Fifth IEEE International Conference on Fuzzy Systems, New Orleans, Louisiana, Vol. 3, pp.1951-1957, Sept.8-11, 1996

Integrated Fuzzy Logic Controller with Continuous Processing

S. Bouras †, K. Suyama ‡, and Y. Tsividis ‡
†Division of Computer Science, National Technical University of Athens
Heroon Polytechniou 9, Zographou 15773, Athens, Greece
‡Department of Electrical Engineering, Columbia University New York, NY 10027, USA

Abstract

A new fuzzy logic controller, which is suitable for a system with analog input and output, has been implemented in a monolithic form. It employs an architecture where time sweeping of variables allows continuous evaluation of fuzzy inferences and defuzzification without having to discretize input and output variables using A/D and D/A converters. Direct processing of the analog input signal is used to obtain the corresponding crisp value; the digital portion is used only for programmability. The controller can handle 3 inputs, 1 output, and 25 programmable fuzzy rules. The test IC chips were fabricated using 0.7 μm CMOS technology. A control problem of stabilizing a ping-pong ball in a tube with a controllable air flow has been successfully demonstrated.

1. Introduction

Fuzzy logic has been applied in a variety of disciplines such as control systems, image recognition, robotics, financial market prediction, etc. in the last two decades (see [1] and [2] for recent examples). Many of these applications often require an integrated circuit (IC) implementation of fuzzy logic processing in a form of either a dedicated IC chip or a part of a large integrated system. Mainly, two realization methods for fuzzy logic processing exist. One method is to digitize the input signals and to process the information in digital fashion throughout the system (for example, [3, 4, 5]). However, in many cases, inputs come from sensors and are analog in form, and outputs must also be analog in order to drive several types of actuators. In applications where such input and output analog signals are many, a large number of A/D and D/A converters with sufficient resolution would be needed if an all-digital processor is to be used. This will increase chip size and power dissipation. To avoid this problem, a mixed analog-digital VLSI circuit can be used in such cases (for example, [6, 7, 8, 9, 10, 11, 12]).

This paper presents the system aspect of a fuzzy logic controller with *continuous* fuzzy inference and defuzzification. Most of the critical fuzzy computation is done by analog IC techniques which have been well developed over the last several years in the context of integrated analog continuous-time signal processors (see, for example, [13] and the many references therein). Digital logic is still used for programmability in this method, but all processing mantains the signals in analog form. Since the output of most sensors changes slowly, sequential processing and time-sharing can be used to keep the power dissipation low. This is in contrast with many other mixed analog-digital techniques reported to date where some form of discretization is used. The controller has 3 inputs and 1 output with 25 possible programmable rules.

In our system, the fuzzy inference algorithm is a "min-max" type and the i^{th} rule can be written as follows:

$$\text{If } x_1 \text{ is } A_{1i} \text{ and } x_2 \text{ is } A_{2i} \text{ and } x_3 \text{ is } A_{3i} \text{ then } y_i \text{ is } B_i \tag{1}$$

where x_1, x_2, and x_3 are the input variables, y_i is the i^{th} output variable from the antecedent, and A_{1i}, A_{2i}, A_{3i}, and B_i are fuzzy sets for the i^{th} rule.

The final crisp output value y_{crisp} is obtained through the center of gravity as follows:

$$y_{crisp} = \frac{\int_{y_{low}}^{y_{high}} y \cdot \mu_B(y) dy}{\int_{y_{low}}^{y_{high}} \mu_B(y) dy}. \tag{2}$$

where $\mu_E(z)$ represents a membership function of fuzzy set E as a function of z and y_{low} and y_{high} define the output universe of discourse. As explained later, the integrals are evaluated continuously without discretizing the output universe of discourse.

0-7803-3645-3/96 $5.00©1996 IEEE

Two IC chips have been fabricated; one handles the fuzzy inference described above, and the other contains the defuzzification circuits. In the next section, the overall architecture of the proposed controller and its operation are described. The circuit realization of the essential circuit blocks is deferred to the journal version of this paper [14]. The chips have been used to stabilize a ping-pong ball in a tube with a controllable air flow where the air turbulence around the ball makes the stabilization complicated.

2. Fuzzy controller architecture

The architecture of the fuzzy controller is shown in Fig. 1. The three inputs and the output of the system are denoted as x_1, x_2, x_3, and y_{crisp}, respectively. The input switch, fuzzy rule processing, and defuzzifier blocks consist of analog circuits. The timing block is implemented using digital circuits and the memory block has a mixture of both analog and digital circuits. The following subsections explain each block.

2.1. Timing block

The whole controller operates in a sequential fashion where each input is sampled and processed one at a time. After three inputs (if all three inputs are present) are processed, the defuzzification is done by time sweeping as it will be shown in Sec. 2.4. Therefore, it is essential that the timing block broadcasts the state of operation to all blocks. The user needs to specify the number of inputs to the timing block through D0 and D1. Then, the block generates appropriate control signals to other blocks. Bits, M0 and M1, are used to indicate to the memory block which one of the inputs (the three inputs and the sweep signal) is being processed so that the memory block can generate appropriate fuzzy lebels.

The timing block is also responsible for generating control signals to the input switch, fuzzy rule processing, and defuzzifier blocks as shown in Fig. 2. The role of the control signals, F1C, F2C, READ1, READ2, READ3, and Rsweep, will be explained in Sec. 2.3.

2.2. Memory block

Depending on the state of operation given by M0 and M1, this block generates appropriate quantities which define membership functions being used in the fuzzy rule processing block. The proposed controller can generate fully programmable triangular and trapezoidal membership functions. Each fuzzy label corresponds to five quantities as shown in Fig. 3. Two voltages, V_{SH1} and V_{SH2}, are used to shift each segment indepedently, two currents, I_{C1} and I_{C2}, control the slope of each segment independently, and I_{MAX} determines the height.

Figure 4 shows the memory. Since we chose to employ 5 membership functions to represent the input and output universe of discourse, there are 5 fuzzy label units (F1 to F5); each unit produces 5 analog quantities (V_{SH1}, V_{SH2}, I_{C1}, I_{C2}, and I_{MAX}) to specify a fuzzy label. The fuzzy label units are connected to the membership function generator circuits in the fuzzy rule processing block (Fig. 1) via 25 switch blocks so that the fuzzy label units can be multiplexed (see the inset of Fig. 4). (Recall that there are 25 possible programmable rules in the controller.) The analog multiplexer is controlled by M0 and M1 through binary RAM. The memory described above is similar in operation to the fuzzy memory device proposed in [15].

2.3. Fuzzy rule processing block

Figure 5(a) shows the fuzzy rule processing and input switch blocks. Note that only one input switch block is necessary for all 25 rules but 25 rule blocks are needed. As explained in Sec. 2.1, the operation of the controller is based on sequential sampling of the 3 inputs. The input switch block connects the inputs one at a time to the membership function generator according to the control signals shown in Fig. 2. Another terminal in the block is used to sweep a voltage signal for a period of T (see Fig. 5(a)). During the sweep phase, the MAX and defuzzifying operations are done (see Sec. 2.4).

The membership function generator (Fig. 5(a)) receives an appropriate fuzzy label (the 5 analog values described in Sec. 2.2) from the memory for the particular input and fuzzy rule it is processing. The membership function generator generates a continuous current, which represents the degree of the membership function within the closed universe of [0, 20 μA]. Each input is represented by a continuous voltage in the universe of $[-1, 0$ V], chosen to provide adequate linearity of various internal blocks with the process and power supply voltages used. The output current from the membership function generator is fed to the fuzzy inference unit.

The block diagram of the fuzzy inference unit is shown in Fig. 5(b). It consists of two MIN circuits, two current copiers, and several analog switches. The current copiers are used as analog memory (see below). Sequential MIN operations are first carried out on 3 fuzzy values, which corresponds to 3 inputs.

For the first input $a1$ (READ1 is "ON") in the i^{th}

rule, the degree of membership is calculated. This current is stored in current copier P2 (see Figs. 2 and 5(b)). Then, the second input is connected to the MIN1 (READ2 is "ON"). In the MIN1, this current is compared with the previous current that has been stored in P2 (the switch F2C is "ON" in this phase). The smaller of the two currents is stored in current copier P1. Once the MIN operation is done, the resultant current is copied into P2 (F1C is "ON"). This "pipeline" approach eliminates the need for having a sample-and-hold circuit. The same procedure is repeated for the third input, if any. At the end of this process, P2 has a stored current which represents the result of the antecedent of the i^{th} rule. Here, the inputs are assumed to change slowly compared to the pipelined inference process.

The consequent ("then" clause) of the i^{th} rule is done by generating the output membership function for the rule. The corresponding fuzzy label is given by the memory block and the resulting continuous current from the membership function generator is available at the input of the fuzzy inference unit. During this phase, Rsweep is "ON". The consequent is completed when this current is compared with the stored current in P2 through MIN2, which represents the result of the antecedent of this rule. It should be noted that a change of variable has occurred from voltage to time. Therefore, it is possible to observe the resultant membership functions in an oscilloscope as shown in Sec. 4.

The currents from all rule blocks are applied to the MAX block as shown in Fig. 5(a). The output of the MAX block is the current that corresponds to the overall fuzzy inference. The current is then applied to the defuzzifier.

2.4. Defuzzifier block

A time-sweeping approach to defuzzification was recently reported elsewhere [12], but it is not suitable for implementation using conventional IC technologies. Here we propose instead a method using time sweeping, which can be implemented in standard CMOS. The defuzzification is based on (2) and is implemented by a multiplier, a divider, and 3 integrators (Fig. 6). We also use a current-to-voltage converter, an attenuator, and a division control unit. The inputs are the current $I_{in}(t)$ that represents the overall fuzzy inference and the sawtooth voltage $V_{sweep}(t)$. The output is y_{crisp}, which is a voltage signal. The sawtooth voltage $V_{sweep}(t)$ is obtained by applying a dc voltage V_s to the integrator O1 in Fig. 6 for a period of T (Fig. 5(a)) and resetting the integrator until the next sweep period.

The attenuator block with a gain of 1/2 is inserted since the nonlinearity of the multiplier becomes significant for signals greater than 0.5V (-1 to -0.5V). The current-to-voltage (I/V) converter is needed since the multiplier block is based on a voltage-mode integrator. The division control unit is used to avoid very small signals applied to the denominator of the divider. When the value V_p becomes smaller than a certain voltage V_L, the output voltage of the unit V_{pc} is kept to V_{CL}. Otherwise, $V_{pc} = V_p$. The defuzzified output voltage is:

$$y_{crisp} = V_{out}(T) = M \cdot A \cdot D \cdot \frac{\int_0^T I_{in} \cdot V_{sweep}(t)dt}{\int_0^T I_{in}(t)dt} \quad (3)$$

where M is the multiplication factor, D is the division factor, A is the attenuator factor, and T is the sweep duration.

3. IC implementation

Many existing and novel analog IC techniques are used in the controller. For example, the fuzzy rule processing block employs current copiers [16], which are used here as an analog memory in the fuzzy inference unit (Fig. 5(b)). Techniques used in continous-time integrated filters [13, 17] are also used in the defuzzification block. Due to the limited space in this paper, we ask the interested readers to refer to the journal version of this paper [14].

A standard 0.7 μm CMOS process was used to fabricate the input switch and fuzzy rule processing blocks on one chip as shown in Fig. 7 (called the "inference" chip) and the defuzzifier on another chip to show the principle. The fuzzy memory and timing blocks were not fabricated at this time and they were built externally. The inference chip is capable of handling 3 inputs, 1 output, and 5 rules. However, the MAX circuit in Fig. 5(a) can accept 25 rules. Therefore, it is possible to extend the current system to handle 25 rules using multiple inference chips. It should be emphasized that the chip set was designed to show the principle, and that no effort was done to optimize the area and power in this case. The sweeping period is 24 μs and the whole cycle is 48 μs. Important design parameters are summarized in Table 1.

4. Measurements

Because of the change of variable inherent in the time sweeping process explained in Sec. 2.3, it is possible to display membership functions easily using a

digital oscilloscope. The top curve (a) in Fig. 8 shows the voltage sweep signal that ramps from -1 V to 0 V. The sweeping time was purposely slowed down for clear displaying in the presence of parasitics due to external connections during testing. The membership functions, which are in the form of current, were converted to voltage signals by external resistors and displayed in Fig. 8(b) and (c). Note that the membership functions look "inverted" due to the way the current signals are converted to voltage signals. Both triangular and trapezoidal functions can be obtained as shown. Figure 9 demonstrates the programmability of membership functions by the membership function generator circuit. Their slopes, heights, and positions can be easily programmed.

Defuzzifier signals can also be observed as shown in Fig. 10. All signals are balanced around -1 V in order to limit the disturbance from spurious signals. In Fig. 10, only the positive side of differential signals is shown. Referring to Fig. 6 for the block diagram of the defuzzifier, the plots from the top of Fig. 10 are the output of the multiplier, the integrators, and the divider, respectively. At the end of the sweep period, the defuzzification computation is completed. In Fig. 10, the divider value "H" is marked to show that it is the crisp value that should be sampled.

5. Demonstration

A control problem of stabilizing the position of a ping-pong ball in a tube with a controllable air flow has been used to demonstrate the functionality of the controller [18]. A fan with controllable voltage is used at the bottom of the tube to blow air which pushes the ball upwards. Ultrasound transducers, mounted at the top of the tube, are used to sense the ball's position and, through differentiation, its speed. Five rules, two inputs (the position of the ball and its derivative), and three triangular membership functions (negative, zero, and positive) are used. The interested readers are asked to refer [14] for more details on the demonstration. The system successfully stabilized the position of the ball even when some disturbances where purposely introduced (e.g., inhibiting part of the air flow by partially covering the top of the tube).

6. Conclusion

A fuzzy logic controller has been designed, built, tested, and demonstrated in a control system. The controller has two custom-made mixed analog-digital chips using a standard 0.7 μm CMOS fabrication process. Through a time sweeping feature that allows a change of variable from voltages and currents to time, continuous fuzzy inferences and defuzzification were employed. This eliminates the need for any quantization in the signal path, and thus no data converters are needed. Taking advantage of various state-of-the-art analog IC techniques (e. g. variable transconductance elements, current copiers, continuous-time integrators, etc.), analog circuits for programmable membership functions and continuous defuzzification have been designed. Although the controller presented in this paper has only 3 inputs, 1 output, and 25 possible rules, the modular design of the architecture can potentially allow more input and output variables as well as a larger number of rules. The continuous-time analog processing feature of this controller will allow it to coexist with integrated sensors within one IC chip where the controller could be used without having to discretize the sensor output.

Acknowledgment

This work was supported in part by PENED Grant 91ED122.

References

[1] *IEEE Trans. Fuzzy Systems.*

[2] *Proc. IEEE Int. Conf. Fuzzy Systems.*

[3] M. Togai and H. Watanabe, "Expert system on a chip: an engine for real-time approximate reasoning," *IEEE EXPERT,* vol. 1, no. 3, pp. 55-62, Aug. 1986.

[4] H. Watanabe, W. D. Dettloff, and K. E. Yount, "A VLSI fuzzy logic controller with reconfigurable, cascadable architecture," *IEEE J. of Solid-State Circuits,* vol. 25, no. 2, pp. 376-382, Apr. 1990.

[5] M. Menke, R. Brunner, H. Eichfeld, and T. Künemund, "A general-purpose 12 bit fuzzy coprocessor," in *Proc. ESSCIRC,* pp. 118-121, (Lille, France), Sep. 1995.

[6] T. Yamakawa, T. Miki, and F. Ueno, "The design and fabrication of the current mode fuzzy logic semicustom IC in the standard CMOS IC technology," in *Proc. IEEE Int. Symp. Multiple-Valued Logic,* pp. 76-82, (Ontario, Canada), May 1985.

[7] T. Kettner, C. Heite, and K. Schumacher, "Analog CMOS realization of fuzzy logic membership functions," *IEEE J. Solid-State Circuits,* vol. 28, no. 7, pp. 857-861, July 1993.

[8] J. Fattaruso, S. S. Mahant-Shetti, and J. B. Barton, "A fuzzy logic inference processor," *IEEE J. Solid State Circuits,* vol. 29, no. 4, pp. 397-402, April 1994.

[9] I. Baturone, S. Sánchez Solano, A. Barriga, and J. L. Huertas, "A switched-capacitor based singleton fuzzy controller," in *Proc. IEEE Int. Conf. on Fuzzy Systems,* pp. 1303-1307, (Orlando, FL), June 1994.

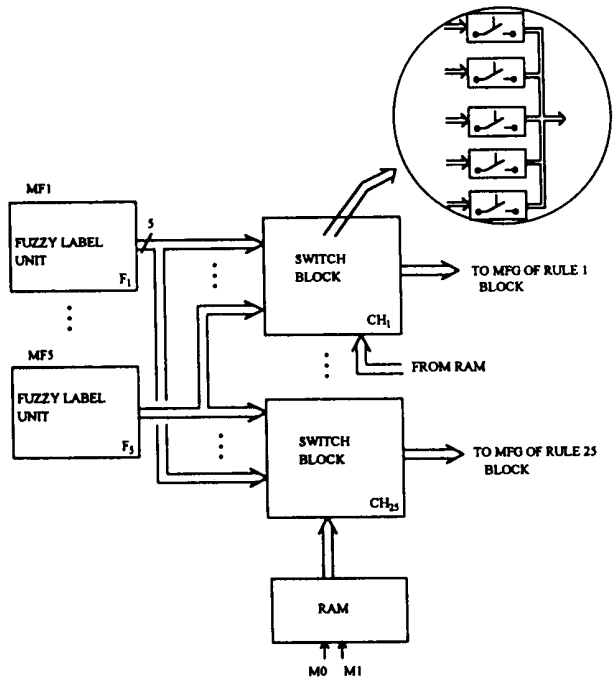
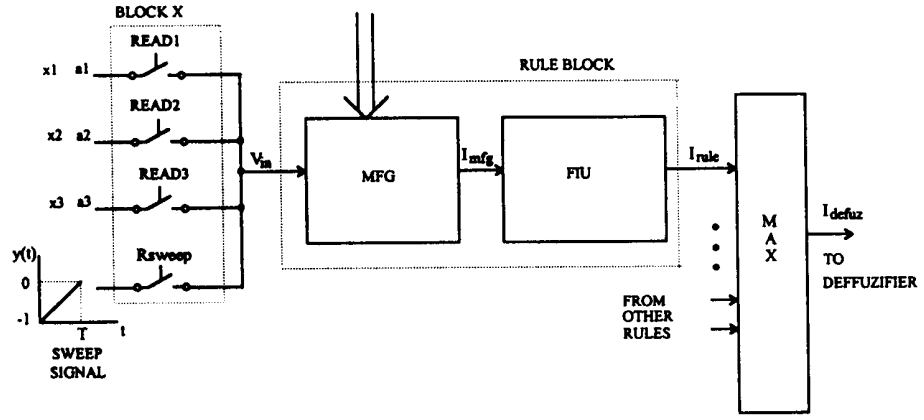
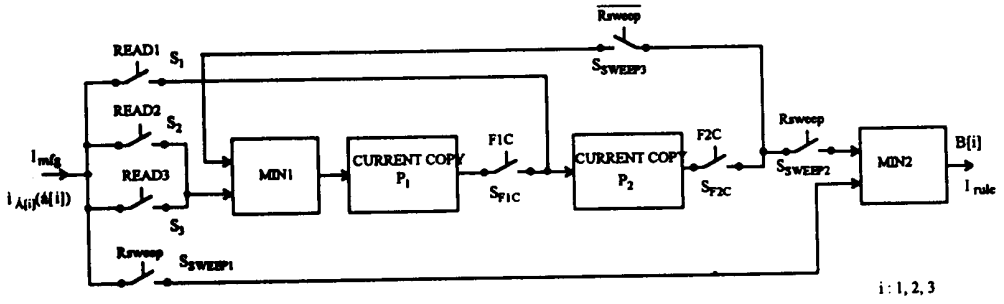


Figure 4. Memory.



(a)



(b)

Figure 5. (a) Input switch and fuzzy rule processing blocks. (b) Fuzzy inference unit.

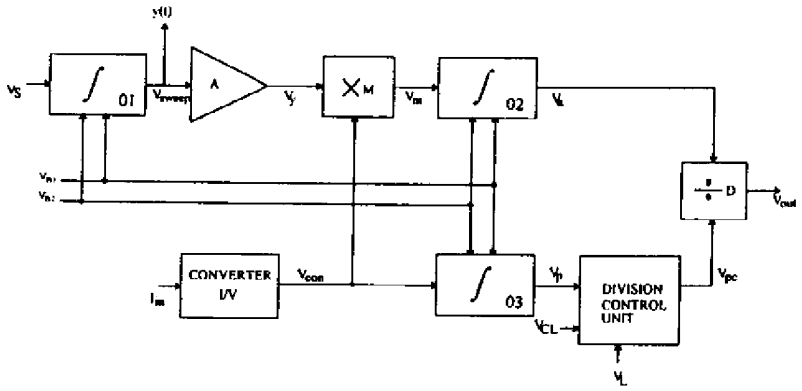


Figure 6. Block diagram of defuzzifier.

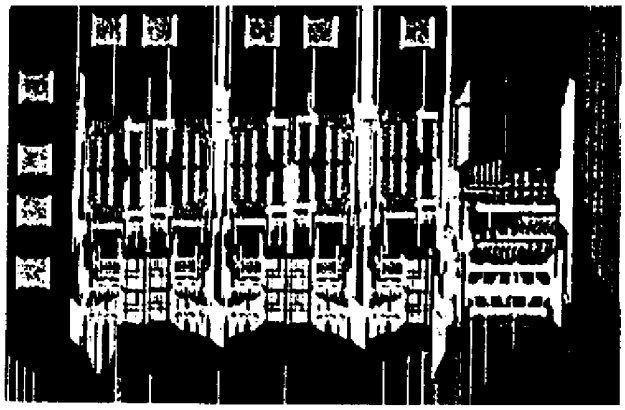


Figure 7. Chip microphotograph of the inference chip.

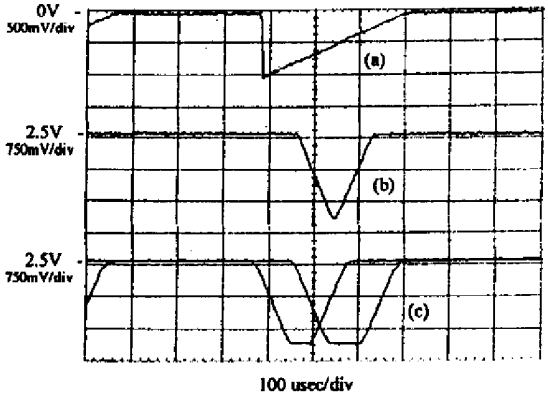


Figure 8. Membership function signals: (a) Sweep signal, (b) triangular membership function, and (c) trapezoidal membership function.

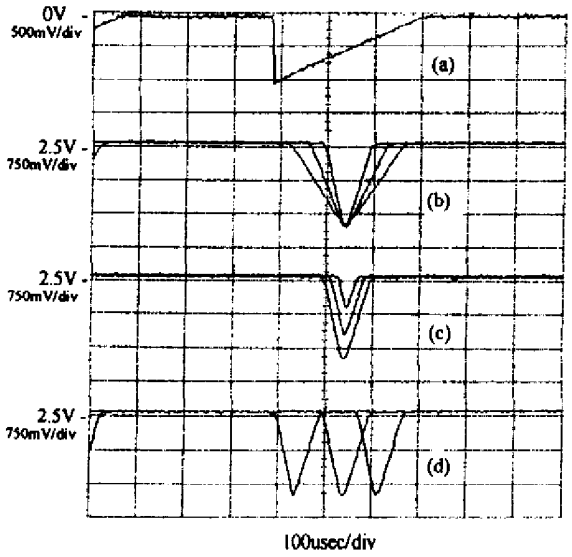


Figure 9. Programmability of membership functions. Curves are (a) sweep voltage, (b) triangular functions with different slopes, (c) triangular functions with different heights, and (d) triangular functions with different locations.

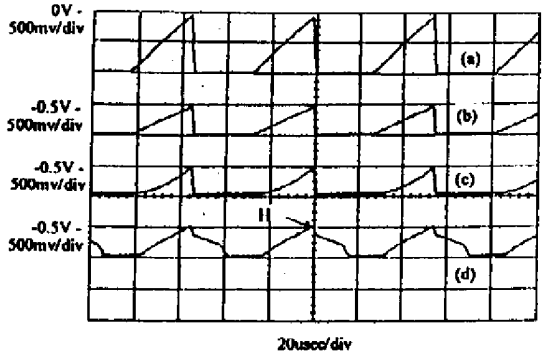


Figure 10. Defuzzifier signals. Referring to Fig. 6, plots are (a) multiplier output, (b) the output of the integrator O3, (c) the output of the integrator O2, and (d) the output of the divider.