Fuzzy Logic Enters the Mainstream

Hardware and Software Application Tools Proliferate

By Bruce Koball, Motion West, Berkeley, CA

From the number of new products and strategic alliances announced this year, it would seem that fuzzy logic is entering the mainstream of microprocessor systems design consciousness, if not actual applications. Fuzzy logic pioneers like Togai InfraLogic and Aptronix have been joined recently by such major players as Hitachi, Intel, Motorola, National Semiconductor, Siemens, VLSI Technology and others in offering everything from fuzzy compilers and graphically-oriented software development environments to fuzzy processors available as logic blocks, and chip- and board-level products.

All these products are an outgrowth of work done by Professor Lotfi Zadeh in the 1960s on fuzzy set theory. The fundamental premise of fuzzy set theory is that, in contrast to classical or "crisp" sets whose elements are either members of a given set or not, the elements of fuzzy sets have varying degrees of membership in a given set, as defined by a membership function. Current fuzzy logic applications are generally implemented as rule-based systems, with the designer using linguistic variables and rules to define the relationship between the system inputs and outputs. In operation, the system performs three basic functions:

- *Fuzzification*—Crisp input values are translated into degrees of membership in some number of fuzzy sets, becoming fuzzy inputs. The membership values are generally expressed as a fractional value from 0 to 1.
- *Rule Evaluation*—Linguistic rules are evaluated, using the fuzzy inputs, deriving fuzzy outputs. The fuzzy outputs are also generally expressed as fractional values from 0 to 1.
- *Defuzzification*—Fuzzy outputs from active rules are combined and translated into crisp output values. The best method of defuzzification is a matter of debate. One widely used approach is called the centroid or center of gravity (COG) method.

In a home heating system, for example, the room temperature—expressed as a crisp value such as "55° F"—is first translated into degrees of membership in fuzzy sets like COLD, COOL, COMFORTABLE, WARM and HOT (fuzzification). Next, system rules like "if TEMP is COLD then CHANGE IN HEAT is POSITIVE GREAT" and "if TEMP is COOL then CHANGE IN HEAT is POSITIVE MODERATE" are evaluated using fuzzy sets (COLD, COOL, etc.) as inputs, deriving fuzzy sets (POSITIVE GREAT, POSITIVE MODERATE, etc.) as outputs. Finally, the fuzzy outputs are translated into crisp outputs to control the heater (defuzzification). (For a more detailed explanation of this process, see "An Introduction to Fuzzy Logic," µPR 2/20/91, p. 11–13.)

Fuzzy logic solutions currently being offered fall into two categories; software-based and hardware-based systems. Software-based systems implement fuzzy logic algorithms using standard microprocessors. While this approach is certainly slower than systems using dedicated fuzzy hardware, it is quite adequate for the vast majority of control-oriented applications. Hardware-based systems employ specialized fuzzy processors or coprocessors to accelerate the fuzzification, rule evaluation, and defuzzification functions. This approach is best suited to applications requiring very high-speed system response or with very large numbers of rules and membership functions.

Current estimates place software-based systems in excess of 90% of total fuzzy logic applications, although some vendors see hardware-based systems picking up as much as 25% of the applications within 5 years. Another provocative prognostication, offered by Motorola's Steve Marsh, is that by 1996 half of all standard microcontroller applications will use some amount of fuzzy logic code. This technology is already appearing in high-volume products, such as Panasonic camcorders using fuzzy logic for image stabilization and 1993 GM Saturn cars with fuzzy controllers for their automatic transmissions.

Software-Based Fuzzy Systems

One popular format in which fuzzy application tools are being offered is as an integrated CASE (computer-aided software engineering) development environment. Several vendors have such packages running on PC-class machines under Windows, including TILShell+ from Togai InfraLogic, Fuzzy Inference Development Environment (FIDE) from Aptronix, NeuFuz from National Semiconductor, and fuzzyTECH 3.0 from Inform GmbH. Most of these packages provide sophisticated, graphically-oriented capabilities for defining membership functions and linguistic rules, and for simulating the fuzzy system operation. Once the fuzzy system is defined, it is usually compiled and emitted as C source code or as assembly code for the target microcontroller. Most of these systems require some amount of additional coding in C or assembler, outside of the development environment, for I/O interfaces.

Some fuzzy tool vendors have struck deals with major microprocessor manufacturers to support various target processors. Aptronix and Motorola have teamed up to provide fuzzy development support for Motorola's 68HC05 and 68HC11 8-bit microcontrollers using Aptronix's FIDE. They will also be supporting Motorola's

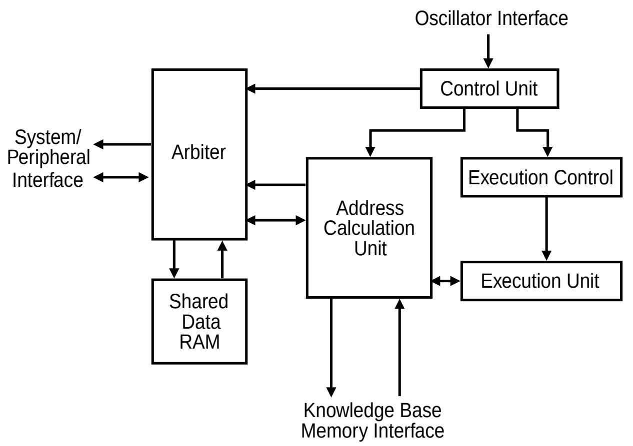

Figure 1. Block diagram of Togai's FC110 Digital Fuzzy Processor.

'HC16 and '332 families with future releases. In addition, Motorola has created an excellent, interactive Windows-based *Fuzzy Logic Education Program* that it is marketing along with a demo version of FIDE at low cost. The package is a serious educational tool, not the typical throw-away marketing demo, and should provide a designer with a working knowledge of fuzzy systems sufficient to begin useful work. Even before its collaboration with Aptronix, Motorola was pursuing fuzzy logic applications for its microcontrollers with its freeware Knowledge Base Generator (KBG.EXE) tool, supporting the 'HC05 and 'HC11.

In business since 1987, Togai is one of the senior players in fuzzy technology and has a large suite of fuzzy tools. Togai's TILShell+, used with its MicroFPL kernel and compiler, currently supports a number of microcontrollers, including Hitachi's H8/300 and H8/500 8-bit devices, its HMCS 400 4-bit parts, Intel's 8051 family, and Mitsubishi's 37450. Togai's Fuzzy-C Development System (FCDS) will generate C code for a fuzzy system defined in TILShell+ and its TILGen automated rulebase generator automates the process of defining fuzzy system rules by using neural networks to analyze the user's input data. Togai has also announced an alliance with Siemens to collaborate on a range of fuzzy devices and software support for microcontrollers.

Late in joining the race, Intel has recently announced that fuzzyTECH Release 3.0, from German systems and software house Inform GmbH, now supports Intel's '196 family of 16-bit microcontrollers, with support for the 8051 and x86 families to follow this coming March. Although similar to Togai's and Aptronix's products in function, fuzzyTECH also has a target-level debugging capability integrated into its fuzzy development system. Neither Aptronix nor Togai support this level of debugging, but both vendors are planning to add it to their next releases, also due in March of 1993.

National Semiconductor has taken an independent path in this competition. Instead of signing on with one of the existing players in fuzzy technology, National is creating its own development tool, called Neufuz (for Neural + Fuzzy), that it plans to ship by next March. Neufuz claims several unique features. Like Togai's TILGen, it uses neural net techniques to automate the development process, but Neufuz actually "learns" the required membership functions and rules by having the designer present system inputs and their corresponding desired outputs to the development system. Neufuz then "generalizes" from these specific examples. In addition, Neufuz doesn't use standard defuzzification techniques like the COG method, relying instead on a proprietary, neural-net-based strategy. Initially, Neufuz will support National's COP-8 8-bit microcontroller family, with support for its HPC+ 16-bit devices to follow shortly.

Another software-based fuzzy tool is Hyperlogic's Cubicalc RTC fuzzy development environment. Like TILShell+/FCDS, Cubicalc provides a graphical user interface and produces C code as output. At the low-cost end of the spectrum is Byte Craft's Fuzz-C preprocessor. Fuzz-C adds keywords for fuzzy sets, membership functions, and rules to the C language, simplifying the process of writing fuzzy applications entirely in C. The resulting code is compatible with most ANSI C compilers.

Fuzzy Hardware

While software-based fuzzy logic applications may dominate the market, there are several interesting offerings in fuzzy logic hardware. Aptronix had introduced a board-level fuzzy processor, the Truth Value Flow Inference Module (TVFI/M) in 1991 (see µPR 3/6/91, p. 11–13), but has since down-played this product, at least in the U.S., in favor of its fuzzy software development tools.

Togai was actually first to market in 1989 with its FC110 Digital Fuzzy Processor (DFP) and claims to have scored a number of significant design wins. It is currently priced at $40 in quantities of 100. The FC110 can operate as a coprocessor or in stand-alone mode with very limited I/O. Togai describes the device as a RISC processor—and the instruction set is indeed sparse—but its six addressing modes seem to be at odds with the traditional definition of RISC.

The FC110 architecture (see Figure 1) contains two separate memory spaces. Variable data is stored in a 256-byte on-chip RAM, with the low 64 bytes dual-ported between the DFP and the host processor for inter-processor communications. Instructions and constant data reside in a 64K × 16-bit off-chip "knowledge base" ROM space. There is also a bank of sixteen 16-bit general-purpose registers that may be referenced as 8-, 16-, 24-, 32-, or 64-bit words. Fuzzy rules and membership functions are maintained at 8 bits of precision.

The system clock is 20 MHz and, while many instruc-

Mnemonic	Description
DEFUZ	Defuzzification
FZAND	Fuzzy AND Connective
FZOR	Fuzzy OR Connective
LHS	Evaluate Left Hand Side ("IF" portion) of rule
RHSC	Evaluate Right Hand Side ("THEN" portion) of rule using Cenroid method
RHSH	Evaluate Right Hand Side ("THEN" portion) of rule using Height method

Table 1. FC110 fuzzy-specific instructions

tions execute in a single cycle, some of the fuzzy-specific instructions could take many hundreds of cycles, depending on the size of their arguments. Although its performance is modest compared to today's high-end microprocessors, Togai claims that, at 100,000 fuzzy rule evaluations per second, the FC110 can provide over 10 times the performance of the equivalent software-based fuzzy model running on a 20-MHz 386. This advantage is due to the FC110's suite of specialized instructions for evaluating fuzzy logic rules (see Table 1). FC110 applications are developed using TILShell+ and Togai's FC110 compiler.

More recently, Togai has begun offering the Fuzzy Computation Acceleration (FCA) portion of its FC110 as a core cell technology for ASIC design. In this form, some sort of additional controller is required, although depending on the application this could be a simple sequencer rather than a microprocessor. Togai has designed its FCA core to be scalable from 8 to 32 bits of resolution.

In yet another strategic partnership announced in September of this year, Togai has granted VLSI Technology an exclusive license to use the FCA core in merchant-market ASIC designs. VLSI is currently offering customers a 12-bit version of the FCA core, designated the VY86C500. This function block, combined with VLSI's 8-bit Z80 or 32-bit ARM6 processor cores, can provide a high-performance, integrated fuzzy logic system.

Another hardware vendor with a strategic alliance and an interesting array of fuzzy logic and neural net products is American NeuroLogix, whose CEO, Joon Lee, is a former Samsung executive. At $11 each in volume, its NLX230 Fuzzy MicroController (Figure 2) is easily the lowest-cost fuzzy part being offered. The NLX230 is not really a microcontroller, in the standard sense, but rather a pipelined fuzzy logic engine. It has 8 multiplexed inputs, 16 parallel fuzzifier circuits, minimum and maximum comparators, rule memory, and an output register. It also has an optional internal loopback circuit that permits the implementation of fuzzy sequencers and state machines.

The NLX230 can operate in master or slave mode. In master mode, on-chip hardware provides all the necessary signals to automatically download configuration data from an external 93C56-series 2048-bit serial EEPROM upon initialization. In slave mode, configuration data is

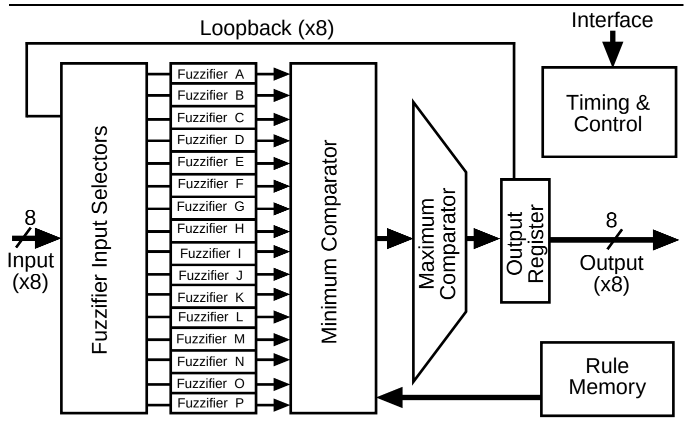

Figure 2. Block diagram of NeuroLogix's NLX230 Fuzzy Microcontroller.

provided by a host processor or other external logic. Input and output data are 8-bit precision. Each input can be connected to one or more of the 16 fuzzifiers. The fuzzifiers calculate the input value's numerical distance from the membership function's ideal value. Rules are 24 bits wide; 16 of those bits associate the rule with one or more fuzzifier outputs. The remaining 8 bits of the rule are called the Action Value and are used to calculate the crisp output. Rules are evaluated by the minimum and maximum comparators. Up to 64 rules may be stored in the on-chip rule memory.

NeuroLogix claims that its NLX230 can evaluate 30,000,000 rules per second. This performance results, however, from the simplified rule evaluation and defuzzification strategy they've taken. In the defuzzification process of most fuzzy systems, all rules that have some truth value contribute to the output. In the NLX230, only the rule with the highest truth value provides the output. This approach greatly reduces the computational requirements for defuzzification. Nonetheless, the device is still quite capable of performing useful functions and its speed is a big plus in real-time applications.

Another chip-level product offered by NeuroLogix is its NLX112 Fuzzy Data Correlator. The use of the term "fuzzy" here is perhaps a bit misleading. The NLX112 is actually a cascadable, parallel 128-bit Hamming comparator. Such devices are often used in pattern matching and image recognition applications, but they are not directly comparable to the rule-based, fuzzy logic technology we've been discussing.

Although we've looked at chips from two smaller companies, the big players are not sitting idly by. Omron, a large manufacturer known mainly for industrial control systems has developed its FP-3000 fuzzy coprocessor and has established a cooperative arrangement with NEC to exploit the technology. In addition, both Motorola and National have indicated that they plan to pursue dedicated fuzzy hardware in the future. SGS-Thomson has also shown an interest in this technology.

Performance Comparisons

We've briefly discussed performance comparisons above and have quoted performance figures in rule evaluations per second. This sort of metric can be very misleading because it depends heavily on the complexity of the rules being evaluated. A more accurate approach would be to specify the number of antecedents (the "IF" side of the rule) and consequents (the "THEN" side of the rule) for each rule. Even this omits the additional overhead of defuzzification as the number of rules in the system grows.

Ultimately, as often found in other attempts at benchmarking the performance of computational hardware, the only meaningful comparisons that can be made are at the systems level with something close to a real application. There are currently some efforts being made to promote such standards for benchmarking fuzzy systems. Until a well-accepted benchmark suite for these systems emerges, however, the numbers will remain slippery.

Conclusions

Proponents of fuzzy logic claim a number of advantages over traditional control system approaches, including a shorter development cycle, more robust system operation, the ability to achieve higher performance with less expensive microcontrollers, ease of system modification and documentation, and, perhaps most important, the ability to solve problems where traditional approaches fail.

Aptronix's Wei Xu, for example, has implemented a two-stage inverted pendulum control system in 950 bytes of 'HC11 code using its FIDE system. This is equivalent to balancing two broomsticks, one on top of the other, from your fingertip. A traditional control-systems solution would require solving fourth-order differential equations in real-time, which is very difficult—if not impossible—with low-cost hardware today. As National's Emdad Khan points out, conventional control systems are very good at linear, time-invariant problems, but not at non-linear, time-variant ones; this is where fuzzy logic shines. Most experts warn that fuzzy logic is not a panacea for every problem and the consensus is that fuzzy logic is simply another tool, albeit a powerful one, that should be in a designer's toolbox.

As to whether fuzzy logic will take hold in the U.S. as it has in Japan, there seems no doubt that it will. The real question is how long it will take. Most optimists see fuzzy systems firmly in the mainstream of control systems design by 1994. Even if Steve Marsh's prediction of 50% penetration in embedded control is short by half, that's still a substantial share of a sizable market. If fuzzy logic takes much longer than a year or two to bloom, however, we could see some attrition among the smaller vendors of this innovative technology.

Price & Availability

The NLX230 Fuzzy MicroController is priced at $9.56 in 1000s. Contact American NeuroLogix, Inc., 411 Central Park Drive, Stanford, FL 32771; 407/322-5608.

The Fuzzy Inference Development Environment (FIDE) costs $1495 in single units. It is available now from Aptronix, 2150 North First Street, Suite 300, San Jose, CA 95131; 408/428-1888.

The Fuzz-C preprocessor costs $149 and will be available in January from Byte Craft Limited, 421 King Street North, Waterloo, Ontario, Canada N2J 4E4; 519/888-6911.

The CubiCalc RTC fuzzy development environment costs $795 and is available now from HyperLogic Corp., 1855 East Valley Pkwy., Suite 210, Escondido, CA 92027; 619/746-2765.

The fuzzyTECH 3.0 development environment for Intel's 80196 family costs $1890 and is available now from Inform Software Corp., 1840 Oak Street, Evanston, IL 60201; 708/866-1838.

The FLED-KT00 Fuzzy Logic Educational Kit ($195) and MC68HXBFIDS Fuzzy Logic Education Kit w/FIDE ($1495) are available now from Motorola, 5050 Barton Springs Road, Austin, TX 78704; 512/505-8108.

National has not released pricing for NeuFuz; availability is promised for 1Q93. National Semiconductor, 2900 Semiconductor Drive, P.O. Box 58090, Santa Clara, CA 95052-8090; 408/721-7136.

Togai's TILShell+/Fuzzy-C Development System for the IBM PC is priced at $4600; the TILShell+/MicroFPL Development System for specified target microprocessors is $2500. The FC110 Digital Fuzzy Processor costs $40 in 100s. Togai InfraLogic, 5 Vanderbilt, Irvine, CA 92718; 714/975-8522.

VLSI's VY86C500 Fuzzy Computational Acceleration Core is available now, but pricing has not been released. VLSI Technology, 1109 McKay Drive, San Jose, CA 95131; 408/434-7912.

Ironically, a major stumbling block in the marketing of fuzzy logic to the design community has been a seemingly arbitrary decision made over 25 years ago: the naming of the technology. The contradictory juxtaposition of the terms "fuzzy" and "logic" has been the lament of many a marketeer. In addition, this intuitive approach to control systems design has proved surprisingly difficult to explain to designers steeped in the Bode plots and root locus diagrams of control theory. As Togai's Camerone Welch puts it, "The company that can explain fuzzy logic in 30 words or less will corner the market." ♦

The author would like to thank David Brubaker, of The Huntington Group in Menlo Park, CA, for showing him where the bodies are buried.

