



- (51) International Patent Classification:
B41J 2/175 (2006.01) B41J 29/38 (2006.01)
- (21) International Application Number:
PCT/US2017/027562
- (22) International Filing Date:
14 April 2017 (14.04.2017)
- (25) Filing Language: English
- (26) Publication Language: English
- (71) Applicant: HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P. [US/US]; 11445 Compaq Center Drive W., Houston, Texas 77070 (US).
- (72) Inventors: MARTIN, Eric; 1070 NE Circle Boulevard, Corvallis, Oregon 97330-4239 (US). ANDERSON, Daryl E.; 1070 NE Circle Blvd., Corvallis, Oregon 97330-4239 (US).
- (74) Agent: MORRIS, Jordan E. et al.; HP Inc., Mail Stop 35 3390 E. Harmony Road, Fort Collins, Colorado 80528 (US).
- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR,

KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- as to the identity of the inventor (Rule 4.17(i))
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))

Published:

- with international search report (Art. 21(3))

(54) Title: DELAY ELEMENTS FOR ACTIVATION SIGNALS

(57) Abstract: In some examples, a fluidic die includes a set of fluid actuators arranged in an order, and a controller to determine, based on input control information relating to controlling actuation of the plurality of fluid actuators, whether a first fluid actuator of the plurality of fluid actuators is to be actuated and whether a second fluid actuator within a specified proximity of the first fluid actuator in the order is to be actuated, and in response to determining that the first fluid actuator is to be actuated and the second fluid actuator within the specified proximity of the first fluid actuator in the order is not to be actuated, activate a delay element associated with the first fluid actuator, the delay element to delay an activation signal propagated to selected fluid actuators of the set of fluid actuators in response to an actuation event.

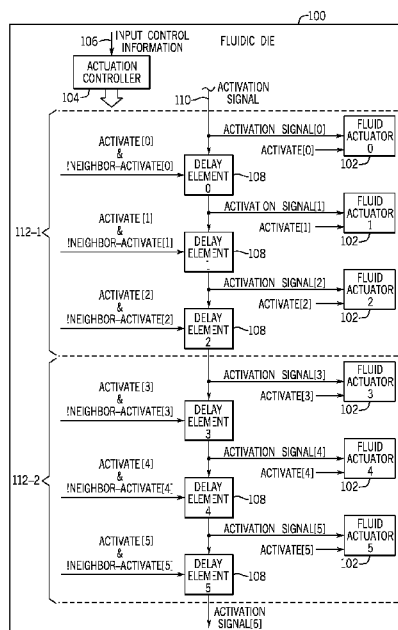


FIG. 1



DELAY ELEMENTS FOR ACTIVATION SIGNALS

Background

[0001] Fluid control devices such as fluidic dies can control movement and ejection of fluid. Such fluidic dies may include fluid actuators that may be actuated to cause displacement of fluid. Some example fluidic dies may include printheads, where fluids used by the printheads can include ink or other types of fluids.

Brief Description of the Drawings

[0002] Some implementations of the present disclosure are described with respect to the following figures.

[0003] Fig. 1 is a block diagram of a fluidic die according to some examples.

[0004] Fig. 2 illustrates an example showing which fluid actuators and which delay elements are activated, according to some examples.

[0005] Fig. 3 is a schematic diagram of a delay element according to some examples.

[0006] Fig. 4 is a timing diagram of delayed instances of activation signals according to some examples.

[0007] Fig. 5 is a block diagram of a fluidic die according to further examples.

[0008] Figs. 6A-6B illustrate shifting of a mask data pattern in a mask register, according to additional examples.

[0009] Fig. 7 is a block diagram of a fluidic die according to some examples.

[0010] Fig. 8 is a block diagram of a fluid control system according to further examples.

[0011] Fig. 9 is a block diagram of a fluid control device according to alternative examples.

[0012] Throughout the drawings, identical reference numbers designate similar, but not necessarily identical, elements. The figures are not necessarily to scale, and the size of some parts may be exaggerated to more clearly illustrate the example shown. Moreover, the drawings provide examples and/or implementations consistent with the description; however, the description is not limited to the examples and/or implementations provided in the drawings.

Detailed Description

[0013] In the present disclosure, use of the term “a,” “an,” or “the” is intended to include the plural forms as well, unless the context clearly indicates otherwise. Also, the term “includes,” “including,” “comprises,” “comprising,” “have,” or “having” when used in this disclosure specifies the presence of the stated elements, but do not preclude the presence or addition of other elements.

[0014] A fluid control device can include multiple fluid actuators that when actuated cause displacement of fluid. For example, the fluid control device can control ejection of a fluid from an orifice of the fluid control device towards a target. In such examples, the fluid control device can be referred to as a fluid ejection device that is able to control ejection of fluids. In some examples, fluid ejection devices can include printheads that are used in two-dimensional (2D) or three-dimensional (3D) printing. In 2D printing, a printhead can eject ink or other printing fluid directed to a target substrate (e.g., paper, plastic, etc.) to print a pattern onto the target substrate. In 3D printing, a printhead can eject a fluid used to form a 3D target object. A 3D printing system can form the 3D target object by depositing successive layers of build material. Printing fluids dispensed from the 3D printing system can include ink, as well as fluids used to fuse powders of a layer of build material, detail a layer of build material (such as by defining edges or shapes of the layer of build material), and so forth.

[0015] In other examples, a fluid control device can include pumps that control fluid flows through respective fluid channels. More generally, a fluid control device can be used in either a printing application or a non-printing application. Examples of fluid control devices used in non-printing applications include fluid control devices in fluid sensing systems, medical systems, vehicles, fluid flow control systems, and so forth. In a printing application, a fluid control device, such as a fluidic die, can be mounted onto a print cartridge, where the print cartridge can be removably mounted in a print system. For example, the fluidic die can be a printhead die that is mounted to the print cartridge. In another example of a printing application, fluid control devices (such as fluidic dies) can be mounted onto a print bar that spans the width of a target medium (e.g., a paper medium or medium of another material) onto which printing fluids are to be dispensed.

[0016] A fluid control device can include multiple fluid actuators that when actuated causes displacement of fluid. As used here, displacement of fluid can refer to movement of fluid within a fluid channel inside the fluid control device, or to ejection of fluid from inside a fluid chamber of the fluid control device through an orifice to a region outside the fluid control device.

[0017] An activation signal (also referred to as a “fire pulse”) can be used to actuate the fluid actuators. The activation signal can be asserted to an active state for a specified time duration (the specified time duration of the active state of the activation signal is the pulse width of the activation signal). When the activation signal is asserted to the active state, selected fluid actuators are actuated, where the selection of fluid actuators is based on input control information as discussed further below. While the activation signal is deasserted to an inactive state, fluid actuators cannot be actuated.

[0018] The multiple fluid actuators of a fluid control device can be partitioned into “primitives” (also referred to as “firing primitives”), where a primitive includes a group of a certain number of fluid actuators. A number of fluid actuators included in a primitive can be referred to as a size of the primitive. Traditionally, primitives of a fluid control device are configured using hardware circuitry, and thus a size of the

primitives used in the fluid control device is fixed. To reduce a peak current when actuating fluid actuators in the primitives, and to minimize power supply transients associated with simultaneous actuation of multiple fluid actuators, a delay can be used to delay the activation signal so that the actuation of fluid actuators between the primitives is correspondingly delayed. In fixed-size primitives, one delay element is provided per primitive. Each fluid actuator of a primitive can be uniquely addressed to select the fluid actuator.

[0019] In accordance with some implementations of the present disclosure, variable-sized primitives can be used in a fluid control device. For a first actuation event (or a first set of actuation events), primitives of a first primitive size can be used, while for a second actuation event (or second set of actuation events), primitives of a second primitive size (different from the first primitive size) can be used. Varying sizes of primitives can be implemented by using different mask data patterns in a mask register of the fluid control device. A first mask data pattern can specify the first primitive size, while a second mask data pattern can specify the second primitive size.

[0020] In arrangements that allow variable-sized primitives according to some implementations of the present disclosure, each fluid actuator can be individually associated with a delay element for delaying an activation signal. The delay elements are daisy chained one to another, so are arranged in series. A delay element is associated with each individual fluid actuator because, in response to a given actuation event, just a respective subset of fluid actuators (where the subset can include just one fluid actuator or some other number of fluid actuators) in each virtual primitive is actuated. For another actuation event, another subset of fluid actuators in each virtual primitive is actuated.

[0021] An actuation event can refer to concurrent actuation of fluid actuators of a fluid control device to cause corresponding fluid displacement.

[0022] To avoid excessive delays from being applied to the activation signal, the delay elements individually associated with the fluid actuators can be selectively

activated and deactivated, based on a determination of whether or not each fluid actuator is to be actuated and also based on whether a neighboring fluid actuator in the primitive is to be actuated. A delay element for an active fluid actuator (a fluid actuator to be actuated) in a given primitive can be activated to delay the activation signal, while a delay element for an inactive fluid actuator (a fluid actuator that is not to be actuated) is deactivated to not delay the activation signal. Note that if the activation signal is subjected to delays of all of the delay elements (arranged in a chain) that are associated with individual fluid actuators, then a large delay can be imposed on the activation signal. Excessive delay of the activation signal can reduce the speed at which fluid displacement operations (e.g., printing operations) can be performed.

[0023] Within a given primitive that includes a group of fluid actuators, it is possible that multiple fluid actuators are actuated in response to an actuation event in some scenarios. For example, multiple fluid actuators in the given primitive can be actuated to increase an effective drop weight. Actuating multiple fluid actuators in a given primitive in response to an actuation event can be performed for other purposes in other examples.

[0024] A drop weight of a fluid can refer to an amount of the fluid that is ejected by a nozzle in response to a single actuation event. In some cases, a drop weight can also be referred to as a drop size. A drop weight is proportional to a drop volume of a fluid. Fluid ejection systems can include fixed drop-weight nozzles, where, for any actuation event, the selected nozzle within a primitive is configured to eject only a single drop weight. Being restricted to a fixed drop weight can reduce flexibility and quality of patterns formed on a target by a fluid dispensed by a fluid ejection system. In other examples, a fluid ejection system includes dedicated nozzles for achieving an increased drop weight. However, use of dedicated nozzles for achieving an increased drop weight can reduce a density of dispensed fluid (such as expressed in dots per inch).

[0025] When multiple neighboring fluid actuators (such as fluid actuators in a given primitive) are being actuated in response to an actuation event, techniques or

mechanisms according to some implementations of the present disclosure implement no delay between the actuation of the multiple neighboring fluid actuators. This allows the multiple neighboring fluid actuators in the primitive to actuate substantially simultaneously, where “substantially simultaneously” can refer to the multiple neighboring fluid actuators actuating at exactly the same time, or actuating at respective time instances that are within a specified time threshold of one another. To achieve increased drop weight, the multiple neighboring fluid actuators that are actuated simultaneously cause respective fluid droplets to be ejected and combine in flight (or on a target) to produce a larger on-target drop. The target can be a paper (or other) medium for 2D printing, or a 3D object for 3D printing, or a different target for a non-printing application. The simultaneous actuation of neighboring fluid actuators can be referred to as a “boost mode” of fluid displacement operation, since the amount of fluid that is displaced responsive to multiple neighboring fluid actuators actuating simultaneously is boosted (i.e., increased) as compared to an example where just one fluid actuator in a group of neighboring fluid actuators is actuated.

[0026] Fig. 1 is a block diagram of an example fluidic die 100. A fluidic die can refer to a structure that includes a substrate on which are provided various layers (e.g., thin film layers) to form fluid channels, orifices, fluid actuators, fluid chambers, electrical conductors, and so forth.

[0027] The fluidic die 100 includes multiple fluid actuators 102. The fluid actuators 102 can be arranged as an array of fluid actuators, which can be a 1-dimensional (1D) array of fluid actuators or a two-dimensional (2D) array of fluid actuators. In other examples, the fluid actuators 102 can be arranged in a different pattern.

[0028] Although Fig. 1 depicts various components of a fluidic die, it is noted that in other examples, similar components can be arranged in other types of fluid control devices.

[0029] In some examples, a fluid actuator 102 can be disposed in a nozzle of the fluidic die 100, where the nozzle may include a fluid chamber and a nozzle orifice in addition to the fluid actuator. The fluid actuator may be actuated such that displacement of fluid in the fluid chamber may cause ejection of a fluid drop through the nozzle orifice. Accordingly, a fluid actuator disposed in a nozzle may be referred to as a fluid ejector.

[0030] A fluid actuator 102 can include an actuator that includes a piezoelectric membrane, an actuator that includes a thermal resistor, an actuator that includes an electrostatic membrane, an actuator that includes a mechanical/impact driven membrane, an actuator that includes a magneto-strictive drive actuator, or other such elements that may cause displacement of fluid responsive to electrical actuation or actuation resulting from another type of input stimulus.

[0031] In some examples, the fluidic die 100 can include microfluidic channels. Microfluidic channels may be formed by performing etching, microfabrication (e.g., photolithography), micromachining processes, or any combination thereof in a substrate of the fluidic die 100. A microfluidic channel may include a fluid channel of specified small size (e.g., of nanometer sized scale, micrometer sized scale, millimeter sized scale, etc.) to facilitate conveyance of small volumes of fluid (e.g., picoliter scale, nanoliter scale, microliter scale, milliliter scale, etc.).

[0032] Some example substrates of fluidic dies can include silicon based substrates, glass based substrates, gallium arsenide based substrates, and/or other such suitable types of substrates for micro fabricated devices and structures. Accordingly, microfluidic channels, chambers, orifices, and/or other such features may be defined by surfaces fabricated in the substrate of the fluidic die 100. The fluid actuators 102 (or a subset of the fluid actuators 102) can be disposed in respective microfluidic channels. In such examples, actuation of a fluid actuator 102 disposed in a microfluidic channel can generate fluid displacement in the microfluidic channel. Accordingly, a fluid actuator 102 disposed in a microfluidic channel may be referred to as a fluid pump.

[0033] The fluidic die 100 includes an actuation controller 104. A “controller” as used herein can refer to any hardware processing circuit, which can include logic circuitry, a microprocessor, a core of a multi-core microprocessor, a microcontroller, a programmable gate array, a programmable integrated circuit device, or any other hardware processing circuit. In further examples, a controller can include a combination of a hardware processing circuit and machine-readable instructions executable on the hardware processing circuit.

[0034] The actuation controller 104 receives input control information 106 relating to controlling an actuation of the fluid actuators 102. Based on the input control information 106, the actuation controller 104 determines which of the fluid actuators 102 are to be actuated. Note that not all of the fluid actuators 102 would be actuated in response to the input control information 106 in some examples.

[0035] As explained further below, the input control information 106 is based on the content of various registers.

[0036] The actuation controller 104 produces various Activate outputs. More specifically, the actuation controller 104 produces N ($N \geq 2$) Activate outputs for N fluid actuators: Activate[0 ... $N-1$]. An Activate[i] output, $i=0$ to $N-1$, is asserted to an active state (e.g., “1”) in response to the input control information 106 selecting a corresponding fluid actuator i for actuation. On the other hand, the actuation controller 104 deasserts the Activate[i] output to an inactive state in response to the actuation controller 104 determining, based on the input control information 106, that the respective fluid actuator i is not to be actuated.

[0037] Each Activate[i] output can be in the form of a signal or any other indication (e.g., a message, an information field, etc.) that can be used to control actuation of the respective fluid actuator i .

[0038] As depicted in Fig. 1, each Activate[i] output is provided to an input of a respective fluid actuator 102. Additionally, in accordance with some implementations

of the present disclosure, each Activate[i] output also controls activation (or deactivation) of a respective delay element 108.

[0039] Fig. 1 shows a chain of delay elements 108 that are to sequentially delay an activation signal 110. The activation signal 110 can be received by the fluidic die 100 from circuitry external of the fluidic die 100, such as from a system controller of a fluid control system. In other examples, the activation signal 110 can be generated internally in the fluidic die 100.

[0040] Each of the multiple delay elements 108 is associated with a respective fluid actuator 102.

[0041] The instance of the activation signal that is received at the input of the chain of delay elements 108 is referred to as activation signal[0]. Activation signal[0] is provided to the input of delay element 0, which can selectively delay (or not) activation signal[0]. The output of delay element 0 is another activation signal instance, referred to as activation signal[1].

[0042] Activation signal[1] is provided to the input of delay element 1, which can selectively delay (or not) activation signal[1]. The output of delay element 1 is another activation signal instance, referred to as activation signal[2]. Activation signal[2] is provided to the input of delay element 2, which can selectively delay (or not) activation signal[2]. The output of delay element 2 is another activation signal instance, referred to as activation signal[3]. Further down the chain of delay elements 108, a further activation signal instance, activation signal[j], is provided to the input of delay element j, which can selectively delay (or not) activation signal[j]. The output of delay element j is another activation signal instance, activation signal[j+1].

[0043] Each fluid actuator i receives the corresponding Activate[i] output from the actuation controller 104 and a respective instance of the activation signal (activation signal[i]) from the chain of delay elements 108. The combination of the respective activation signal[i] (being at an active state) and the respective Activate[i] output

(being asserted to an active state) causes an activation circuit in the respective fluid actuator i to actuate fluid actuator i .

[0044] In accordance with some examples of the present disclosure, the activation or deactivation of a respective delay element i is based on a corresponding Activate[i] output and the state of !Neighbor-Activate[i], where the “!” symbol indicates a logical invert operation. The signal !Neighbor-Activate[i] is active (true) if a neighboring fluid actuator within a specified proximity of fluid actuator i along a specific order of fluid actuators is not to be actuated. The state of !Neighbor-Activate[i] can be set by the actuation controller 104.

[0045] The fluid actuators 102 are arranged in an order, such as an order within a column or other collection of fluid actuators 102.

[0046] In some examples, a neighboring fluid actuator that is within a specified proximity of fluid actuator i can refer to a fluid actuator that is a specified number (1, 2, 3, or any other number) of fluid actuator(s) downstream in the chain from fluid actuator i in a group of fluid actuators. In such examples, the order of the fluid actuators 102 starts upstream in the chain and continues downstream. A first fluid actuator is upstream of a second fluid actuator in the chain of fluid actuators 102 if the first fluid actuator is to be activated by a first activation signal instance that is earlier in time than a second activation signal instance for activating the second fluid actuator.

[0047] In other examples, a neighboring fluid actuator that is within a specified proximity of fluid actuator i can refer to a fluid actuator that is a specified number (1, 2, 3, or any other number) of fluid actuator(s) upstream in the chain from fluid actuator i in a group of fluid actuators. In such examples, the order of the fluid actuators 102 starts downstream in the chain and continues upstream.

[0048] Fluid actuators 102 (such as those within a column or other collection) can be partitioned into multiple groups (e.g., primitives), where each group can include multiple fluid actuators. In Fig. 1, two groups 112-1 and 112-2 are shown, where

each group includes three fluid actuators 102. Group 112-1 includes fluid actuators 0, 1, and 2, and group 112-2 includes fluid actuators 3, 4, and 5.

[0049] Within a group of fluid actuators, the actuation controller 104 is able to control a number of fluid actuators to actuate, where the number can be 1 for fluid actuator actuation in a non-boost mode. However, the actuation controller 104 can cause simultaneous actuation of multiple (2 or 3 in the example of Fig. 1) fluid actuators in a boost mode, such as to increase drop weight.

[0050] In some examples, when just one fluid actuator within group 112-1 is to be actuated, fluid actuator 1 can be actuated while fluid actuators 0 and 2 are not actuated. When two fluid actuators within group 112-1 are to be actuated simultaneously, fluid actuators 0 and 2 can be actuated simultaneously, while fluid actuator 1 is not actuated. Alternatively, all three fluid actuators within group 112-1 can be actuated simultaneously.

[0051] In different examples, different combinations of fluid actuators within a group of actuators can be actuated simultaneously in boost mode. Moreover, although Fig. 1 shows an example where a group (112-1 or 112-2) of fluid actuators includes three fluid actuators, it is noted that in other examples, a group can include a different number of fluid actuators.

[0052] In the example of Fig. 1, !Neighbor-Activate[0] is true if neither downstream fluid actuator 1 or 2 is to be actuated. !Neighbor-Activate[1] is true if downstream fluid actuator 2 is not to be actuated. !Neighbor-Activate[2] is set to true since it is the last fluid actuator in the group 112-1.

[0053] In a different example, !Neighbor-Activate[2] is true if neither upstream fluid actuator 0 or 1 is to be actuated. !Neighbor-Activate[1] is true if upstream fluid actuator 1 is not to be actuated. !Neighbor-Activate[0] is set to true since it is the first fluid actuator in the group 112-1.

[0054] More generally, !Neighbor-Activate[i] is set to a respective state (true or false) by looking ahead or looking behind within a group of fluid actuators. If there

are M ($M \geq 2$) fluid actuators within a group, then $\text{!Neighbor-Activate}[i]$ for delay element i for the group is set to a state based on looking at the remaining fluid actuator(s) ahead (or behind) in the group that is (are) to be actuated.

[0055] Delay element i is activated in response to both the corresponding $\text{Activate}[i]$ output being asserted to an active state and $\text{!Neighbor-Activate}[i]$ being true. Activated delay element i delays a corresponding activation signal instance, $\text{activation signal}[i]$, by a target delay amount (as provided by a delay circuit in delay element i), and outputs the next activation signal instance, $\text{activation signal}[i+1]$. In contrast, a delay element i is deactivated (such that delay element i does not delay $\text{activation signal}[i]$ by the target delay amount) in response to either the $\text{Activate}[i]$ output being deasserted to an inactive state or $\text{!Neighbor-Activate}[i]$ being set to a false state.

[0056] Thus, in response to determining that a first fluid actuator is to be actuated and a second fluid actuator(s) within the specified proximity of the first fluid actuator in an order of fluid actuators is (are) not to be actuated, the actuation controller 104 activates a delay element associated with the first fluid actuator.

[0057] When a given fluid actuator 102 is not to be activated, then the respective delay element 108 remains inactive, such that the deactivated delay element 108 does not delay the activation signal 110 by the target delay amount of a delay element. However, when multiple fluid actuators within a group of fluid actuators are to be actuated simultaneously, only one of the delay elements corresponding to the multiple fluid actuators in the group are activated.

[0058] Each activation signal instance produced in the chain of delay elements 108 can be delayed a different amount relative to the input activation signal 110 ($\text{activation signal}[0]$) depending on how many delay elements upstream in the chain of delay elements 108 were active.

[0059] Fig. 2 shows an example where 12 fluid actuators (such as in a column of fluid actuators) are partitioned into four virtual primitives 0, 1, 2, and 3 (or more

generally four groups). Column 202 in Fig. 2 identifies the 12 fluid actuators as fluid actuators 0 to 11. Column 204 indicates whether or not a respective fluid actuator is to be actuated. In the example of Fig. 2, in virtual primitive 0, fluid actuators 0 and 2 are to be actuated (while fluid actuator 1 is not actuated); in virtual primitive 1, no fluid actuators are to be actuated; in virtual primitive 2, all three fluid actuators 6, 7, and 8 are to be actuated; and in virtual primitive 3, fluid actuator 1 is to be actuated (while fluid actuators 0 and 2 are not actuated).

[0060] Column 206 indicates whether a respective delay element associated with each fluid actuator is activated. The rule applied in determining whether or not a respective delay element of a given virtual primitive is activated is that the actuation controller looks at the downstream two fluid actuators, within the given virtual primitive, to determine whether either of the two downstream fluid actuators in the given virtual primitive are to be actuated. If not, then the respective delay element is activated assuming that the fluid actuator associated with the respective fluid actuator is to be actuated. In the example of Fig. 2, for virtual primitive 0, delay elements 0 and 1 are not activated, while delay element 2 is activated; for virtual primitive 1, none of delay elements 3, 4, and 5 are activated; for virtual primitive 2, delay elements 6 and 7 are not activated, while delay element 8 is activated; and for virtual primitive 3, delay elements 9 and 11 are not activated, while delay element 10 is activated.

[0061] Fig. 3 is a schematic diagram of a delay element 108 according to some examples. The delay element 108 includes a delay circuit 302, which receives as input an activation signal[i] (which corresponds to an activation signal instance along the chain of delay elements 108). The delay circuit 302 can be implemented with any or various types of circuitry. For example, the delay circuit 302 can include a combination of a resistor and a capacitor that in combination causes a delay in the transition of a signal. In other examples, the delay circuit 302 can include a series of inverters or buffers, where the series of inverters or buffers adds a delay to activation signal[i]. As yet another example, the delay circuit 302 can be a flip-flop that is clocked by a clock signal. This causes the delay time to be the period of the clock.

[0062] The output of the delay circuit 302 is provided to the “1” input of a multiplexer 304, while activation signal[i] is provided to the “0” input of the multiplexer 304. A “multiplexer” can refer to any logic that is able to select from among multiple inputs, where the selected input is provided to the output of the multiplexer.

[0063] Selection of the “0” input or the “1” input of the multiplexer 304 is controlled by the combination (e.g., an AND) of an Activate[i] output from the actuation controller 104 and an !Neighbor-Activate[i] signal. A Delay-Activate signal that is derived from the AND of the Activate[i] output and the !Neighbor-Activate[i] signal is provided to the select control input of the multiplexer 304. If the Delay-Activate is set to an inactive state (e.g., “0”), then the “0” input of the multiplexer 304 is selected, and activation signal[i] is propagated through the multiplexer 304 to the output of the multiplexer 304 as output activation signal[i+1]. Selecting the “0” input of the multiplexer 304 effectively bypasses the delay circuit 302, such that activation signal[i] is not delayed by the target delay amount of the delay circuit 302.

[0064] On the other hand, if the Delay-Activate signal is asserted to an active state (e.g., “1”), then the “1” input of the multiplexer 304 is selected, and the output of the delay circuit 302 is selected and propagated through the multiplexer 304 to the output of the multiplexer 304 as output activation signal[i+1].

[0065] In other examples, activation signal[i] can be connected to the “1” input of the multiplexer 304, while the output of the delay circuit 302 is connected to the “0” input of the multiplexer 304. The Activate[i] input to the select control input of the multiplexer 304 would be inverted in such examples. In yet further examples, different logic for selectively delaying or not activation signal[i] can be used in the delay element 108.

[0066] Fig. 4 is a timing diagram showing various activation signal instances: activation signal[0], activation signal[1], and activation signal[2]. In Fig. 4, activation signal[0] corresponds to the (un-delayed) activation signal 110 that is input to the chain of delay elements 108 shown in Fig. 1.

[0067] In the example of Fig. 4, it is assumed that delay element 0 is not activated. As a result, activation signal[1] that is output from delay element 0 is not delayed by the delay amount of the delay circuit 302 (Fig. 3) of delay element 0, as shown in Fig. 4 (note that there may be a slight delay of activation signal[1] relative to activation signal[0] due to the signal passing through the logic of delay element 0, including the multiplexer 304).

[0068] It is assumed in the example of Fig. 4 that delay element 1 (which receives as input activation signal[1] and outputs activation signal[2]) is activated. Fig. 4 shows activation signal[2] delayed by the delay amount of the delay circuit 302 (Fig. 3) of delay element 1. The activation signal instances are successively propagated through respective delay elements in the chain, where some of the activation signal instances may be delayed by activated delay elements while others are not delayed by deactivated delay elements.

[0069] Fig. 5 is a schematic diagram of a fluidic die 500 according to further examples. Fig. 5 shows logic associated with controlling the activation of three respective fluid actuators (that are part of virtual primitive 0). It is noted that further logic is provided for actuating additional fluid actuators in other virtual primitive(s).

[0070] In Fig. 5, the actuation controller 104 includes a number of AND functions 502 and 503. Each AND function 502 receives actuation data from an actuation data register 504 and mask data from a mask register 506. In some examples, the input control information 106 of Fig. 1 includes the actuation data in the actuation data register 504 and the mask data in the mask register 506. A "register" can refer to any storage element that can be used to store data. For example, a register can be part of a portion of a memory device, such as a dynamic random access memory (DRAM), a static random access memory (SRAM), a flash memory, or any other type of memory device. Alternatively, a register can refer to a storage buffer, a data latch, or any other data holding device that can temporarily or persistently store data.

[0071] Each AND function 503 receives the output of a respective AND function 502, and a respective !Neighbor-Activate[j] signal. In other examples, the AND

functions 502 and 503 can be combined into an AND function that receives actuation data from the actuation data register 504, mask data from the mask register 506, and the respective !Neighbor-Activate[i] signal.

[0072] An AND function receives multiple inputs, and produces an active output if all of the multiple inputs are at the active state. Although AND functions are depicted in Fig. 5, it is noted that in other examples, other logic in the actuation controller 104 for producing the Activate[0 ... N-1] outputs based on the actuation data and the mask data can be used. The concept is that an Activate[i] output for actuating a respective fluid actuator is set to an active value in response to all of the corresponding actuation data bit (or other value) in the actuation data, the mask data bit (or other value) in the mask data, and the respective !Neighbor-Activate[i] signal being set to an active value. More generally, the actuation controller 104 is to combine a value in the actuation data register 504 with a corresponding value in the mask register 506 and the respective !Neighbor-Activate[i] signal to determine whether a respective fluid actuator is to be actuated.

[0073] Note that the !Neighbor-Activate[0 ... N-1] signals can be generated by the actuation controller 104.

[0074] The actuation data register 504 can store actuation data that indicates each fluid actuator to actuate for a set of actuation events. Actuating a fluid actuator refers to causing operation of the fluid actuator to perform fluid displacement in the fluidic die 100. As noted above, an actuation event can refer to concurrent actuation of fluid actuators of the fluidic die 100 to cause fluid displacement. An actuation event can be responsive to a command issued to the fluidic die, or a command issued in the fluidic die, to cause fluid displacement to occur. A "set of actuation events" can refer to any sequence or collection of events that can cause respective different groups of fluid actuators 102 to actuate.

[0075] Assuming there are N ($N \geq 2$) fluid actuators 102, the actuation data stored in the actuation data register 504 includes N values that correspond to the N fluid actuators 102. In some examples, each value (represented as "A" in Fig. 5) of the N

values can be provided by a single bit, where a first state of the bit indicates that the corresponding fluid actuator 102 is to be actuated, and a different second state of the bit indicates that the corresponding fluid actuator 102 is to remain un-actuated. In other examples, each value of the N values in the actuation data can be represented using multiple bits, where a first value of the multiple bits indicates that a corresponding fluid actuator 102 is to be actuated, and a different second value of the multiple bits indicates that the corresponding fluid actuator 102 is to remain un-actuated.

[0076] The mask register 506 can store a mask data pattern that indicates a subset of the fluid actuators 102 that is (are) enabled for actuation for a respective actuation event or the set of actuation events. Enabling a fluid actuator for actuation can refer to allowing the fluid actuator to be activated in response to a value of the actuation data in the actuation data register 504 specifying that the fluid actuator is to be actuated.

[0077] A mask data pattern stored in the mask register 506 can have N values that correspond to N fluid actuators 102. Each value (represented as "M" in Fig. 4) of the N values in the mask data pattern can be provided by a single bit or can be provided by multiple bits.

[0078] If a value of a mask data pattern indicates that a particular fluid actuator is not enabled for actuation, then the particular fluid actuator will not be actuated even though the actuation data stored in the actuation data register 504 specifies that the particular fluid actuator 102 should be actuated. On the other hand, if a mask data pattern specifies that the particular fluid actuator is enabled for actuation, the particular fluid actuator is actuated only if the actuation data stored in the actuation data register 504 specifies that the particular fluid actuator is to be actuated. More specifically, a given fluid actuator 102 is to be actuated in response to both a value ("A") of the actuation data register 504 specifying that the given fluid actuator 102 is to be actuated, and a corresponding value ("M") of the mask data pattern enabling actuation of the given fluid actuator 102.

[0079] In the example of Fig. 5, an “A” bit from the actuation data register 504 is provided to a first input of a respective AND function 502 in the actuation controller 104, and an “M” bit from the mask register 506 is provided to a second input of the respective AND function 502. If both input bits are active (e.g., “1”), then the AND function 502 asserts its output to an active state, where this output is provided to an input of the AND function 503.

[0080] Fig. 5 shows the activation signal 110 propagated through a chain of delay elements 108. In Fig. 4, a first (un-delayed) activation signal instance, activation signal[0], and the Activate[0] output from the actuation controller 104 are provided to fluid actuator 0, a second (possibly) delayed activation signal instance, activation signal[1], and the Activate[1] output are provided to fluid actuator 1, a third (possibly) delayed activation signal instance, activation signal[2], and the Activate[2] output are provided to fluid actuator 2, and so forth. Each delay element 108 causes a specified respective delay to be applied to the activation signal 110 when the delay element 108 is activated by the corresponding Activate[i] signal being active.

[0081] Fig. 5 further shows a data parser 508 that receives input data 510. The input data 510 can be provided by a fluid control system to the fluidic die 500. In different phases of operation, the data parser 508 causes loading of the actuation data register 504 and the mask register 506. The data parser 508 is a form of data loading logic to control loading of data into respective registers. The data parser 508 writes column actuation data 512 into the actuation data register 504 during a fluid displacement phase, during which the fluidic die 500 causes displacement of fluid (e.g., eject fluid during a printing operation). The data parser 508 writes a mask data pattern 514 into the mask register 506 during a mask register write phase, which can be part of initialization of the fluidic die 500, as well as in subsequent phases when updating of the mask data pattern in a mask register is to be performed. The mask register 506 is dynamically updateable to provide different drop weights of fluid dispensed by the fluid control device towards a target.

[0082] In some examples, different mask data patterns can be written to the mask register 506. One example use case of writing different mask data patterns to the

mask register 506 is to set a different primitive size. For example, for a first set of actuation events, a first mask data pattern can be written to the mask register 506 to set a first primitive size, for a second set of actuation events, a second mask data pattern can be written to the mask register 506 to set a second primitive size, and so forth.

[0083] In other examples, instead of using just one mask register 506, multiple mask registers can be included in the fluidic die 500, where the multiple mask registers can store different mask patterns. A multiplexer (not shown) can be provided to select from among the multiple mask registers to select the mask data pattern to use.

[0084] As shown in Figs. 6A-6B, a fluidic die can further include a mask register controller 600, which controls a shifting operation of the mask register 506. In some examples, within a given virtual primitive, just a subset of the fluid actuators (where a subset can include one fluid actuator or multiple fluid actuators) of a virtual primitive is actuated in response to a respective actuation event. To actuate all of the fluid actuators of the virtual primitive, a set of actuation events are provided, where each successive actuation event of the set corresponds to actuation of a next subset of fluid actuators of the virtual primitive.

[0085] Figs. 6A-6B show an example where a mask data pattern (in the mask register 506) indicates a primitive size of four, i.e., each virtual primitive has four fluid actuators. Also, the mask data pattern sets a drop weight of two, since two bits are set to "1" for each virtual primitive, and the remaining two bits are set to "0" in the virtual primitive. Assuming a column of 12 fluid actuators, the column is divided into three virtual primitives 1, 2, and 3 (as shown in Fig. 6A). A set of two actuation events (actuation event 0 and actuation event 1) is provided to cause actuation of the four fluid actuators in each virtual primitive at two successive times.

[0086] Fig. 6A shows actuation event 0, in which addresses 0 and 1 are selected by the mask data pattern in the mask register 506. The fluid actuators in the three virtual primitives assigned addresses 0 and 1 are enabled for actuation. The

actuation data register 104 contains all "1"s in the example, while the drop weight pattern of the selected mask register 506 contains the following drop weight pattern: 110011001100. An "F" indicates the respective fluid actuator (associated with each of address 0 and address 1) in each of the three virtual primitives 1, 2, and 3 that is actuated in response to the combination of an actuation data bit and a drop weight pattern bit.

[0087] For actuation event 1, as shown in Fig. 6B, the mask register controller 600 causes a first shift operation 602-1 to occur in the mask register 506. In the example of Fig. 6B, the head of the mask register 506 is shifted into the tail of the mask register 506, and the mask data pattern bits in the mask register 506 are shifted by two bit positions in the example shown. Shifting by two bit positions means that each bit in the mask register 506 is shifted along the shift direction by two positions in the mask register 506. In the example of Fig. 6B, the shift operation 602-1 in response to actuation event 1 causes addresses 2 and 3 to be selected in each virtual primitive. An "F" in Fig. 6B indicates each fluid actuator (associated with address 2 or 3) in each virtual primitive that is actuated.

[0088] If a primitive size were larger than four, then further shift operations in response to further successive actuation events can cause further shifting of the mask data pattern bits by respective two-bit positions.

[0089] More generally, the mask register controller 600 is to shift the mask data pattern in the mask register 506 in response to each actuation event of a set of actuation events, where the shifting is to cause enabling of a different set of fluid actuators for each successive actuation event. The shifting of the mask data pattern in the selected mask register 506 can include a circular shift (as shown in Figs. 6A-6B), or another type of shift, such as a bi-directional shift, first-in-first-out (FIFO) shift, or any other type of movement of bits in the mask register.

[0090] Fig. 7 is a block diagram of a fluidic die 700 according to further examples. The fluidic die 700 includes a set of fluid actuators 102 arranged in an order. The fluidic die 700 further includes an actuation controller 104 to determine, based on

input control information 106 relating to controlling actuation of the fluid actuators 102, whether a first fluid actuator (which can be any of the fluid actuators 102) is to be actuated and whether a second fluid actuator within a specified proximity of the first fluid actuator in the order is to be actuated. In response to determining that the first fluid actuator is to be actuated and the second fluid actuator within the specified proximity of the first fluid actuator in the order is not to be actuated, the actuation controller 104 activates a delay element 108 associated with the first fluid actuator.

[0091] Fig. 8 is a block diagram of an example fluid control system 800, which can be a printing system or any other system in which fluid displacement can be controlled. The fluid control system 800 includes a system controller 802. In a printing system, the system controller 802 is a printer controller.

[0092] The fluid control system 800 further includes a fluidic die 804, which includes a set of fluid actuators 102 arranged in an order, multiple delay elements 108 associated with the fluid actuators 102, where the delay elements if activated are to delay an activation signal 110.

[0093] The fluidic die 804 further includes a register 806 (e.g., the actuation data register 504 and/or the mask register 506 of Fig. 5) to store input control information (which can be provided by the system controller 802) relating to controlling actuation of the fluid actuators 102.

[0094] The fluidic die 800 also includes an actuation controller 104 to determine, based on the input control information, which fluid actuators 102 are to be actuated. The actuation controller 104 determines, based on the input control information, which fluid actuators of the set of fluid actuators 108 are to be actuated. The actuation controller 104 activates delay elements associated with first fluid actuators that are to be actuated, if respective neighboring fluid actuators of the first fluid actuators are not to be actuated, deactivates delay elements associated with second actuators that are to be actuated, if respective neighboring fluid actuators of the second fluid actuators are to be actuated, and deactivates delay elements associated with third fluid actuators that are not to be actuated.

[0095] Fig. 9 is a block diagram of a fluid control device 900 that includes a fluid actuators 102, associated delay elements 108, an actuation data register 504 to store actuation data, a mask registers 506 to store a mask data pattern, and an actuation controller 104 to determine, based on the actuation data and the mask data pattern, whether a given fluid actuator of the set of fluid actuators is to be actuated, and in response to determining that the given fluid actuator is to be actuated and a neighboring fluid actuator within a specified proximity of the given fluid actuator is not to be actuated, activate a delay element 108 associated with the given fluid actuator.

[0096] As noted above, in some examples, certain logic (such as the various controllers) can be implemented as either a hardware processing circuit or as a combination of a hardware processing circuit and machine-readable instructions (software or firmware) executable on the hardware processing circuit.

[0097] In examples where machine-readable instructions are employed, the machine-readable instructions can be stored in a non-transitory machine-readable or computer-readable storage medium.

[0098] The storage medium can include any or some combination of the following: a semiconductor memory device such as a dynamic or static random access memory (a DRAM or SRAM), an erasable and programmable read-only memory (EPROM), an electrically erasable and programmable read-only memory (EEPROM) and flash memory; a magnetic disk such as a fixed, floppy and removable disk; another magnetic medium including tape; an optical medium such as a compact disk (CD) or a digital video disk (DVD); or another type of storage device. Note that the instructions discussed above can be provided on one computer-readable or machine-readable storage medium, or alternatively, can be provided on multiple computer-readable or machine-readable storage media distributed in a large system having possibly plural nodes. Such computer-readable or machine-readable storage medium or media is (are) considered to be part of an article (or article of manufacture). An article or article of manufacture can refer to any manufactured single component or multiple components. The storage medium

or media can be located either in the machine running the machine-readable instructions, or located at a remote site from which machine-readable instructions can be downloaded over a network for execution.

[0099] In the foregoing description, numerous details are set forth to provide an understanding of the subject disclosed herein. However, implementations may be practiced without some of these details. Other implementations may include modifications and variations from the details discussed above. It is intended that the appended claims cover such modifications and variations.

What is claimed is:

- 1 1. A fluidic die comprising:
2 a set of fluid actuators arranged in an order; and
3 a controller to:
4 determine, based on input control information relating to controlling
5 actuation of the plurality of fluid actuators, whether a first fluid actuator of the plurality
6 of fluid actuators is to be actuated and whether a second fluid actuator within a
7 specified proximity of the first fluid actuator in the order is to be actuated, and
8 in response to determining that the first fluid actuator is to be actuated
9 and the second fluid actuator within the specified proximity of the first fluid actuator in
10 the order is not to be actuated, activate a delay element associated with the first fluid
11 actuator, the delay element to delay an activation signal propagated to selected fluid
12 actuators of the set of fluid actuators in response to an actuation event.

- 1 2. The fluidic die of claim 1, wherein the controller is to:
2 in response to determining that the first fluid actuator is to be actuated and the
3 second fluid actuator within the specified proximity of the first fluid actuator in the
4 order is to be actuated, deactivate the delay element associated with the first fluid
5 actuator such that the activation signal is not delayed by the delay element.

- 1 3. The fluidic die of claim 1, wherein the controller is to:
2 in response to determining that the first fluid actuator is not to be actuated,
3 deactivate the delay element associated with the first fluid actuator such that the
4 activation signal is not delayed by the delay element.

- 1 4. The fluidic die of claim 1, wherein the first and second fluid actuators are to be
2 actuated simultaneously in response to the actuation event to increase a drop weight
3 of fluid dispensed onto a target.

- 1 5. The fluidic die of claim 1, further comprising:
2 a plurality of delay elements individually associated with respective fluid
3 actuators of the set of fluid actuators.
- 1 6. The fluidic die of claim 1, further comprising:
2 an actuation data register to store actuation data that indicates each fluid
3 actuator of the set of fluid actuators to actuate,
4 wherein the input control information comprises the actuation data.
- 1 7. The fluidic die of claim 6, further comprising:
2 a mask register to store a mask data pattern indicating respective fluid
3 actuators of the plurality of fluid actuators enabled for actuation for the actuation
4 event,
5 wherein the input control information further comprises the mask data pattern.
- 1 8. The fluidic die of claim 7, wherein the controller is to combine a value in the
2 actuation data register with a corresponding value in the mask register to determine
3 whether a respective fluid actuator of the plurality of fluid actuators is to be actuated.
- 1 9. The fluidic die of claim 7, wherein the mask data pattern defines a primitive
2 size corresponding to a number of fluid actuators in a primitive, the set of fluid
3 actuators partitioned across multiple primitives each of the primitive size.
- 1 10. The fluidic die of claim 7, wherein the mask register is to be loaded with
2 different mask data patterns to provide respective different drop weights of fluid
3 dispensed by the fluidic die.

- 1 11. The fluidic die of claim 1, further comprising:
2 a mask register to store a mask data pattern indicating a respective subset of
3 fluid actuators of the set of fluid actuators enabled for actuation for the actuation
4 event, wherein the input control information comprises the mask data pattern,
5 wherein the controller is to shift the mask data pattern in the mask register in
6 response to each actuation event of a set of actuation events, the shifting to cause
7 enabling of another subset of fluid actuators.
- 1 12. A fluid control system comprising:
2 a system controller; and
3 a fluidic die comprising:
4 a set of fluid actuators arranged in an order;
5 a plurality of delay elements associated with the fluid actuators, the
6 delay elements if activated to delay an activation signal;
7 a register to store input control information relating to controlling
8 actuation of the set of fluid actuators; and
9 an actuation controller to:
10 determine, based on the input control information, which fluid
11 actuators of the set of fluid actuators are to be actuated,
12 activate delay elements associated with first fluid actuators that
13 are to be actuated, if respective neighboring fluid actuators of the first fluid actuators
14 are not to be actuated,
15 deactivate delay elements associated with second actuators that
16 are to be actuated, if respective neighboring fluid actuators of the second fluid
17 actuators are to be actuated, and
18 deactivate delay elements associated with third fluid actuators
19 that are not to be actuated.

1 13. The fluid control system of claim 12, wherein the register comprises an
2 actuation data register to store actuation data that indicates each fluid actuator of the
3 plurality of fluid actuators to actuate, the fluidic die further comprising:

4 a mask register to store a mask data pattern indicating a respective set of fluid
5 actuators of the set of fluid actuators enabled for actuation for an actuation event,

6 wherein the actuation controller is to determine which fluid actuators of the set
7 of fluid actuators are to be actuated further based on the mask data pattern.

1 14. A fluid control device comprising:

2 a set of fluid actuators arranged in an order;

3 an actuation data register to store actuation data that indicates each fluid
4 actuator of the plurality of fluid actuators to actuate;

5 a mask register to store a mask data pattern indicating a respective subset of
6 fluid actuators of the set of fluid actuators enabled for actuation for a respective
7 actuation event; and

8 a controller to:

9 determine, based on the actuation data and the mask data pattern,
10 whether a given fluid actuator of the set of fluid actuators is to be actuated, and

11 in response to determining that the given fluid actuator is to be
12 actuated and a neighboring fluid actuator within a specified proximity of the given
13 fluid actuator is not to be actuated, activate a delay element associated with the
14 given fluid actuator, the delay element to delay an activation signal propagated to
15 selected fluid actuators of the set of fluid actuators in response to an actuation event.

1 15. The fluid control device of claim 14, wherein the mask data pattern in the
2 mask register is dynamically updateable to provide different drop weights of fluid
3 dispensed by the fluid control device towards a target.

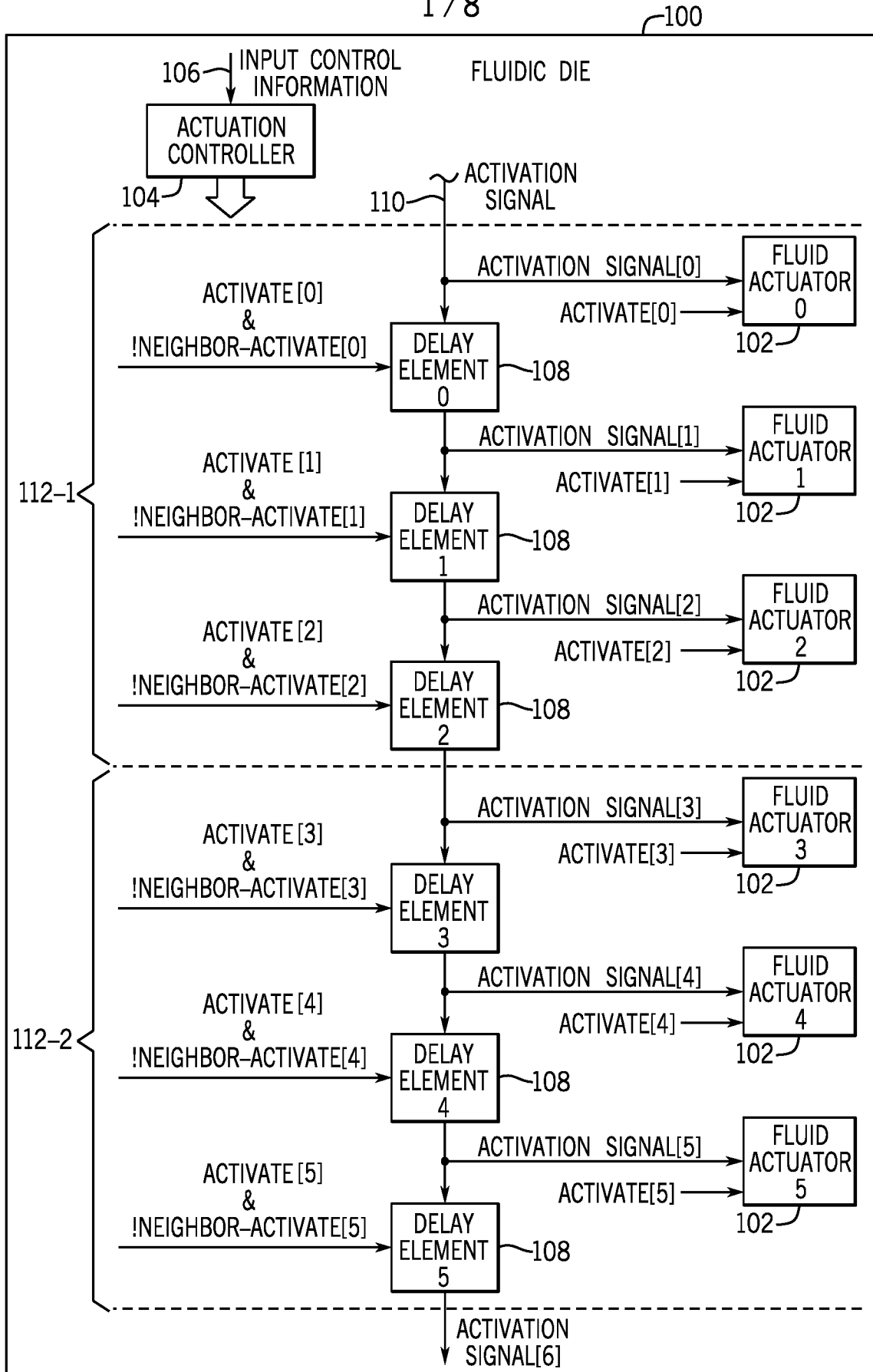


FIG. 1

		ACTUATE	DELAY ACTIVE
VIRTUAL PRIMITIVE 0	FLUID ACTUATOR 0	YES	NO
	FLUID ACTUATOR 1	NO	NO
	FLUID ACTUATOR 2	YES	YES
VIRTUAL PRIMITIVE 1	FLUID ACTUATOR 3	NO	NO
	FLUID ACTUATOR 4	NO	NO
	FLUID ACTUATOR 5	NO	NO
VIRTUAL PRIMITIVE 2	FLUID ACTUATOR 6	YES	NO
	FLUID ACTUATOR 7	YES	NO
	FLUID ACTUATOR 8	YES	YES
VIRTUAL PRIMITIVE 3	FLUID ACTUATOR 9	NO	NO
	FLUID ACTUATOR 10	YES	YES
	FLUID ACTUATOR 11	NO	NO

202
204
206

FIG. 2

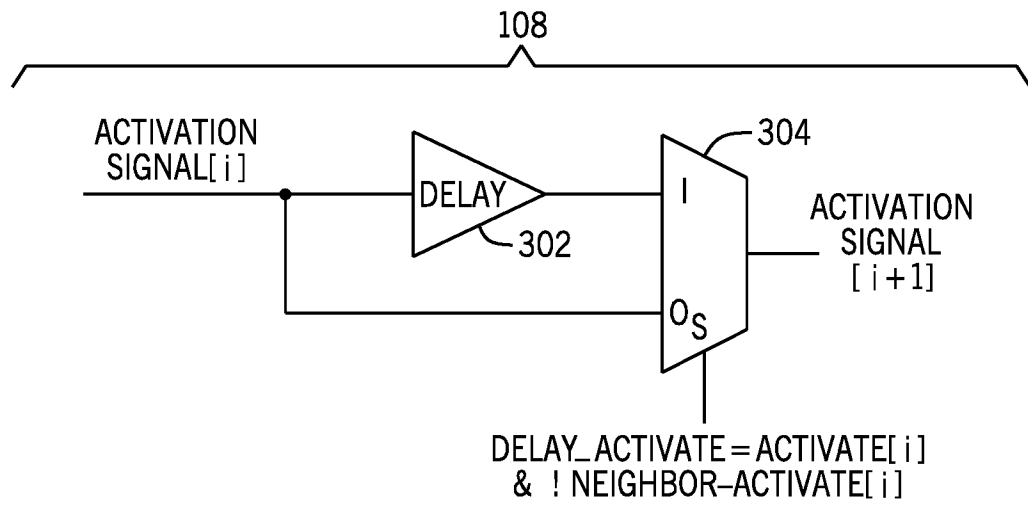


FIG. 3

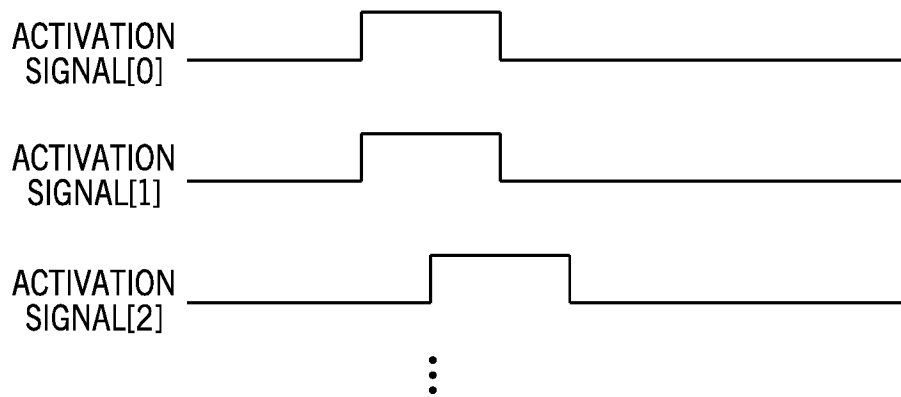


FIG. 4

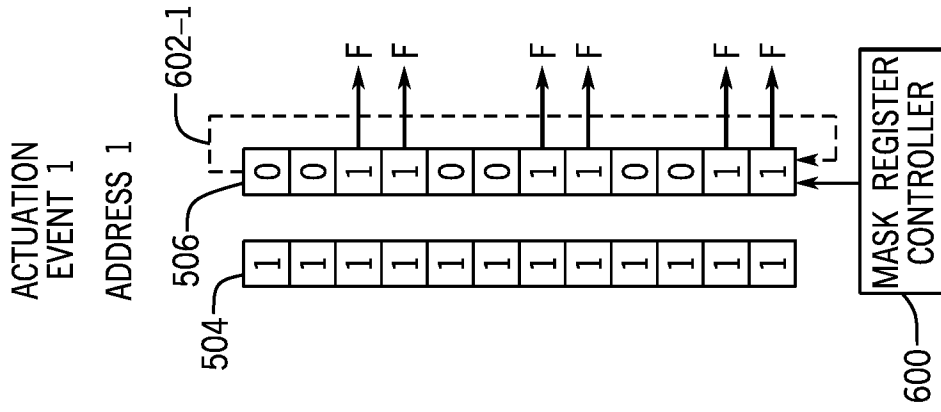


FIG. 6A

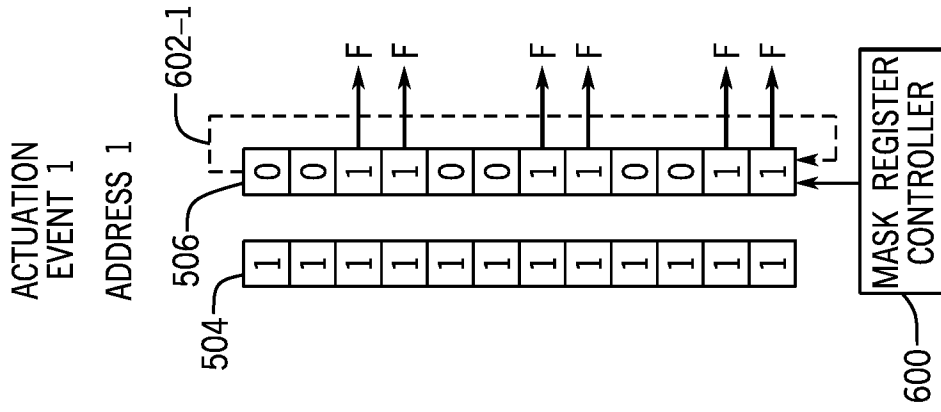


FIG. 6B

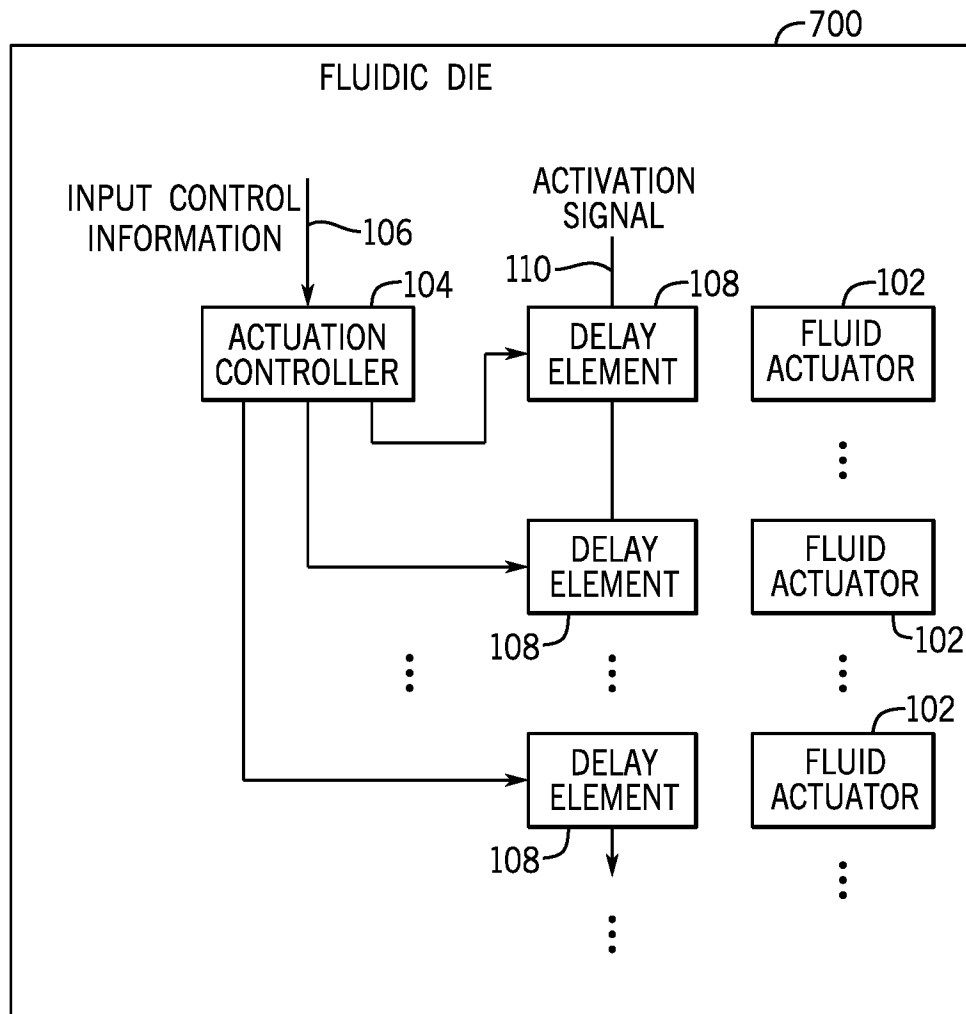


FIG. 7

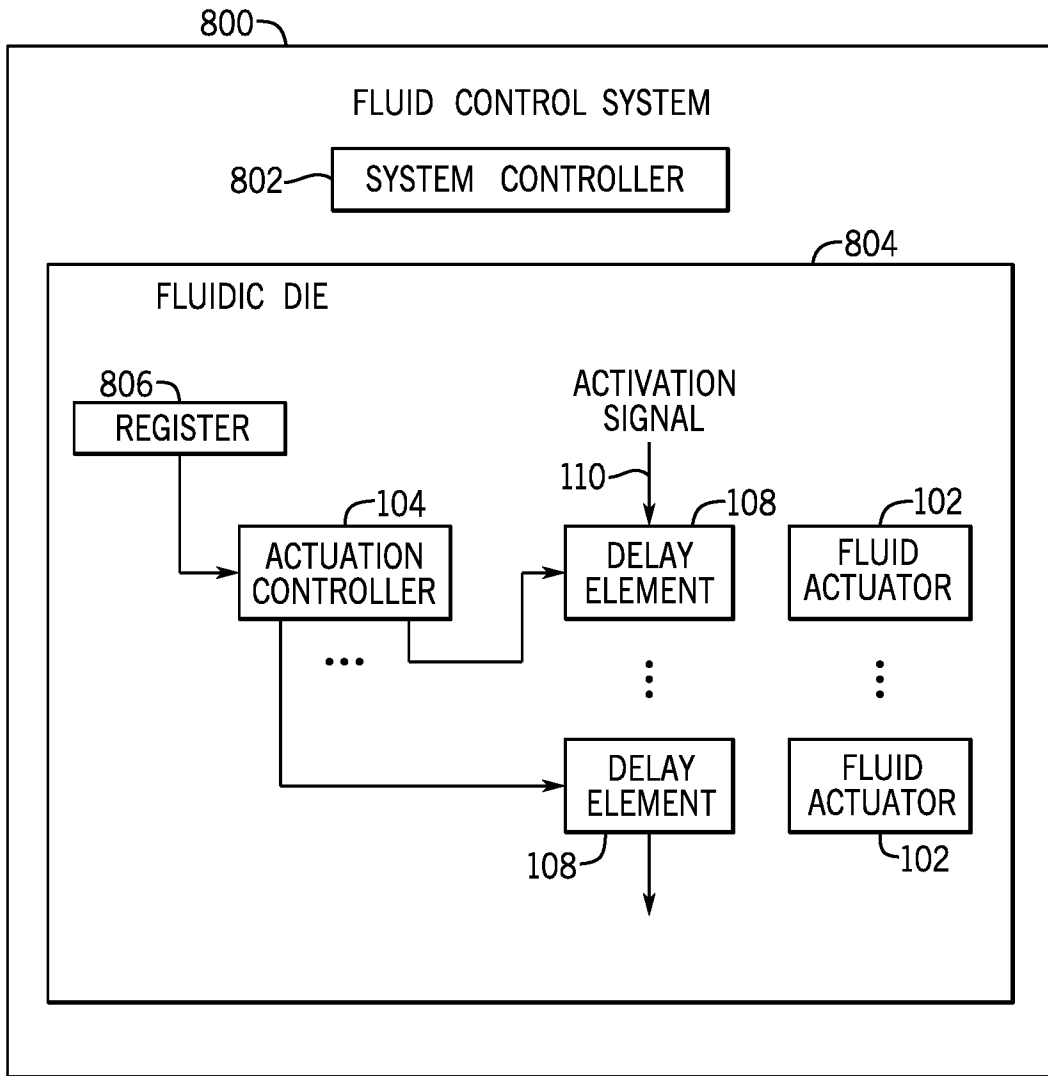


FIG. 8

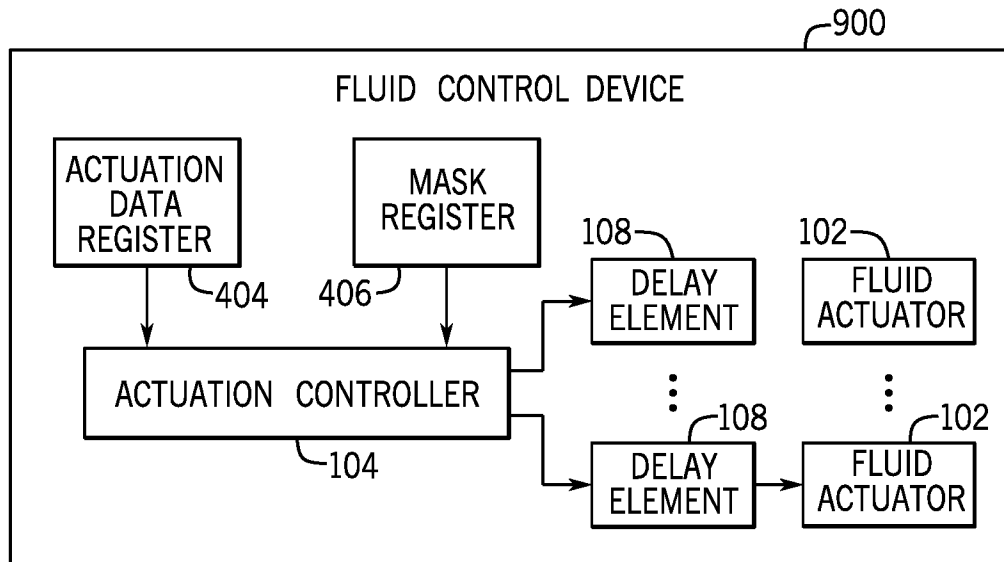


FIG. 9

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US 2017/027562

A. CLASSIFICATION OF SUBJECT MATTER		
<i>B41J 2/175 (2006.01)</i> <i>B41J 29/38 (2006.01)</i>		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols)		
B41J 2/00, 2/215, 29/00, 29/38, 2/175		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)		
PatSearch (RUPTO internal), USPTO, PAJ, Esp@cenet, Information Retrieval System of FIPS		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2012/0120138 A1 (NEEL BANERJEE et al) 17.05.2012, abstract, fig. 2,6-8, paragraphs [0023],[0027]-[0028],[0032],[0034]-[0043],[0052]	1-15
A	US 8348374 B2 (TOSHIBA TEC KABUSHIKI KAISHA) 08.01.2013	1-15
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/> See patent family annex.		
*	Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family
"A"	document defining the general state of the art which is not considered to be of particular relevance	
"E"	earlier document but published on or after the international filing date	
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	
"O"	document referring to an oral disclosure, use, exhibition or other means	
"P"	document published prior to the international filing date but later than the priority date claimed	
Date of the actual completion of the international search		Date of mailing of the international search report
29 September 2017 (29.09.2017)		30 November 2017 (30.11.2017)
Name and mailing address of the ISA/RU: Federal Institute of Industrial Property, Berezhkovskaya nab., 30-1, Moscow, G-59, GSP-3, Russia, 125993 Facsimile No: (8-495) 531-63-18, (8-499) 243-33-37		Authorized officer S. Zhuravlev Telephone No. (495) 531-64-81