Spike processing model of the brain

Robert Alan Brown

8 Foster Street

Mattapoisett, Massachusetts, 02739

E-mail: robert.brown21@comcast.net

Abstract

The timing of a short burst of energy (spike) within a specific time period is used to identify a place in space (input terminal) and/or detect changes in energy or position in the environment. Spike timing is used to determine the motion of an actuator or the activation of a place in space (output terminal). The timing of a spike is specified by a sensor or a time delay memory cell that is preset (predetermined) or set through experience (empirical). Time delay memory cells are arranged in decoding networks that activate specific output terminals based upon the timing of incoming spike trains, or arranged in encoding networks that generate spike trains from activated input terminals. These spike trains form semi-axes that can transmit large quantities of information in one direction through a single conductor. These spike train semi-axes are essential in the transmission of information from peripheral neurons to and from the brain through the spinal chord. Means are shown that merge multiple semi-axes in a multi-dimensional collective network in the brain to produce coordinated behavior. To optimize performance, the collective network is divided into a set of collective networks that contain semi-axes that act at the same time.

KEYWORDS: Collective networks, decoding networks, empirical networks, encoding

networks, neural spikes, predetermined networks, sparcification, spike-timing dependent plasticity (STDP), temporal semi-axes, time delay memory cells.

1. Introduction

The present view of the brain is that memory and behavior are established by the “strength” of connections at the synapses between neurons. Learning and memory are attributed to physical or chemical changes in these neuronal synapses, and/or the growth of new connections. This paper proposes an alternative theory; that memory and behavior are determined by small changes in the timing of the spikes produced by neurons within repeating periods. These spike timings are established by time delays stored within memory neurons. These time delays are predetermined genetically, or are established through experience, and are sufficient to explain the memory and plasticity of the brain without requiring changes in synaptic connections.

Studies of STDP (spike-timing dependent plasticity) have focused upon changes in long-term potentiation (LTP) and long-term depression (LTP) of a synapse according to the time of arrival of a spike in relation to the time of its post synaptic spike. This paper shows a simple and practical control system based upon the use of timers alone.

2. Methods

Starting with the design of a spike generator, this paper develops a simple and logical model of a neural based information processing system based upon spike timing alone. Each system is a complete sensor/actuator network capable of performing some useful function. Small additions or alterations are made to each system that widen and improve their functionality, much in the same way that organisms evolve, resulting in a comprehensive brain model.

3. Results

The spike processing systems in this paper show that timers can form a viable control system that appears similar to vertebrate nervous system. The memory and plasticity of a system can be changed while the memory cells in this system remain connected. These changes are due to changes in the spike time delays stored within memory cells. These stored spike times change (decode) the path of activation of memory cells according to input spike trains, and create (encode) new spike trains. These spike trains can transmit many different messages over a single conductor in one direction using very little energy, forming a semi-axis. Semi-axes can be grouped into a collective network that forms a multi-dimensional system. The collective network works best when all of its semi-axes are active at a given time. Thus, the collective network needs to be divided up into a set collective networks made up of semi-axes that are active at the same time. This parceling process can be carried out by an algorithm in which semi-axes that never or seldom occur together are deleted from a given collective network.

The systems shown in this paper are based upon a single spike start signal frequency and variations in input spike timing that occur in roughly the period of the start signal frequency. Progress has been made in designing multi-axes systems using frequencies that are multiples of a base frequency to deal with longer and shorter input spike periods. Also, much progress has been made in the design of inertial navigation and voice recognition systems using this spike process.

4. Discussion

The first part of this paper shows systems that are capable of functioning in a predictable and desirable way by using networks of memory cells having predetermined time delay memory settings and spike timings. The second part shows how the time delay memory settings in these networks can be established through experience, also.

4.1 Sensor/actuator machine

All living and non-living machines sense conditions in their environment and produce actions that contribute to their continued operation. This section shows how a sensor/actuator machine can operate using spikes with just three components: a spike generator, timing unit, and actuator.

4.1.1 Spikes

An electrical spike can be generated by the one-shot spike circuit shown in figure 1.

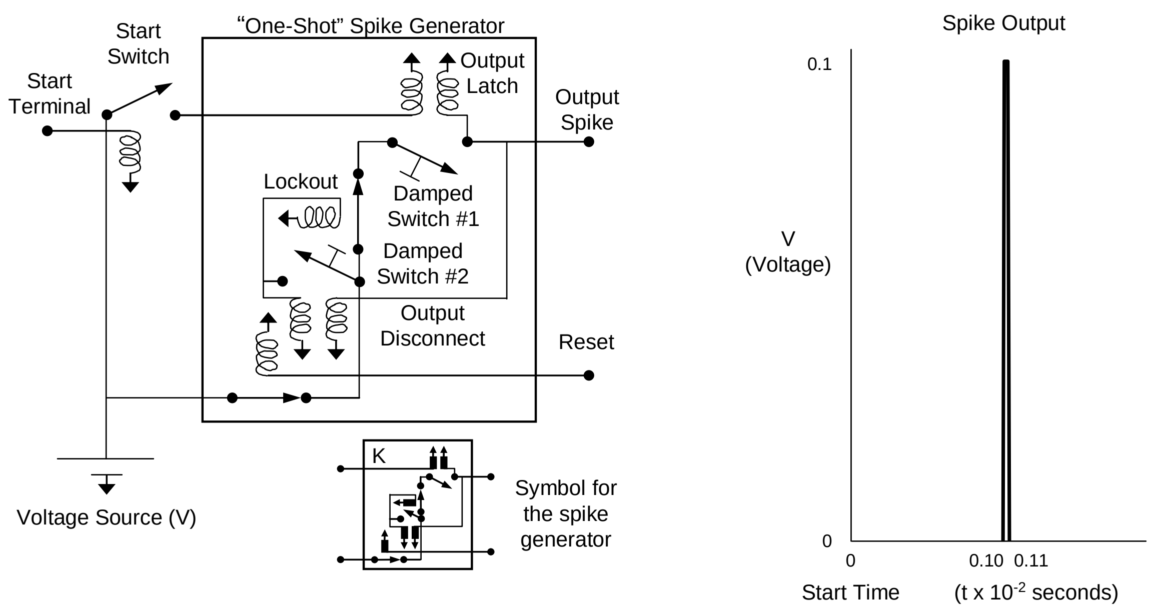


Figure 1. The *spike generator* produces a voltage spike of high power over a short time period that consumes little energy.

When a voltage is introduced at the start terminal, or the normally open start switch is closed physical, output latch switch #1 is closed and the voltage (V) of the voltage source appears at the output terminal. The output latch keeps the contacts from rebounding, and causes the lockout circuit to interrupt the voltage to the output spike terminal within some short time period according to the degree of damping in switch #2.

This sequence of actions creates the spike output curve shown on figure 1. The time of the spike after the start signal is determined by damper #1, and the width of the spike is determined by damper #2. If the width if the spike is very short, say 0.0001 of a second, a small amount of electrical energy produces a spike of high power. The lockout uses the voltage source to keep open its normally closed switch. This prevents a second spike from occurring if the start signal or physical contact is maintained longer than the spike time. A reset is needed to restore the lockout to its normally closed position, in preparation for the next spike. This basic process of turning something on and then having it turn itself off nearly instantly is used throughout all of the networks in this paper. The low energy, high power characteristics of spikes[1] makes them ideal for detecting and processing changes in energy, particularly when these energy changes occur at high frequency.

4.1.2 Bi-directional actuator

The timing of several spikes can be used to determine the net displacement of the actuator shown in figure 2.

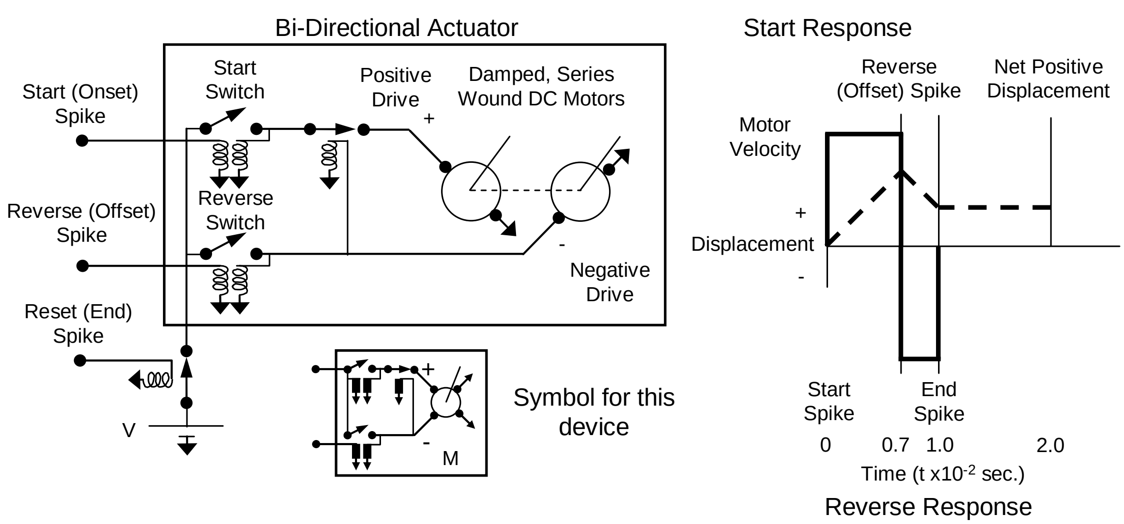


Figure 2. The displacement of this *bi-directional actuator* is determined by the timing of a start (onset) spike, a reversing (offset) spike, and a reset (end) spike.

Two series wound direct current motors[2] can be connected as shown in figure 2. The left side motor starts to produce a positive displacement when the start spike switch is closed. When the reverse spike switch contact is closed, the voltage from the start switch is interrupted, and the right side motor begins to produce a negative displacement that continues until the end spike. The shafts of the two motors are connected so they run together in either direction. Thus, the net positive or negative displacement of the motors is determined by the timing of the three spikes.

The low inertia motors are highly damped electrically, and produce the maximum torque at zero speed. So they reach their maximum speed nearly instantly, and run at a constant speed determined by the input voltage. A symbolic representation of the bi-directional actuator is shown wherein the two motors are considered as one.

4.1.3 Using the bi-directional actuator as a timer

The bi-directional actuator in figure 2 can replace the damped switch #1 in the spike generator in figure 1, creating the adjustable timer in figure 3.

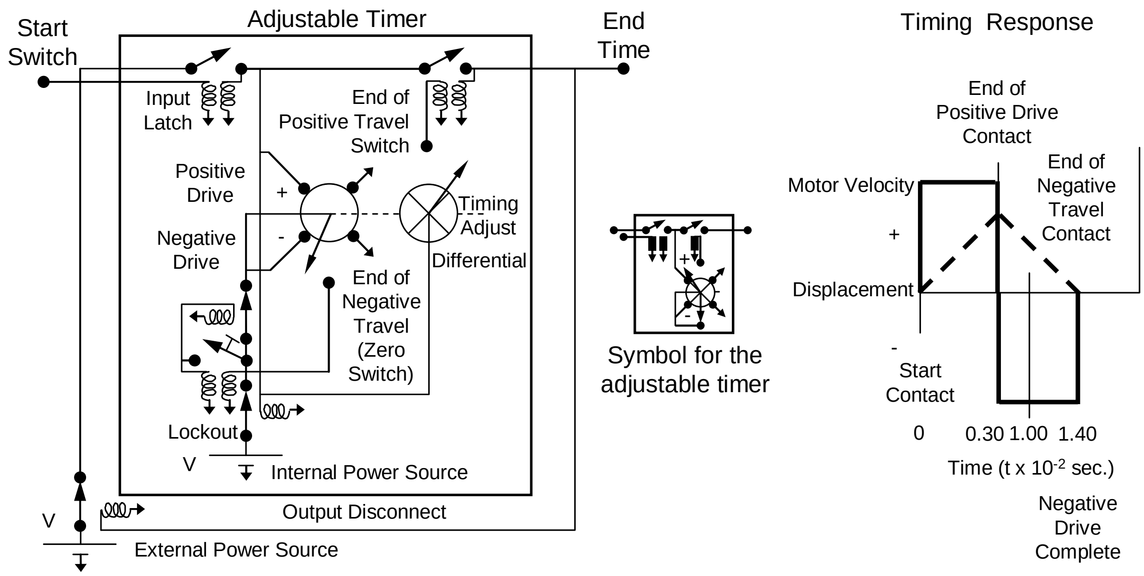


Figure 3. The *adjustable timer* is the basic memory element in a temporal process.

A signal at the start switch closes the input latch, which disconnects the internal voltage source and causes the motor to rotate in the clockwise, positive (+), direction, driving the side member of the differential in the counter-clockwise, negative (-), direction. The other side member of the differential is held fixed by the timing adjustment. This causes a contact mounted upon the differential shaft to rotate in the negative direction until it contacts the end of positive travel switch. This produces an end time output signal, which is used to release the input latch through the output disconnect. This shuts off the positive drive circuit, and allows the negative circuit to drive the motor in the negative direction until its negative drive contact hits the end of negative travel switch. This opens the negative drive disconnect switch, which terminates the motion of the motor.

The time delay between the occurrences of the start signal and the end of the positive travel contact is determined by the speed of the motor and the angle between the positive travel contact arm and negative travel contact arm. When the end of positive travel contact and end of negative travel contact are 135 degrees apart (measuring counter-clockwise), the motor rotates 45 degrees before contact is made with the end of positive travel switch. The maximum time delay occurs when the contact arms are slightly more than 180 degrees apart, and the minimum time delay occurs when they are slightly less than 180 degrees apart.

Since the motor drives one side member of the differential, and the other side member is restrained by the timing adjustment, the distance that the positive travel contract arm moves is determined by the position of the restraining side member of the differential. Thus, the time delay is determined by the position of it retaining side member, which can be set manually to any position, and left in that position for as long as is desired, forming a predetermined memory cell.

Later, it will be shown that the position of the retaining side member can be adjusted by a system of sensors, actuators, and timers to form an empirical memory cell. However, the purpose of this section is to show systems that allow these predetermined memory cells to produce useful outcomes.

4.1.4 Sensor/actuator system

The damped switch #1 in spike generator shown in figure 1 can be replaced with the ballistic switches in the motion sensors K0 and K1 shown in figure 4 that work together like a telegraph key.

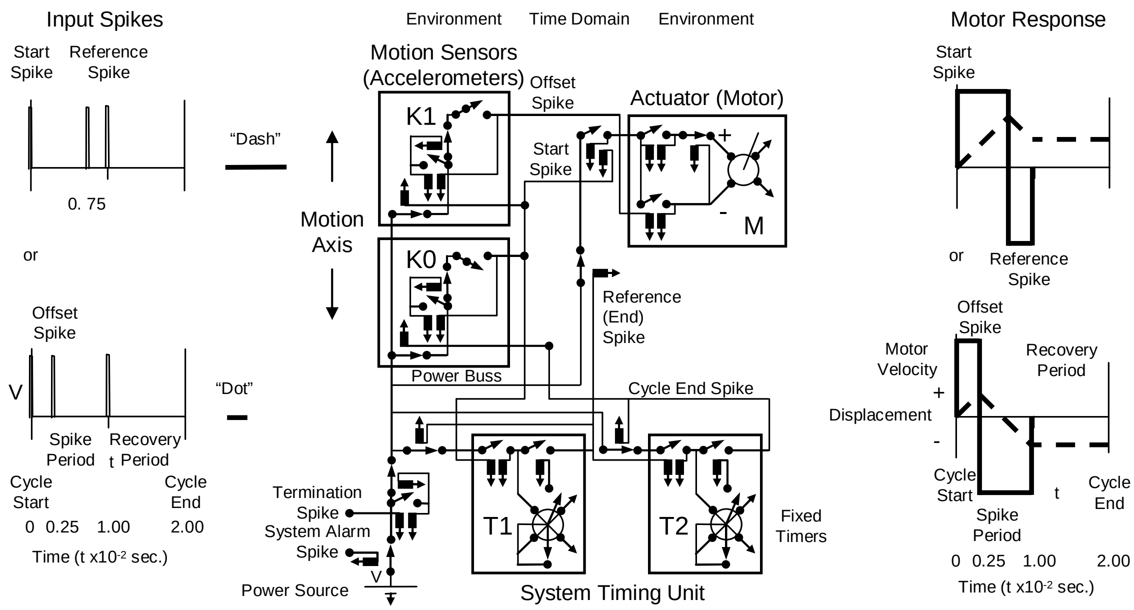


Figure 4. The *sensor/actuator system* produces an action according to the timing of input spikes.

The K0 motion sensor produces a spike when its ballistic switch is closed by a sudden downward movement. This produces a start spike that starts the first fixed timer (T1) in the system timing unit, releases the lockout in the K1, and starts the positive drive of the actuator (M). When there is a sudden upward motion, K1 sends an offset spike to the actuator that reverses its motion until T1 times out and produces a reference spike that shuts off the actuator, starts T2, and resets itself. When T2 times out it produces a cycle end spike that resets K0 so it can produce a new start spike, and resets itself.

The time between the start spike and the offset spike determines the positive displacement of the actuator that can be represented by a "dot" or a "dash". The time between the offset spike and the reference spike determines the negative drive of the actuator. The ratio of the positive drive to the negative drive determines the net output displacement of the actuator. Thus, the actuator response is determined by the input spike timing and the physical characteristics of the electrical and mechanical elements of this machine only.

4.2 Decoding networks

An animal or machine may need to change the way it responds to its environment. This can be done by transforming an input spike pattern into a new spike pattern. However, this transformation requires that the spike timing of the motion sensors be decoded first into a value of space (place) variable that represents a unique input spike timing. Then this value of the place variable can be used to generate (encode) the new spike timing.

Decoding is the process of representing a particular combination of many quantities by one quantity. In telegraphy, a series of dots and dashes are converted to a single letter or number. In this section, spikes will be generated by a telegraph key or a sensor that reacts to events or energy in the environment, such as acceleration. The timing of these spikes are measured and decoded into a position in space such as a typewriter key that can be used to represent an object such as a letter or a number.

4.2.1 Concurrence cell

Different spike times can be distinguished by the concurrence cell in figure 5 that measures the maximum time difference (non-concurrence) among a set of spike timings. This is accomplished by adding the bi-directional actuator shown in figure 2 to a concurrence logic network shown in figure 5.

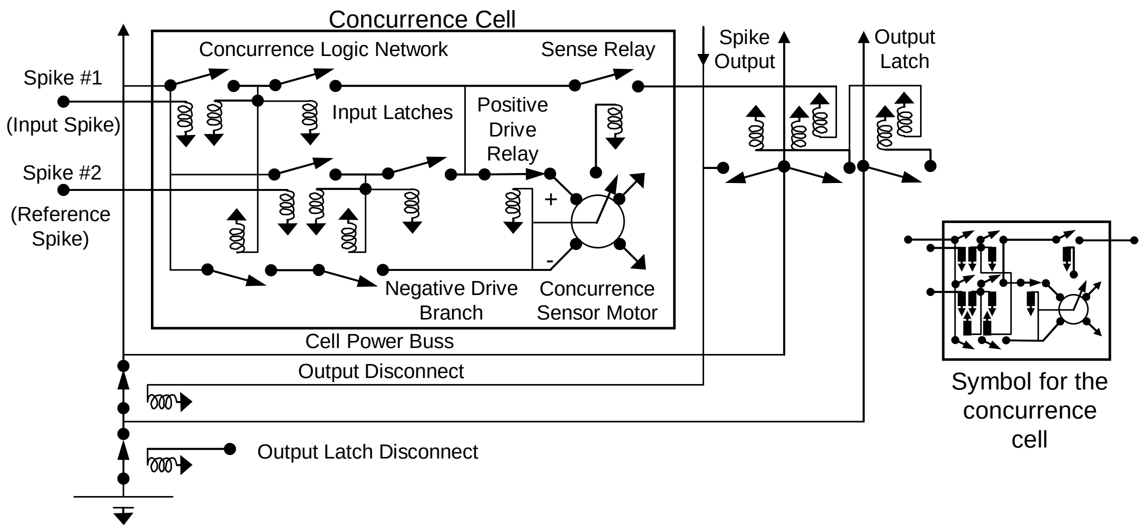


Figure 5. The *concurrence cell* measures the maximum time difference among incoming spikes.

Please assume that the contact of the concurrence sensor motor is at the sense relay terminal. When an input spike (#1) occurs, the first two switches in the top row of switches and one of the bottom switches will close, causing the motor to run some distance in the clockwise direction away from the sense relay contact. Then when both spikes have occurred, all of the bottom row of switches will close, causing the negative drive branch to take over, and run the motor back to the sense relay contact. This causes an output to occur that produces a spike that shuts down the cell, leaves the motor contact at the sense relay, and leaves a voltage on the output latch.

If spike #1 and spike #2 occur close together in time, the motor will run very little, closing the sense relay contacts very soon after the two spikes. Thus, the concurrence sensor measures the absolute time difference between the two spikes. Spike (#2) is designated as the reference spike that is made to occur at the end of the spike period by the system timer T1 shown in figure 4. Thus, if the input spike #1 is made to occur close to the reference spike #2, the concurrence unit will produce an output very close to the end of the spike period.

4.2.2 Decoding network for measuring concurrency

Different spike times can be identified by a set of memory cell timers that are set to different spike times, and concurrence measuring cells that identify which set comes closest to the input spike timing. This can be accomplished by replacing the actuator in figure 4 with a set of decoding and concurrence cells, as shown in the sensor decoding system in figure 6.

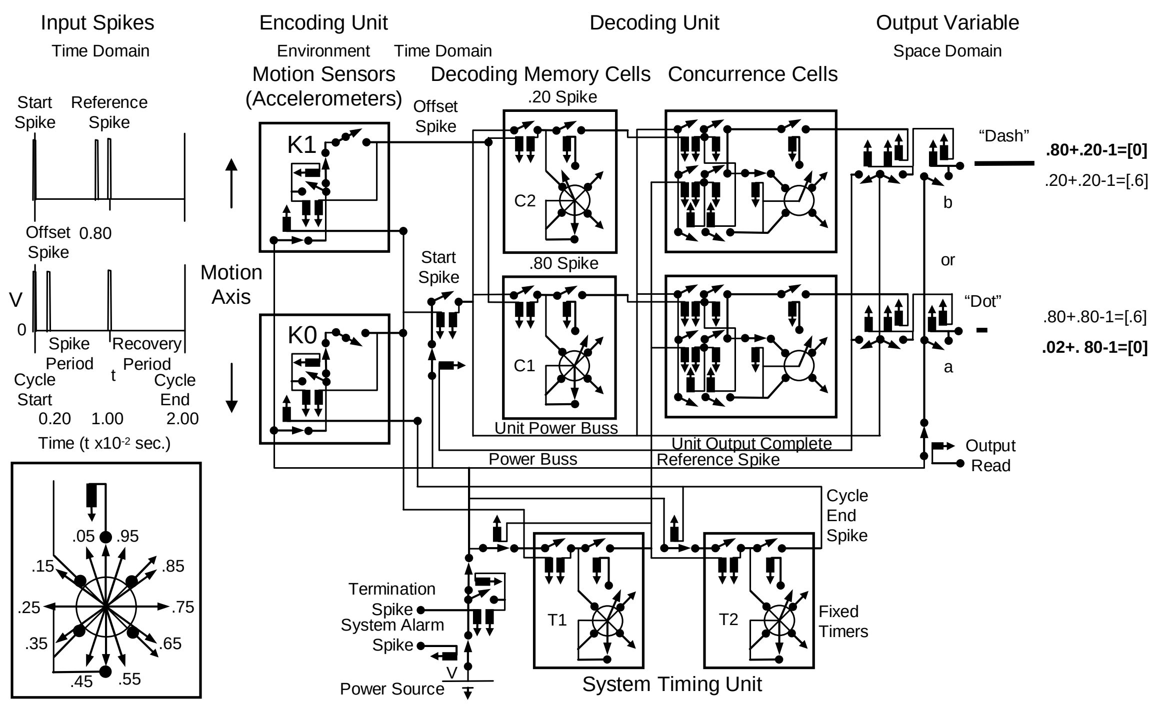


Figure 6. The *sensor decoding system* selects an output terminal that best represents an input spike pattern.

The decoding memory cells (C1 and C2) are connected to the network so that they are energized by the start spike from K0, but their timers is not started until they are triggered by an input spike from the motion sensor K1. The time delay of a decoding memory cell is set to the converse (reference spike time – input spike time) of an input spike. This allows the decoding memory cell to produce an output spike as close a possible to the reference spike when that input spike appears.

Thus, when the input spike occurs early in the spike cycle, the cell in the row in which the memory cell is set for an early spike, having a long delay, will produce a spike closest to the time of the reference spike, and cause the concurrence cell to select an output, in this case (a). This terminates the selection of that unit in that spike period so that only one output terminal can be selected in a spike period, and leaves a voltage on the output terminal (a).

If the input spike occurs late in the spike cycle, the row with the memory cell set for a short delay will produce an output closest to the reference spike, causing its concurrence cell to produce an output first, leaving a voltage on the output terminal (b). Thus, the decoding memory and concurrence cells in a decoding unit select the output terminal that represents a spike time closest to the input spike. This transforms a given spike timing into a unique value of place (space) variable. The “Output Read” reset spike releases the latched output so any new output can be selected in the next spike cycle.

4.2.3 Divergent (decoding) network

The decoding unit in figure 6 can be duplicated multiple times and connected into the divergent network shown in figure 7.

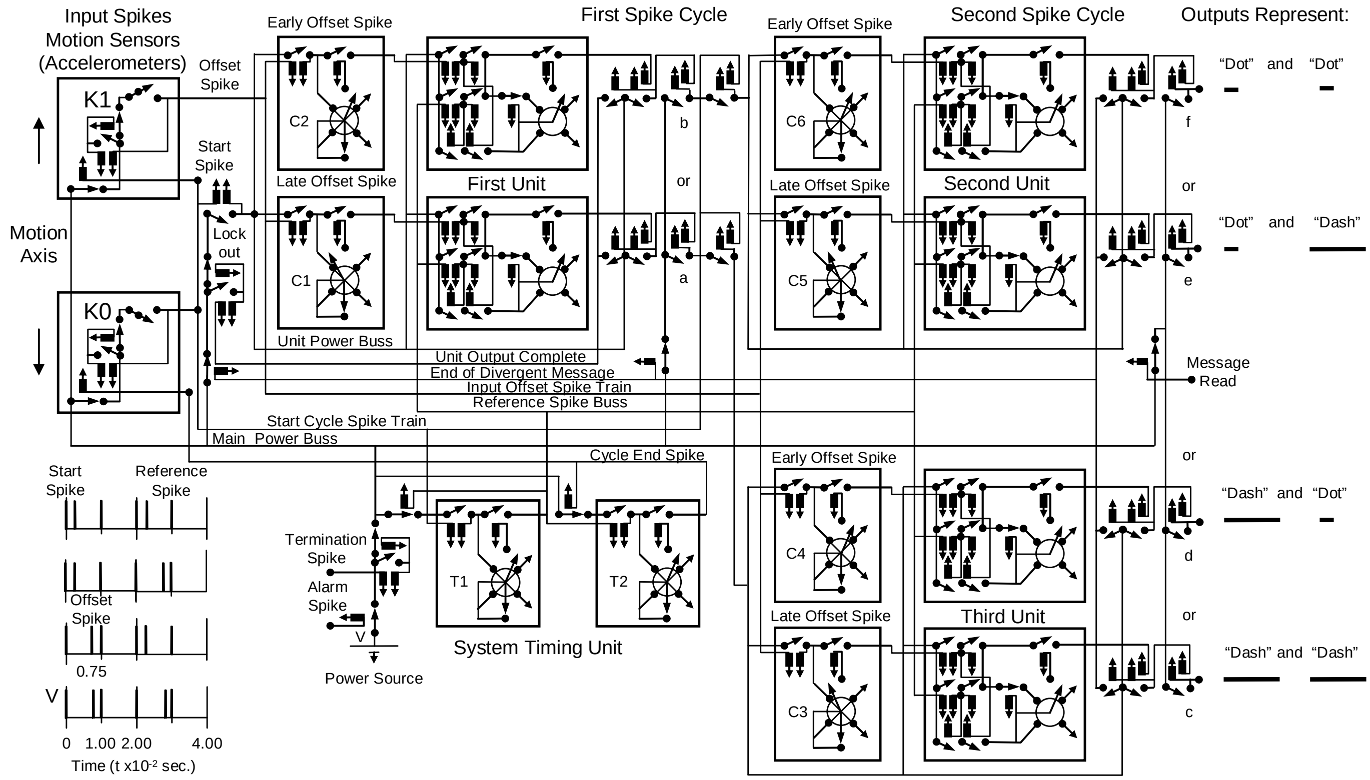


Figure 7. The *divergent network* produces an output that represents a unique sequence of spikes.

The output (a) or (b) of the first unit activates one or the other of the second set of units, one of which selects an output according to the second spike timing. Each unit resets itself after it produces an output, but a lockout is needed to deactivate the first unit until the last unit produces and output so that only one unit is active in any spike period. As shown in this figure, the dichotomic divergent network can be used to decode a temporal code similar to Morse code.

4.3 Encoding Networks

Encoding is the process of representing a single quantity by some combination of many quantities. In telegraphy, a single quantity such as a letter or number is represented by a specific set of dots and dashes, each of which can be produced by a spike generated at a particular moment in time within a specified time period. This section deals with the generation of spike trains by activating a value of a place variable that is represented by a unique timing memory cell, and using its stored spike time value to control motion and transmit information.

4.3.1 Memory/actuator system

The motion sensors K0 and K1 in figure 4 can be replaced by the adjustable timers C1 and C2 shown in figure 3, and a spike generator added, forming the memory/actuator system shown in figure 8.

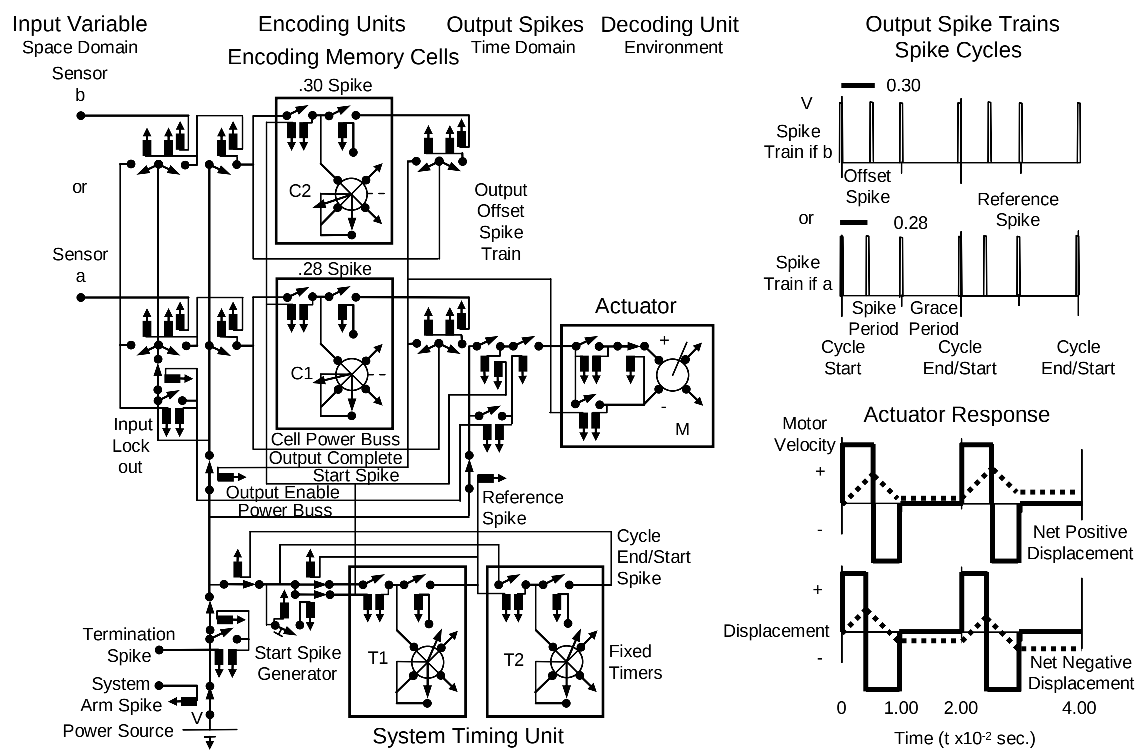


Figure 8. The *memory/actuator system* drives the actuator to a particular position according the stored spike timing in a selected (activated) encoding memory cell.

A system arm spike starts the start spike generator and the system timing unit, which cycles through the start/end spike and reference spike repeatedly, forming a series of blank spike periods and grace periods. When an input signal occurs at one of the place terminals (a) or (b), its encoding memory cell is turned on (activated), the lockout is energized which prohibits any addition input signals from being accepted in that spike period, and an output enable spike is generated that allows the actuator to operate when it receives the next start spike. When the system timing unit T2 causes this next start spike, the timer in the activated encoding memory cell and the positive drive of the actuator are started. This memory cell then produces an output (offset) spike on its output complete buss according to its stored time delay, which resets the lockout and ends the positive displacement of the actuator by reversing the rotation of the actuator, and the actuator runs in this direction until the system timing unit T1 produces a reference spike, as shown in figure 4.

The spike timing of the encoding cell can be represented by a line originating at the start of the spike cycle, and ending at the time of the offset spike, as shown in figure 8. It moves in one direction from the present (start), only. Thus, a single spike system can be considered a half dimension like time, rather than a full dimension that allows a point to move in both directions. This half dimension will be referred to as a semi-axis.

The resulting spike cycles and actuator responses are shown for arbitrarily selected predetermined time delay settings. The system will produce output spikes and displacements as long as an input terminal (a) or (b) is energized, or until the termination terminal is energized.

4.3.2 Convergent network

The encoding unit in figure 8 can be duplicated multiple times and used to form the dichotomic convergent network in figure 9.

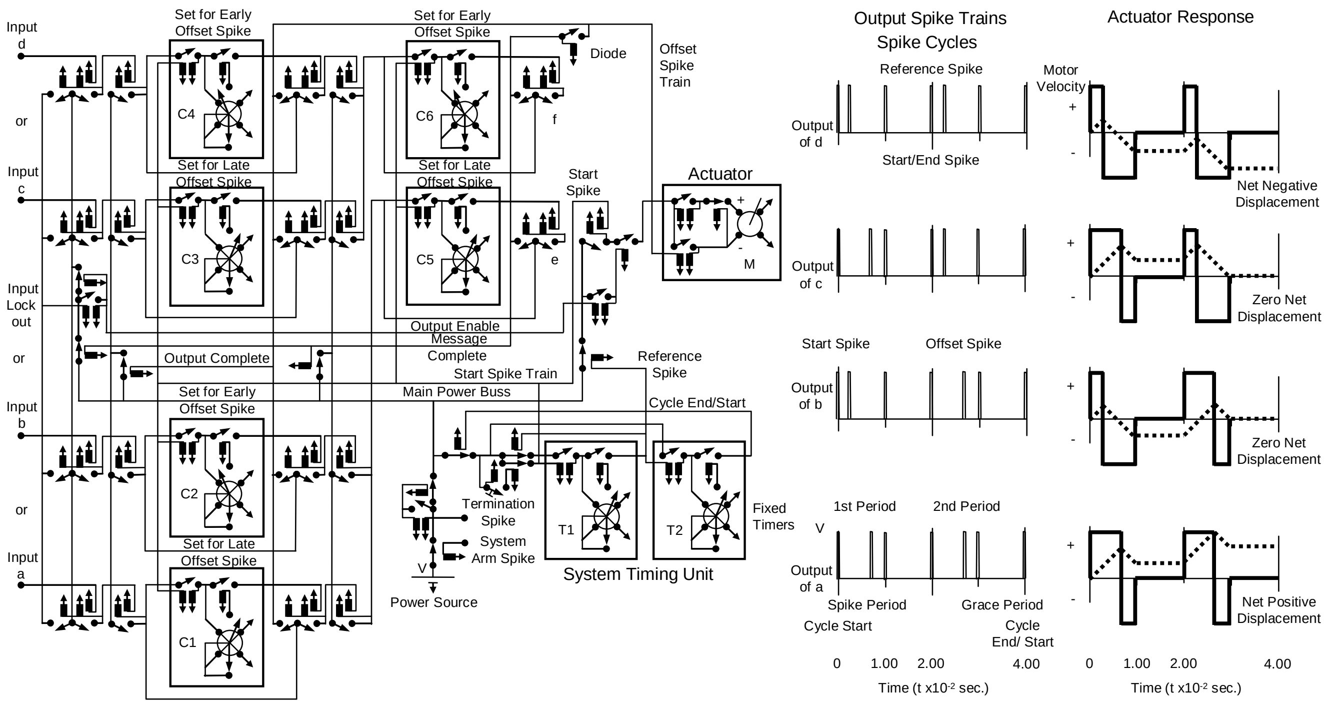


Figure 9. The *convergent network* produces a unique spike train according to spike time stored in the path of the activated memory cells.

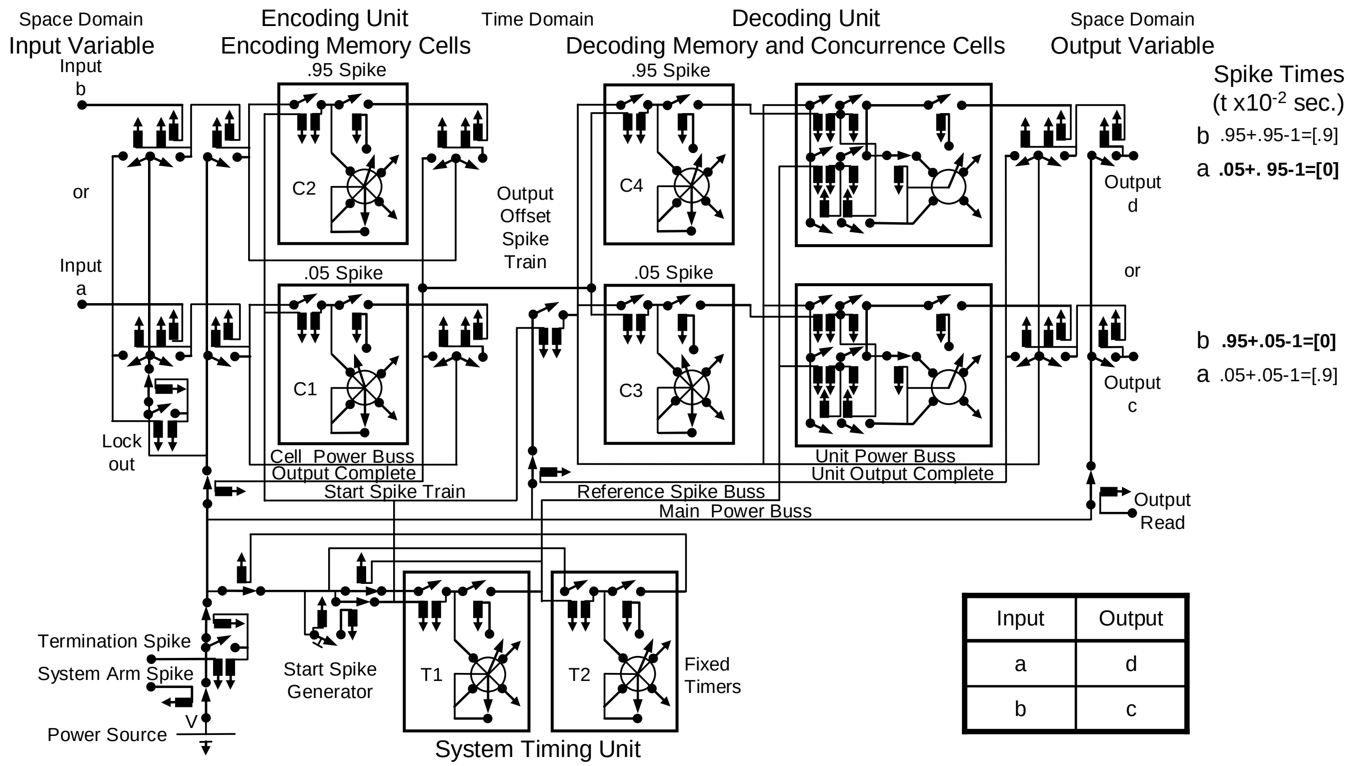
The activation of an encoding memory cells in the first column causes the activation of one memory cell in the second column in a fixed convergent pattern. The diode between the output complete buss in the first column and the second column prohibits the spike generated in the first column from appearing on the output buss (message complete) of the second column. Thus, the spike train produced by the convergent network is a unique spike train that represents the input terminal (a), (b), (c), or (d).

4.4 Encoding/decoding networks

An encoding network can be used to generate spikes in place of the motion sensors shown in figure 6, forming an encoding/decoding network that transforms a value of an input place variable into a unique place value of an output variable. This creates a system capable of transmitting values of place variables (letters and numbers) over great distances over a single wire using a temporal code.

4.4.1 *Encoding/decoding system*

The encoding system in figure 8 can be connected to the decoding system in figure 6 by a single conductor, forming the encoding/decoding system shown in figure 10.



Input	Output
a	d
b	c

Figure 10. The *encoding/decoding system* transmits information in one direction over a single conductor.

An input at (a) or (b) produces an output at (c) or (d) within one spike cycle according to the time delay settings of the encoding and decoding memory cells. The two units are synchronized by the same start time and reference signal created by the system start spike generator and fixed timers T1 and T2. If the inputs and outputs are two keys of two electric typewriters, an input key can be set to activate a given output key by setting the memory timers appropriately. An example of the transition between input values and output values is shown in the table in figure 10.

4.4.2 *Convergent/divergent network*

The encoding and decoding units in figure 10 can be expanded, forming the convergent/divergent network in figure 11.

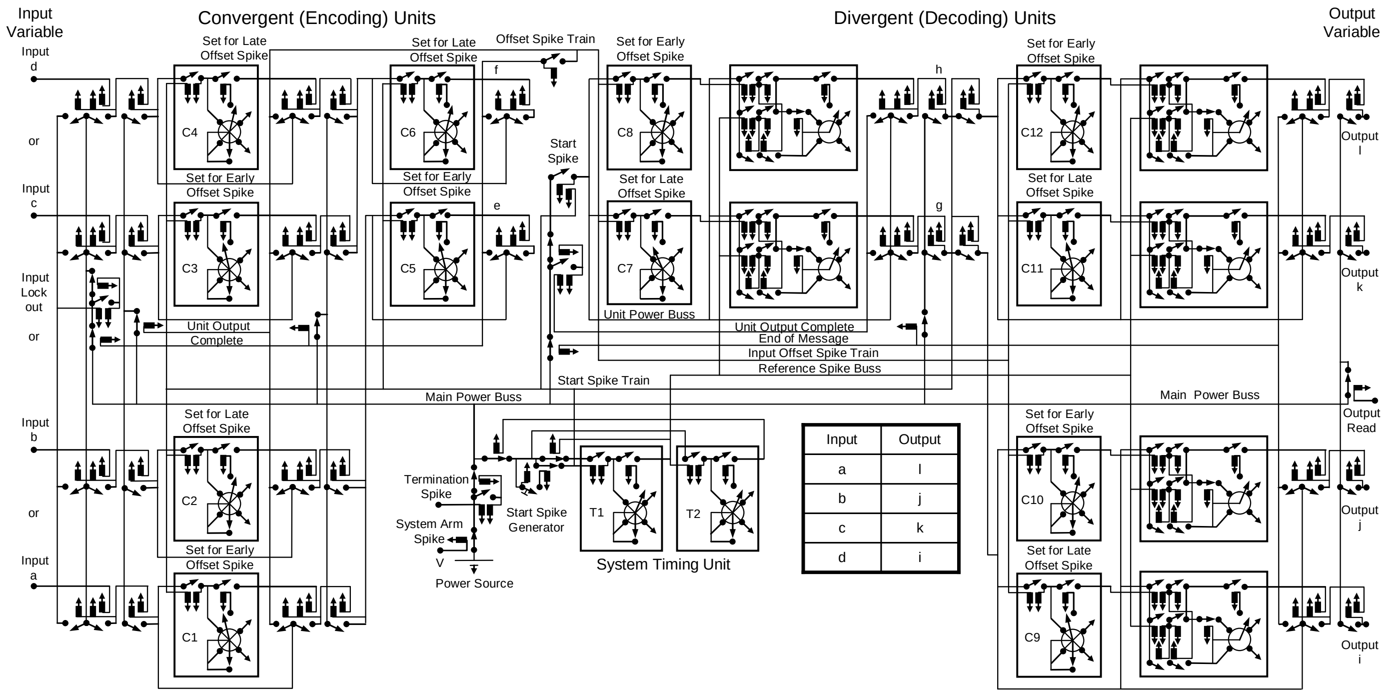


Figure 11. The *convergent/divergent network* can be expanded to transmit a large amount of information from one place to another over a single conductor.

The convergent network produces a unique offset spike train from a given value (a), (b), (c), or (d) of the input place variable according to the time delay settings of its memory cells. According to the time delay settings of its memory cells of the divergent network, this offset spike train produces a value (i), (j), (k), or (l) of the output place variable, as shown in the truth table. This transformation is carried out using only six memory cells. A square connection matrix requires sixteen memory cells to accomplish the same transformation.

4.4.3 Convergent/divergent network for sending and receiving Morse code

The convergent/divergent shown in figure 11 can be expanded to form the binary, six-level convergent/divergent network shown in figure 12.

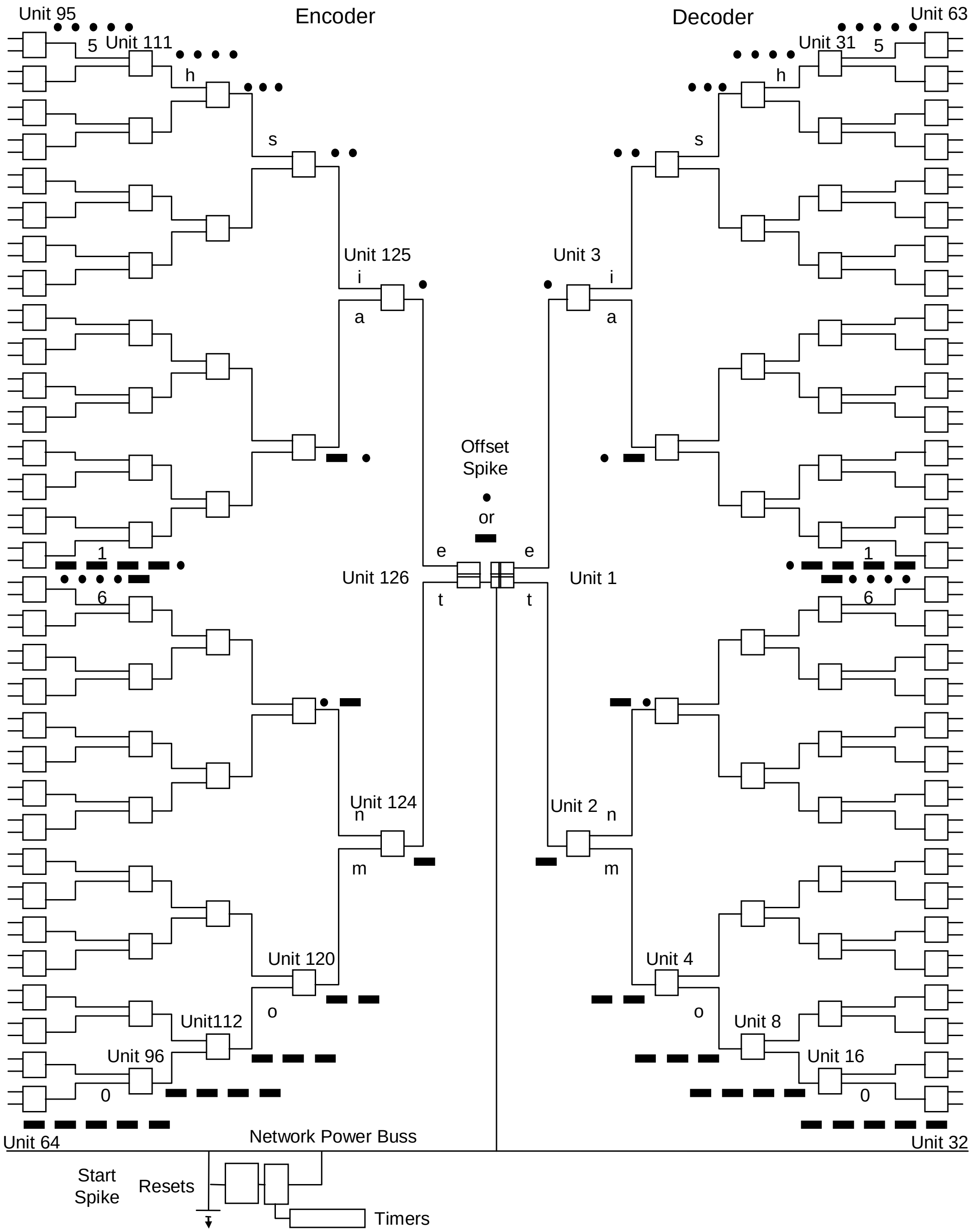


Figure 12. The *convergent/divergent network* can send and receive Morse type code in one direction, automatically.

A unique spike train is created by the convergent network from each input terminal on the left side, according the stored spike timing in each encoding cell. The divergent network produces an output at a unique output terminal according to the spike train created by the convergent network. The peripheral input and output terminals can be connected to electric typewriters, forming an automatic teletype system in which an operator types in letters or numbers that are encoded and reproduced at a second typewriter at the output of the decoding network. The operators do not need to know how to send and receive Morse code. In fact any other code can be used without the operators having any knowledge of the code.

This convergent/divergent network can be used to send information from millions of peripheral neurons to the brain through very few neurons in the spinal cord. A decoding network in the brain can decode this signal to a unique place (neuron) in the brain that identifies each peripheral neuron.

4.5 Decoding/encoding networks

Animals need to sense sound patterns in their environment, and create unique sound patterns in response. This requires a decoding/encoding network that converts spike patterns into chosen actions.

4.5.1 Sensor decoding/encoding system

The sensor decoding system shown in figure 6 can be connected to the actuator drive system shown in figure 8, creating the sensor decoding/encoding system shown in figure 13 that produces a spike-to-spike transformation.

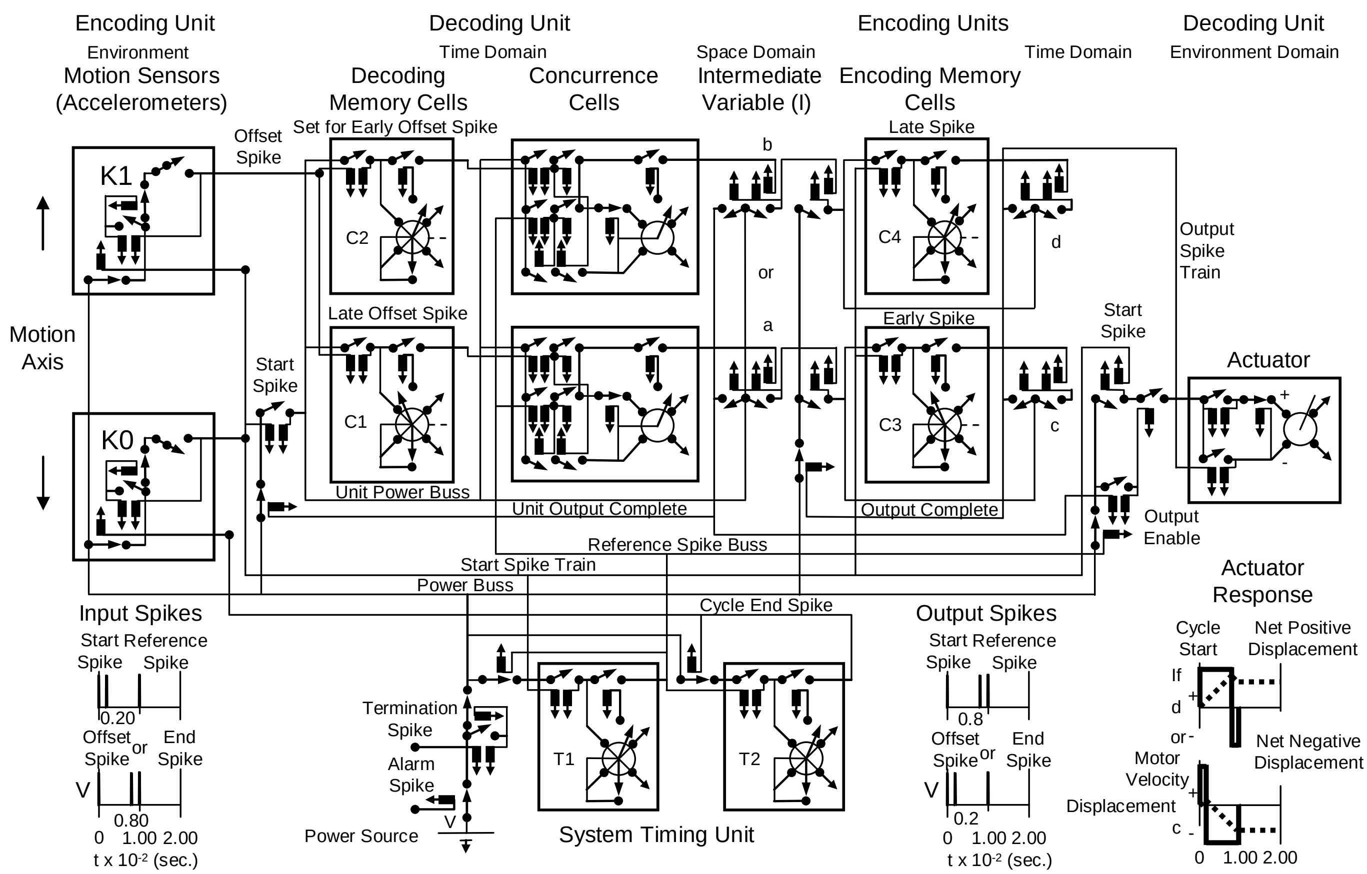


Figure 13. The *sensor decoding/encoding system* converts an input spike pattern into a new spike pattern according to the time values stored in its memory cells.

The spike timing generated by the motion sensors K0 and K1 is converted into a value (a) or (b) of the intermediate variable (I), which then energizes an encoding memory cell that produces a unique spike according to its timer setting. This spike is then decoded by the actuator into a unique position. Thus, conditions in the environment are converted into a spike pattern that is used to influence conditions in the environment. Changes in this relationship are obtained solely by changing the time delay settings of the intervening memory cells.

4.5.2 Divergent/convergent network for transforming spike trains

The decoding and encoding memory units in figure 13 can be expanded into the divergent/convergent network shown in figure 14.

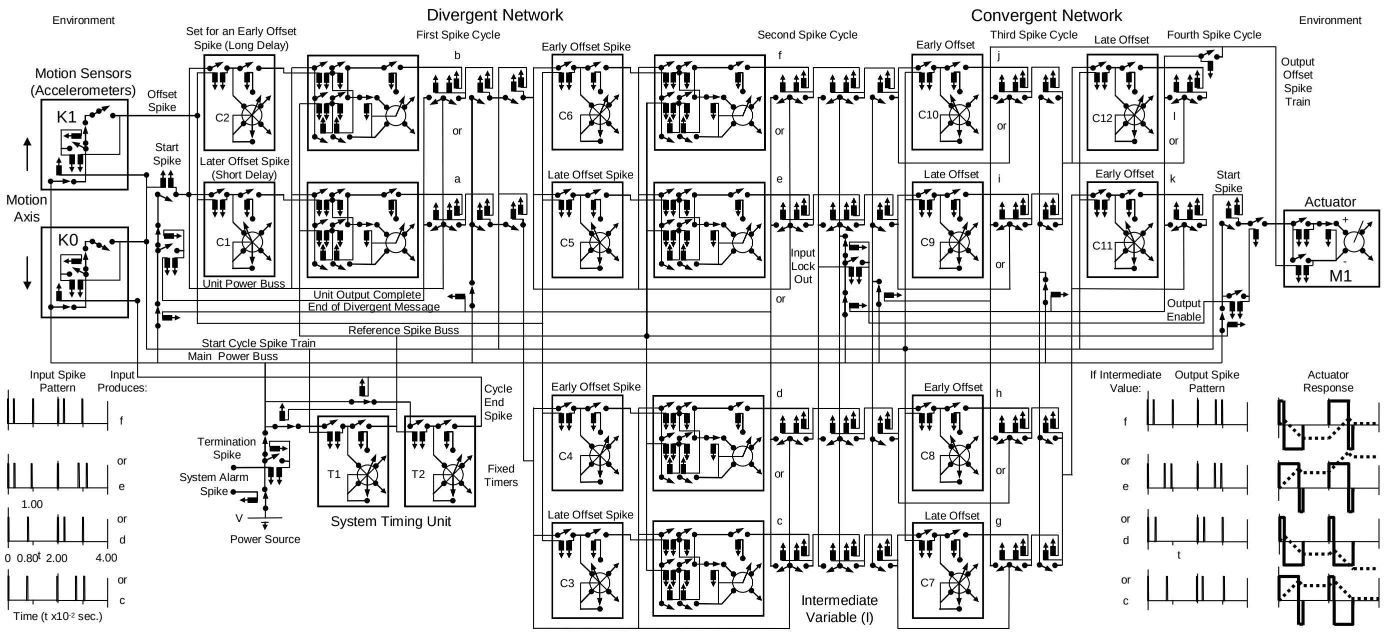


Figure 14. The *divergent/convergent network* identifies a unique sequence of spikes, and produces a new unique sequence of spikes.

The temporal pattern of a sequence of two input spikes determines the temporal pattern of a sequence of two output spikes that produce a unique sequence of motions. Since the system can read (identify) two different input spikes, and reads two spike cycles, it identifies four states that are represented by the four values (c, d, e, and f) of the intermediate place variable (I). Each of these four states can produce a unique spike train and output motion sequence according to the time delays set into the encoding memory cells.

4.5.3 Divergent/convergent network for encrypting Morse code

The divergent/convergent network in figure 14 can be expanded to form the network for encrypting temporal codes shown in figure 15.

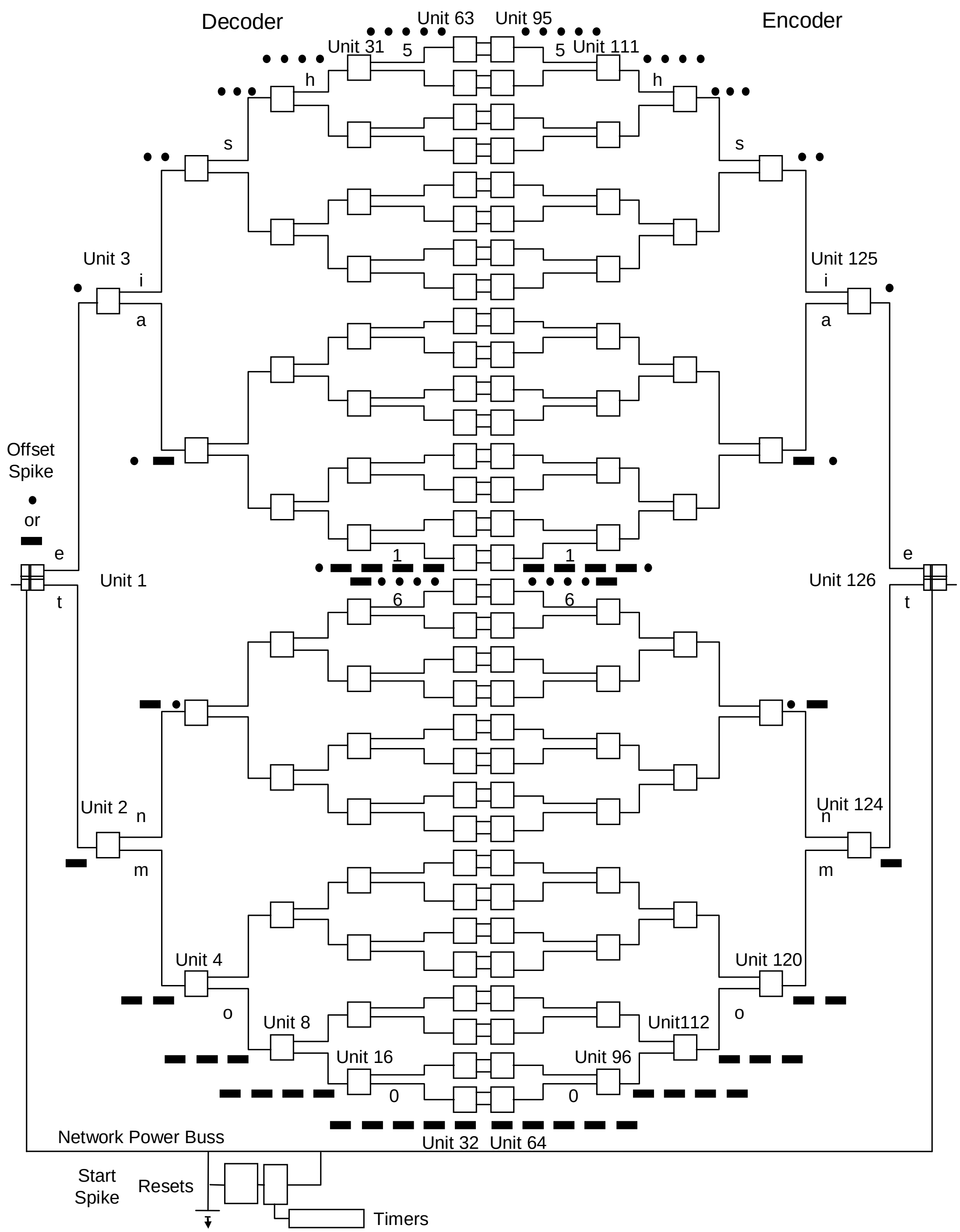


Figure 15. The *divergent/convergent encryption network* can change one temporal code into another.

Please note that these spike processing networks either decode spikes or encode values of place variables, and decoders can always be connected to encoders or encoders and be connected to decoders, as shown in figures 12 and 15. So, two divergent/convergent networks can be placed between the convergent network in figure 9 and the divergent network in figure 7 forming an automatic message encrypting system that can be read correctly only by senders and receivers with the same encryption networks.

4.6 Multi-Spike Systems

So far we have been dealing with systems that operate with just one variable spike per spike cycle; the two other spikes being the start spike and the reference spike that are constant. Animals and machines may need to respond differently to different combinations of two or more events. This requires a spike processing system that makes decisions based upon the spike timing of two or more variable spike trains.

4.6.1 Network for detecting concurrency of two variable spikes

An additional motion sensor and actuator can be added to the sensor decoding and encoding system in figure 13 forming a system with two concurrent spike trains, as shown in figure 16. Note that the motion up sensor (K2) and the motion down sensor (K1) together define changes in motion along the vertical axis, and K0 defines the start time for both.

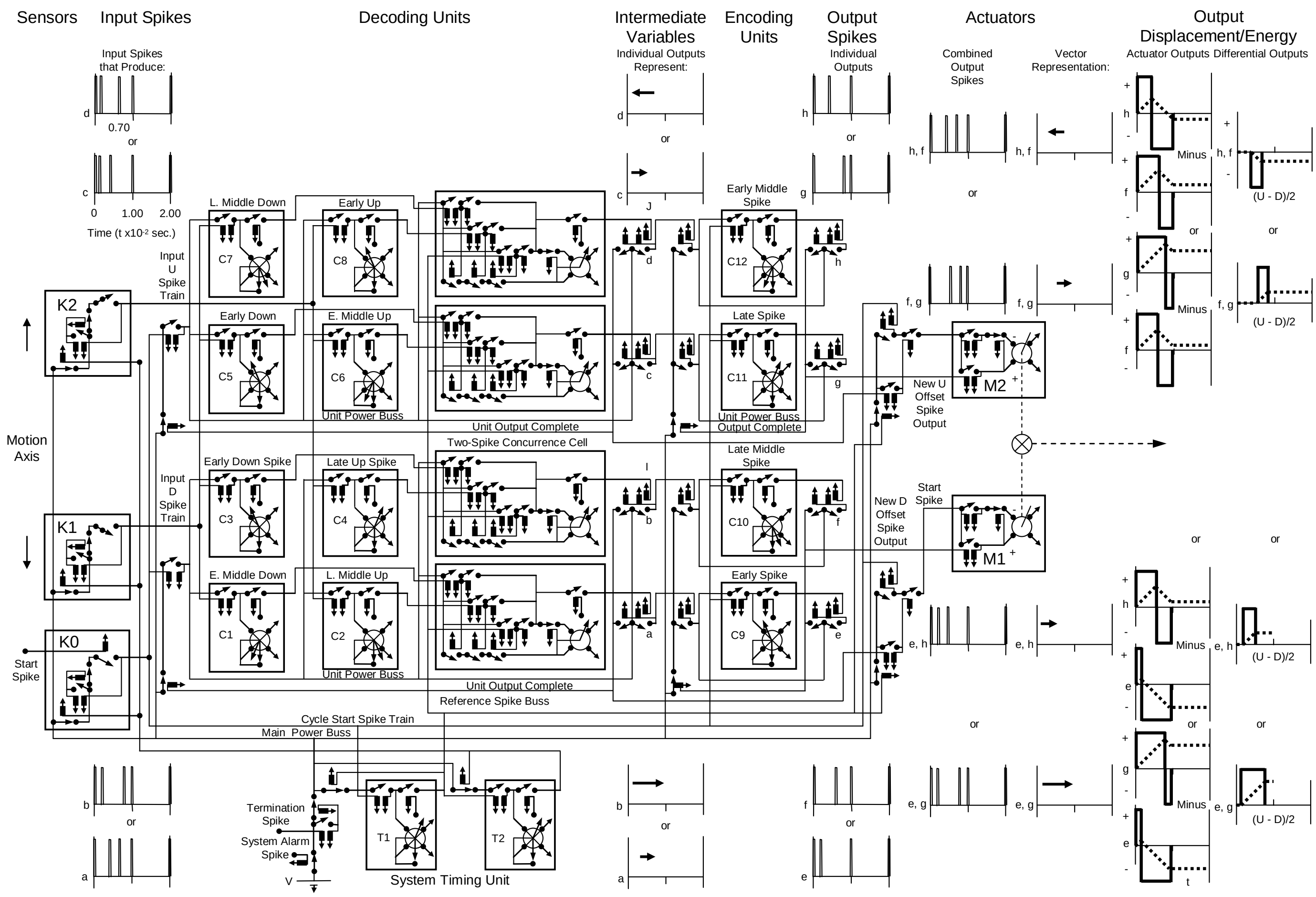


Figure 16. The *network for decoding/encoding two semi-axes* forms a complete single axis system that identifies and produces any beginning and end point of a line in both directions in a single axis.

The concurrence cell shown in figure 5 can be modified to include a second variable spike, as shown in the two-spike concurrence cell in figure 16. Since the concurrence sensor motor in the two-spike concurrence cell is connected to two input encoding cells, it will run in the positive direction until their two input spikes and the reference spike have occurred, at which time the concurrence sensor motor will run in the negative direction until its contact arm reaches the sense relay. This closes the sense relay, producing an output spike that closes the output latch. As before, the running time of the concurrence motor is a measure of the lack of concurrence in the timing of spikes at the three spike terminals, and the concurrence cell in each unit of two cells that receives spikes closest to the reference spike select values (a) or (b) of the intermediate variable (I), and (c) or (d) of the intermediate variable (J). Each intermediate place value (a) or (b), and (c) or (d) activates an encoding memory cell, forming two new spike paths. The additional actuator (M2) is added so that each spike path ends up at an actuator.

Animal muscle cells are connected in series and parallel so that the total displacement and force produced by the muscle is the sum of the displacement and force of its individual cells. By connecting the output displacement of the two actuators with a differential, which sums their outputs and divides by two, the output of the differential creates a displacement curve that has up to two variable inflection points , as shown in the two-spike system in figure 16.

As shown in figure 16, a line that represents an output motion can start at an onset up or down spike that comes some time after the start spike, and ends at an up or down offset spike that occurs some time before the reference spike. Since the two spike system allows a representation to be modified in two directions it can be called a full, one-dimensional system. These two spikes must occur on different conductors as shown. If they were to appear on one conductor there would be no way to tell the onset spike from the offset spike.

4.6.2 Divergent/convergent collective network for decoding/encoding two semi-axes

The decoding/encoding network for two semi-axes in figure 16 can be expanded into the divergent/convergent collective network for two semi-axes as shown in figure 17.

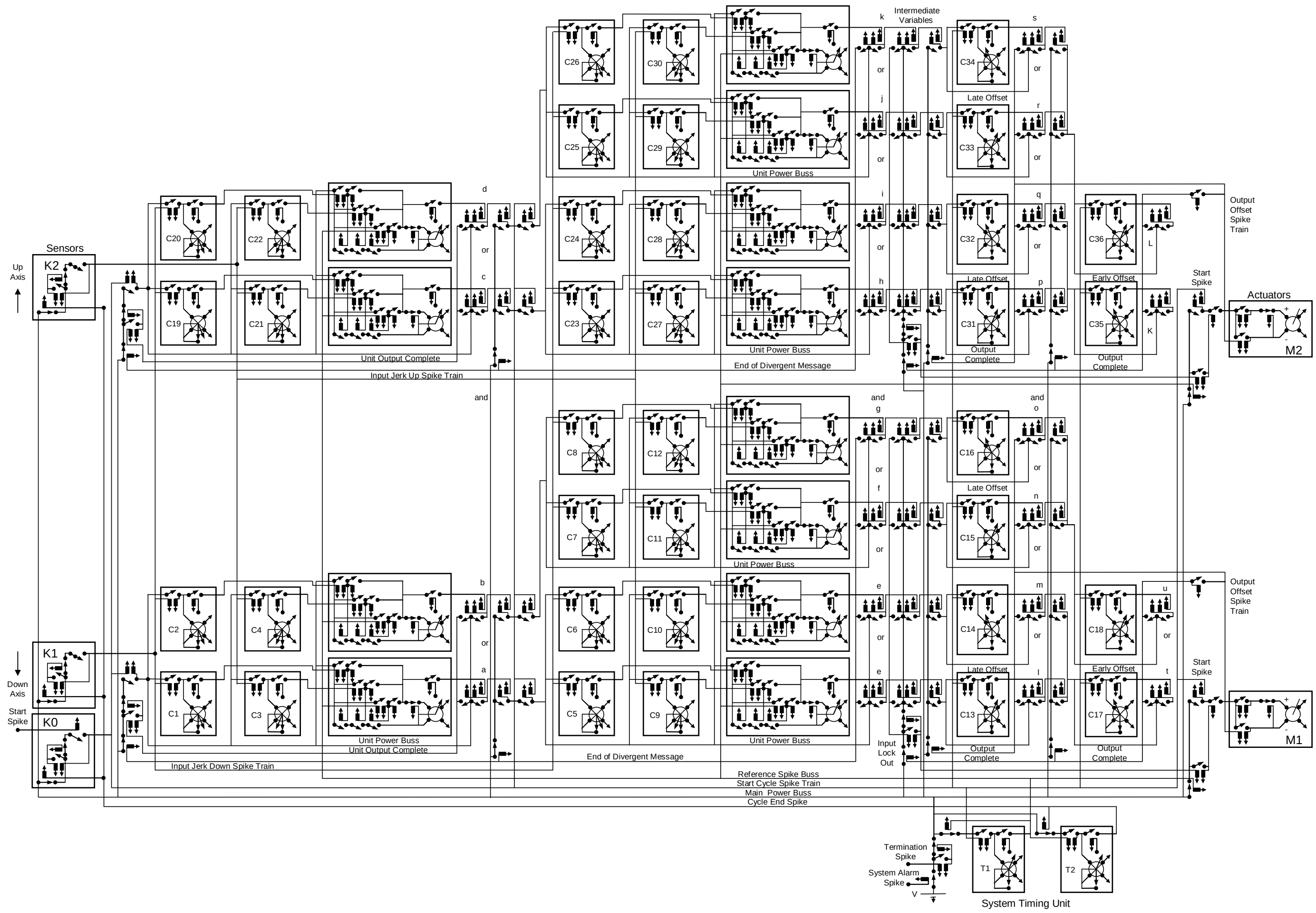


Figure 17. The *divergent/convergent collective network* for decoding and encoding two semi-axes produces two output spike trains that are determined by the combination of two concurrent input spike train timings and the time settings in the intervening memory cells.

The principal task of a decoder/encoder network is to produce an output spike train to control speech from input spike trains generated by voice messages. Multi semi-axes networks are called collective because they depended upon other semi-axes to produce an output. Two of the collective networks in figure 17 are needed to define a two axes system, and require four spike concurrence cells.

4.6.3 Place-to-Place Transformation of Two Spike Trains

The encoding/decoding network in figure 10 can operate on two spikes by using the two-spike concurrence cells shown in figure 16, as shown in figure 18.

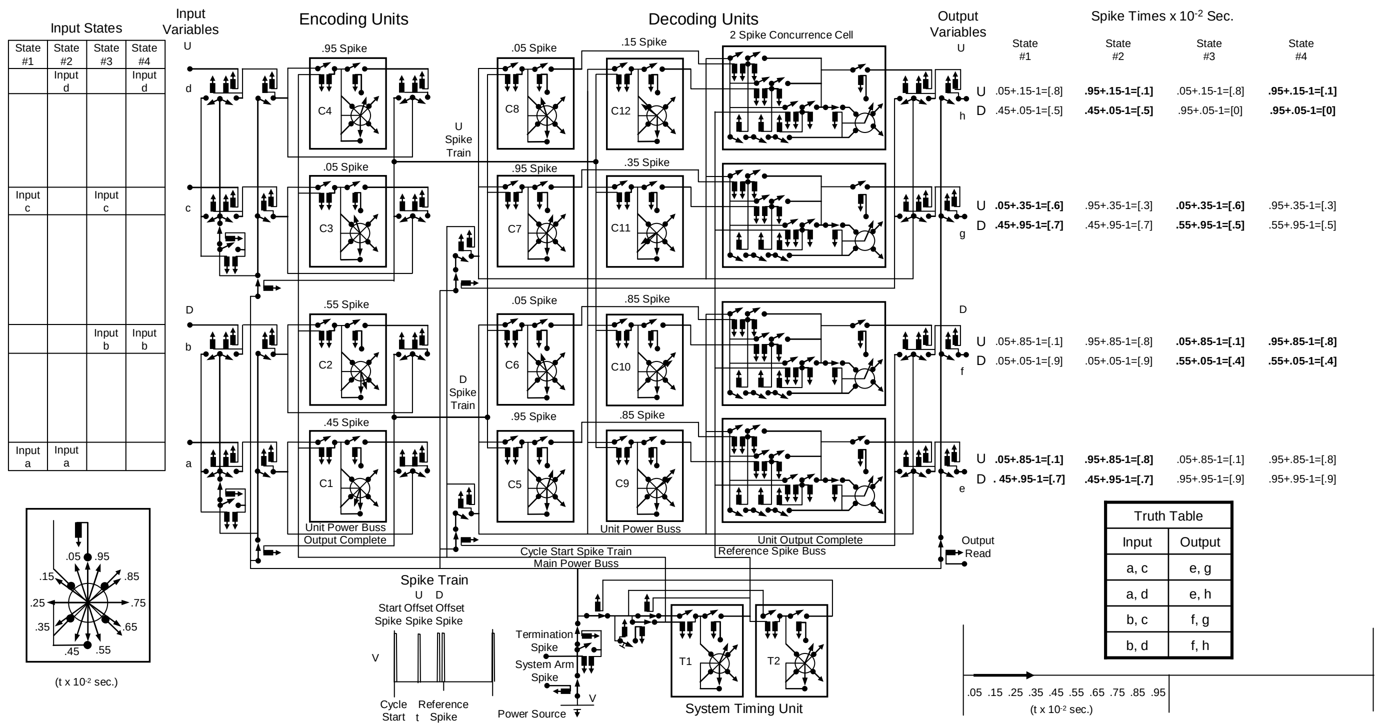


Figure 18. The *network for encoding/decoding two semi-axes* can produce a unique set of output values for a unique set of input values by some combination of time settings in its memory cells.

The outputs occur at the output terminals of the concurrence cells that measure the least mismatch between the spikes from the activated encoding cells and the reference spike.

4.6.4 Convergent/divergent collective network for encoding and decoding two semi-axes

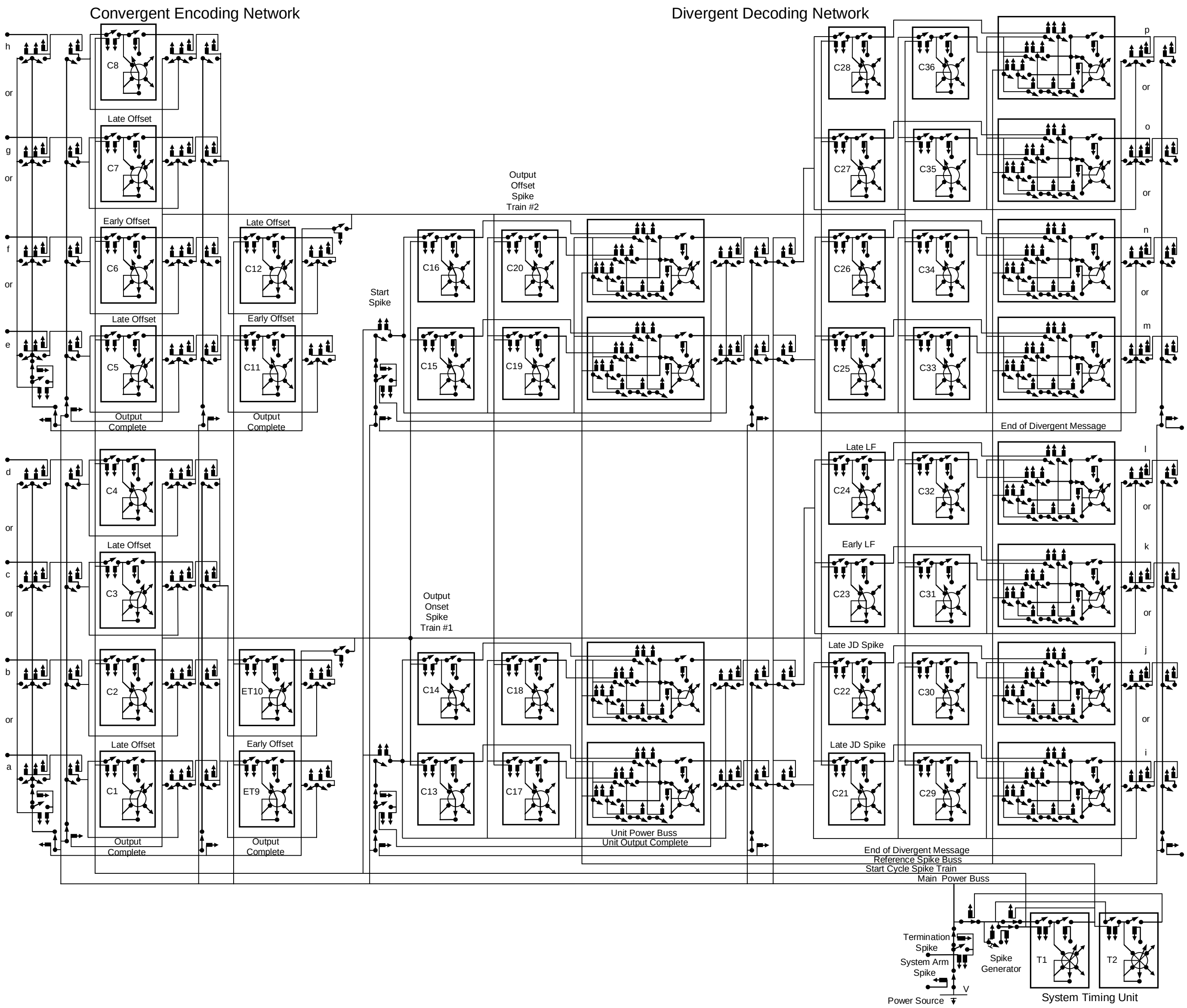


Figure 19. The convergent/divergent collective network for encoding and decoding two semi-axes forms the basis of the central brain in a predetermined spike processing system.

The place-to-place network in figure 18 can be expanded into the dichotomic convergent/divergent network of two semi-axes shown in figure 19. This network can transform any point on a line into any other point on a line. Three or more spike trains can be combined by three or more sets of encoding cells, and be connected to a set of three or more spike decoding cells for higher dimensional systems.

The multi-spike collective (distributed) networks compete to produce pathways of activation. They can cooperate or hinder the selection of a pathway and the generation of spike trains. The dichotomic collective networks differ from the fully structured logic networks, like a quaternary system required for two semi-axes that are solid masses of memory cells like the solid mass of stone in an Egyptian pyramid, which provide little or no living space inside. The individual axes in dichotomic networks are like the single dimension joists, rafters and studs in a balloon structured house, providing just the amount of support needed with minimum mass and volume.

A system with four semi-axes defines a point in a plane. Thus the encoding/decoding four semi-axes networks can specify any output point in a plane given an input point in a plane. The four individual semi-axes of dichotomic networks are like the matrix of two lane roads found in most cities, where cars can get anywhere while moving in one-way lanes. A fully structured logic network of four semi-axes is like a parking lot where cars can move in all directions.

Six-spike networks can be used to control a three-dimensional system such as an industrial robot. And a twelve semi-axes system is needed to control the six degrees of freedom of an aircraft or submarine direction control system. More semi-axes are needed to control speed as well.

4.6.5 Multi-Axes Model of a Predetermined Brain

Figure 20 shows a model of a predetermined nervous system consisting of many peripheral or sensor neurons that are encoded by a set of single spike convergent networks into a spike trains that allow a relatively few afferent spinal neurons to transmit the afferent information through the spinal cord. Since these signals flow in just one direction, each can be carried in a single semi-axis network.

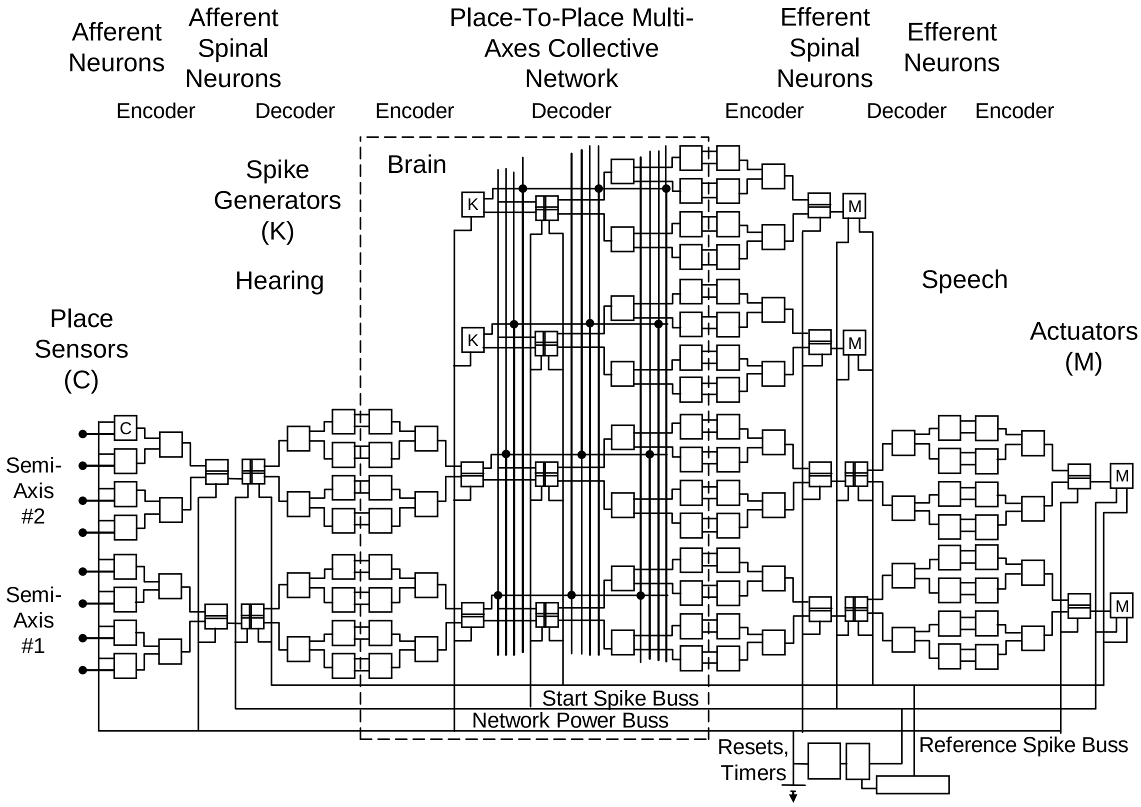


Figure 20. The *model of a predetermined brain* shows how all of the networks fit together.

These spike trains are then decoded by divergent networks into the many central brain neurons that represent the peripheral neurons. These brain neurons are then encoded by convergent networks into a new spike trains that are mixed together and transformed in the multi-axes, place-to-place collective network like the one shown in figure 19. These brain neurons are then encoded by convergent networks into a relatively few efferent spinal neurons, which are then decoded by divergent networks into the set of many efferent neurons needed to produce action.

The sets of peripheral neurons on the left can be made to overlap, so two or more efferent spinal neurons represent a single sensor state (place). Thus the set of input encoders form a larger encoder system that increases the resolution of the input states using a fewer number of spinal neurons. Likewise, the afferent neurons on the right can be made to overlap, so that two or more afferent neurons represent a single afferent state. Thus the output decoders form a larger decoder system that increases the resolution of the afferent states with a fewer number of afferent spinal neurons.

4.7 Empirical Systems

The predetermined memory cells shown above can be made into empirical (learning) memory cells by adding circuits that adjust their timing using a different algorithm depending upon whether they are encoding or decoding memory cells. These empirical memory cells can replace predetermined cells, or be placed among predetermined cells to form systems that produce useful behavior from predetermined (innate) and learned spike time settings. Many predetermined cells may need to remain, particularly in the one-spike networks used to encode and decode spike trains from and to peripheral neurons.

4.7.1 Empirical decoding/encoding network for two spike trains

The predetermined memory cells in the decoding/encoding network in figure 16 can be replaced with empirical decoding and encoding memory cells as shown in figure 21.

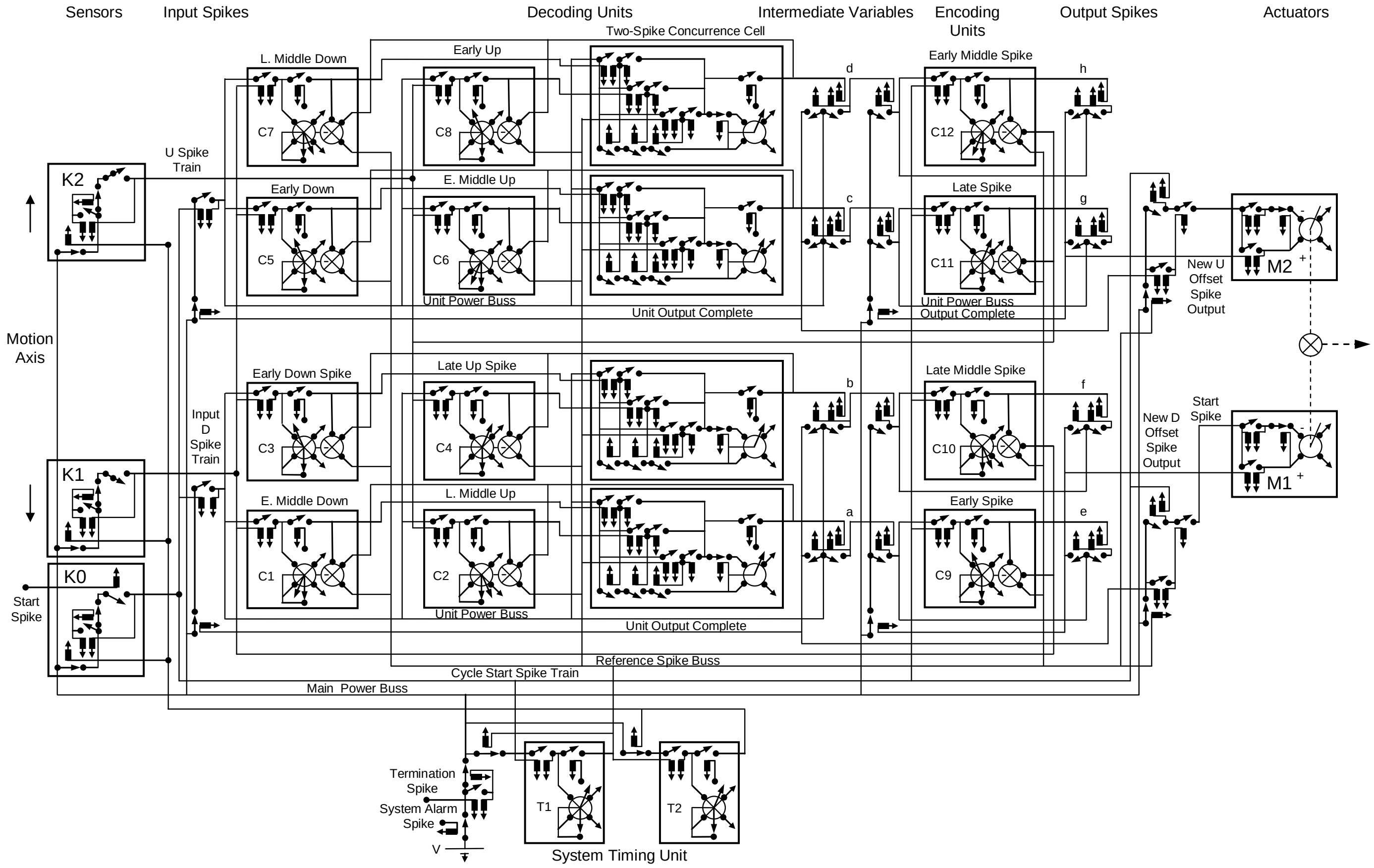


Figure 21. The *empirical decoding/encoding network* learns to produce the most common set of spike trains that occur together.

An empirical decoding and encoding network mimics spikes that occur together. If a set of empirical decoding cells (C1 and C2) and their concurrence co-cell within an energized unit of cells produces (selects) their output (a), (because it is the first cell in the unit to fire), that set of decoding memory cells decreases their time delay by changing the position of the side member shown in figure 3 if they produce an output after the reference spike, or increases their time delay if they produces an output before the reference spike. If that set of cells does not produce the output because another set of cells (C3 and C4) in its concurrence unit selects (b) before it, there is no change made in the timing setting in this unselecting set of decoding memory cells.

Once an encoding memory cell is energized by an input (a, b, c, or d), it will produce an output spike. If this spike occurs before the input spike from K1 or K2, the empirical encoding cell increases its time delay by changing the position of its differential side member. If the empirical encoding cell produces a spike after the input spike, it decreases its time delay. The timing is not changed in any cells that are inactive. This algorithm causes the spike from the encoding cell to occur more closely to the timing of the spike produced by the motion sensor when that memory cell is activated.

The timings of the cells can be changed incrementally by an amount proportional to the difference in time between the output spike of and cell and the reference spike, in the case of the decoding cell, or the input spike to the cell in the case of the encoding cell. This results in a logarithmic change in settings[3] that provides reduced corrections for small timing errors. This feature provides the highest timing adjustment resolution. If the difference in the timing of the spike is measured and changed exactly, the cells will reproduce the new timings exactly. The empirical encoding cell will repeat that input timing whenever it is energized by an input terminal, even without an input spike, and the decoding cell will produce a spike at the time of the reference spike whenever it is energized and receives the same input spike.

The empirical decoder/encoder network can be taught to produce an output spike train to control speech for example from input spike trains generated by voice messages. This is basically the same as teaching the empirical to reproduce a given input code. Since an “e” is a dot in Morse code, a dot is inputted until it energizes a decoder output, and an addition dot gets recorded on the encoder cell connected to the decoder output. Thereafter, a dot is produced when a dot is inputted.

4.7.2 *Empirical divergent/convergent network*

The empirical decoding and encoding networks in figure 21 can be duplicated multiple times, and connected in the empirical divergent and convergent network in figure 22.

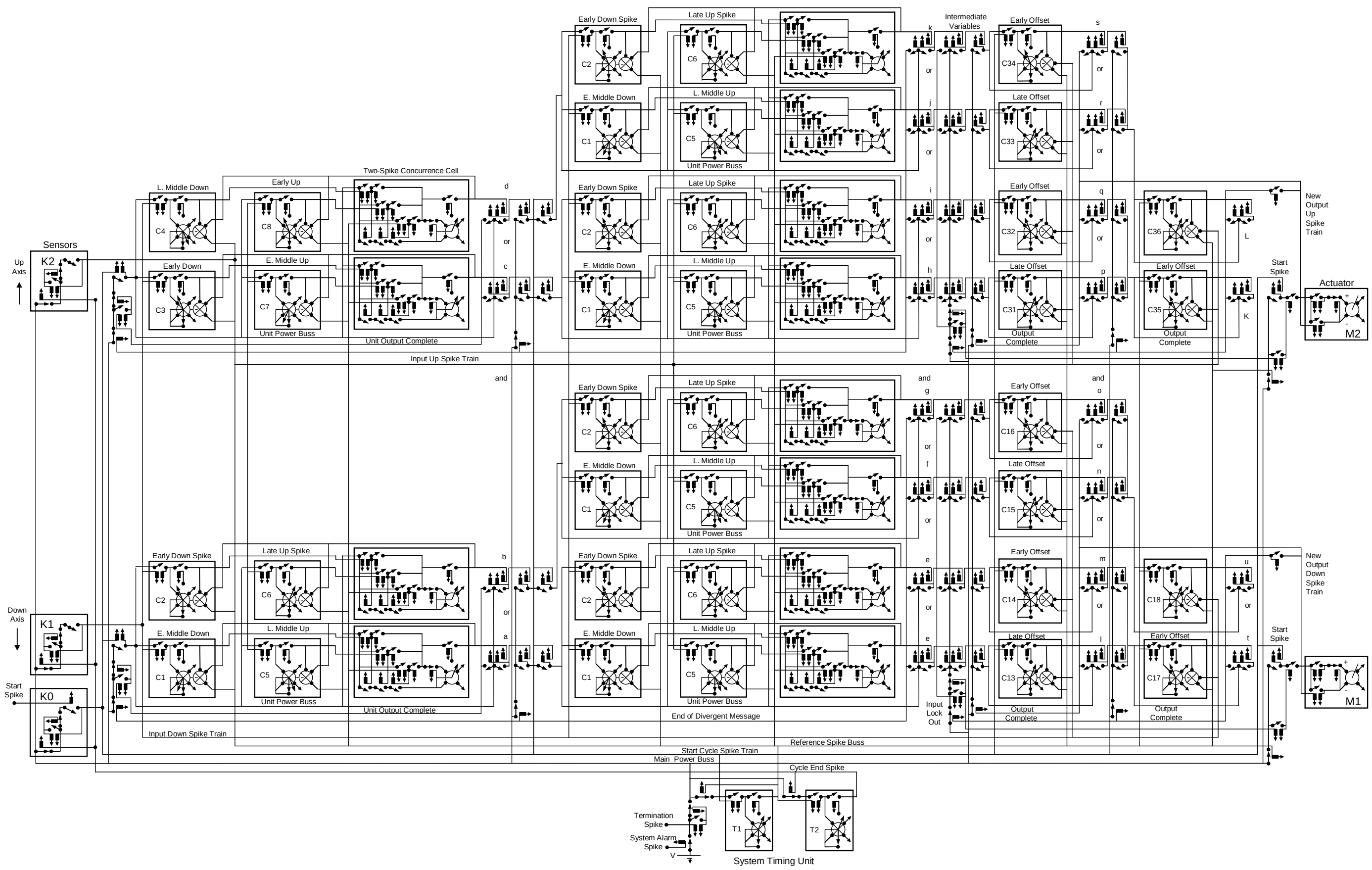
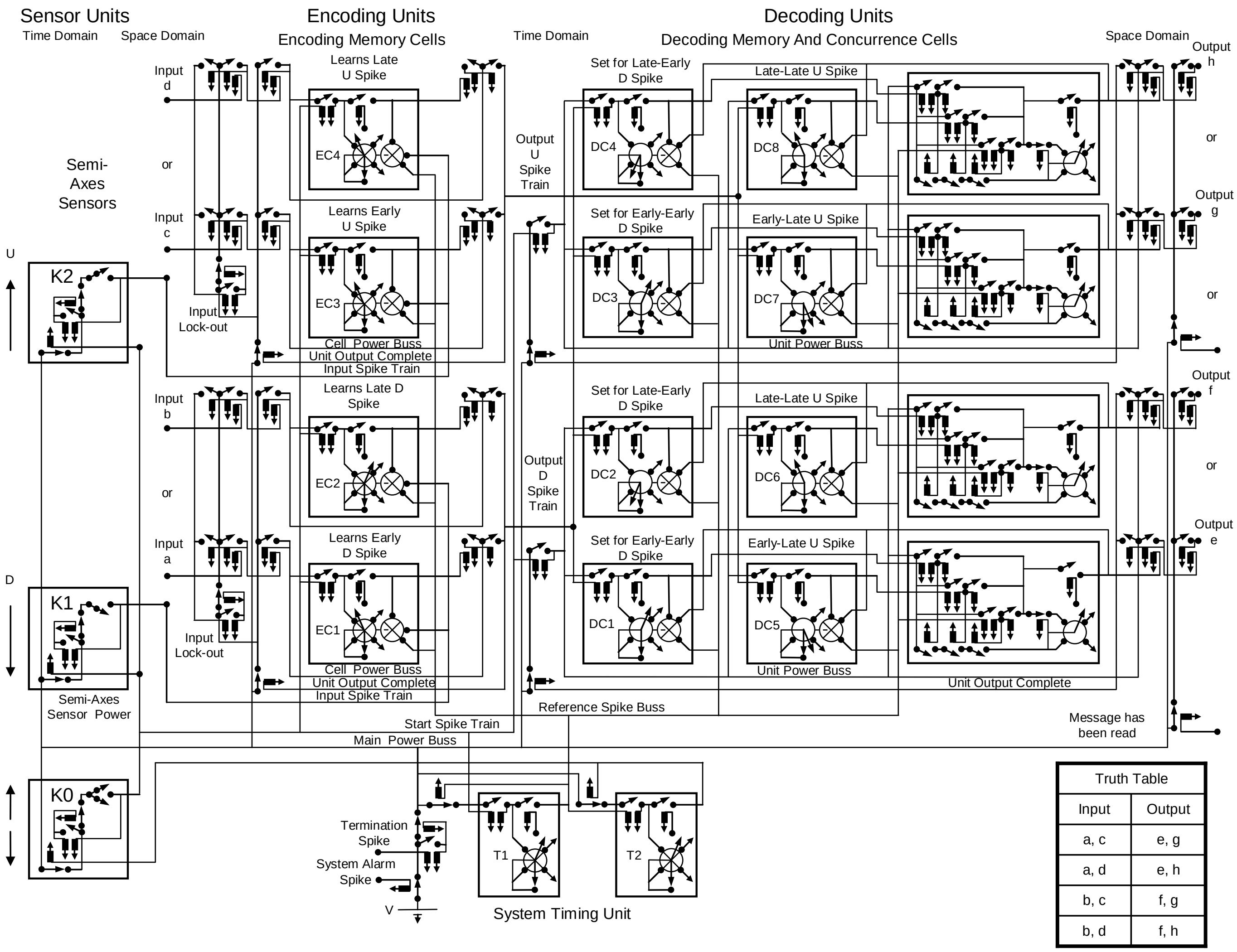


Figure 22. The *empirical divergent/convergent network* is the primary network for decoding voice messages and producing speech.

The empirical divergent/convergent can learn to mimic whole spike trains when their memory cells are set to change instantly and exactly to the timing of incoming spikes.

4.7.3 *Empirical encoding/decoding network for two spike trains*

Empirical cells can be substituted for the predetermined memory cells in the convergent/divergent network in figure 18, and spike generators such as the motion sensor K0, K1, and K2 can be added to the encoding network, forming the empirical encoding/decoding network in figure 23.



Truth Table	
Input	Output
a, c	e, g
a, d	e, h
b, c	f, g
b, d	f, h

Figure23. The *empirical encoding/decoding network* for two semi-axes learns the timing of spikes that occur when an encoding cell is activated.

This empirical encoding/decoding network can be taught to produce a given output (e, f, g, or, h) for an input place value (a, b, c, or d). For example, if the teacher wants input (a) to produce output (e) the teacher finds the spike time that produces (e), and then inputs (a) and that input spike timing.

Note that the spike train from the motion sensor does not pass through the encoding network. The output of the encoding network is made up of the stored time delays only. This encoding network can learn or copy input spike patterns instantly and exactly, storing the external world in the brain. Thus, everything that we feel, hear, and see can be stored and reproduced according to this brain model.

4.7.4 Two Level, Empirical Convergent/Divergent Network of Two Semi-Axes

The empirical convergent/divergent network of two semi-axes in figure 23 can be expanded to form the two-level, convergent/divergent network of two semi-axes shown in figure 24.

Figure 24. An *empirical convergent/divergent collective network* of two semi-axes is the basis of the central brain of an empirical spike processing system.

The two-level convergent/convergent network can produce any one of sixteen output states for any one of sixteen input states. A two-level, three semi-axes network can produce any one of 64 output states for any one of 64 input states. A four semi-axes network forms a two dimensional system that can control a point in a plane given some input point in a plane. It can produce any one of 256 points in a plane according to any one of 256 points in a plane. A two level, six semi-axes convergent/divergent network can produce any one of four-thousand output states for any one of four-thousand input states.

The number of semi-axes can be increased. The four-semi-axes system corresponds somewhat to the original four-bit microcomputers in the 1970's. These were upgraded to eight-bit, sixteen-bit, thirty-two, and sixty-four bit computers we use today. Mainframes and supercomputers use 128 bit processors and higher. This determines how many bits are processed in one computer cycle. The number of semi-axes processed in one spike cycle is determined by the size of the concurrence cells used in the network. For example, a system that can deal with 128 semi-axes requires concurrence cells with 128 inputs. The brain may require many more than 128 semi-axes. In this case, each semi-axis may be partially connected to many thousands of other multiple semi-axes in collective networks, with some upper limit to the number of semi-axes in a given collective network.

A single collective network is like the warp and weft of a fabric sheet. It consists of many parallel semi-axes in one direction crossed by many parallel connecting spike trains, as shown in figure 24. In general, the collective network does not work well unless all of its semi-axes are active at given time. This means that the collective network needs to be divided into sets of semi-axes that are active together. The process of parceling up the collective network is similar to the process of tailoring a sheet of fabric into a garment. It is cut into pieces and sewn together. The single collective network can be cut into patches and sewn together to form the shape required by the user. Ideally, semi-axes of the collective network can be shortened or extended to intersect the other semi-axes that act together. A garment made in this way would acquire the required shape without seams, much like a knitted sock.

The parceling of the collective network can be done by a process of changing the connections between semi-axes by the algorithm of deleting or sparcifying connections that seldom or never active together. This process improves the performance of the system, but does not change the memory of the system.

4.7.5 Brain Model

The predetermined encoding and decoding networks like those shown in figure 11 can generate spike trains containing a great deal of spike and place information from peripheral neurons that can be sent over a few conductors through the spinal cord, as shown in the brain model in figure 25.

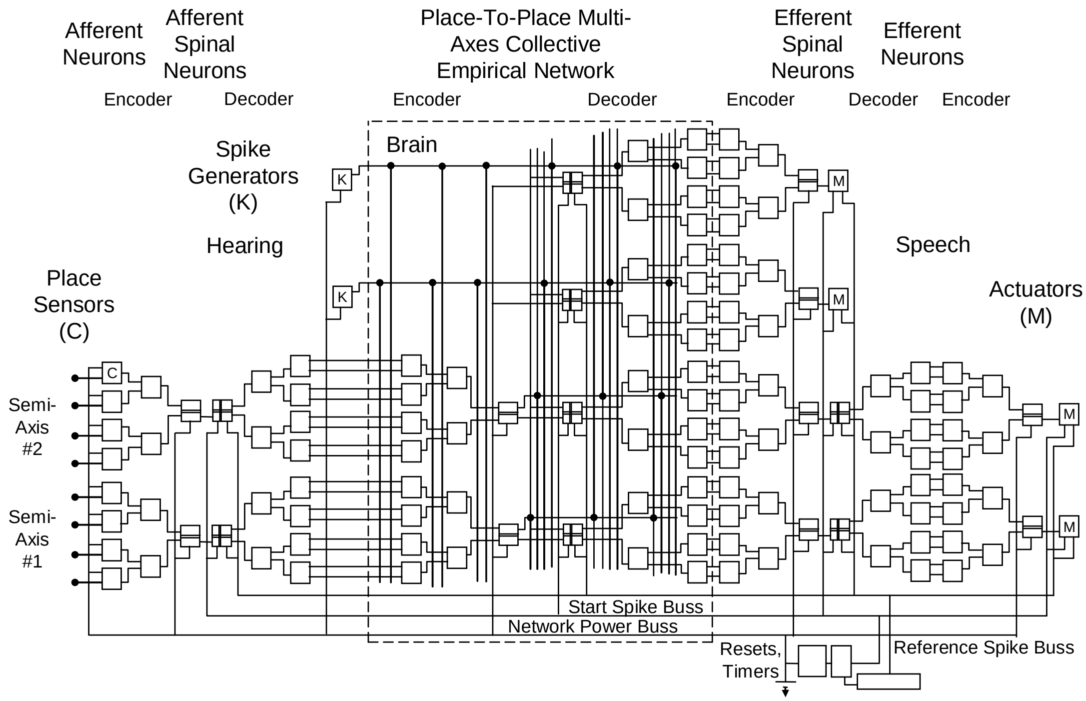


Figure 25. The *brain model* shows the basic elements of an empirical spike processing system

Empirical decoding and encoding networks like those shown in figure 22 can receive spike signals from places like the eyes and ears that are located in the head with the brain, as well as information from peripheral neurons to create new spike trains that associated with information from the peripheral neurons. These spike trains are mixed in the empirical decoding networks to activate place values. These place values can then be encoded by predetermined convergent networks for transmission back through the spinal cord and then decoded by predetermined divergent networks into the place values and/or encoded into the spike values needed to produce behavior.

Since the empirical encoding memory cells can be made to mimic, copy, or record an incoming spike train exactly in one trial, teaching a set of encoding memory cells consists of exposing the cells to the desired spike trains while activating specific input encoding cells. If the spikes are created in auditory sounds such as in a song, they will be copied exactly as each encoding cell is energized by the encoding cell before it. Then this spike train is played back exactly when the initial encoding cell is energized. Since the brain identifies a particular peripheral neuron by the spike train through a convergent network, a convergent network that is set to a particular spike train, such as a strain of music, may be identify by the brain as some place, just as it would identify the place where a thorn had pricked your finger.

We may not have access to most of the place values in the brain, but we do have access to the encoding neurons involved with the sense of touch and audible spike trains. I predict that if a spike train such as a simple melody is presented to a subject in conjunction with a particular touch sensation, that particular touch sensation will facilitate the recall of the melody. We see this kind of behavior when people are trying to remember something. They often touch or rub different places on their face and neck to help them remember.

Like the empirical encoding cell, an empirical decoding cell also learns the timing of the spikes to which it is exposed, all be it the time converse of the spike. Thus, every spike that produces an output of a concurrence cell is learned and recorded. This can be a problem in animal learning. For example, telegraph operator sending Morse code with a key may begin sending dots later in the time period. If two operators communicate exclusively, they may adjust to these changes in protocol and inadvertently modify the code. They may not notice this until they try to communicate with other operators, who could have difficulty understanding their unorthodox usage. This problem is seen extensively in our spoken language. It causes isolated groups of people to acquire distinctly different accents over time, introduce new (slang or vernacular) words and expressions, and abandon (obsolete) words and expressions used previously.

4.8 Biological Considerations

Relays, timing motors, voltage sources, and conductors in this paper are used to describe a temporal control process. Other devices would be used in a biological system to carry out the same temporal process. For example, the stored time delays in the memory cells responsible for the selection of behavior of the systems shown in this paper can be obtain by means other than electrical timing motors. A pathway of a given length that conducts a spike at a specific rate (speed) serves the same purpose as a timing motor. If the rate of conduction or the length of the conductor can be varied, these delay lines can be used as the time memories needed in a temporal spike processing control system.

Studies show that the rate of transmission of impulses (spikes) through a white matter neuron[4] is determined by the ratio of thickness of the myelin sheath of the neuron to the diameter of the neuron. Also, white matter constitutes the greatest portion of the weight of the brain according to these studies, and the thickness of the myelin appears to change with learning. Indeed, the physical changes in the brain responsible for all learning and memory may be due to the increase or decrease of the thickness of the myelin sheaths of individual neurons making up the white matter in the brain.

If it can be shown that changes in synapses result in changes in the time delay of spikes between neurons, then the synapses may serve as the timers shown in the networks in this paper. Otherwise, connections between neurons do not have to be made, broken, or changed in any way in the spike the timing system shown in this paper to change behavior. Indeed, a system that relies upon making connections only, without transmitting message information through these connections, is like a telephone system that allows anyone to call any other phone, but does not allow anyone to talk.

Acknowledgements

I wish to thank William D. Penny and Karl J. Friston, Director of the Wellcome Trust Centre for Neuroimaging, University College London for their patience and advice.

References

1. Hopfield, J.J., & Brody, Carlos D. (2000). What is a moment? “Cortical” sensory integration over a brief interval. USA: Proc. Natl. Acad. Sci.

2. Motor (electric). (1968). Van Nostrand’s Scientific Encyclopedia. New York: D. Van Nostrand.

3. Brown, Robert Alan. (1994). *Machines that learn* (p. 320). New York: Oxford University Press.

4. Fields, R. Douglas. March 2008. White matter matters. Scientific American, Vol. 298, Number 3, 54.