a class of appliances which will cause difficulties for any nonintrusive algorithm. This is the set of appliances in which the power level and other signature components can vary continuously as a function of the operator's control. Light dimmers, power tools, and ham radios are of this nature. As far as we can see at this point, this class of appliances can not be recognized satisfactorily. The best we can do is to have a recognition algorithm which can throw out time periods in which continuous changes occur. We are continuing to consider this problem.

3.7 Definition of Appliance ON and OFF

Most discussions of appliance-usage statistics tacitly assume that appliances are either ON or OFF. This assumption can fail to hold in two different ways. The first is that an appliance can have more that one ON state. The second is that there are two distinct levels of description to which the terms ON and OFF are appropriate. These two topics are taken up in turn.

The problem of more than one ON state is clarified by the FSM of Section 2.2.2. Many appliances can operate in a range of states, all of which are considered not OFF. For appliances with more than just an ON and OFF state, two solutions are possible to the resulting problem of describing the appliance usage. One is to calculate total energy usage in any state, and total ON-time in any state other that OFF. The second is to calculate energy usage and ON-time in each state separately. The choice depends on whether or not the extra data provided in a state-by-state breakdown is of sufficient interest to load researchers to justify preservation.

Note that the calculation of total ON-time in any state other than OFF requires that the algorithm be able to locate the OFF state in the FSM. Generally speaking this is simple to determine. A steady-state real and reactive power can be associated with each state, as described in Section 5. The state associated with zero power is the OFF state. A complication arises if there is more than one such state. The constraint of noncollapsibility, defined in Section 5, should eliminate this problem in most cases. The problem could still arise in the case of a washing machine, or other sequential event appliance, in which there is a waiting state in which very little power is drawn. The effect of these states on total ON-time statistics is slight enough to be ignored. Note that the energy-related appliance usage statistics do not require the identification of the OFF state.

An appliance can be described as ON or OFF from two points of view. To the occupant cooking dinner, the stove is ON from the time the switch is turned ON until it is turned OFF. From the standpoint of power usage, however, the stove might be cycling thermostatically ON and OFF dozens of times in the same time interval, and is actually ON for a much shorter period of time than the user knows. It is not clear which notion of ON-time is of more use to load researchers. The power-related notion is the one which is directly calculable by the recognition algorithm. An algorithm might be devised to connect together the separate cycles as an estimate of the occupant-related ON-time, but it is not clear that this can be determined in principle. It is likely that any reasonable algorithm would estimate that the refrigerator is always ON. This may or may not agree with the occupants perspective.

Another appliance which is misrepresented by a simple choice of only ON and OFF is the instant-picture TV. These TVs draw power even when turned OFF to keep the tube filiments warm so they will be ready at all times to provide an instant picture. Because only changes in appliance state can be recognized, the energy usage in the OFF state can not be identified by a nonintrusive device unless the TV is frequently plugged in and out.

3.8 Appliances wherein OFF is not negative of ON

In most appliances which we have observed, the transition associated with turning ON is not the negative of the transition associated with turning OFF. An appliance such as the coffee pot discussed in Section 3.1.1 is the exception in which the admittance remains constant during operation. Most heating elements, such as stoves, ovens and clothes driers, heat up sufficiently to affect their resistance. This causes a gradual drop in power within each cycle of the appliance. Most large motors also show a drop in power within a cycle. (See the discussion of the refrigerator in Section 3.1.1. for example.) Here the cause may be a combination of an actual power drop and the weakness of the edge detector discussed in Section 3.3 in which a small but nonzero derivative is considered to be steady state before the spike has completely decayed.

There are several ways in which this can be modeled. In the FSM ARS there is no difficulty. A model like the two-state FSM of Fig.2-17, but allowing two different transitions, would be used. If the SSR ARS is used, the regions can be made large enough to include turn-on transitions and also the negative of the turn-off transitions. Although simple to implement, this is likely to result in unacceptable overlap of appliance representaions. A more satisfactory solution within the SSR ARS is to allow the recognition algorithm to move the region closer to the origin if necessary when considering turn-off transitions. (These are easily identified by their negative real power components.) An approach similar to this is used in the recognition algorithm of Section 4. It has the weakness that all appliances are treated equally, whether or not their power level changes.

A more complex ARS in which the ON and OFF transitions for each appliance can be specified individually is a better solution. If each appliance were modeled as two regions of signature space, one for the turn-on transition and one for the turn-off, then this effect could be handled properly. A constraint on the ARS would be that the regions must be nearly negatives, in a way that could be made precise after further examination of appliances.

3.9 Sequential Event Appliances

Sequential-event appliances are those like a dishwasher or washing maching which automatically pass through a sequence of states (e.g., wash, rinse, spin, dry) during a single user cycle. These appliances provide a strong argument for the necessity of the FSM ARS. Examination of the washing machine cycle of Fig. 3-15 shows that it can be modeled as two passes through a six-state cycle. The rinse cycle repeats the exact sequence of transitions as the wash cycle. The machine which models this is shown in Fig. 3-18. Only five states would be necessary if the 100 W transitions were ignored. The dishwasher of Fig. 3-14 is somewhat more complex because of the ramps and the fact that the heater is only ON during certain cycles. We omit an FSM for it because of the difficulty introduced by the ramps.

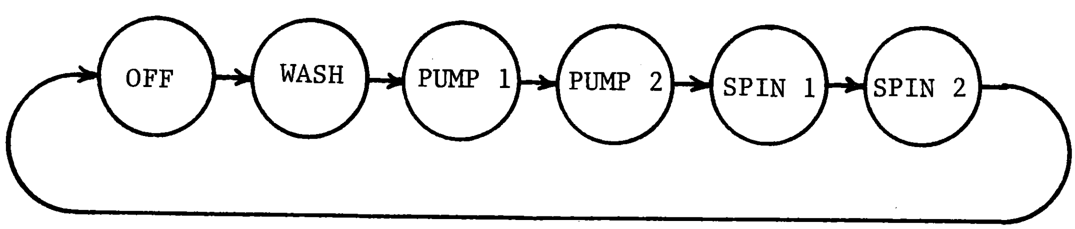


Fig. 3-18. FSM for washing machine.

The clothes drier shown in Fig. 3-19 provides another example of a sequential-event appliance. Each time the heating element is energized, it is introduced by a long spike of the type discussed above in which the resistance changes with temperature. The spike lasts several seconds and is modeled with a state labeled SPIKE in Fig. 3-20. The very first spike is larger because it includes a motor start-up load. This spike is represented with the state STARTING SPIKE. The transition indicated with the dotted line in Fig. 3-20 is not indicated by the data in Fig. 3-19. This transition would be observed if the door of the drier were opened during the heating portion of the cycle. As such, it is a special case of a general class of transition to OFF which any appliance can take from any state if it interrupted in its cycle by opening a door or being unplugged, for example. A special provision will have to be made in the recognition algorithm to allow for this and to allow resumption in the interrupted state if the cycle is continued.

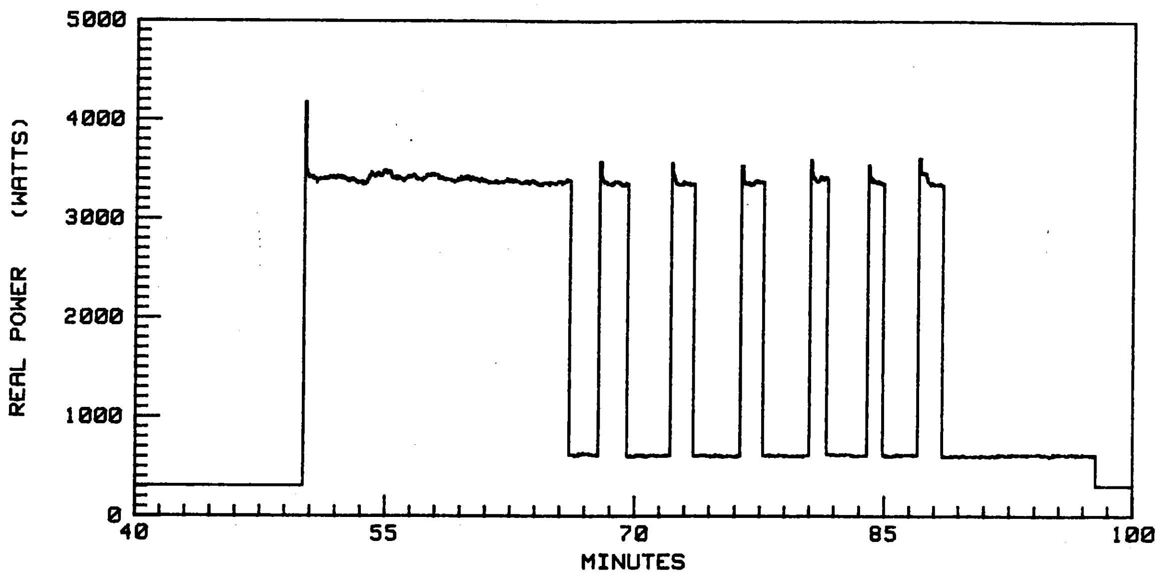


Fig. 3-19. Drier power usage.

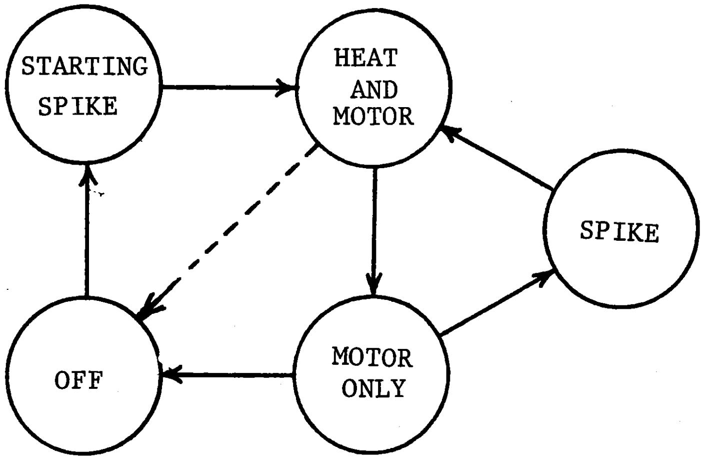


Fig. 3-20. FSM for drier.

To model a sequential-event appliance without the FSM ARS is difficult, but not impossible. In principle these appliances can be modeled as a set of independent individual appliances, e.g., the motor, the heater and the spike. The difficulty with this approach is that often several of these components change states simultaneously. For example, when the drier turns ON, the transition would have to be analyzed into three simultaneous components. It is not clear that the methods of Section 3.6 are capable of doing this consistently. The advantage of the FSM approach is that it allows this to be learned and represented.

3.10 Multiple-Speed Appliances

Like sequential-event appliances, multiple-speed appliances support the need for the FSM ARS. The generic HIGH, MEDIUM, LOW appliance of Fig. 2-16 is the paradigm for all similar appliances. The parameters for specifying such a FSM are the number of states and the connections provided by the transitions. Of all the possible connectivities imaginable, only three seem to be realized in actual appliances. The first we call cyclic. The three-way light of Fig. 2-15 is of this type. Each state must be passed in turn before returning to any given state. The washing maching FSM of Fig. 3-18 is a six-state machine with this connectivity. The second type of connection we call ordered. An example is given in Fig. 3-21. It is typified by a multiple-position switch which can be turned forward or backward, but can not pass over intermediate states. The final connectivity we have observed in multiple-speed appliances is the fully connected type shown in Fig. 2-16. Any state can be entered from any other.

All machines can be considered to be of the fully connected type if desired. From this point of view, certain transitions just happen not to be traversed. This may be a reasonable way to model all appliances but we suspect it will cause too many transitions to be under consideration at any given time. One difficulty with fully connected appliances is that the storage requirements necessary to represent them increase rapidly with the number of states. For example, a push-button blender with seven speeds and OFF state allows for 56 transitions. It is not likely that they would all be observed.

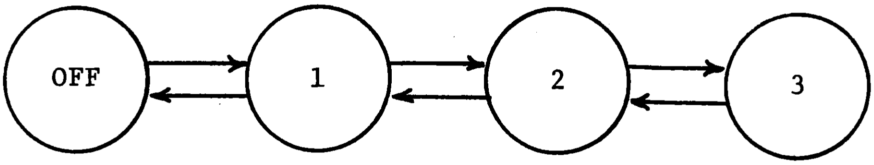


Fig. 3-21. FSM with ordered states.

3.11 Identical Appliances

A difficult problem for a nonintrusive appliance load determination algorithm is the case of two identical appliances. The two appliances might be similar, such as two different 1200 W heating devices, or they might literally be identical, such as two burners of the same size on a stove, or two 100 W light bulbs. In the case of similar appliances, the situation may be rectified by the addition of additional components to the signature space, but the case of truly identical appliances will always remain. They are likely to appear identical using any possible signatures.

The only information that can be determined for the individual appliances is of a statistical nature. A record can be kept of how often neither is ON, how often one is ON, and how often both are ON. During the time that only one is ON, it would not be known which one was ON, but it is not clear that this matters. If the SSR ARS is used, this information could be kept with the use of a counter for each appliance indicating how many instances of that appliance are ON at any given time. The count would be incremented when an ON transition was detected and decremented after the OFF transition. If the count did not return preiodically to zero, this would indicate a problem with the appliance representation. In the FSM ARS, the FSM of Fig. 3-21 would serve the same purpose. Each state indicates how many instances of the modeled appliance are ON. Statistics would be kept which record what fraction of the time is spent in each state.

From this data, individual appliance statistics could be determined if the assumption of independence was made. For example, if F1 and F2 are the actual fractions of the time that burner 1 and burner 2 are ON, then the probability that both are ON is F1*F2, and the probability that exactly one of the two is ON is F1+F2-F1*F2 if they are independent. These two probabilities would be measured, and the actual fractions, F1 and F2, could be solved for in a quadratic equation. This probably will not be done, however, because the directly measured statistics are considered to be sufficient for load research purposes and because of the weakness of the assumption of independence.

3.12 Portable Appliances

Portable 120 V appliances, such as a hair drier or vacuum cleaner which is sometimes used in one room and sometimes in another, introduce a special problem to the nonintrusive algorithm. The problem occurs when the appliance switches from one leg of the 240 V service to the other. The signature then becomes rotated in the signature space from one half of the components to the other. This requires a special test in the recognition or learning algorithm which checks for the leg switch. We expect that the algorithm will start out by learning the appliance as two separate appliances. After waiting a suffiently long period in which it is noted that both appliances are never ON at the same time, the learning algorithm would fuse the two models into a single appliance representation which includes a flag indicating portability.

A minor point to note about portable appliances is that their measured admittance would vary somewhat according to which branch circuit of the house that the appliance was plugged in to. The admittance of an appliance as measured from outside the house includes the resistance of the wiring between the sensor and the appliance. This will vary as the appliance is moved, but should not significantly affect the representation. These changes ought to be far less than the variation due to other causes such as those discussed in the connection with the refrigerator in Section 3.1.1.

3.13 Appliance Evolution

The set of appliances in a residence can change in two ways. One way is that the inventory can change. Appliances are purchased and discarded. The second way is that particular appliances change their properties. In the first case, the learning algorithm will be expected to learn that new appliances have been introduced in the same way that it learns about any appliance. The decision that an appliance has left the inventory is somewhat harder because seasonal appliances may be out of use for almost a year, and then return. The removal of appliances can be handled in three ways. They can be simply removed from the residence model, which would require learning them again if they returned. Alternatively, they could be removed from the model, but kept around in a list of inactive appliances for use by the learning algorithm to speed up the learning process if they do re-appear. A third choice is simply never to discard appliances, but this approach is not likely to allow the appliance inventory to converge properly.

A particular appliance can change its properties in two ways. A sudden discontinuous change, such as happens when something burns out or breaks, would be perceived as a change in the appliance inventory. The new properties of the appliance could not be related to the previous model. A second type of change is more gradual; motor bearings wear out, for example. This could result in a continuous change of the signatures over time. Gradual change of this sort can be followed by continuously updating the appliance representation based on changes in the recognized transitions. The program discussed in the next section has the capacity to track these changes. It compares the average of the recognized transitions for each appliance to the center of the transition SSR. The regions can then be allowed to wander so that the center follows the observed center. The need for this procedure has not yet been observed, however.

4.0 AN APPLIANCE RECOGNITION PROGRAM

As discussed in Section 2.3, the purpose of an appliance recognition program is to calculate the appliance state vector, given the residence model and the residence transitions. We have written and operated an appliance recognition program which is fairly successful according to the criteria of Section 2.6. The program runs in real time, in the house of the author, on an HP9845B computer using a Digital AC Monitor as a sensor device. The hardware arrangement is described in Section 2.1.4.

The program has been given a residence model which consists of seventeen appliances. Each appliance is represented by a region of the four-dimensional signature space, as described in Section 2.2.1. The signature components used are normalized real and reactive power on the two legs of the 240 V utility service, measured at a point between the kWh meter and the distribution panel. The SSR geometry which was selected is that of rectangles (or more precisely, four dimensional rectangular hyper-parallelepipeds). It would be a minor change to convert the program to use ellipsoidal regions. It would not be too difficult to convert the program to use FSM appliance representations.

The appliance representations in the signature space are shown in Fig. 4-1. More precisely, Fig. 4-1 collapses the four-dimensional signature space into the two-dimensional complex power plane. 120 V appliances on the two legs are shown with solid and dotted lines. The other leg of each 120 V appliance is modeled as a region of the same size, but centered around zero real and reactive power. 240 V appliances are shown (using a heavier line) as just one leg. The second leg is identical to the first for all the appliances modeled. These appliance representations were determined by turning the appliances ON and OFF individually and noting the signatures detected by the AC Monitor. Note that there is a phase-angle error created in the current transformers which provide the input signal to the AC Monitor. The transformers are oversized for the application, which

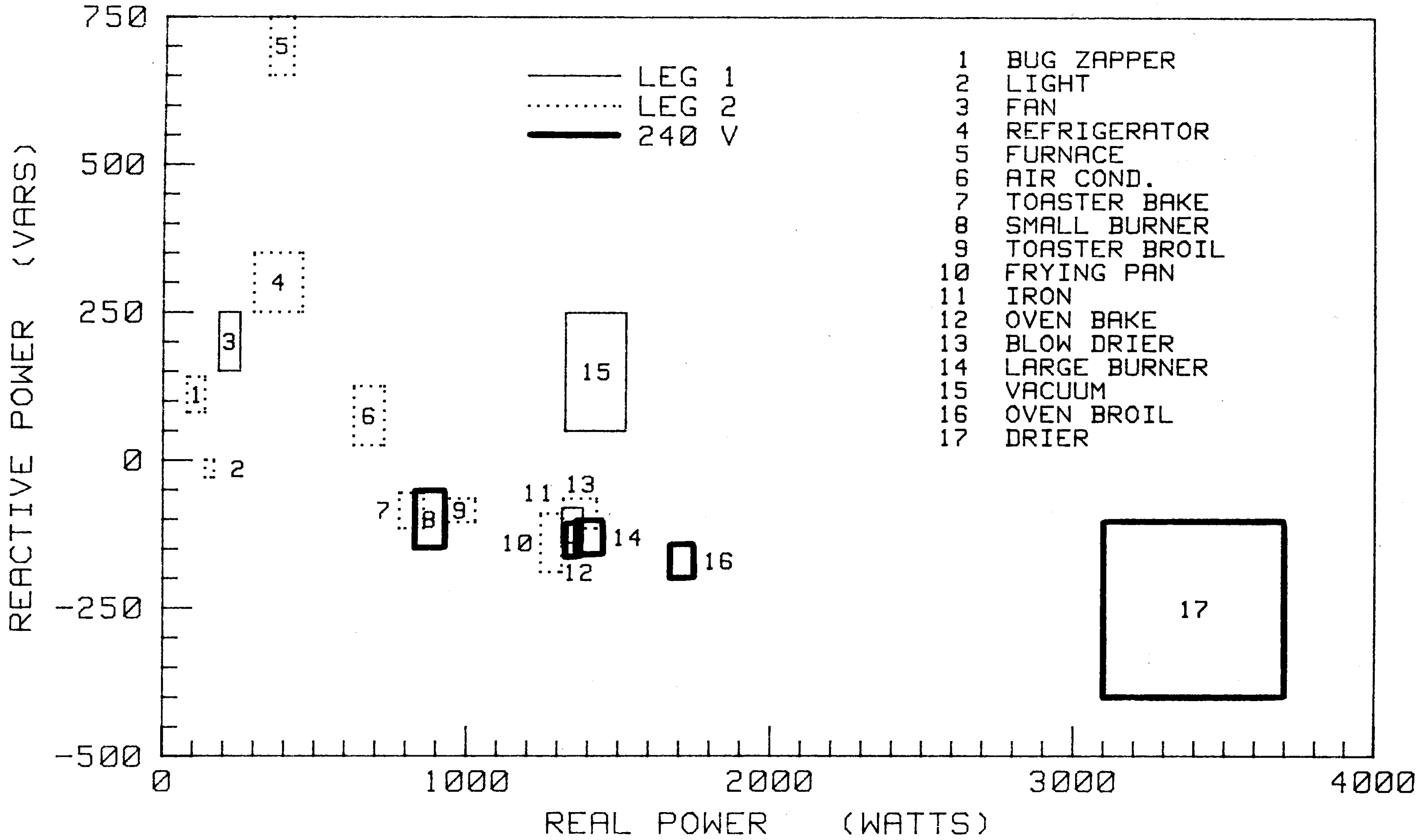


Fig. 4-1. Appliance representations for recognition program.

results in an output current which leads the input current by a few degrees. (The correct current transformers have been ordered but have not been installed as of this writing.) This does not disturb the algorithm, however, because it is a linear transformation of the signature space as discussed in Section 3.1. (It might cause trouble to an identification algorithm, however, if it misidentified some of the resistive heaters because of the inaccurately measured nonzero reactive power consumption.)

The operation of the algorithm is sketched in Fig. 4-2. The Digital AC Monitor measures the current and voltage at the service entrance and continuously calculates real and reactive power on the two separate legs. Every half second, the average value of these measurements over the half-second period is transmitted to the HP9845B. In addition, the average line voltage is transmitted. (The half-second period was determined empirically; the HP could not always process the transitions at a higher sampling rate when the load was active.) The power four-vector is then normalized to 125 V to correct for utility line-voltage variations, as described in Section 3.1.1. (This is equivalent to using conductance and susceptance as the signature components.) This four-vector is then passed through the spike-passing edge detector described in Section 3.3 to determine the residence transitions. The signatures, changes in admittance (or normalized power), are then compared with a table which describes the regions of Fig. 4-1. If a region is found which includes the signature or its negative, then the residence state vector (a list of ON or OFF status for each appliance) is updated. If the signature is not found, two special checks are performed. One allows the region to be expanded temporarily if the appliance was ON and is now turning OFF. The other checks if the signature can be the sum of two appliance signatures for any combination of the appliances, as described in Section 3.5.

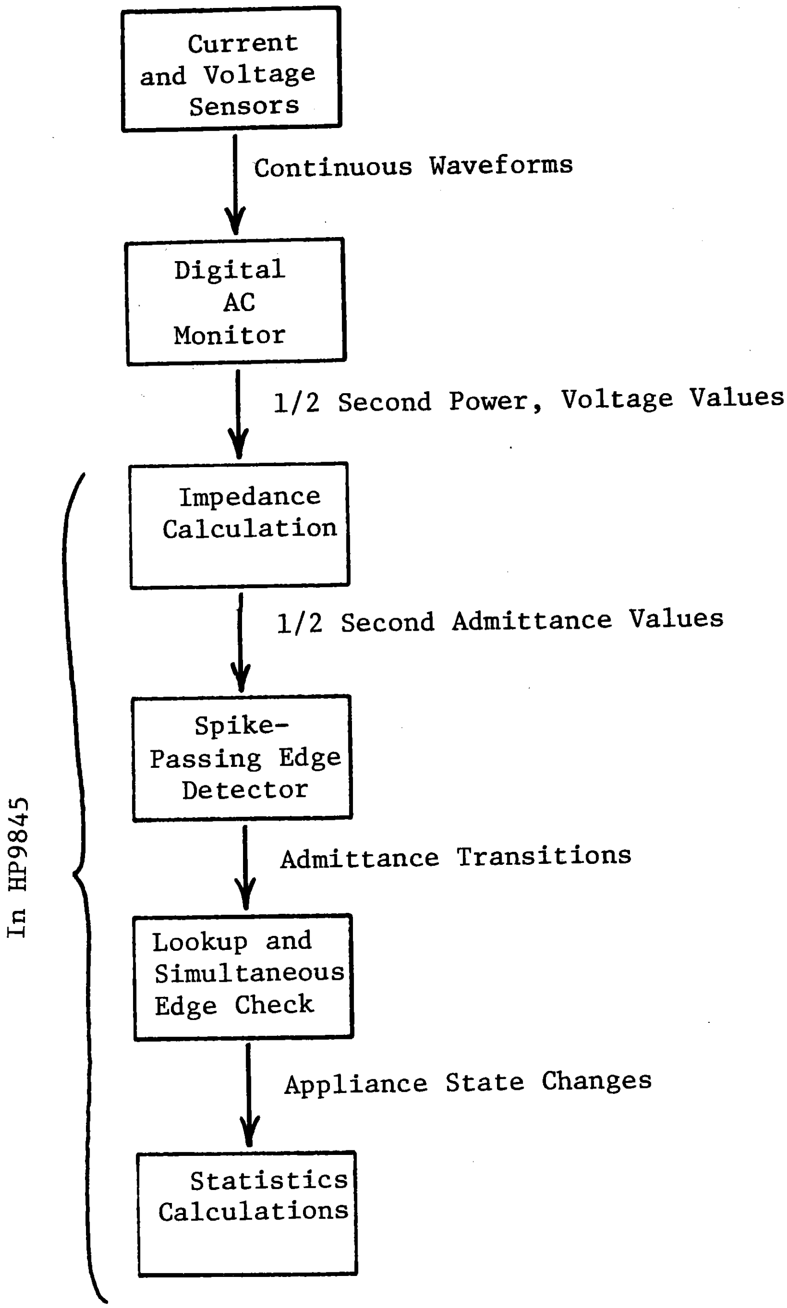


Fig. 4-2. Informations flows in recognition program.

Elapsed Time: 68:57:30

APPLIANCE	#ON	#OFF	KWH	CORRECTION VECTOR				DUR.	START	%ON
REFRIGERATOR	184	182	11.5	-.362	-.0694	-14.6	-2.52	9.95	1.85	43.8
FURNACE	48	48	.885	.411	-.137	.0761	-19.6	2.94	1.98	3.34
VACUUM										
FAN										
BLOW DRIER	3	3	.113	0	0	-21.4	-3.34	1.64	1.67	.119
AIR COND.	1	2	.427	0	0	8.27	-8.36	37.7	3	.911
STOVE (sm)	205	205	1.87	-9.85	10.6	-21.4	13.1	.313	.959	1.54
STOVE (lrg)	1	1	.0031	37.5	-7.9	2.5	4.25	.0667	1.5	.0016
OVEN (bake)										
OVEN (broil)										
TOASTER TOP	21	21	.249	0	0	-3.04	-.655	.727	.143	.369
TOASTER BAKE	19	19	.167	0	0	-.263	.682	.643	0	.296
FRYING PAN										
IRON	14	12	.101	-23	-10.6	.762	0	.374	1.32	.108
DRIER	6	7	1.69	-82	-15.7	-65.8	-39.1	2.48	11.8	.36
B.R. LIGHT	29	28	.851	.093	.0491	.458	-2.78	11.8	1.07	7.96
BUG ZAPPER	3	2	.579	.2	0	-14.2	7.78	158	1.83	7.64

30.5% Energy   94.5% Transitions   39 Pairs   8 Near   62 Misses

Fig. 4-3. Output of recognition program.

The computer CRT displays the status of the appliances, the statistics of their usage, and the statistics of the algorithm's success. Figure 4-3 is a printout of the CRT display. The notion of an appliance <u>cycle</u> is important in interpreting the statistics. The algorithm considers an appliance to have cycled when it detects the appliance turning OFF, and the most recent previous activity for the appliance was to turn ON. The time between the consecutive turn-OFF and turn-ON is the <u>cycle time</u>. The top line of the printout indicates that the statistics are cumulative over an elapsed time period of about three days (69 hours). The bulk of the printout lists individual appliance statistics. The columns are interpreted as follows:

<u>Appliance</u>: This is the common name of the appliance. Note that the program did <u>not</u> determine these names. The identification problem has not been attempted. The names were specified by the programmer.

# On:	This is the number of times the algorithm determined that the appliance turned ON. The actual number of times may have been slightly higher if some transitions were not properly interpreted due to noise or some other problem. Lines with this and other entries blank indicate that no activity was detected for that appliance.
# Off:	This entry indicates the number of times the algorithm determined that the appliance turned OFF. If the program were 100% accurate, this entry would be equal to the number of ON transitions. The discrepancies of some entries are discussed below.
kWh:	This is the total energy used by the appliance during observed cycles, in kilowatt-hours. It is calculated by multiplying the average power at the center of the appropriate rectangle of Fig. 4-1 by the sum of the cycle-times for the appliance.
Correction Vector:	This is a four-component signature space vector that could be used to correct the rectangle of Fig. 4-1. It is calculated by subtracting the center of the specified rectangle from the average of the signature vectors which were observed for the appliance. Thus, if this vector is added to the corresponding SSR, the resulting region would be exactly centered on the observed transitions. This has been calculated with the intention that in the final algorithm the correction would be automatically applied to track drift in appliance properties, as discussed in Section 3.13. This program does not use it for that purpose; it only displays it for informative reasons.
Duration:	The average observed cycle time, in minutes, is displayed.

Start: The start column displays the length of time, in seconds, that the peak-passing edge detector took to go from steady state before the transition, to steady state after the transition. This was examined with the intention that it could be a signature component, since some appliances would take more time to reach steady-state than others. It was discarded, however, as being too variable within a single appliance when repeated turn-ons were observed.

% On: The final column of Fig. 4-3 displays the percentage of the total elapsed time that the appliance was observed to be in a cycle.

Some of the statistics of Fig. 4-3 deserve addressing; the appliances are discussed line by line. The refrigerator is by far the largest energy user in the house, by almost an order of magnitude. The program missed two turn-offs, and hence two cycles, so the actual energy use would have been slightly higher. It is not clear why the two turn-offs went unrecognized. Perhaps unmodeled appliances, such as the dishwasher, washing machine, or lighting, happened to change state at the same time. An unmodeled defrost cycle is also a likely candidate for explaining the discrepancy. The furnace was recognized with complete accuracy; it was ON for 48 cycles, maintaining hot water. Although the blow drier was correctly recognized to have one cycle per morning during the three-day period, this is partly an artifact of the author's use of the appliance. It has four separate heat/fan settings, and only the commonly used one is modeled by the SSR. If other settings had been used, errors would have occurred. A more complex ARS is needed to model this multistate appliance. The air conditioner is a similar case. One turn-ON was missed because only one of 3 speeds is modeled in the program, and transitions through the other operating levels result in recognition errors. The two burners of the stove were recognized with high accuracy. The single short (4-second) cycle of the large burner is accurate. It was turned ON mistakenly and immediately turned OFF.

The toaster oven contains two elements which are listed here as separate appliances. The controls are such that both elements are ON for bake or toast settings, but only the upper one is ON for broil. The bake and toast states are therefore good tests for the special test in the algorithm that detects simultaneous changes in two appliances. Each of the 38 times that the toaster turned ON or OFF resulted in a transition which is not listed in the residence model. When this occurs the algorithm tries two special checks, one of which is to see if the transition can be the sum of two appliance transitions which happened to occur at the same time. If a unique pair can be found, the algorithm updates its tables as if they happened separately. An exception here is that the start column, which records the time required to reach steady state, is not updated because the two separate start times can not be determined from the simultaneous transition. This accounts for the start entry of zero for the bake (lower) element. The top element does have a start entry because it was ON twice, on broil, without the lower element.

Two turn-offs of the iron were missed because the rectangular region was not centered properly. The coordinates in Fig. 4-1 were measured by quickly turning the iron ON and OFF without giving it time to heat up. The turn off of the hot iron is actually from a lower power level, due to a resistance change. Note how the first entry in the correction vector, a negative power component, indicates that this change is required. The first turn-on of the drier was missed because it includes the motor start-up and is outside of the region in the table. It is not known why the bathroom light and bug zapper turn-offs were missed; other simultaneous events resulting from unlisted appliances is the most likely reason.

The bottom line of the printout in Fig. 4-3 summarizes the program's success statistics. The values shown are typical for similar periods in which the program has operated. Of all the transitions in which the power components of the two legs summed to over 200 W, 94.5% were identified as belonging to one of the seventeen appliances. We believe that no misidentifications occurred, although this can not be known for certain for some of the appliances. This results in a 94.5% transition percentage, according to the method described in Section 2.6. Even though the vast majority of transitions were identified, only 30.5% of the energy consumed by the house was accounted for by the program. This difference is in part due to the fact that the Digital AC Monitor and HP9845B computer together consume 250 W continuously, which accounts for 28.8% of the energy use during the period. The program therefore accounted for 42.8% of the actual load. Most of the remaining discrepancy is due to appliances which are not included in the seventeen listed.

The entry "39 Pairs" of the statistics line indicates that otherwise unrecognized transitions were resolved into two simultaneous transitions 39 times. Thirty-eight of these resulted from the toaster. The other involved the refrigerator and the small burner.

The entry "8 Near" indicates that unrecognized transitions were accounted for by a second type of special test on eight occasions. This test has the effect of somewhat expanding the rectangle. The program allows a rectangle to be expanded by 10% to account for occasional "noise." The noise is expected to be the simultaneous transition of a small, unmodeled appliance. When an unrecognized transition occurs, the algorithm checks if the appliance in the signature space which is closest to the observed transition (using the Euclidean metric) is in the ON state. If so, then the algorithm allows the rectangle to be expanded by up to 50%. This is motivated by the observation that if there is good evidence of a large appliance being ON and a similar but unrecognized large appliance going OFF, then it is likely to be the same appliance with noise added.

To conclude this section, we note that accuracy statistics over 90% are quite typical for the multiday time periods during which this program has operated. The percent of recognized transitions would have been higher if the unmodeled washing machine and dishwasher had been left idle during the period.

5.0 CONSTRAINTS ON THE FINITE STATE MACHINE APPLIANCE REPRESENTATION SYSTEM

The FSM ARS presented in Section 2.2.2 has too much descriptive power to be applied in the non-intrusive appliance load data acquisition algorithm. This section gives several examples demonstrating the problems that arise, and then suggests some reasonable constraints which do not limit the applicability of the system to practical appliances. There are two ways in which an ARS can be demonstrated to be underconstrained. The first way is that it can be underconstrained in principle. The second is that it can be underconstrained in practice. To show tht the ARS is underconstrained in principle, we show that it is ambiguous. By this we mean that two distinct models of the household appliance inventory can always be given which are indistinuishable from any data available at the service entrance. This section presents a series of counter-examples and arguments which show that that the FSM ARS is underconstrained in principle. The ideal situation would be to be able to prove that some set of constraints, such as those presented below, were sufficient so that only one set of FSMs could describe any given residence, and that any other set would be invalidated by the residence transitions given enough time. Such a proof is not yet at hand.

Even after constraints are developed which reduce the FSM ARS to an ARS that is no longer underconstrained in principle, it is likely that it would still allow too many possibilities for any learning algorithm. If this occurs, then we say that the ARS is underconstrained in practice, although it is not clear that it will occur or that we could prove it to be the case if it did. In this case, additional constraints would have to be proffered in order that an algorithm could be developed. As an example of such constraints, this section ends with a highly constrained sample FSM ARS which only allows a small number of state connection topologies. Further constraints of this nature are not considered here, as we have not reached a point in either the constraining of the FSM ARS or in the development of learning algorithms where it need be considered.

A simple example which shows that the FSM ARS is underconstrained in principle is the one-state FSM of Fig. 5-1. This FSM models the entire residence as a single appliance which is always ON. Any observed transition leads the appliance back to the ON state. This model will always fit the data in a trivial manner. There are many ways in which the ARS could be constrained so that this is not a valid appliance model. One approach is to put a maximum on the number of transitions which can enter or leave any state or to set a maximum total number of transitions for any single appliance. These constraints do not get to the essential difficulty with Fig. 5-1; however, such methods may be necessary at some point to eliminate problems in which an ARS is underconstrained in practice. A reasonable constraint to impose on the FSM ARS is that no transition lead from any state back to the same state. This is reasonable because if a signature is worth noting, it must be because the appliance changed state. If the appliance does not change state, the signature can be ignored.

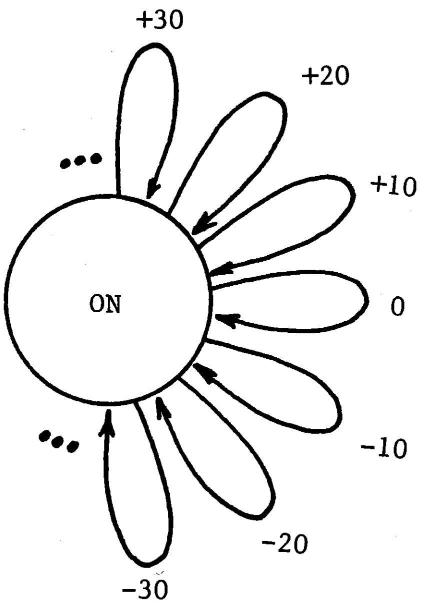


Fig. 5-1. One-state whole-house FSM.

A second constraint which would just as easily eliminate the FSM of Fig. 5-1 is to require that every FSM contain at least two states. This is reasonable because if the appliance can not change from one state to another, then it can produce no signature worthy of note and would not be detected. This is not to say that there can not be an appliance which exhibits only one state. A doorbell transformer might be wired so that it is never OFF. However, the properties of such appliances can not be determined by a nonintrusive algorithm.

A second example which demonstrates the need for a constraint in principle is given in Fig. 5-2. In this example, every observed transition from the time the algorithm begins creates an additional state. This one appliance models the entire house, which is always in the most recently created state. This FSM can be eliminated in practice by setting a maximum number of states that one FSM can contain. (This does not solve the problem, however, because when the maximum number of states was reached, a second appliance could be created, leaving the first appliance in its last state, then a third, and so on. A practical limit on the number of appliances would stop this eventually, but this does not get to the heart of the problem.) A reasonable constraint which would eliminate this from the ARS is to require that all appliance models be fully traversable. We define this to mean that there be a path of transitions from any state of any appliance model to every other state of that model. As a consequence of this definition, there can be no dead ends or unbounded open chains, and every state must have at least one exit and one entrance. This definition also implies the existence of circuits through every state. A circuit is a path of transitions starting at one state, proceeding through other states, and returning (for the first time) to the starting state.

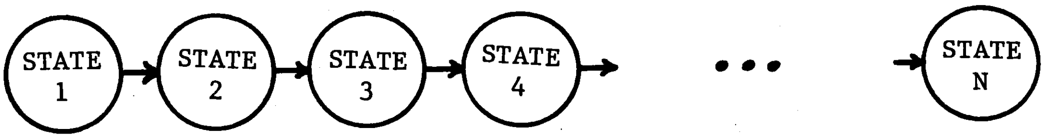


Fig. 5-2. Infinite chain FSM.

The next constraint which we present concerns zero-vectors as transition signatures. Consider any transition in any FSM, such as the one labeled X, between states A and B in the left-hand side of Fig. 5-3. Such a transition can always be replaced with two transitions and an intervening state. The transition X becomes the sequence 0, A', X in the right-hand side of this example. From the point of view of a recognition algorithm, these two FSMs are indistinguishable, because the zero-vector transition can always be located. As this would result in an unacceptable ambiguity, we conclude that zero-vectors as transition signatures must be eliminated in principle. This is a perfectly reasonable constraint given the purpose of the FSM representation. If no signature was detected, then there is no reason to conclude that any appliance changed its state. If we label the transitions with regions of signature space, rather than single points, then the constraint is that the region can not contain the zero vector.

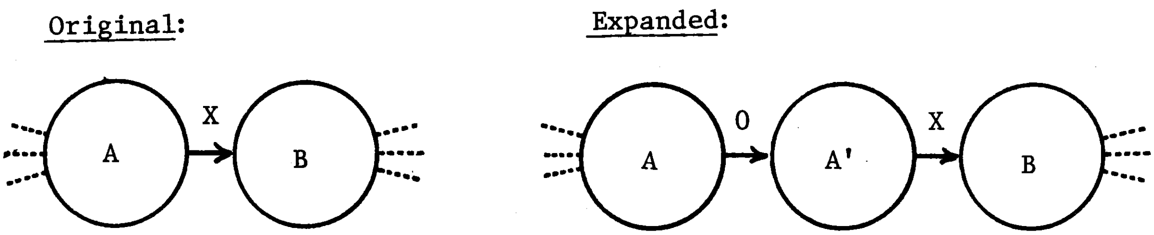


Fig. 5-3. Zero-vector expansion of transition X.

We next consider the example of Fig. 5-4, which is analogous to Fig. 5-1, but contains two states. This one appliance FSM can trivially model an entire appliance inventory. The FSM simply alternates between the two states. No constraint presented so far eliminates this FSM, so an additional constraint is necessary. To develop this constraint, we first consider the subspace of the signature vector-space which contains only the steady-state components, as defined in Section 3.1. Real and reactive power are examples of components that could be in this subspace. Although it is not stated explicitly in the FSM diagram, we can associate with each state of an appliance model a power level which the appliance draws when operating in that state. Most, but not all, appliances have a state we would label OFF in which the power level is zero. Any transition out of the OFF state is accompanied by a signature in which the power component increases to the level associated with the state that is entered. In general, the power component of a transition is equal to the difference between the power components of the operating levels of the two associated states.

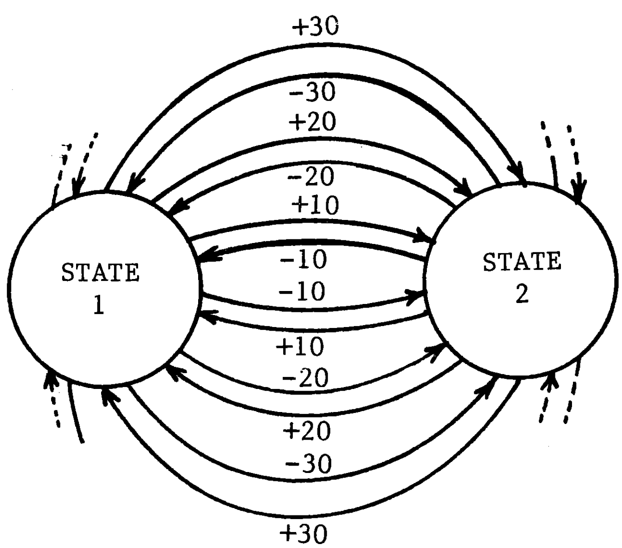


Fig. 5-4. Two-state whole-house FSM.

Generalizing further, we associate with each state of an FSM an operating level which is a point in the steady-state subspace of the signature space. The projection of any transition signature into this subspace is then the difference between the operating levels of the state entered and the state left. This difference operation is the discrete analog to the continuous process of taking the gradient of a scalar potential. It results in a path-independent conservative law for the transition vectors, analogous to Kirchhoff's law of voltages around circuit paths. <u>The projection into the steady-state subspace of the sum of the transition signatures around any circuit of any appliance model is zero</u>. This constraint eliminates the two-state whole-house FSM of Fig. 5-4 from our ARS.

In reality the situation is not quite as clear-cut as the above paragraph indicates. One complication arises from the fact that SSRs, rather than points, should label the transitions. This results in the much weaker constraint that the circuit-sum be a region around zero, rather than zero exactly. To be useful, we would like to keep the region as small as possible. A second complication is that we might want a state to represent a range of operating levels, as in the case of the appliances discussed in Section 3.8 in which the OFF to ON transition is not the negative of the ON to OFF transition. This will also require a loosening of the zero criterion. Ramps also introduce a problem here.

The next constraint which we propose involves notions of expanding and collapsing FSMs. Any FSM can be <u>expanded</u> in an infinite number of ways into more complex FSMs which accept the same set of transitions. For example the two-state FSM of Fig. 5-5a can expanded into the six-state FSM of Fig. 5-5b. In this case the FSM has been tripled. The general process involves repeating the FSM a given number of times and then permuting the transitions between the repeated FSMs. Figure 5-6 demonstrates the general expansion process. The arbitrarily selected FSM of Fig. 5-6a has been repeated three times in Fig. 5-6b. The transitions can then be permuted in many ways; one is given in Fig. 5-6c.

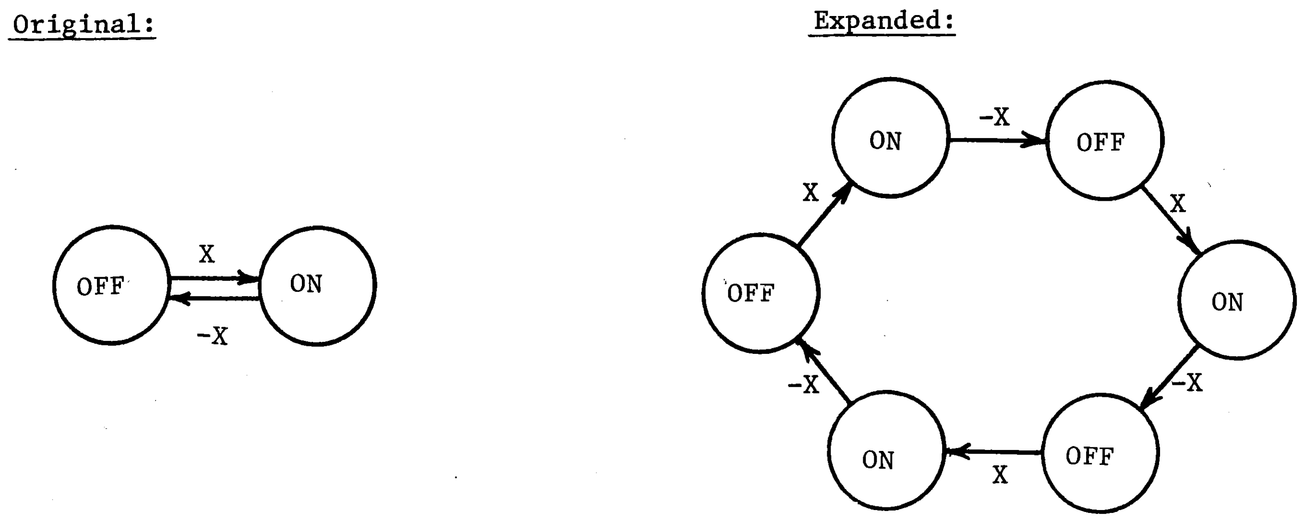


Fig. 5-5. Six-state expansion of two-state FSM.

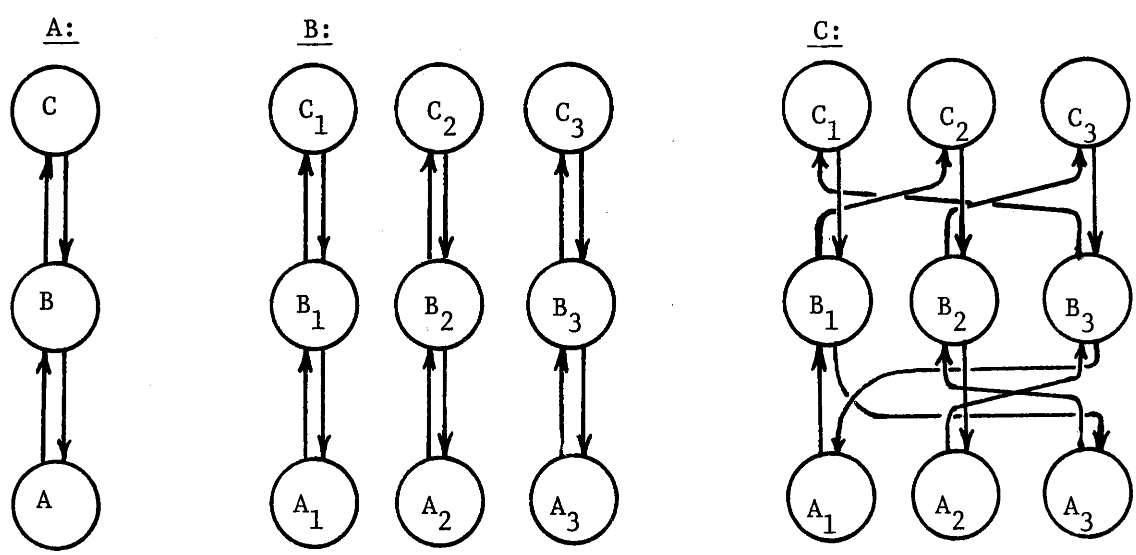


Fig. 5-6. Example of generalized expansion process.

We call the inverse procedure to expansion collapsing an FSM. If expansions of FSMs were allowed in the residence model, then the resulting ambiguity would cause the ARS to be underconstrained in principle. We therefore require that all FSMs not be collapsible. We note in passing that an algorithm for collapsing an expanded FSM is straightforward.

The final constraint which we propose here requires the development of an algebra of FSMs and a notion of prime and composite FSMs. Consider the two separate appliances of Fig. 5-7. The appliance models, which we call F and G, have two and three states, respectively. A residence model with these two separate FSMs is equivalent in terms of signature transitions to a model with the single six-state FSM of Fig. 5-8. We define the FSM of Fig. 5-8 to be the product of the FSMs F and G. Although formally simple, it is not necessary to present the general definition of FSM product here. It should be clear that the product of F and G has one state for each pair of states, one from F and one from G. The transitions of the product are merely the transitions of the factors, repeated as many times as there are states in the other factor. Composite FSMs can always be laid out on a rectangular grid such as Fig. 5-8 with no "diagonal" transitions. To remove the ambiguity that product FSMs introduce in an appliance inventory, we require in principle the constraint that all FSMs be prime, i.e., not factorable within the ARS.

We note in passing that the factoring of composite FSMs is computationally straightforward. If a composite FSM were detected, it could be factored into FSMs for the independent appliances which it represents. The factoring operation could also be used in the breakdown approach to the learning problem described above in Section 2.4.

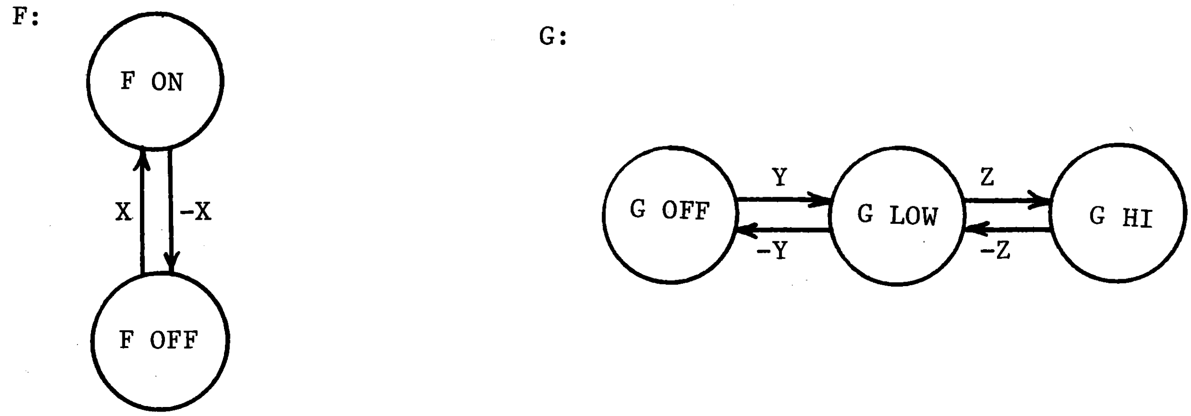


Fig. 5-7. FSMs for appliances F and G.

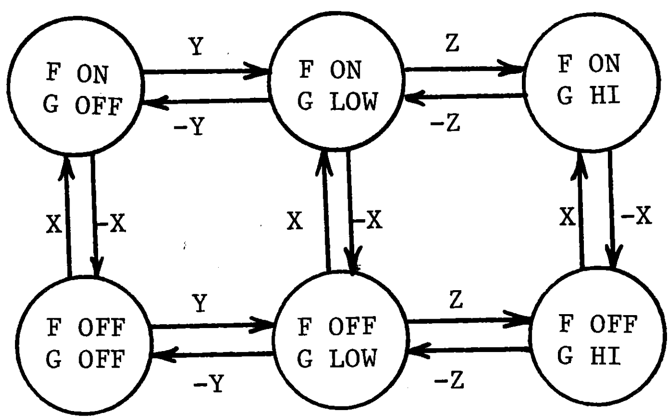


Fig. 5-8. FSM product of F and G.

We summarize now the constraints in principle which have been presented for restricting the FSM ARS. Note that all the appliance models presented in Sections 3 and 2.2.2 satisfy these criteria.

- No transition can begin and terminate at the same state, (This is now derivable from other constraints).
- Each FSM must contain at least two states.
- Every FSM must be fully traversable.

- No transition signature can be the zero-vector (or a region which contains the zero vector).
- The projection into the steady-state subspace of the sum of the transition signatures about any circuit is the zero-vector (or a region which contains the zero-vector).
- No FSM can be collapsible.
- All FSMs must be prime in the ARS.

It is not clear at this juncture exactly how constraints of this sort would be implemented in the learning algorithm. Most likely, they would not be stored as constraints and used as "filters" to eliminate hypothesized FSMs created by the algorithm. Instead, it would be much more effective to have the structure of the algorithm be such that it could only generate FSMs which satisfy the constraints.

In addition, practical constraints limiting the number of states, transitions and appliances will be required by the finite memory of any physical computer, and should be useful for eliminating unwieldly residence models.

As an example of a highly constrained FSM ARS which may be of practical value, we are considering the set of seven FSMs in Fig. 5-9. A set like this, which includes the two-state FSM and three transition topologies for FSMs with between three and some small fixed number of states, should be sufficient for most appliances. (The three transition topologies were presented in Section 3.10.) The transition signatures would have to be restricted so that the above constraints hold. The actual utility of this set will only be determined by experiment. To be of value, a learning algorithm must be developed which can not only handle appliances which fit these templates, but in addition, is not adversely affected by the existance of appliances which do not fit.

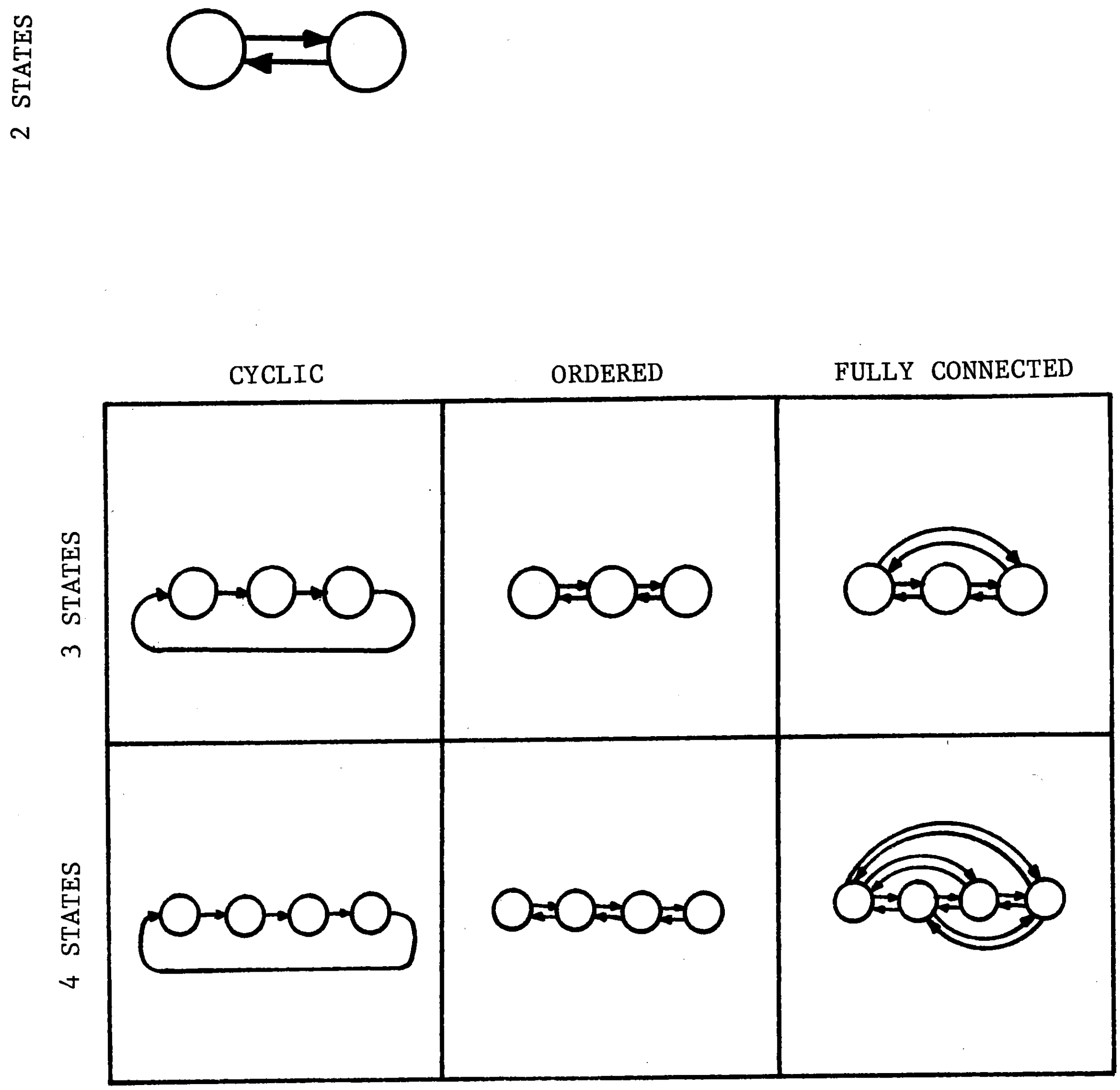


Fig. 5-9. Highly constrained FSM ARS allowing seven arrangements.

6.0 APPLIANCE USAGE DATA MARGINAL VALUE

Any nonintrusive appliance monitoring method is certain to fail for appliances below some size threshold. This is a consequence of the proliferation of appliances at low power levels and problems of measurement and noise. The exact level at which a method fails (for any given residential appliance mix) will depend upon details of the algorithm. Additional effort may improve an algorithm to identify a wider range of appliances. The decision to expend this extra effort requires some knowledge of the value of the additional data that may be collected.

To approach this question, we look at the annual energy consumption of typical household appliances compared with their operating power usage. Table 6-1 lists this data for 70 appliances as calculated from data in References 1 and 2. The first column of numbers contains the operating power usage for a typical instance of the appliance type. When interpreting this data, it must be noted that individual appliances will vary widely from this data. In the case with appliances with more than one ON state, the entry is generally the maximum power usage. For most appliances there is only one ON state, so the given power level is equal to the size of the power transition which must be detected in order to recognize the appliance state changes. The second column of the table gives the average energy consumption on an annual basis for a single instance of the appliance; particular residences will vary from this average, of course. This data is presented in graphical form in Figure 6-1. (Only appliances which consume over 0.5 kWh/day are labeled.)

Table 6-1 Appliance Power Usage and Energy Consumption

APPLIANCE	POWER (Watts)	ENERGY EACH (kWh/Day)	ENERGY/RESIDENCE (kWh/Day) U.S. Avg
Electric Heating	25000	18.03	2.698
Electric Range and Oven	12200	1.74	0.579
Separate Oven	7800	0.74	0.093
Heat Pump	5000	32.39	3.263
Electric Clothes Dryer	4856	2.83	1.219
Water Heater (Quick)	4474	13.18	1.713
Central Air Conditioner	3600	5.88	1.540
Cooktop Range	3600	1.00	0.120
Electric Water Heater	2475	11.56	1.502
Fondue Set/Wok	1448	0.23	0.013
Portable Heater	1322	0.48	0.133
Popcorn Popper	1250	0.00	0.000
Dishwasher	1201	0.71	0.284
Toaster Oven	1200	0.50	0.108
Waffle Iron	1200	0.06	0.018
Coffeepot	1200	0.38	0.287
Griddle	1196	0.27	0.028
Skillet	1196	0.27	0.153
Toaster	1146	0.11	0.096
Broiler/Roaster	1140	0.23	0.043
Swimming Pool Pump	1008	5.50	0.238
Microwave Oven	1000	0.73	0.028
Room Air Conditioner	860	1.71	1.129
Food Grinder	720	0.06	0.003
Stationary Metal Working	696	0.00	0.000
Vacuum Sweeper	630	0.13	0.153
Hair Dryer	600	0.07	0.040
Portable Woodworking	556	0.02	0.028
Clothes Washer	512	0.24	0.210
Garbage Disposal	445	0.02	0.001
Decorative Lights	400	0.07	0.017
Trash Compactor	400	0.14	0.001
Refrigerator/Freezer	375	4.15	4.464
Attic/Exhaust Fan	370	0.80	0.449
Garden Tools	360	0.00	0.000
Slide/Movie Projector	360	0.01	0.003
Blender	300	.00	0.002
Hot Tray	300	0.04	0.007

Table 6-1 Continued

APPLIANCE	POWER (Watts)	ENERGY EACH (kWh/Day)	ENERGY/RESIDENCE (kWh/Day) U.S. Avg
Sun Lamp	279	0.04	0.004
Humidifier/Dehumidifier	217	0.74	0.250
Freezer	200	3.58	1.361
Garage Door Openner	200	.00	.000
Crockpot	200	0.10	0.032
Vaporizer	177	0.45	0.142
Electric Blanket	177	0.40	0.151
Refrigerator	170	3.60	0.363
Flood Lights - Manual	150	0.03	0.014
Color TV	145	2.18	2.044
Mixer - Regular	127	0.01	0.002
Mixer - Portable	127	0.01	0.003
Home Entertainment Ctr.	109	0.30	0.022
Console Stereo, HI FI	109	0.30	0.245
Black and White TV	100	1.26	1.053
Electric Knife	92	0.02	0.009
Portable Fan	88	0.12	0.091
Other Shop Tools	85	0.00	0.000
Electric Lighting	75	2.52	2.520
Sewing Machine	75	0.03	0.021
Table Radio	71	0.24	0.361
Heating Pad	65	0.03	0.002
Juicer	60	0.00	0.000
Massager/Vibrator	40	0.01	.000
Curling Iron	40	0.00	0.000
Knife Sharpener	25	0.00	0.000
Opener/Sharpener	25	0.00	0.000
Tape Deck	25	0.00	0.000
Can Opener	25	0.00	0.000
Shaver	15	.00	.000
Electric Clock	2	0.05	0.100
Toothbrush	1.1	.00	.000

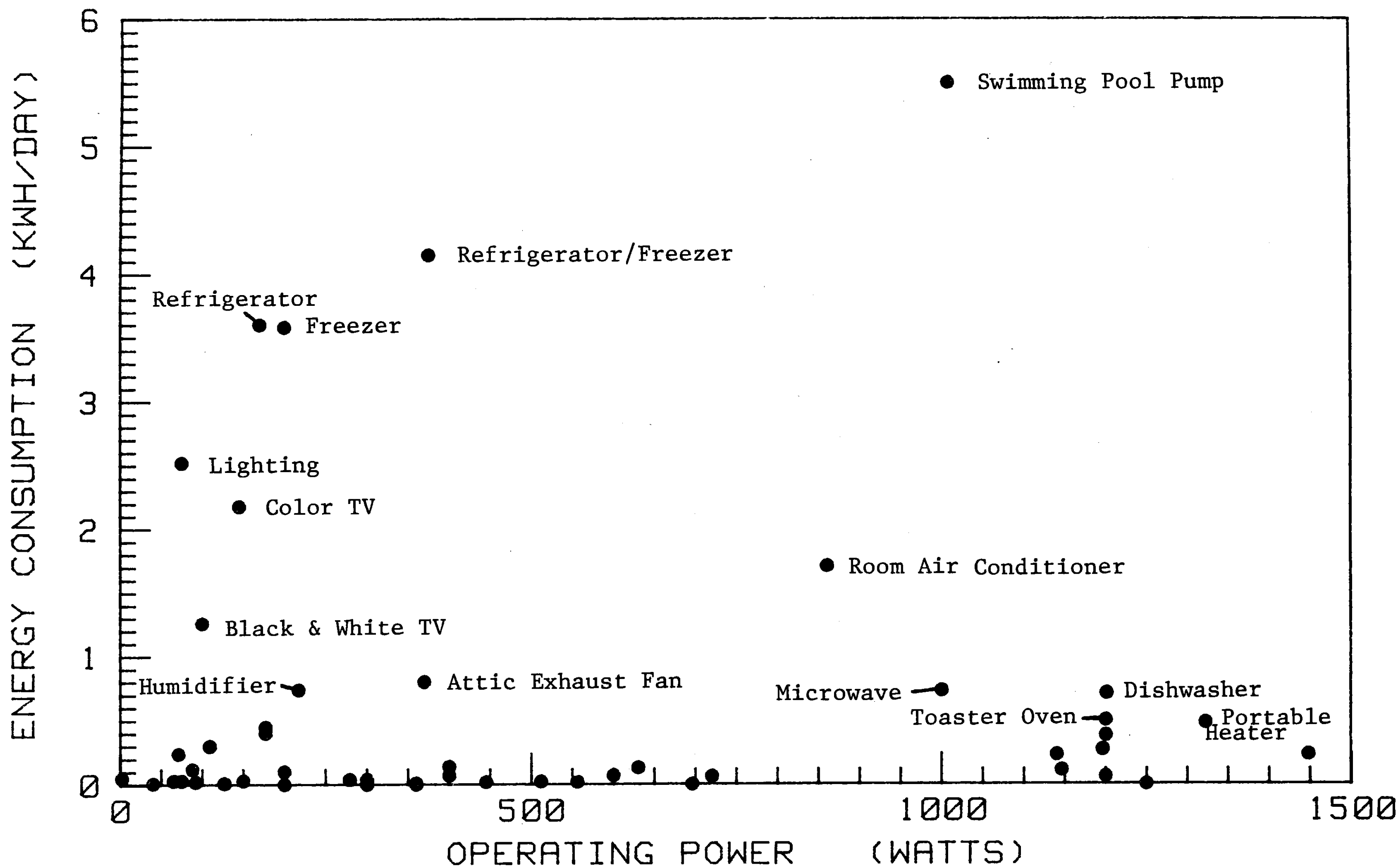


Fig. 6-1. Appliance annual energy consumption vs. power usage.

In order to use this data meaningfully, the energy consumption must be weighted according to how many of each appliance are found in the average U.S. residence. The third column of Table 6-1 specifies the annual average energy consumption of each appliance weighted (multiplied) by the average number of appliances per household in the United States. For example, the first appliance entry, electric heat, typically uses 25 kW when turned on. In the average house with electric heat, 18.03 kWh/day is consumed by the heater as an annual average. Because only 15% of U.S. houses have electric heating, it accounts for 2.698 kWh/day in the average U.S. house. In some cases (e.g., refrigerator/freezer and table radio), the average house has more than one instance of the appliance, so the U.S. average energy consumption per residence is greater than the energy consumption per appliance.

A comparison of the first and third columns of Table 6-1 can be used to estimate the portion of the U.S. residential electric energy consumption which can be identified by a nonintrusive method. This estimate is based on the assumption that the limiting factor which ultimately prevents identification is the size of the power transitions that can be recognized. For example, the total energy which can be identified if all transitions over 200 W are recognized is the sum of the energy consumption of all appliances with power usage greater than 200 W. This turns out to be slightly over 70 % of the total based on the data in Table 6-1. Figure 6-2 presents this energy vs. power relationship in a cumulative form. The abscissa ranges over appliance operating power levels. The ordinate indicates the percentage of the total annual residential energy usage which is consumed by appliances equal to or larger in size than the abscissa. From this we see that if the nonintrusive method can be refined to the point where it includes lighting (plotted at 75 W), then on the average, 98% of residential energy usage could be accounted for. The relative steepness of the curve between 0 and 200 W suggests that the use of current harmonic signatures for the recognition of smaller appliances may be worth the effort (see Section 3.1.2).

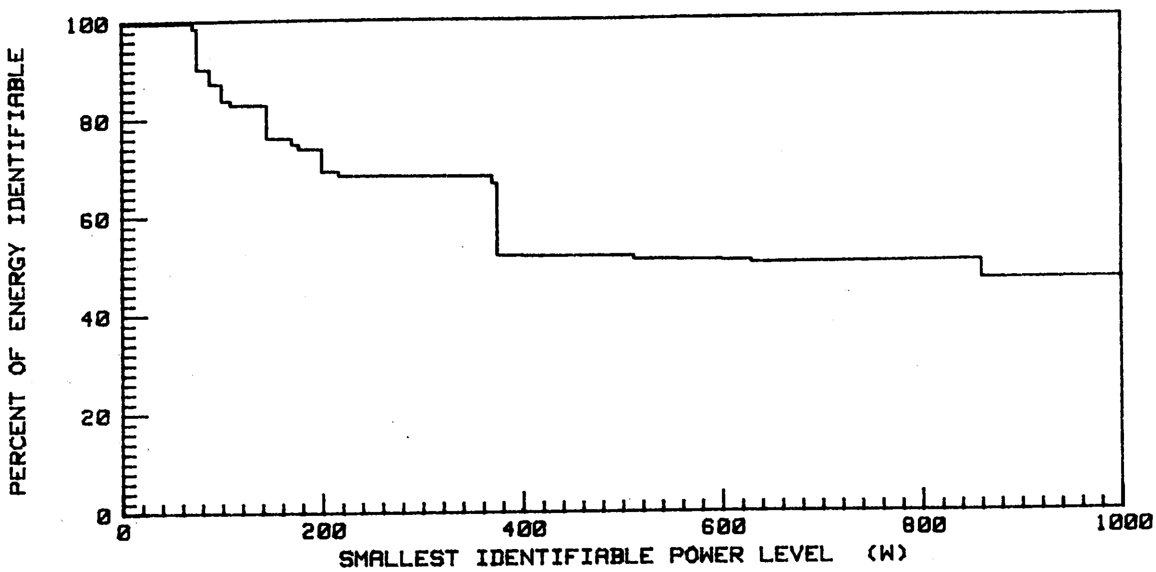


Fig. 6-2. Effect of minimum power threshold on identifiable energy.

Although this analysis suggests that a large fraction of the total energy is identifiable, this contrasts with the results of the case study presented in Section 4, in which less than 50% of the energy was identified. The difference is attributable to weaknesses of that algorithm, the small number of appliances considered in Section 4, and differences between the particular appliance inventory and usage patterns of the residence and the U.S. average.

<u>References</u>:

[1] <u>Statistical Abstract of United States</u>, 1982-1983, Number 1359, p. 758.

[2] "Annual Energy Requirements of Household Electrical Appliances," Edison Electrical Institute, Report #75-61.

APPENDIX. Extracts of Digital AC Monitor Operating Manual*

I. Introduction

A general purpose Digital AC Monitor with the following capabilities and features has been developed.

1. Continuous "simultaneous" measurement of up to 8 AC circuits.
2. Measures RMS voltage, RMS current, real power, reactive power, apparent power, power factor, frequency, harmonic distortion, load impedance and admittance, dc voltage, dc current. It can also separate the real and reactive power flows into or out of a circuit, analogously to ratcheted Kilowatthour and Kilovarhour meters.
3. Full programmability. Any combination of measurements can be taken on any sets of input circuits independently.
4. Programming and output is in ASCII characters over a standard RS-232 line at speeds up to 19200 baud. Baud rate, parity, stop bits and character echo are switch selectable.
5. Output is directly readable in engineering units (volts, amperes, watts, hertz...).
6. The AC Monitor can be used as a data logger or test instrument by connecting it with a computer terminal, or it can be used as a sensor peripheral to any computer equipped with a serial RS-232 port.
7. It can output digitized current and voltage waveforms showing 64 samples per cycle from which "oscilliscope tracings" of current and voltage waveforms can be reconstructed by the user's software.
8. It can be programmed to output either automatically at intervals, or when polled by the user.
9. It can be programmed either to output sampled values of the inputs or averaged values, over periods of up to 18 hours.

The AC Monitor is designed to be a flexible instrument which is cost effective in a wide range of applications where 60-Hz ac circuits are to be instrumented and the results interfaced to a computer for data acquisition or control purposes. It is designed to function as a data acquisition system, replacing analog transducers and a data logger. Although the monitor was designed with photovoltaic power systems in mind, it is an instrument of far wider applicability.

---

*Additional information is available from the MIT Energy Laboratory.

VIII. Method

Figure 4 shows a block diagram of the information flow through the essential parts of the AC Monitor. The analog to digital converter (A/D) can measure any one of the sixteen inputs via the multiplexer. The microprocessor controls which inputs are sampled, and what calculations are made, in accordance with the instructions it receives over the RS-232 interface. It also transmits it outputs via the RS-232 line.

For each circuit for which any measurement commands (other than frequency) were given, the voltage input and the current input are each sampled with the eight-bit A/D sixty-four times during a one-sixtieth of a second cycle. A single A/D converter is used, sampling alternately between the current and voltage inputs at a rate of 7680 samples (2 * 64 * 60) per second. One extra current reading is taken so that there is a current value spaced equally before and after each voltage reading. The digitized values range from -127 to +127. In the formulas below, V and I represent these individual A/D readings. An A/D reading must be multiplied by a conversion constant, C, to obtain the actual input voltage.

$$C = \frac{5}{128}$$

The dc voltage (and current) is calculated by simply averaging the 64 sampled values.

$$VD = \left[\frac{C}{64}\right] \text{AVG} \left| \sum V \right|$$

The function, AVG $\left|\ldots\right|$ represents the averaging which occurs between outputs. The part of the computation between the braces is performed repeatedly (as each cycle is measured). When it is time to output results, the average of these computations is determined and multiplied by the conversion constant in the brackets to determine the average dc voltage in units of volts. This is then converted to decimal and output.

The RMS voltage is computed as:

$$VR = \left[\frac{C}{\sqrt{64}}\right] \text{AVG} \left\{ \sqrt{\sum V^2} \right\}$$

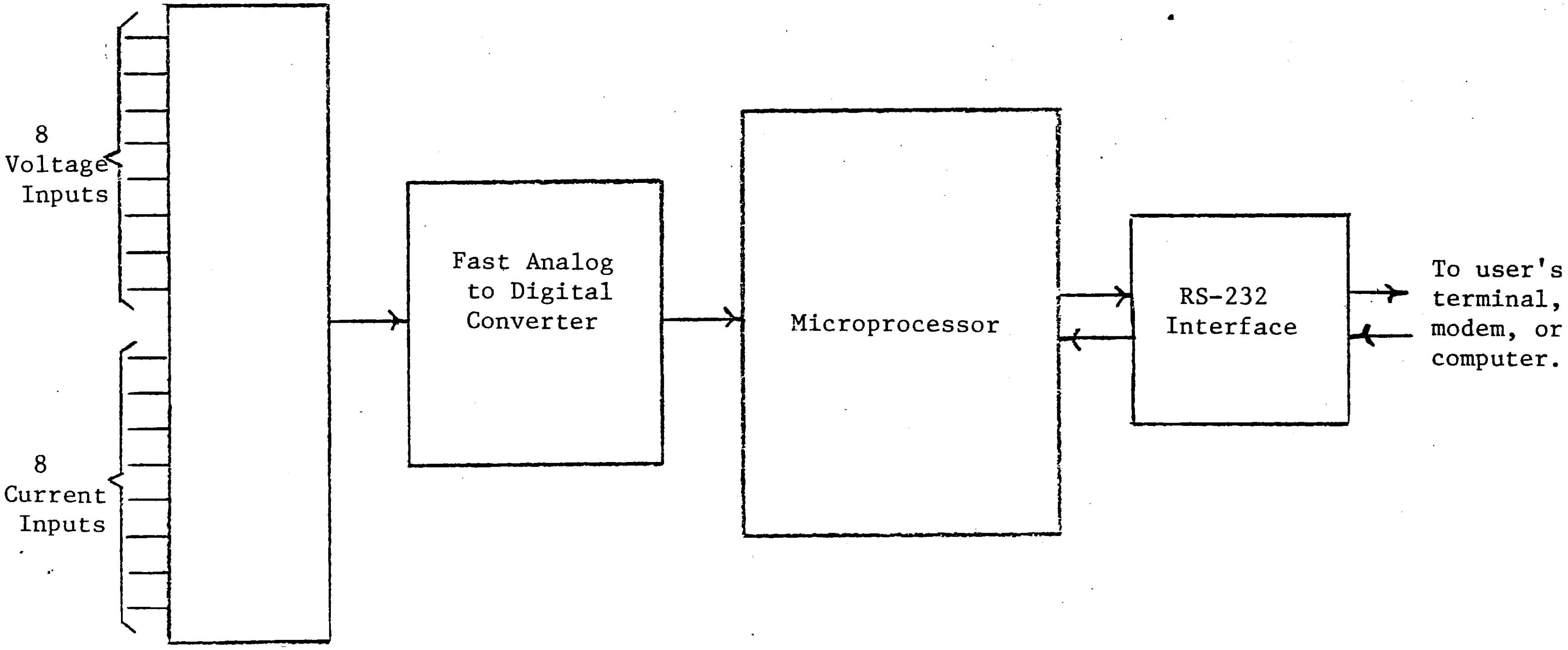


Figure 4 Digital AC Monitor Block Diagram

Note that this results in a true RMS reading, whatever the waveform shape, taking into account thirty harmonics. The RMS current is similar. It uses only sixty-four of the sixty-five sampled values.

To compute the power, an integral of I*V is performed. The current value at a time synchronous with any voltage value can be approximated by averaging the current values immediately preceding it and following it. The division by two required in the averaging is factored out so power is computed as:

$$PR = \left[\frac{C^2}{2*64} \right] \text{AVG} \left\{ \sum V_i \left(I_{i-\frac{1}{2}} + I_{i+\frac{1}{2}} \right) \right\}$$

Note that in the presence of harmonic distortion this gives the total real power at all frequencies (up to the thirtieth harmonic).

Power into a circuit is calculated from the real power by treating negative powers as zero. We define a positive power function, f^+, by

$$f^+(x) = \left\{ \begin{array}{ll} x & \text{if } x>0 \\ 0 & \text{if } x<=0 \end{array} \right\}$$

Then

$$PI = \text{AVG} \left\{ f^+(PR) \right\}$$

Power out of a circuit is computed at output-time as

$$PO = PI - PR$$

Although this is computed all at once, it gives the same results as if it were computed by averaging only the negative values of power:

$$PO = \text{AVG} \left\{ f^+ (-PR) \right\}$$

Note that if power flows into the circuit during part of the averaging period, and out of the circuit during other parts of the period, the following results hold:

$$PI > 0$$

$$PO > 0$$

$$PR = PI - PO$$

Reactive power is calculated analogously to real power, integrating the product of the voltage and the current 90° (16 samples) later:

$$RP = \left[\frac{C^2}{2*64}\right] \ AVG \left\{ \sum V_i \left(I_{16+i-\frac{1}{2}} + I_{16+i+\frac{1}{2}} \right) \right\}$$

where the subscript calculations are performed modulo 64.

This gives the reactive power at sixty hertz according to most definitions, but note that by some definitions this is not the reactive power if harmonic distortion is present. Reactive power into and out of the circuit are completely analogous to real power:

$$RI = AVG \left\{ f^{+}(RP) \right\}$$

$$RO = RI - RP$$

Frequency is computed by timing three consecutive voltage waveform cycles, starting and stopping at upward zero crossings. The algorithm requires that the voltage signal at the AC Monitor have an amplitude of at least 0.5 volts. If no signal is present, or the signal has too small an amplitude, or if the frequency is out of the 40-80 Hz range, a value of zero is output.

$$FR = \frac{3}{AVG \left\{ \text{Period of 3 cycles} \right\}}$$

Total harmonic distortion of a current or voltage waveform is calculated by first computing the RMS value of the 60 Hz fundamental, A_{60}, using Fourier integrals, and then "subtracting it out" from the RMS value of the waveform, A_{RMS}. It is then normalized to the fundamental, and multiplied by 100 to give a percentage. The calculations are equivalent to the following:

$$A^2_{60} = \frac{2}{64^2} \left[\left(\sum I_i \ SIN\left(\frac{2\pi i}{64}\right) \right)^2 + \left(\sum I_i \ COS\left(\frac{2\pi i}{64}\right) \right)^2 \right]$$

$$THD = AVG \left\{ 100 \sqrt{\frac{A^2_{RMS} - A^2_{60}}{A^2_{60}}} \right\}.$$

The calculated THD has a limited range of accuracy because the analog to digital converter only attains eight-bit resolution. The measured waveforms are digitized to the nearest 1% so distortion less than one percent can not be accurately determined. The minimum accurate THD percentage increases as the waveform amplitude decreases because the digitization step becomes a larger percentage of the RMS value, so it is important that the input signal be scaled to between 3 and 3.5 V RMS if THD is to be measured.

The six derived quantities are calculated based on the averaged values of RMS voltage, RMS current, real power, and reactive power. The following formulas use the abbreviations given in Table I.

Apparent power	AP = IR * VR
Power Factor	PF = PR / (IR * VR)
Resistance	ZR = PR / IR^2
Reactance	ZX = RP / IR^2
Conductance	YG = PR / VR^2
Susceptance	YB = - RP / VR^2

Note that these formulas are only valid for sinusoidal voltages and currents. In the presence of harmonics, for example, the calculated impedance and admittance might not be reciprocals.

The standard sign convention is followed throughout. An inductive load will be reported to consume a positive reactive power, and will have a positive impedance angle. This assumes of course that consistent input polarity was observed, as described in Section III.

Table I Measurement Summary

Measurement	Command Characters	Output Units	Derived Quantity	Min Value	Max Value	Scale Factor
DC Voltage	VD	Volts	no	-4.96	4.96	V
DC Current	ID	Amperes	no	-4.96	4.96	I
RMS Voltage	VR	Volts	no	0	4.96	V
RMS Current	IR	Amperes	no	0	4.96	I
Real Power	PR	Watts	no	-24.6	24.6	VI
"" Into circuit	PI	Watts	no	0	24.6	VI
"" Out of circuit	PO	Watts	no	0	24.6	VI
Reactive Power	RP	Vars	no	-24.6	24.6	VI
"" Into circuit	RI	Vars	no	0	24.6	VI
"" Out of circuit	RO	Vars	no	0	24.6	VI
Apparent Power	AP	Volt Amperes	YES	0	24.6	VI
Power Factor	PF	Dimensionless	YES	-1	1	none
Load Resistance	ZR	Ohms	YES	-25	25	V/I
Load Reactance	ZX	Ohms	YES	-25	25	V/I
Load Conductance	YG	Mhos	YES	-25	25	I/V
Load Susceptance	YB	Mhos	YES	-25	25	I/V
THD of Voltage	VH	Percent	no	1	100	none
THD of Current	IH	Percent	no	1	100	none
Frequency	FR	Hertz	no	40	80	none
Voltage Waveform	VW	Hexadecimal	no	-127	127*	5V/128
Current Waveform	IW	Hexadecimal	no	-127	127	5V/128

*Note that a voltage or current waveform maximum value of 127 analog-to-digital-converter-counts is reported as a hexadecimal value 7F which corresponds to 4.96 volts input. Similarly, -127 counts is 81 hex, corresponding to -4.96 volts input.