Fault Diagnosis of the Intracellular Microprocessor
Overline: Computational Biology

One sentence summary: An engineering approach reveals the weakest links in cellular signaling networks.

Editor's Summary

Finding Vulnerability
Vulnerability assessment methods, which are commonly used to test digital circuits, were applied to biological signaling networks to identify the molecules that when dysfunctional, would be most likely to disrupt the function of the network. Two signaling well-characterized networks were analyzed (one leading to caspase3 activation and apoptosis and a second leading to activation of p53) and the molecules known to be critical to the response were properly identified. In a third network of neuronal activation of the transcription factor CREB, this analysis lead to the identification of \(G_\alpha_i\) and the P/Q-type calcium channel, which were then experimentally validated as critical molecules in this network.

Fault Diagnosis Engineering of Digital Circuits Can Identify Vulnerable Molecules in Complex Cellular Pathways

Ali Abdi\(^1\), Mehdi Baradaran Tahoori\(^2\), and Effat S. Emamian\(^3\)\(^*\)\(^†\)

\(^1\) Center for Wireless Communications and Signal Processing Research
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
323 King Blvd
Newark, NJ 07102, USA

\(^2\) Department of Electrical and Computer Engineering
Northeastern University
307 Dana Research Building
Boston, MA 02115, USA

\(^3\) Advanced Technologies for Novel Therapeutics (ATNT)
211 Warren Street, Newark, NJ 07103, USA

\(^†\) Work initiated at The Rockefeller University, Box 296
1230 York Avenue
New York, NY 10021, USA
Abstract

The application of complex system engineering approaches to cell signaling networks should lead to novel understandings and subsequently new treatments for complex disorders. In the area of circuit fault diagnosis engineering, there are various methods to identify the defective or vulnerable components of complex digital electronic circuits. In biological systems, however, knowledge is limited regarding the vulnerability of interconnected signaling pathways to the dysfunction of each specific molecule. By developing proper biologically-driven digital vulnerability assessment methods, the vulnerability of complex signaling networks to the possible dysfunction of each molecule can be determined. To demonstrate the utility of this approach, we analyzed three well-characterized signaling networks-- a cellular network that regulates the activity of caspase-3, a network that regulates the activity of p53, and a central nervous system network that regulates the activity of the transcription factor CREB. We found important differences among the vulnerability values of different molecules. Most of the identified highly vulnerable molecules are functionally related, and known key regulators of these networks. Experimental data confirmed the ability of digital vulnerability assessment to correctly predict key regulators in the CREB network. Because this approach may provide insight into key molecules that contribute to human diseases, it may aid in the identification of critical targets for drug development.

Introduction

Over the past few decades, a large amount of information has been collected regarding the function of individual signaling molecules and many detailed individual molecular mechanisms that regulate cellular function have been identified. Systems biologists have started to integrate these individual interactions and components, to analyze the properties and functions that emerge from these complex biological systems (1).
Each cell in the human body includes many biomolecules, which are interacting with each other through a network of many cellular signaling pathways. Dysfunction of some molecules involved in these pathways may interfere with the efficacy and efficiency of the signal transduction within the network, which can eventually result in a transition from the normal function (physiological condition) to a dysfunctional system (diseased or pathological condition). For some of hereditary human diseases, such as Huntington's disease, neurofibromatosis, and polycystic kidney disease, variations in a single gene cause the pathology. However, for some human diseases, including cancer, as well as some neurodegenerative and psychiatric brain disorders, a single gene does not cause the disorder. Instead, the disease may result from the dysfunction of several molecules in different pathways. In such complex trait disorders, it is not clear which molecules have causative effects, and how much each molecule may contribute to the development of the pathology.

The core idea of this paper is to conceptualize a disorder at the molecular level as a faulty system, in which one or more molecules in the complex intracellular signaling network are dysfunctional. Although genome- and proteome-wide expression analysis of biological systems provides a valuable picture of the “expression levels” of the molecules, it is the “functionality” of each molecule that determines the overall performance of the molecular system. We develop useful molecular fault models, similar to some fault models used in digital circuits (2), to quantify the functionality of different molecules in a network.

Results
There are many similarities between digital electronic circuits and genetic or signaling networks. In a manufacturing facility, a digital circuit is manufactured based on a particular design and is supposed to provide a specific function. However, during the fabrication process, physical defects--faulty transistors, open and short wires, and such--may occur, causing the manufactured circuit not to function according to the design specifications. Testing of digital circuits and systems allows defective manufactured parts to be separated from the nondefective ones, in order to guarantee the production of fault-free products. The test is an assessment of the manufactured circuit, according to a set of criteria. During the lifetime operation of electronic systems, the correct functionality is a key aspect and is typically referred to as reliability.

To determine the most vulnerable molecules in a network of molecules, we take advantage of a class of electronic circuit reliability analysis techniques known as vulnerability assessment methods. Such methods provide numerical values for the vulnerability of the operation of the entire molecular system to the dysfunction of each individual molecule. A high vulnerability for a molecule means that with high probability, the whole signaling network does not operate correctly, if that particular molecule is dysfunctional. Identification of such important molecules, those with high vulnerability values, in signaling networks implicated in disease is a major step towards understanding the molecular basis of complex human diseases. From the drug development perspective, vulnerability assessment provides a set of candidate molecules to target.

To calculate the amount of the vulnerability of a network of interconnected pathways to the dysfunction of each molecule, one needs a model for the network. There are different types of models, such as Boolean, Bayesian, differential equations, algebraic,
and graph theory. We chose the Boolean framework, where each molecule is either active (on) or inactive (off) (3, 4) thus by analogy with digital electronic circuits, the state of a molecule is either 1 or 0, respectively. The Boolean framework and binary logic has been extensively used to explore different characteristics of signaling and genetic networks (3). Through application of digital circuit fault and reliability analysis methods (2) and Boolean models of pathways, we show that the molecular system vulnerability to the dysfunction of each molecule within the system can be computed.

To illustrate this approach, we generated circuit diagrams for a "toy" molecular network (Fig. 1A), which has seven molecules. In Fig. 1A, $A$ and $B$ are the input nodes (molecules), and $G$ is the output node (molecule). Activation and inhibition are shown by lines ended to $\rightarrow$ and $\nabla$, respectively. The activity of each molecule can be shown by a binary value assigned to the name of that molecule. For example, $A = 0$ indicates that $A$ is inactive (off), whereas $A = 1$ means that $A$ is active (on). To assess the vulnerability of a network, it is necessary to derive a binary logic equation for each molecule of the network. Based on the known physiological mechanisms by which different regulators control the activity of signaling molecules, we propose two rules to derive the logic equations of a molecular network. Rule #1 states that if a molecule has no inhibitory input, then the activation of at least one of its activatory inputs in enough to activate the molecule. Rule #2 states that if a molecule has at least one inhibitory input, then that molecule will be inactive, if at least one of its inhibitory inputs is activated. Moreover, the molecule can be active, only if all the inhibitory inputs are inactive.

Based on the definitions of Boolean OR, AND, and NOT (inversion), Rule #1 implies the $OR$ of activatory inputs; whereas Rule #2 is the $AND$ of inverted inhibitory
inputs. When a molecule has both activatory and inhibitory inputs, the two rules are combined, which means the AND of “inverted inhibitory inputs” with the “OR of activatory inputs”. Using the above two rules, we derived logic equations for the toy network (Fig. 1B) and based on the five logic equations, we generated a digital electronic circuit for the toy network (Fig. 1C). In digital circuits, there are four types of "gates": AND, OR, BUFFER, and NOT. The following symbols are used to represent these gates: A semi-ellipse represents the AND operation, a bullet-like shape depicts the OR operation, a triangle with a bubble on the right vertex represents the NOT operation (inhibition), a triangle with no bubble is called a BUFFER (activation). The input and output of a BUFFER have the same logic values; if the input is on or off, then the output is on or off, respectively.

To test the validity and feasibility of this approach in cellular signaling networks, we apply this logic to a well-characterized network (Fig. 2A), for which the interactions between the molecules are extensively characterized and experimentally verified (5,6). Janes et al. (5) studied the individual and joint effects of the three input nodes, the ligands epidermal growth factor (EGF), insulin, and tumor necrosis factor (TNF), on the activity of the two output nodes caspase3 and forkhead-related transcription factor (FKHR). Their study also showed how the activity of several intracellular intermediate molecules is modulated to regulate the activity of caspase3 and FKHR. The caspase3-FKHR network investigated by Janes et al. has 22 nodes, with 27 interactions among them. Using the two rules that we previously defined, we derived a single logic equation for each node in the network (Supplementary Table 1) that includes all the regulatory inputs to each node.

With the logic equations derived for the caspase3-FKHR network, we generated a caspase3-FKHR digital circuit (Fig. 2B) using AND, OR, BUFFER, and NOT gates.
Using the logic equations, the binary logic values of the outputs, FKHR and caspase3, can be determined for any combination of inputs. The resulting “truth table” (Table 1) shows in the first three columns all possible input scenarios and in the last three columns show the calculated outputs, according to the circuit logic equations (Supplementary Table 1). Consistent with the experimental findings (5), digital circuit analysis showed that as long as EGF or insulin is active, caspase3 is inactive, indicating that apoptosis will not occur. However, when both EGF and insulin are inactive and TNF is active, then caspase3 is activated, which can lead to apoptosis. The consistency of the experimental data (5) with the results of the digital circuit analysis validates the biological application of this binary logic engineering approach and suggests that this level of abstraction, the Boolean on/off model, has a coarse predictive power that can be verified experimentally.

To perform the vulnerability assessment for any circuit, the fault model must be specified (2). We chose the class of "stuck-at-0" and "stuck-at-1" fault models, because they seem to be more biologically relevant. To place this model into a biological perspective, consider that each molecule is supposed to change its active or inactive state, according to its input signals. However, if the molecule is “stuck” at a particular state, due to mutations or other structural or functional abnormalities, it can not respond properly to the input signals. The state of such a dysfunctional molecule will not change, even though it may receive stimulatory or inhibitory commands from its surrounding molecules. Thus, a node with the stuck-at-0 fault means that the logic value of that node (molecule) is always 0 (inactive), irrespective of its inputs. Similarly, a node with the stuck-at-1 fault implies that the node (molecule) is always 1 (active), and its state does not change if the inputs change.
In the caspase3-FKHR example, if AKT is "stuck-at-1", then according to the logic equation for caspase3 [caspase3 = AKT' × (caspase8+JNK1+MK2); (Supplementary Table 1)], caspase3 is 0. This is because 1'=0 and 0 × (caspase8+JNK1+MK2) = 0. Thus “stuck-at-1” AKT removes the only chance for caspase3 activation, the presence of TNF in the absence of insulin and EGF, and the subsequent apoptosis under normal conditions (Table 1, compare last two columns). This dependence of caspase3 activation and by inference, apoptosis, on the state of AKT is supported by the experimental findings showing that the hyperactivity of AKT (stuck-at-1 in the digital circuit fault analysis) is associated with malignant transformation (7).

The next step in digital circuit fault analysis is calculating the vulnerability of a node. By definition, the vulnerability value of a node is the probability that the system fails (incorrect system output), if that particular node is faulty (dysfunctional). To determine the vulnerability of the network to the dysfunction of each individual molecule, we considered the input signals to be statistically independent, such that each input molecule takes 0 and 1 with the same probability of 0.5. By applying the error probability propagation (EPP) method (8,9) to the caspase3-FKHR circuit (Fig. 2B), we computed the vulnerability values of all the molecules in the caspase3-FKHR network (Table 2).

In this type of analysis, the vulnerabilities of the output nodes are always 1, because if the output nodes are dysfunctional, by definition the network is not functioning correctly. The caspase3-FKHR network shows the highest vulnerability to the dysfunction of AKT, which may be because AKT is the immediate upstream regulator of both the output nodes (Fig. 2A). To clarify this issue, we removed the output node FKHR from the network and recalculated the vulnerabilities of all the nodes (Table 2). Even in this case, AKT still has the highest vulnerability value of all nonoutput molecules, which
implies that AKT plays a critical role in this network. Using the three-input one-output caspase3 network, a biological interpretation for the AKT vulnerability value 0.87 means that if AKT is dysfunctional, then on average, for 87% of all possible ligand binding incidents (input signals), the cell will not correctly regulate the activity of caspase3 (the output node). The computed vulnerabilities (Table 2) are insensitive to the presence or absence of some nodes that have minor effects on the vulnerabilities of the rest of the nodes in the network. For example, removal of p38 from the caspase3-FKHR network, which allows the MKK3 node to activate the MK2 node directly, did not change the vulnerability values of the remaining components. Removal of the NF-κB node or both the p38 node and the NF-κB node also did not change the vulnerability values.

Using the approach of Ma’ayan et al. (11), we constructed a p53 cellular network containing numerous intermolecular interactions (Fig. 3A). The p53 network was constructed from pairs of experimentally-verified molecular interactions reported in the literature (see Supplementary Materials for citations). This network was analyzed to test the validity and capability of circuit fault diagnosis engineering to correctly predict the main regulators of p53, which is a tumor suppressor that is a transcriptional activator of several genes that ultimately control cell cycle arrest, cellular senescence, or apoptosis. p53 has been found mutated or functionally inactivated in more than half of all the human cancers (12). The resulting network, called the p53 network, has a total of 49 molecules, with 94 intermolecular interactions. The input nodes are the two ligands insulin and platelet-derived growth factor (PDGF), and the output node is the transcription factor p53. The specific interactions among the molecules of this network are derived from the literature and publicly available databases (see Supplementary Materials). A single specific logic equation for every individual molecule in the p53 network was derived on
the basis of the logic rules (Supplementary Table 2) and these logic equations were used to generate the p53 digital circuit (Supplementary Fig. 2), and the vulnerability values of all the molecules in the network were computed (Table 3).

The p53 network shows the highest vulnerability (more than 0.5) to the dysfunction of phosphatidylinositol-(4,5)-bisphosphate (PIP2), AKT, caspase3, and protein phosphatase 2A (PP2A) (Fig. 3B). Previous studies have experimentally shown that these molecules are known to be the key regulators of p53 (32, 33, 34, 35, 36, 37, 38). Thus, the analysis predicts that if any one of these molecules is dysfunctional, out of 100 incidents of ligand binding, over 50 times p53 will not function properly. The distribution of the vulnerability values of the molecules is nonuniform (Fig. 3C). There were four highly vulnerable nodes (>0.5), six moderately vulnerable nodes (values between 0.1 and 0.5), and a majority of the nodes (38 out of the 49) exhibited very low vulnerability values (less than 0.1) (Fig. 3B).

The four molecules that had the highest vulnerability values in the p53 network are known to be key regulators of the cellular functions for which p53 is responsible. For AKT and caspase3, there is clear evidence that these two molecules play crucial roles in regulating a number of p53-regulated functions, such as cell survival and apoptosis (12, 13). More importantly, there is experimental evidence supporting a key role for AKT in proper activation of p53 (32, 33, 34). Although PIP2 have not been directly connected to p53, it is a regulator of AKT and thus this connection to AKT may explain its high vulnerability value. The experimental evidence that PP2A and caspase3 plays a major role in regulating p53 activity comes from (35, 36, 37). Thus, molecules diagnosed as faulty by this approach are experimentally known to contribute to failure of the system.
and cause pathology in humans. This can confirm the reliability of this novel methodology.

We also constructed a complex neuronal network, following the same approach of Ma’ayan et al. (11). The output node is the transcription factor CREB (cAMP responsive element binding protein) and the input nodes are seven major ligands in nervous system--glutamate, dopamine, GABA, serotonin, ACh, adenosine, and enkephalin. The CREB network (Fig. 4A) is comprised of 64 molecules and 152 intermolecular interactions (see Supplementary Materials). The logic equations for the CREB network (Supplementary Table 3) were derived using Rule #1 and Rule #2, and the corresponding digital electronic circuit, was constructed (See Supplementary Fig. 4). Calculation of the vulnerabilities of all the molecules (Table 4) once again revealed a nonuniform distribution (Fig. 4C). Indeed, dysfunctional in 41 molecules out of 64 would not contribute to the failure of CREB circuit (vulnerability values less than 0.1).

The molecules with a vulnerability of more than 0.5, which indicates that their dysfunction can result in the failure of CREB function, are calmodulin, calcium, cAMP, Gαi (abbreviated Galphai in the network diagram), adenlyate cyclase 1 (AC1), AC2, AC5, protein kinase A (PKA), P/Q-type calcium channel (abbreviated P/QCaCh in the network diagram), and PP2A (Table 4 and Fig. 4B). These molecules can be grouped into elements of the cAMP-dependent signaling (cAMP, Gαi, PKA, and the AC isoforms) and elements of calcium signaling (calcium, calmodulin, and P/Q-type calcium channels). Furthermore, the distribution of the vulnerability values in the CREB network is highly
non-uniform (Fig. 4C). Fig. 4B shows the molecules with the highest vulnerability values in the CREB network are functionally related molecules, and some of them are already known as main physiological regulators of CREB function. Indeed the name CREB is based on the identification of the protein as a cAMP responsive element binding protein. This engineering analysis has identified cAMP and the molecules directly related to cAMP function, such as AC1, AC2, AC5, and PKA, as the most critical molecules for the regulation of CREB. The crucial role of PKA in the regulation of CREB is well known (14). Important functions of the cognitive and executive human brain, such as learning and memory, are directly regulated by cAMP-dependent CREB functions (14). In pathological terms, direct evidence for deregulation of PKA signaling has been reported in human disorders manifested by memory dysfunction, such as Alzheimer Disease (18) or schizophrenia (19, 31). Vulnerability assessment of the CREB circuit has also identified some elements of calcium signaling as playing a major role in the function of CREB. This observation is also physiologically and pathologically relevant and consistent with experimental data (20). Furthermore, several pathological conditions associated with memory dysfunction can arise from deregulation of calcium-dependent signaling (21-23). Although it is not yet clear how CREB is involved in the pathogenesis of these disorders, the physiological role of CREB in neuronal mechanisms underlying the memory function of mammalian brain has been known for many years (14).

Because many of the identified molecules are already experimentally known regulator of CREB, we provide experimental evidence for only two, Ga\textsubscript{i} and P/Q-type calcium channels. We altered the activity or abundance of endogenous P/Q-type calcium channels and Ga\textsubscript{i} in primary cortical neurons from rats and then assayed for the changes in endogenous CREB activity or abundance (see Methods). We used primary neuronal
culture as a model system because it has the most similar CREB signaling network to the in vivo signaling networks of the mammalian brain.

Short-term (2.5 h) treatment of primary neurons with the selective P/Q-type calcium channel blocker, ω-Agatoxin IVA, increased the phosphorylation of CREB at Ser\textsuperscript{133}, without changing the total abundance of CREB (Fig. 5A). In contrast, long-term (12 h) treatment decreased the phosphorylation of CREB at Ser\textsuperscript{133}, as well as decreased the abundance of CREB (Fig. 5A). Furthermore, we targeted the expression of P/Q-type calcium channels with three unique adenoviral shRNA constructs and measured their effect on the endogenous total protein and the phosphorylation levels of CREB at Ser\textsuperscript{133}. Three unique adenoviral shRNA expression vectors (CACNA1A V1, V2, and V3), which target the expression of transcript variants 1 and 2 of the alpha 1A subunit of P/Q-type calcium channels were used to knockdown the expression of the channels in neurons. The V2 vector caused a substantial decrease in the abundance of P/Q-type calcium channels, the V1 vector caused a smaller decrease in the abundance, and V3 did not affect the abundance of the channel in primary neurons. We observed a decrease in the abundance of CREB and in the proportion of Ser\textsuperscript{133} phosphorylated CREB in cells with the V2 vector. However, the V3 vector, which did not alter P/Q-type calcium channel abundance, did not affect the abundance or phosphorylation of CREB. These experiments show that the activity or abundance of P/Q-type calcium channels can affect both the abundance and phosphorylation state of CREB. This observation is consistent with the findings of Sutton et al. (15), which shows exogenous expression of P/Q-type calcium channels can induce transcription of specific genes related to the synaptic function. Whether P/Q-type calcium channels are altering calcium signaling to affect CREB activity or whether they are functioning through a nonconducting mechanism, similar to
that described for other types calcium channels (L and N type) (16), remains to be
determined. It is also an open question as to why there are opposing effects of short-term
and long-term blockage of P/Q-type calcium channel activity on CREB phosphorylation.
A possible biological explanation for the different response between the short and long
term treatments is related to the compensatory mechanisms that re-regulate the initial
cellular response. The existing feedbacks in the circuit may explain the role of
compensatory mechanisms that occur within different time intervals in biological
systems. More detailed studies are needed to precisely address the differences among
short and long term responses.

To verify the importance of $G_{\alpha i}$ in the regulation of CREB we inhibited $G_{\alpha i}$ with
pertussis toxin (17) (see Methods) and found an increase in the abundance CREB and an
increase in Ser$^{133}$ phosphorylation. In contrast, activation of $G_{\alpha i}$ by MAS-7, an active
mastoparan analog that stimulates $G_{\alpha i}$ (17), caused a decrease in the abundance of CREB
and a decrease in Ser$^{133}$ phosphorylation (Fig. 5C). The decrease in abundance of CREB,
along with the decrease in Ser$^{133}$ phosphorylation, is consistent with an overall decrease
in the activity of CREB, as phosphorylation of CREB at Ser$^{133}$ is required for activity
(14). Thus, the active/inactive Boolean framework used for modeling and vulnerability
analysis lead to discovery of both regulation by changes in abundance and
phosphorylation state. These experiments demonstrate the ability of the fault diagnosis
engineering approach to correctly identify key regulators of signaling pathways.

With the primary neuronal cultures, we tested the ability of the constructed Boolean
network to correctly predict the output activity, following an increase in the concentration
of each input molecules. More specifically, we experimentally verified the accuracy of the following table, obtained using the logic equations of the CREB network (Supplementary table 3). This table shows the binary logic values of the output CREB, when different inputs are applied to the circuit. The first five columns include the input scenarios, whereas the last three columns are computed by simulating the circuit logic equations in ModelSim®, a simulation and debug software tool for digital circuits.

<table>
<thead>
<tr>
<th>Input ligands</th>
<th>Output</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Serotonin</td>
<td>Glutamate</td>
<td>Dopamine</td>
<td>GABA</td>
<td>Adenosine</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

According to this table, serotonin is the only input that activates CREB, when PKA is functional. To verify this prediction experimentally, we treated primary neurons that express these receptors (39) with 10 μM activatory concentration of serotonin (40), glutamate (42), dopamine(41), GABA(43), and adenosine (44) for 15 minutes then measured the activity of CREB by monitoring Ser133 phosphorylation (14). As shown in Fig. 5D, treatment with serotonin had the most robust effect on the activity of CREB, stimulating phosphorylation to a similar extent as did the PKA activator forskolin. However, exposure of the neurons to the other ligand failed to robustly activate CREB (Fig. 5D). This is consistent with the output predicted by the logic equations and listed in the above table: Serotonin should be the most effective activator of CREB. Previous studies have also reported that serotonin can induce Ser133 CREB phosphorylation (26,
40), thus the engineering model not only correctly predicts this effect of serotonin, but also shows the specificity of serotonin’s effect, compared to the other ligands.

We also tried to verify the biological relevance of the stuck-at-1 and stuck-at-0 fault models, which are included in the last two columns of the above table. We tested the effect of serotonin on the activity of CREB when PKA, a highly vulnerable molecule in this network, is either activated (stuck-at-1) or inhibited (stuck-at-0), by forskolin or H-89, respectively. When PKA is stuck-at-1 or 0, the model predicts that serotonin should no longer activate CREB (See the last two columns of the above above). To verify this experimentally, we treated primary neurons with serotonin in the presence or absence of either forskolin or H-89 (Fig. 5E). Whereas individual treatments with serotonin or forskolin activate CREB, treatment with serotonin following activation of PKA by forskolin attenuated the effect of forskolin on the activity of CREB, as compared to treatment with serotonin only. This is while the general expectation in biology is that addition of two activators of CREB, i.e., serotonin and forskolin, should enhance the activatory effect. Thus, when the highly vulnerable molecule PKA is stuck-at-1, the network output is not correctly regulated by the input. Inhibition of PKA by H-89 also prevents serotonin from stimulating the activity of CREB (Fig. 5E). Immunofluorescent analysis of CREB phosphorylation in primary neuron cultures exposed to vehicle, forskolin, or serotonin individually, or to serotonin following treatment with forskolin, was consistent with the results obtained by Western blotting for CREB phosphorylation. The cells were triple labeled with antibodies against Ser\textsuperscript{133} CREB (as a measure of CREB activity), Map-2 (as a marker that specifically labels neurons), and DRAQ-5 (as a nuclear marker that stains the nucleus of neuronal and nonneuronal cells in primary culture). Treatment of the neuronal primary cultures with forskolin increased the activity
of CREB in both neuronal and non-neuronal (glial) cells in primary culture. However, treatment with serotonin only increased the activity of CREB in neurons, not in the nonneuronal cells. This specific activation in the neuronal cell population by serotonin was expected, because neurons have receptors for serotonin whereas the nonneuronal cells in the culture do not. Consistent with the Western blot data (Fig. 5E), the effect of serotonin on the activity of CREB in neurons is attenuated, when serotonin was added after treatment with forskolin. These two experiments suggest that dysfunction of the network output occurs when PKA is stuck in either an active or inactive state. Thus, the data support the prediction of the model that serotonin fails to activate CREB when PKA is faulty, which is entirely consistent with PKA having a high vulnerability value in the network.

**Discussion**

Taken together, our experimental data indicates that fault diagnosis engineering can identify new critical regulators, and also correctly predict previously known regulators of the output molecules. Furthermore, we have provided experimental evidence that a reconstructed Boolean network can correctly predict the activity of output molecules, based on the activity levels of input signals. Finally, our experimental data supports the theoretical finding of the proposed fault diagnosis approach, specifically the experiments confirm that for proper propagation of the input signals to regulate the activity of output molecules, the normal function and activity level of highly vulnerable molecules are necessary. When these highly vulnerable molecules become faulty, stuck-at-1 or 0, the
interconnected pathways can not correctly propagate the signals from input to the output, and the molecular network does not function properly.

Boolean modeling has certain applications and it is not the claim of the paper that it can be used to precisely model all characteristics of signaling networks. The usefulness of Boolean modeling depends on the goal of the study. In the present study, Boolean framework provides a model that captures those essential characteristics of a molecular network that we needed, to determine the vulnerability of the network function to the dysfunction of each molecule. Boolean approach has provided biologically-relevant results in the paper, consistent with both the experimental findings of other research groups and the experiments of the paper. The proposed approach can be extended in several directions, to broaden its scope. For example, two rules were introduced in the paper, to build digital circuit equivalents of large signaling molecular networks. When looking at each individual molecule in the networks we studied, the two rules were found to be sufficient. Verification of the computational results with experiments further validated the approach. However, it is possible that the activation or inhibition of a particular molecule may differ from what considered in the two rules. Examples may include molecules that need sequential phosphorylation to become activated, or competitive inhibition mechanisms of some molecules. In such scenarios, certainly an appropriate mixture of AND/OR/NOT operations can be used, to properly model the activity of the molecule in terms of its inputs.

There are a number of different modeling approaches such as differential equations, Boolean models, graphs, Bayesian models, etc. We chose the Boolean modeling framework as it provides a picture of cell signaling which was sufficient for the goal we had, as confirmed by the experimental results of the paper. It also has a certain prediction
power which was validated by experiments as well. When it comes to finding important molecules, one may think of graph theory and sensitivity analysis. Graph theoretical approaches make conclusions based on the topology and connections that exist among the nodes of a graph that represents a molecular network. Based on the existing literature and to the best of our knowledge, since these assertions are not compared with experiments, we cannot comment on their usefulness regarding the identification of critical molecules.

Sensitivity analysis methods are typically used in conjunction with differential equation models of molecular networks, to determine the sensitivity of the concentration levels to the variations of kinetic parameters and rate constants (30). So, to implement such sensitivity analysis methods in large signaling networks, the nominal values of a large number of kinetic parameters and concentration levels need to be determined first. The proposed Boolean vulnerability assessment framework, however, does not use these parameters and can identify vulnerable molecules using less detailed information. Experiments are conducted and reported in the paper, to verify the usefulness of the proposed approach.

In summary, this paper takes advantage of the concepts of electronic circuit fault diagnosis engineering, to identify the vulnerable molecules that play crucial roles in the dysfunction of molecular networks. The vulnerable molecules identified in caspase-3, p53, and CREB signaling networks are functionally related sets of molecules, with physiological and pathological relevance to the specific function of each network. This indicates the usefulness of the proposed approach. Application of this technique can improve the physiological understanding of the functionality of biological systems,
identify key regulatory components, and potentially identify important targets for drug discovery.

Materials and Methods

Constructing the networks

For the caspase3-FKHR network, we used the experimentally-verified networks of Janes et al. (5,6). For the p53 and CREB networks, we employed the approach of Ma’ayan et al. (11), as follows. First the inputs nodes (ligands) and the output node (transcription factor) of interest were specified, as well as the intermediate molecules that transmit the input signals to the output. Then the type of the interactions among the molecules (activatory or inhibitory) were determined, using the existing literature and databases (See Supplementary Materials for details).

Identification of the feedback paths

We used the depth-first search (DFS) algorithm (10) to identify the feedback paths. This step is done after a digital circuit is created. First the digital circuit model is converted into a graph model. Each node in the digital circuit is modeled as a node in the corresponding graph and connections between the logic gates are modeled as edges in the graph. The DFS traversal is applied to the graph model to identify the feedback loops (10). This traversal visits all nodes of the graph one by one in the topological order by traversing the paths in the graph. Whenever a node is revisited (i.e., visited twice) in a path, it indicates the existence of a loop in the graph. Then, the loop is broken by
removing the edge and replacing the corresponding connection in the digital circuit with a flip-flop. A flip-flop is a one-bit digital logic memory unit. This process is repeated until no further loop can be identified, i.e., DFS traversal of the modified graph can be concluded without revisiting any node.

**Vulnerability analysis of the logic circuits**

We present a methodology for reliability analysis (vulnerability analysis) of logic circuits and its application to biological signaling networks to calculate the vulnerability values for the molecules in the network.

We extract the error propagation probability (EPP) of the internal nodes, which is the probability that an erroneous value on that node propagates from the error site to system outputs and results in an observable error in the system. We developed an EPP computation approach, which improves the runtime of previous EPP methods by several orders of magnitude. The presented approach uses the signal probabilities (SP) of all nodes in the combinational part and then computes EPPs based on the topological structure of the logic circuit. The signal probability (SP) of a line $l$ indicates the probability of $l$ having logic value 1 (24). Experiments on benchmark circuits and comparison of the results with the fault injection method based on random simulation show the effectiveness and the accuracy of the presented approach (9).

We first extract the structural paths from the error sites to all reachable primary outputs and then traverse these paths to compute the propagation probability of the erroneous value to the reachable primary outputs or to the reachable flip-flops. Based on the error site, we categorize nets and gates in the circuit as follows. An *on-path* signal is a net on a path from the error site to a reachable output. Also, an *on-path gate* is defined as
the gate with at least one on-path input. Finally, an off-path signal is a net that is not on-path and is an input of an on-path gate (See Supplementary Fig. 5).

For error propagation probability calculation, as we traverse the paths, we use signal probability for off-path signals and use our propagation rules for on-path signals. SP calculation and estimation techniques have been presented previously (25). The problem statement can be described as follows:

*Given the failure probability in node \( n \), calculate the probability of the propagation of this error to Primary Outputs (POs) or Flip-Flops (FFs) (that is, system failure).*

Errors can be directly propagated to a primary output and cause a system failure at the same clock cycle, or they can be propagated to flip-flops repeatedly, and finally manifest as errors at a primary output several clock cycles later.

First, consider a simple case when there is only one path from the error site to an output. As we traverse this path gate by gate, the error propagation probability from an on-path input of a gate to its output depends on the type of the gate and the signal probability of other off-path signals. In the example shown in Supplementary Fig. 6, the error propagation probability to the output of the gate C (AND gate) is the product of the probability of the output of gate A being 1 and the error probability at the PI (\( 1 \times 0.2 = 0.2 \)). Similarly, EPP at the output of the gate D (OR gate) is calculated as \( 0.2 \times (1-SP_B) = 0.2 \times 0.7 = 0.14 \).

In the general case in which reconvergent paths (one signal directly or indirectly drives more than one input of a logic gate) might exist, the propagation probability from the error site to the output of the reconvergent gate depends on not only the type of the gate and the signal probabilities of the off-path signals, but also the polarities of the
propagated error on the on-path signals. In the presence of errors, the status of each signal can be expressed with four values:

- **0**: no error is propagated to this signal line and the signal has an error-free value of 0.
- **1**: no error is propagated to this signal line and it has logic value of 1.
- **a**: the signal has an erroneous value with the same polarity as the original erroneous value at the error site (denoted by $a$).
- **$\bar{a}$**: the signal has an erroneous value, but the erroneous value has an opposite polarity compared to the erroneous value at the error site (denoted by $\bar{a}$).

Based on this four-value logic, we can redefine propagation rules for each logic gate. These probabilities, denoted by $P_a(U_i)$, $P_{\bar{a}}(U_i)$, $P_1(U_i)$, and $P_0(U_i)$, are explained as follows:

- $P_a(U_i)$ and $P_{\bar{a}}(U_i)$ are defined as the probability of the output of node $U_i$ being $a$ and $\bar{a}$, respectively. In other words, $P_a(U_i)$ is the probability that the erroneous value is propagated from the error site to $U_i$ with an even number of inversions, whereas $P_{\bar{a}}(U_i)$ is the similar propagation probability with an odd number of inversions.
- $P_1(U_i)$ and $P_0(U_i)$ are defined as the probability of the output of node $U_i$ being 1 and 0, respectively. In these cases, the error is blocked and not propagated.

Note that for on-path signals, $P_a(U_i) + P_{\bar{a}}(U_i) + P_1(U_i) + P_0(U_i) = 1$, and for off-path signals $P_1(U_j) + P_0(U_j) = 1$. Because we have considered the polarity of error effect propagation, this will take care of reconvergent points. The error propagation calculation rules for elementary gates are shown in Supplementary Table 4. To illustrate how to employ the propagation rules for reconvergent paths, consider the example shown in Supplementary Fig. 7. In this example, the error propagation probability from the output of gate A to PO is calculated. Assume that the gate A becomes erroneous. So, initially,
we set $P_0(A)=1$, $P_\bar{a}(A)=0$, $P_1(A)=0$, and $P_0(A)=0$. Then, we propagate these probabilities through gates D, E, and H. As an example, we do the following steps to compute the error propagation probability of the erroneous value to the output of gate H.

\[
P_0(H) = P_0(C) \times P_0(D) \times P_0(G) = 0.7 \times 0.8 \times 0.3 = 0.168
\]

\[
P_a(H) = (P_0(C) + P_a(C)) \times (P_0(D) + P_a(D)) \times (P_0(G) + P_a(G))
\]

\[
P_\bar{a}(H) = (P_0(C) + P_\bar{a}(C)) \times (P_0(D) + P_\bar{a}(D)) \times (P_0(G) + P_\bar{a}(G))
\]

\[
P_1(H) = 1 - (0.168 + 0.042 + 0.392) = 0.398
\]

\[
\Rightarrow P(H) = 0.042(a) + 0.392(\bar{a}) + 0.168(0) + 0.398(1)
\]

Finally, the EPP of gate A to outputs can be computed as:

\[
\text{EPP}_{A \rightarrow PO} = [P_a(H) + P_\bar{a}(H)] = (0.042 + 0.392) = 0.434
\]

In general, the following algorithm shows how we can extract and then traverse all paths from a given error site to all reachable outputs and how we apply the propagation probability rules as we traverse the paths.

**The Main Algorithm**

For every node, $n_i$, do:

1) **Path Construction**: Extract all on-path signals (and gates) from $n_i$ to every reachable primary output $PO_j$ and/or flip-flop $FF_k$. This is achieved using the forward Depth-First Search (DFS) algorithms (10).

2) **Ordering**: Prioritize signals on these paths based on their distance level using the *topological sorting* algorithm (10). Topological sort of a directed acyclic graph is an ordered list of the vertices such that if there is an edge $(u, v)$ in the graph, then $u$ appears before $v$ in the list.
3) **Propagation Probabilities Computation**: Traverse the paths in the topological order and apply propagation rules to compute the probability for each on-path node based on propagation probability rules (Supplementary Table 4).

Using the above formulation, error propagation probabilities from an arbitrary error site to any flip-flop, primary output, or both can be computed in just one pass starting from the error site to reachable output. As a result, the complexity of this approach is linear to the size of the circuit (number of logic gates). In other words, the exponential path enumeration problem is not observed in this algorithm. After computing the EPP of each node to all outputs, the overall EPP of an arbitrary node A to all primary outputs at the first clock cycle can be computed as:

\[
EPP_{c=1}(A) = 1 - \prod_{i=1}^{k} 1 - EPP_{A \rightarrow PO_i}
\]

where \(k\) is the number of primary outputs. Note that \(EPP_{c=1}(A)\) computes the EPP of node A at the first clock cycle. The transient error, however, can be captured in flip-flops and propagated to primary outputs in the next clock cycles. To compute EPP of a node in the next clock cycles \((c > 1)\), the same error propagation rules can be repeated in the next cycles. Note that in multiple cycle simulations \((c > 1)\), the transient error disappears from the original error site. In this case, flip-flops are considered as possible error sites. Using the error propagation rules \(EPP_{c=2}(A), EPP_{c=3}(A), EPP_{c=4}(A)\) are computed accordingly. Our experiments show that \(EPP_{c=x}(A)\) for \(x > 4\) becomes very close to 0 such that we can ignore these probability values without sacrificing any accuracy. Finally, we compute the overall EPP of a node A according to the following equation:
$EPP(A) = EPP_{c=1}(A) +$
\[ (1 - EPP_{c=1}(A)) \times EPP_{c=2}(A) + \]
\[ (1 - EPP_{c=1}(A)) \times (1 - EPP_{c=2}(A)) \times EPP_{c=3}(A) + \]
\[ (1 - EPP_{c=1}(A)) \times (1 - EPP_{c=2}(A)) \times (1 - EPP_{c=3}(A)) \times EPP_{c=4}(A) \]

In order to run our vulnerability analysis methodology for the biological circuits of interest, we first need to convert their set of Boolean equations (Supplementary Table 1, Supplementary Table 2 and Supplementary Table 3) into sequential circuits, without any combinational feedback (the original set of equations contain combinational feedbacks). For this purpose, the feedback paths of these circuits are extracted using the DFS algorithm (10). Then, appropriate flip-flops are inserted at each feedback path, identified by backward edges in the DFS, to convert the original circuits to sequential circuits (Supplementary Fig. 2 & 4). Because the Boolean equations of Supplementary Table 1 do not have any combinational feedback, no flip-flop is inserted in the corresponding combinational circuit (Fig. 2B). Finally, the vulnerability analysis algorithm is applied to the circuits, to extract the SP and EPP values for all the nodes.

**Developing the EPP method for the toy molecular network**

As an example, we explain the EPP method for the digital electronic circuit (Fig. 1C) of the toy molecular network (Fig. 1A). First, we need to obtain the signal probabilities (SP) of all nodes. Staring with input signal probabilities of 0.5, this yields the following:

$SP(A) = 0.5$, $SP(B) = 0.5$

$SP(D) = SP(A) = 0.5$

$SP(C) = 1 - SP(B) = 0.5$

$SP(E) = 1 - [(1 - SP(B) \times (1 - SP(D))] = 0.75$

$SP(C+D) = 1 - [(1 - SP(C) \times (1 - SP(D))] = 0.75$
SP(E') = 1 – SP(E) = 0.25

SP(F) = SP(E') × SP(C+D) = 0.185

SP(G) = 1 – [(1 – SP(F) × (1 – SP(E))] = 0.796875

Once we have the signal probability values of all nodes in the circuit, we can apply the error propagation probability rules (4) to the logic gates in the circuit, starting from input towards the output, to obtain EPP values for the output.

The topological order of this circuit is as follows:

A, B : level 0
C, D : level 1
E : level 2
F : level 3
G : level 4

Given an error in input A (dysfunction in the corresponding molecule), the probability that this error affects the output (causing the system to fail) is obtained as follows:

Since A is erroneous, Pa(A)=1, P\(\bar{a}\)(A)=0, P\(_1\)(A)=0, and P\(_0\)(A)=0.

Pa(B)=0, P\(\bar{a}\)(B)=0, P\(_1\)(B)=0.5, P\(_0\)(B)=0.5

Pa(C)=0, P\(\bar{a}\)(C)=0, P\(_1\)(C)=0.5, P\(_0\)(C)=0.5

Pa(D) = Pa(A)=1, Pa(D) = Pa(A)=0, P\(_1\)(D) = P\(_1\)(A)=0, P\(_0\)(D) = P\(_0\)(A)=0

P\(_0\)(E) = P\(_0\)(B)× P\(_0\)(D) = 0

P\(_d\)(E)=[0.5×1] – 0 = 0.5 ,

P\(_d\)(E)=0,
P\(_1\)(E)=1 – (0+0.5+0) = 0.5,
P\(_d\)(E')=0, P\(_d\)(E')=0.5, P\(_1\)(E')=0, P\(_0\)(E')=0.5

P\(_0\)(C+D)= P\(_0\)(C)× P\(_0\)(D) = 0
\[ P_a(C+D) = [0.5 \times 1] - 0 = 0.5, \]

\[ P_a(C+D) = 0, \]

\[ P_1(C+D) = 1 - (0 + 0.5 + 0) = 0.5, \]

\[ P_1(F) = P_1(C+D) \times P_1(E') = 0, \]

\[ P_a(F) = [0 \times 1] - 0 = 0, \]

\[ P_a(F) = [0.5 \times 0.5] - 0 = 0.25, \]

\[ P_0(F) = 1 - (0 + 0 + 0.25) = 0.75, \]

\[ P_0(G) = P_0(E) \times P_0(F) = 0 \]

\[ P_a(G) = [0.5 \times 0.75] - 0 = 0.375, \]

\[ P_a(G) = 0, \]

\[ P_1(G) = 1 - (0 + 0.375 + 0) = 0.625. \]

The probability of error propagation from A to output G is calculated as

\[ \text{EPP}(A \rightarrow G) = P_a(G) + P_a(G) = 0.375. \]

Similarly, the error propagation probabilities from all other nodes to the output can be computed and the signals can be ranked based on their EPPs.

**Primary cortical culture**

Primary cortical cultures were prepared from brains of embryonic day 17 to 18 Sprague-Dawley rats (Charles River). After trituration of cortical sections with a glass pipette, 2 to 4 \( \times 10^5 \) neurons were plated on a coverslip (diameter: 12 mm) precoated with poly d-lysine (BD Biocoat). For biochemical analysis, primary cortical cells were plated in 35 mm dishes precoated with poly d-lysine (~3 \( \times 10^6 \) neurons/dish). Neurons were grown in
NEUROBASAL Medium supplemented with 0.5 mM L-glutamine, B27 (2%), and N2 (1%) supplements.

**Protein extraction, and immunoblot analysis**

Primary neurons were cultured as described. Cells were homogenized in ice cold lysate buffer (0.25 M Tris, pH 7.5) containing protease inhibitors (Protease Inhibitor Cocktail tablets, Boehringer Mannheim) and phosphatase inhibitors (Phosphatase Inhibitor Cocktails I & II, Sigma) and lysed through three cycles of freezing (in liquid nitrogen) and thawing (in 37°C water bath). Protein concentration was measured by Bio-Rad's protein assay and spectrometry at 595 nm. Equal amounts of total protein were loaded on 4–12% gradient Bis-Tris gels, separated using the NuPAGE system (Invitrogen) and transferred onto nitrocellulose membrane. The membrane was probed with primary and secondary antibodies and signals were detected by chemilluminescence followed by autoradiography. The following antibodies were used: anti-CREB antibody (Cell Signaling, 1:1000), anti-phospho Ser133 CREB antibody (Cell Signaling, 1:1000), anti P/Q type calcium channel antibody (Chemicon, 1:1000), and anti-actin antibody (Sigma, 1:1000).

**Immunofluorescent studies**

Cortical neurons were grown on coverslips, fixed for 10 min in PBS plus 3.7% formaldehyde, and permeabilized for 2 min with cold acetone. Coverslips were coated with 100 μl of primary antibody diluted in PBS (anti Phospho Ser133 CREB, Cell Signaling, 1:200, anti Map-2 antibody, Upstate Biotechnology, 1:250). Coverslips
washed three times and labeled with Alexa 568 anti-rabbit antibody (1:500), Alexa 488 anti-mouse antibody (1:500), and the nuclear marker DRAQ-5 (1:10,000).

**Analysis of CREB regulation by P/Q-type calcium channels or Gαi**

Primary cortical neurons were cultured as described. We followed the methods of Dolmetsch et al. (27) to analyze the effect of calcium channel blockers on CREB activity. We characterized the signaling properties of the endogenous P/Q-type calcium channels by monitoring the endogenous activity and total protein levels of CREB. In order to minimize the effect of other elements of calcium or G protein signaling following neuronal depolarization, all experiments were performed without depolarizing the neurons. Previous time-course studies have shown that short-term Ser\textsuperscript{133} phosphorylation of CREB are transient events and prolonged Ser\textsuperscript{133} phosphorylation (more than 40 minutes) is required for transcriptional activity (27, 28, 29). Therefore, in these experiments we analyzed the endogenous Ser\textsuperscript{133} phosphorylation and total CREB protein abundance following at least 2 hours treatment with agonist or antagonists of P/Q-type calcium channels and 2 hours treatment with agonist or antagonist of Gαi. w-Agatoxin IVA (Calbiochem), pertussis toxin (Sigma), and Mas-7 (Calbiochem) were dissolved in the appropriate vehicle and added to the medium. Following the treatment, cortical neurons were harvested, lysed, and subjected to Western blot analysis as described.

**Adenoviral gene knockdown**

We used AdenoSilence\textsuperscript{TM} RNAi viral vector kit (Millipore, Cat# GAL10021) to target the expression of P/Q-type calcium channels in primary neuronal cultures. This kit
consisted of three unique adenoviral shRNA constructs targeting the transcript variants 1 and 2 of the alpha 1A subunit of P/Q-type calcium channels (CACNA1A). Following the kit instruction, $60 \times 10^6$ VPU (Viral Particle Unit) from the crude virus of each construct was added to 50 ul of complete medium and added to 35-mm dishes of primary neuronal cultures. Cells were harvested 48 hours after viral transduction, protein was extracted, and the lysates were subject to Western blot analysis.

**Supplementary Materials**

**Supplementary Table 1.** Logic equations of the caspase3-FKHR network.
**Supplementary Figure 1.** Intermolecular interactions of the p53 network.
**Supplementary Table 2.** Logic equations of the p53 network.
**Supplementary Figure 2.** The digital p53 circuit.
**Supplementary Figure 3.** Intermolecular interactions of the CREB network.
**Supplementary Table 3.** Logic equations of the CREB network.
**Supplementary Figure 4.** The digital CREB circuit.
**Supplementary Figure 5.** A typical path between an erroneous node to primary outputs and flip-flops.
**Supplementary Figure 6.** A simple path between an erroneous input to a primary output.
**Supplementary Figure 7.** Applying error propagation rules for a reconverging path.
**Supplementary Table 4.** Computing probability at the output of a gate in terms of its inputs.

**Supplementary References**

**References**


45. **Acknowledgement.** We would like to thank Mr. H. Asadi (Northeastern University, Boston) for helping us with the analysis of digital circuits. We also thank Dr. A. Ma’ayan and Dr. R. Iyengar (Mount Sinai School of Medicine, New York) for useful comments regarding the use of their SAVI software tool.

**Fig. 1. A toy molecular network.** (A) The input molecules of the seven node toy model are $A$ and $B$, the intermediate molecules are $C$, $D$, $E$, $F$, and finally $G$ is the output molecule. The activatory and inhibitory signals are shown by lines ended to $\rightarrow$ and $\cdot\cdot\cdot$, respectively. (B) Logic equations of the molecular network of Fig. 1A, used to generate the equivalent digital electronic circuit in Fig. 1B. The symbols $\times$, $+$ and $'$ in these equations stand for the binary logic operations AND, OR and NOT, respectively. (C) The
digital electronic circuit derived for the molecular network of Fig. 1A. The function of each logic gate (circuit component) is specified in the square box. Names of some molecules in the circuit diagram may appear few times, for the convenience of the reader, to easily recognize the inputs and outputs of the logic gates. A small black dot is used to show where a wire is branched out from another wire.

**Fig. 2. The caspase3-FKHR network.** (A) The caspase3-FKHR network is based on (5, 6) and has a total of 22 nodes. The input molecules are EGF, insulin, and TNF, and the output molecules are caspase3 and FKHR. The node ComplexI includes TNFR and TRADD-RIP-TRAF2 (5), whereas the node ComplexII stands for TRADD-RIP-TRAF2 and FADD (5). (B) The digital electronic caspase3-FKHR circuit, derived based on the caspase3-FKHR molecular network.

**Fig. 3. The p53 network.** (A) This network has a total of 49 molecules. The input molecules are insulin and PDGF, and the output molecule is p53. All the 94 intermolecular interactions of this network are listed in the Supplementary Materials, along with literature citations. (B) The vulnerability bar chart of the p53 circuit. The vulnerabilities of PIP2, AKT, caspase3, and PP2A are greater than 0.5, whereas those of caspase8, phosphoinositide 3-kinase (PI3K), Abelson leukemia tyrosine kinase (Abl), phosphatidylinositol 3,4,5 triphosphate (PIP3), the lipid phosphatase PTEN, and protein kinase C (PKC) are between 0.1 and 0.5. The vulnerabilities of the rest of the molecules are less than 0.1. (C) The histogram of the vulnerability values of the p53 circuit. The
number at the top of each bar represents the total number of molecules whose vulnerabilities fall within the range specified by the location of that bar.

**Fig. 4. The CREB network.** (A) This network has a total of 64 molecules. The input molecules are glutamate, dopamine, GABA, serotonin, ACh, adenosine, and enkephalin, and the output molecule is CREB. All the 152 intermolecular interactions of this network are listed in the Supplementary Materials, along with literature citations. (B) The vulnerability bar chart of the CREB circuit. The vulnerabilities of calmodulin, Ca$^{2+}$, cAMP, G$\alpha_i$, AC2, AC1, AC5, PKA, P/Q-type calcium channels, and PP2A are greater than or equal to 0.5, whereas those of G$\beta\gamma$, calcium/calmodulin dependent protein kinase II (CaMKII), protein phosphatase 2B (PP2B), calcium/calmodulin dependent protein kinase type IV (CaMKIV), calmodulin-dependent protein kinase kinase (CaMKK), cAMP response element modulator (CREM), N-type calcium channels (abbreviated N-type CaCh in circuit), N-methyl-D-aspartate type glutamate receptor (NMDAR), PI3K, PIP3, the kinase PDK1, and the kinase RSK are between 0.1 and 0.5. The vulnerabilities of the rest of the molecules are less than 0.1. (C) The histogram of the vulnerability values of the CREB circuit. The number at the top of each bar represents the total number of molecules whose vulnerabilities fall within the range specified by the location of that bar.

**Fig. 5. Experimental verification of the fault diagnosis findings.** (A) Western blot analysis on the protein extracts from primary neuronal culture after 2.5 and 12 hours treatment with either vehicle or 1 $\mu$M concentration of the selective P/Q-type calcium
channel blocker, ω-Agatoxin IVA. Each blot shown in this figure is a representative blot of three independent experiments. (B) 48 hours following adenoviral transduction using three unique shRNA constructs targeting P/Q-type calcium channels. Lane 1 shows the protein size by a protein marker (Magic Marker from Invitrogen). Lanes 2 and 6 are loaded with protein extracts from two control plates (mock) and lanes 3, 4 and 5 are treated with V1, V2, and V3 adenoviral vectors, respectively. Top blot shows a decrease in P/Q-type calcium channel abundance with V2 vector (lane 4) and a smaller decrease with the V1 vector (lane 3). Middle blots show the total protein and Ser\textsuperscript{133} phosphorylation of CREB in the same set of samples. The same membrane of the top blot was stripped and re-probed with anti-CREB antibody. Bottom blot shows total actin protein of the same membrane, as the loading control. (C) After 2 hours treatment with 0.1 or 0.2 μg/ml of the G\alpha\textsubscript{i} antagonist, pertussis toxin, and 5 or 10 μM of G\alpha\textsubscript{i} agonist, MAS-7. Blots were first probed with antibody against phospho-CREB (Ser\textsuperscript{133}) (top) then stripped and re-probed with antibody against CREB (middle) and the antibody against actin (bottom), as the loading control for confirming equal loading. (D) The activity of CREB following treatment with different neurotransmitter ligands, including serotonin (STN), glutamate (GLT), dopamine (DPM), GABA, and adenosine (ADN). Top blot shows phosphorylation of CREB at Ser\textsuperscript{133}, 15 minutes after treatment with vehicle, 10 μM forskolin (FSK), serotonin, glutamate, dopamine, GABA, and adenosine, respectively. Middle blot shows total CREB protein on the same blot. Bottom blot shows actin as loading control. (E) The effect of treatment with serotonin on the CREB activity when PKA is activated with forskolin or inhibited with H-89. Cells were treated with vehicle, 10 μM forskolin or H-89 for 30 minutes followed by 15 minutes treatment with 10 μM
serotonin. (F) Immunofluorescent analysis of primary cortical culture following treatment with vehicle, forskolin, serotonin, or serotonin and forskolin. Red represents the phosphorylated CREB (Ser^{133}) as a measure of CREB activity. Green represents the Map-2 staining as a specific neuronal marker and blue represent DRAQ-5 nuclear staining of all the cells in the primary culture (both neuronal and glial cells). Images were captured with the same confocal parameters for the four different treatment conditions.

**Table 1.** Truth table of the caspase3-FKHR digital circuit (Fig. 2B). It shows the binary logic values of the outputs, FKHR and caspase3, when different inputs are applied to the circuit. The first three columns include all possible input scenarios, whereas the last three columns are computed according to circuit logic equations (Supplementary Table 1) and the basic properties of binary operations as described in the text.

**Table 2.** The vulnerability values of the caspase3-FKHR circuit and the caspase3 circuit in which the output node FKHR and its preceding BUFFER are removed. The vulnerabilities are calculated using the EPP algorithm (8,9), and are sorted from high to low. In both circuits, AKT has the highest vulnerability value.

**Table 3.** The vulnerability values of the p53 circuit, calculated using the EPP algorithm and sorted from high to low.

**Table 4.** The vulnerability values of the CREB circuit, calculated using the EPP algorithm and sorted from high to low.
Abdi et al., Fig. 1
Abdi et al., Fig. 2
Abdi et al., Fig. 4
<table>
<thead>
<tr>
<th>EGF</th>
<th>INSULIN</th>
<th>TNF</th>
<th>FKHR</th>
<th>CASPASE3</th>
<th>CASPASE3 (AKT IS STUCK AT 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

*Abdi et al., Table 1*
<table>
<thead>
<tr>
<th>NODE</th>
<th>VUL. (WITH FKHR)</th>
<th>VUL. (WITHOUT FKHR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASPASE3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>FKHR</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>AKT</td>
<td>1</td>
<td>0.874</td>
</tr>
<tr>
<td>EGF</td>
<td>0.5</td>
<td>0.249</td>
</tr>
<tr>
<td>EGFR</td>
<td>0.5</td>
<td>0.249</td>
</tr>
<tr>
<td>Insulin</td>
<td>0.49</td>
<td>0.245</td>
</tr>
<tr>
<td>MEKK1ASK1</td>
<td>0.249</td>
<td>0.249</td>
</tr>
<tr>
<td>TNF</td>
<td>0.249</td>
<td>0.249</td>
</tr>
<tr>
<td>ComplexI</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>ComplexII</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>ERK</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>IRS1</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>JNK1</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>MEK</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>MK2</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>MKK3</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>MKK7</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>CASPASE8</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>p38</td>
<td>0.125</td>
<td>0.125</td>
</tr>
<tr>
<td>IKK</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>NFKB</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cFLIPL</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

*Abdi et al., Table 2*
<table>
<thead>
<tr>
<th>NODE</th>
<th>VUL.</th>
<th>NODE</th>
<th>VUL.</th>
<th>NODE</th>
<th>VUL.</th>
</tr>
</thead>
<tbody>
<tr>
<td>p53</td>
<td>1</td>
<td>Dok</td>
<td>0.067</td>
<td>ROCK</td>
<td>0.022</td>
</tr>
<tr>
<td>PIP2</td>
<td>0.88</td>
<td>RasGAP</td>
<td>0.067</td>
<td>IRS1</td>
<td>0.021</td>
</tr>
<tr>
<td>AKT</td>
<td>0.88</td>
<td>AC2</td>
<td>0.064</td>
<td>Gab1</td>
<td>0.019</td>
</tr>
<tr>
<td>CASPASE3</td>
<td>0.87</td>
<td>ILK</td>
<td>0.062</td>
<td>RyR</td>
<td>0.018</td>
</tr>
<tr>
<td>PP2A</td>
<td>0.64</td>
<td>Vav</td>
<td>0.061</td>
<td>PDGF</td>
<td>0.017</td>
</tr>
<tr>
<td>CASPASE8</td>
<td>0.43</td>
<td>PLCβ</td>
<td>0.05</td>
<td>PDGFR</td>
<td>0.017</td>
</tr>
<tr>
<td>P13K</td>
<td>0.35</td>
<td>Ca²⁺</td>
<td>0.046</td>
<td>IP3</td>
<td>0.015</td>
</tr>
<tr>
<td>Abl</td>
<td>0.29</td>
<td>AC5</td>
<td>0.042</td>
<td>DAG</td>
<td>0.013</td>
</tr>
<tr>
<td>PIP3</td>
<td>0.23</td>
<td>Rho</td>
<td>0.038</td>
<td>PMCA</td>
<td>0.011</td>
</tr>
<tr>
<td>PTEN</td>
<td>0.23</td>
<td>Grb10</td>
<td>0.036</td>
<td>CASPASE9</td>
<td>0.007</td>
</tr>
<tr>
<td>PKC</td>
<td>0.12</td>
<td>IRS2</td>
<td>0.036</td>
<td>Grb2</td>
<td>0.0052</td>
</tr>
<tr>
<td>cAMP</td>
<td>0.086</td>
<td>PLCγ</td>
<td>0.033</td>
<td>Cdc42</td>
<td>0.0015</td>
</tr>
<tr>
<td>PKA</td>
<td>0.086</td>
<td>PIP5K</td>
<td>0.032</td>
<td>Rac</td>
<td>0.0015</td>
</tr>
<tr>
<td>Insulin</td>
<td>0.082</td>
<td>IP3R</td>
<td>0.029</td>
<td>APS</td>
<td>0.00025</td>
</tr>
<tr>
<td>IR</td>
<td>0.082</td>
<td>PAK</td>
<td>0.026</td>
<td>NCK</td>
<td>0.00012</td>
</tr>
<tr>
<td>SHP2</td>
<td>0.073</td>
<td>JNK</td>
<td>0.023</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDK1</td>
<td>0.071</td>
<td>JIP</td>
<td>0.023</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Abdi et al., Table 3*
<table>
<thead>
<tr>
<th>NODE</th>
<th>VUL.</th>
<th>NODE</th>
<th>VUL.</th>
<th>NODE</th>
<th>VUL.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREB</td>
<td>1</td>
<td>RSK</td>
<td>0.12</td>
<td>A2AR</td>
<td>0</td>
</tr>
<tr>
<td>Calmodulin</td>
<td>0.74</td>
<td>Adenosine</td>
<td>0.05</td>
<td>CaMKI</td>
<td>0</td>
</tr>
<tr>
<td>CA²⁺</td>
<td>0.6</td>
<td>Serotonin</td>
<td>0.05</td>
<td>CBP</td>
<td>0</td>
</tr>
<tr>
<td>cAMP</td>
<td>0.59</td>
<td>5-HT1AR</td>
<td>0.05</td>
<td>cJun</td>
<td>0</td>
</tr>
<tr>
<td>Galphai</td>
<td>0.58</td>
<td>A1R</td>
<td>0.045</td>
<td>DAG</td>
<td>0</td>
</tr>
<tr>
<td>AC2</td>
<td>0.58</td>
<td>Dopamine</td>
<td>0.04</td>
<td>5-HT2AR</td>
<td>0</td>
</tr>
<tr>
<td>AC1</td>
<td>0.57</td>
<td>Ach</td>
<td>0.035</td>
<td>5-HT4R</td>
<td>0</td>
</tr>
<tr>
<td>AC5</td>
<td>0.57</td>
<td>Enkephalin</td>
<td>0.035</td>
<td>GABABR</td>
<td>0</td>
</tr>
<tr>
<td>PKA</td>
<td>0.5</td>
<td>mGluR7</td>
<td>0.035</td>
<td>Galphas</td>
<td>0</td>
</tr>
<tr>
<td>P/Q type CaCh</td>
<td>0.5</td>
<td>D1R</td>
<td>0.025</td>
<td>Grb2</td>
<td>0</td>
</tr>
<tr>
<td>PP2A</td>
<td>0.5</td>
<td>D2R</td>
<td>0.025</td>
<td>GSK3</td>
<td>0</td>
</tr>
<tr>
<td>Gbeta gamma</td>
<td>0.43</td>
<td>D3R</td>
<td>0.025</td>
<td>ILK</td>
<td>0</td>
</tr>
<tr>
<td>CaMKII</td>
<td>0.43</td>
<td>D0R</td>
<td>0.025</td>
<td>M1R</td>
<td>0</td>
</tr>
<tr>
<td>PP2B</td>
<td>0.27</td>
<td>K0R</td>
<td>0.025</td>
<td>PIP2</td>
<td>0</td>
</tr>
<tr>
<td>CaMKIV</td>
<td>0.27</td>
<td>M2R</td>
<td>0.025</td>
<td>AKT</td>
<td>0</td>
</tr>
<tr>
<td>CaMKK</td>
<td>0.27</td>
<td>M4R</td>
<td>0.025</td>
<td>PKC</td>
<td>0</td>
</tr>
<tr>
<td>CREM</td>
<td>0.27</td>
<td>M0R</td>
<td>0.025</td>
<td>PLCβ</td>
<td>0</td>
</tr>
<tr>
<td>N-type CaCh</td>
<td>0.25</td>
<td>N0R</td>
<td>0.025</td>
<td>PLCγ</td>
<td>0</td>
</tr>
<tr>
<td>NMDAR</td>
<td>0.24</td>
<td>Galphaz</td>
<td>0.02</td>
<td>RasGAP</td>
<td>0</td>
</tr>
<tr>
<td>P13K</td>
<td>0.12</td>
<td>Glutamate</td>
<td>0.01</td>
<td>SAM68</td>
<td>0</td>
</tr>
<tr>
<td>PIP3</td>
<td>0.12</td>
<td>mGluR1</td>
<td>0.01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDK1</td>
<td>0.12</td>
<td>GABA</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Abdi et al., Table 4*
**Supplementary Materials**

**Vulnerability assessment algorithm**

Here we summarize our molecular vulnerability assessment algorithm, through which the vulnerability of a cellular signaling network to the dysfunction of its components can be calculated. The details of each step are explained in the Method Section.

1- Specify the inputs nodes (such as ligands, receptors, secondary messengers, etc.) and the output nodes (such as different transcription factors relevant to the input signal), as well as the intermediate molecules that allow the input signals to propagate from the inputs to the outputs. Then specify the type of the interactions among the molecules (stimulatory or inhibitory), using the existing literature.

2- Use Rule #1 and Rule #2, to derive a binary logic equation for every intermediate molecule and the output molecules, using the interactions specified in Step 1.

3- Construct the digital circuit of the network from the binary logic equations of Step 2, using the AND, OR, NOT and BUFFER digital circuit elements.

4- Identify the feedback paths of the digital circuit of Step 3, using the depth-first search (DFS) algorithm. If there is no feedback path, proceed directly to the next step.

5- Finally, apply the EPP algorithm to the circuit obtained in the previous step, to calculate the vulnerability levels of all the input and intermediate nodes (the vulnerability of the output node is always 1, since if the output node is dysfunctional, the network will not operate efficiently anyway).
**Deriving the logic equations**

The logic equation of each molecule is a symbolic Boolean expression that shows how the activity of the molecule is regulated by its inputs. Using Rule #1 and Rule #2, a binary logic equation for each intermediate molecule and the output molecule were obtained, in terms of the input stimulators and inhibitors. These two rules are devised based on the known physiological mechanisms that different regulators employ to control the activity of signaling molecules. Rule #1 applies the Boolean OR of activating inputs to a signaling network. Rule #2 applies the Boolean AND of inverted inhibitory inputs to the network. In the derived equations for the networks, \( ', + \) and \( \times \) stand for the binary logic operations NOT, OR and AND, respectively.

**Constructing the digital circuits**

To obtain the digital circuit schematic of a set of logic equations for a particular network, the \( ', + \) and \( \times \) operations were represented in the circuit by NOT, OR and AND circuit components, respectively. Equations of the form \( X = Y \), where \( X \) and \( Y \) are two different molecules and \( X \) is activated by \( Y \), were implemented using the BUFFER circuit component.
**Supplementary Table 1. Logic equations of the caspase3-FKHR network.** Each logic equation specifies the input signals to a molecule using the logic operations ‘’, + and ×, which represent NOT, OR and AND, respectively. These equations are used to generate the digital electronic caspase3-FKHR circuit (Fig. 2B).

<table>
<thead>
<tr>
<th>Molecule</th>
<th>Logic equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AKT</td>
<td>AKT=EGFR+Insulin</td>
</tr>
<tr>
<td>Caspase3</td>
<td>Caspase3=AKT’×(Caspase8+JNK1+MK2)</td>
</tr>
<tr>
<td>Caspase8</td>
<td>Caspase8=cFLIP_1’×(ComplexII+ERK)</td>
</tr>
<tr>
<td>cFLIP_1</td>
<td>cFLIP_1=NFκB</td>
</tr>
<tr>
<td>ComplexI</td>
<td>ComplexI=TNF</td>
</tr>
<tr>
<td>ComplexII</td>
<td>ComplexII=TNF+ComplexI</td>
</tr>
<tr>
<td>EGFR</td>
<td>EGFR=EGF</td>
</tr>
<tr>
<td>ERK</td>
<td>ERK=MEK</td>
</tr>
<tr>
<td>FKHR</td>
<td>FKHR=AKT</td>
</tr>
<tr>
<td>IKK</td>
<td>IKK=ComplexI</td>
</tr>
<tr>
<td>IRS1</td>
<td>IRS1=Insulin</td>
</tr>
<tr>
<td>JNK1</td>
<td>JNK1=MKK7</td>
</tr>
<tr>
<td>MEK</td>
<td>MEK=EGFR+IRS1</td>
</tr>
<tr>
<td>MEKK1ASK1</td>
<td>MEKK1ASK1=ComplexI</td>
</tr>
<tr>
<td>MK2</td>
<td>MK2=p38</td>
</tr>
<tr>
<td>MKK3</td>
<td>MKK3=MEKK1ASK1</td>
</tr>
<tr>
<td>MKK7</td>
<td>MKK7=MEKK1ASK1</td>
</tr>
<tr>
<td>NFκB</td>
<td>NFκB=IKK</td>
</tr>
<tr>
<td>p38</td>
<td>p38=MKK3</td>
</tr>
</tbody>
</table>
Supplementary Fig. 1. Intermolecular interactions of the p53 network.
This network (Fig. 3A) includes the above 94 interactions, listed alphabetically, according to the name of “source” molecules. At least one representative reference from the national library of medicine (Pub Med) is listed for each individual interaction. These references are listed at the end of the Supplementary Materials.

<table>
<thead>
<tr>
<th>Molecule</th>
<th>Logic equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abl</td>
<td>Abl=NCK+PKC+PAK</td>
</tr>
<tr>
<td>AC2</td>
<td>AC2=PKC</td>
</tr>
<tr>
<td>AC5</td>
<td>AC5=(Ca^{2+})×PKC</td>
</tr>
<tr>
<td>APS</td>
<td>APS=PDGFR</td>
</tr>
<tr>
<td>Ca^{2+}</td>
<td>Ca^{2+}=PMCA^{1+}×(IP3R+RyR)</td>
</tr>
<tr>
<td>cAMP</td>
<td>cAMP=AC2+AC5</td>
</tr>
<tr>
<td>CASPASE3</td>
<td>CASPASE3=PIP2′×(CASPASE8+CASPASE9)</td>
</tr>
<tr>
<td>CASPASE8</td>
<td>CASPASE8=PIP2</td>
</tr>
<tr>
<td>CASPASE9</td>
<td>CASPASE9=PIP2′×AKT</td>
</tr>
<tr>
<td>Cdc42</td>
<td>Cdc42=Vav</td>
</tr>
<tr>
<td>DAG</td>
<td>DAG=PLCβ+PLCγ</td>
</tr>
<tr>
<td>Dok</td>
<td>Dok=Abl</td>
</tr>
<tr>
<td>Gab1</td>
<td>Gab1=SHP2′×Grb2</td>
</tr>
<tr>
<td>Grb2</td>
<td>Grb2=IRS1+SHP2</td>
</tr>
<tr>
<td>Grb10</td>
<td>Grb10=IR</td>
</tr>
</tbody>
</table>
Fault Diagnosis Engineering of Cellular Signaling Pathways  Abdi et al.

<table>
<thead>
<tr>
<th>Molecule</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILK</td>
<td>ILK=PIP3</td>
</tr>
<tr>
<td>IP3</td>
<td>IP3=PLCβ+PLCγ</td>
</tr>
<tr>
<td>IP3R</td>
<td>IP3R=PKA’×IP3</td>
</tr>
<tr>
<td>IR</td>
<td>IR=Insulin</td>
</tr>
<tr>
<td>IRS1</td>
<td>IRS1=JNK’×PKC’×IR</td>
</tr>
<tr>
<td>IRS2</td>
<td>IRS2=IR</td>
</tr>
<tr>
<td>JIP</td>
<td>JIP=ROCK+JNK</td>
</tr>
<tr>
<td>JNK</td>
<td>JNK=JIP’</td>
</tr>
<tr>
<td>NCK</td>
<td>NCK=IRS1</td>
</tr>
<tr>
<td>p53</td>
<td>p53=AKT’×(Abl+PKA+JNK)</td>
</tr>
<tr>
<td>PAK</td>
<td>PAK=Cdc42+NCK+Rac</td>
</tr>
<tr>
<td>PDK1</td>
<td>PDK1=PIP3</td>
</tr>
<tr>
<td>PDGFR</td>
<td>PDGFR=PDGF</td>
</tr>
<tr>
<td>PI3K</td>
<td>PI3K=Gab1+Grb10+IRS1+IRS2+SHP2</td>
</tr>
<tr>
<td>PIP2</td>
<td>PIP2=PI3K’×PLCβ’×PLCγ’×PIP5K</td>
</tr>
<tr>
<td>PIP3</td>
<td>PIP3=PTEN’×PI3K</td>
</tr>
<tr>
<td>PIP5K</td>
<td>PIP5K=CASPASE3’×Rho</td>
</tr>
<tr>
<td>PKA</td>
<td>PKA=cAMP</td>
</tr>
<tr>
<td>AKT</td>
<td>AKT=PP2A’×(ILK+PDK1+PIP2+PIP3+RasGAP)</td>
</tr>
<tr>
<td>PKC</td>
<td>PKC=PDK1+PIP3+DAG+Ca^{2+}</td>
</tr>
<tr>
<td>PLCβ</td>
<td>PLCβ=PKA’×PKC’</td>
</tr>
<tr>
<td>PLCγ</td>
<td>PLCγ=PKA’×PDGFR</td>
</tr>
<tr>
<td>PMCA</td>
<td>PMCA=PKA’×PKC’×PIP2</td>
</tr>
<tr>
<td>PP2A</td>
<td>PP2A=CASPASE3’</td>
</tr>
<tr>
<td>PTEN</td>
<td>PTEN=CASPASE3’</td>
</tr>
<tr>
<td>Rac</td>
<td>Rac=Vav</td>
</tr>
<tr>
<td>RasGAP</td>
<td>RasGAP=Dok</td>
</tr>
<tr>
<td>Rho</td>
<td>Rho=Vav</td>
</tr>
<tr>
<td>ROCK</td>
<td>ROCK=Rho</td>
</tr>
<tr>
<td>RyR</td>
<td>RyR=PIP2</td>
</tr>
<tr>
<td>SHP2</td>
<td>SHP2=PDGFR+PKC</td>
</tr>
<tr>
<td>Vav</td>
<td>Vav=Abl+APS+Grb2+PIP3</td>
</tr>
</tbody>
</table>

**Supplementary Table 2. Logic equations of the p53 network.** Each logic equation specifies the input signals to a molecule using the logic operations ’, + and ×, which represent NOT, OR and AND, respectively. These equations are used to generate the digital p53 circuit (Supplementary Fig. 2).
Supplementary Fig. 2. The digital electronic p53 circuit. There are seven feedback paths in this circuit, initiated from the following nodes: Ca\(^{2+}\), JNK, caspase3, Gab1, PLC\(\beta\), Abl, and PKC. These feedbacks are identified using the DFS algorithm, and then one flip-flop (FF) is inserted in each path (FF is a one-bit digital logic memory unit). For example, there is an FF at the upper right corner of the circuit, with PKC and PKC_FF as its input and output, respectively. The name PKC_FF appears again at the lower left corner of the circuit, which means that PKC is “fed back” from the right side of the circuit to the left side. The feedback wire itself is not shown, to make the circuit diagram easier to read.
Fault Diagnosis Engineering of Cellular Signaling Pathways

Supplementary Fig. 3. Intermolecular interactions of the CREB network.

This network (Fig. 4A) includes 152 interactions, listed alphabetically according to the name of “source” molecules. At least one representative reference from the national library of medicine (Pub Med) is listed for each individual interaction. These references are listed at the end of the Supplementary Materials.
Molecule | Logic equation
---|---
A1R | A1R=Adenosine
A2AR | A2AR=Adenosine
AC1 | AC1=Gbetagamma’×Galphaz’×(Galphas+Calmodulin)
AC2 | AC2=Galphi’×(Gbetagamma+Galphas+PKC)
AC5 | AC5=Galphi’×Galphaz’×(Ca^{2+})’×(PKC+Galphas)
PP2B | PP2B=Calmodulin
Ca^{2+} | Ca^{2+}=NMDAR+N-typeCaCh+P/QtypeCaCh
Calmodulin | Calmodulin=Ca^{2+}
CaMKI | CaMKI=PP2A’×CaMKK
CaMKII | CaMKII=Calmodulin+NMDAR
CaMKIV | CaMKIV=PP2A’×PP2B’×(Calmodulin+CaMKK)
CaMKK | CaMKK=PKA’×Calmodulin
cAMP | cAMP=AC1+AC2+AC5
CBP | CBP=SAM68’×PKA
cJun | cJun=GSK3’×CBP
CREB | CREB=PP2A’×PP2B’×CREM’×(GSK3+PKA+cJun+RSK+CaMKII +CaMKIV+CaMKI)
CREM | CREM=CaMKIV
D1R | D1R=PKA’×Dopamine
D2R | D2R=Dopamine
D3R | D3R=Dopamine
DAG | DAG=PLCβ+PLCγ
DOR | DOR=Enkephalin
5-HT1AR | 5-HT1AR=Serotonin
5-HT2AR | 5-HT2AR=Serotonin
5-HT4R | 5-HT4R=Serotonin
GABABR | GABABR=GABA’
Galphai | Galphai=A1R+D1R+D2R+D3R+M4R+M2R+mGluR_{γ}+DOR+KOR +MOR+NOR+5-HT1AR
Galphas | Galphas=A2AR+D1R+5-HT1AR+5-HT4R
Galphaz | Galphaz=D2R+5-HT1AR
Gbetagamma | Gbetagamma=A1R+A2AR+M1R+M4R+M2R+D1R+D2R+D3R +GABABR+mGluR_{γ}+mGluR_{γ}+DOR+KOR+MOR+NOR +5-HT1AR+5-HT2AR+5-HT4R
Grb2 | Grb2=D3R
GSK3 | GSK3=ILK’×AKT’×PKC’×PP2A
ILK | ILK=PIP3
KOR | KOR=Enkephalin
M1R | M1R=Ach
M2R | M2R=Ach
M4R | M4R=Ach
mGluR_{1} | mGluR_{1}=PKC+Glutamate
### Supplementary Table 3. Logic equations of the CREB network.

Each logic equation specifies the input signals to a molecule using the logic operations ’, + and ×, which represent NOT, OR and AND, respectively. These equations are used to generate the digital electronic CREB circuit (Supplementary Fig. 4).

<table>
<thead>
<tr>
<th>Molecule</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>mGluR7</td>
<td>mGluR7=Calmodulin’×(PKC+PKA+Glutamate)</td>
</tr>
<tr>
<td>MOR</td>
<td>MOR=Enkephalin</td>
</tr>
<tr>
<td>NMDAR</td>
<td>NMDAR=PKC’×PP2A’×Calmodulin’×PP2B’×(PKA+Glutamate)</td>
</tr>
<tr>
<td>NOR</td>
<td>NOR=Enkephalin</td>
</tr>
<tr>
<td>N-type CaCh</td>
<td>N-typeCaCh=Gbetagamma’×(PKC+CaMKII)</td>
</tr>
<tr>
<td>PDK1</td>
<td>PDK1=PIP3</td>
</tr>
<tr>
<td>PI3K</td>
<td>PI3K=Gbetagamma</td>
</tr>
<tr>
<td>PIP2</td>
<td>PIP2=PI3K’×PLCβ’×PLCγ’</td>
</tr>
<tr>
<td>PIP3</td>
<td>PIP3=PI3K</td>
</tr>
<tr>
<td>PKA</td>
<td>PKA=cAMP</td>
</tr>
<tr>
<td>AKT</td>
<td>AKT=PP2A’×(RasGAP+ILK+PIP3+PDK1+PIP2+CaMKK)</td>
</tr>
<tr>
<td>PKC</td>
<td>PKC=PDK1+PIP3+DAG+Ca²⁺</td>
</tr>
<tr>
<td>PLCβ</td>
<td>PLCβ=PKA’×PKC’×(Gbetagamma+Calmodulin)</td>
</tr>
<tr>
<td>PLCγ</td>
<td>PLCγ=PKA’×NMDAR</td>
</tr>
<tr>
<td>PP2A</td>
<td>PP2A=CaMKII’</td>
</tr>
<tr>
<td>P/Q type CaCh</td>
<td>P/Q type CaCh=PKA’×Gbetagamma</td>
</tr>
<tr>
<td>RasGAP</td>
<td>RasGAP=Gbetagamma</td>
</tr>
<tr>
<td>RSK</td>
<td>RSK=PDK1</td>
</tr>
<tr>
<td>SAM68</td>
<td>SAM68=Grb2</td>
</tr>
</tbody>
</table>

**Supplementary Fig. 4. The digital electronic CREB circuit (see next page).** There are five feedback paths in this circuit, initiated from the following nodes: PLCγ, PP2A, calmodulin, PKA and PKC. These feedbacks are identified using the DFS algorithm, and then one flip-flop (FF) is inserted in each path. For example, there is an FF at the upper right corner of the circuit, with PKA and PKA_FF as its input and output, respectively. The name PKA_FF is appeared again at the upper left corner of the circuit, which means that PKA is “fed back” from the right side of the circuit to the left side. The feedback wire itself is not shown, to make the circuit diagram easier to read.
Supplementary Fig. 5

Supplementary Fig. 5. A typical path between an erroneous node to primary outputs and flip-flops.

Supplementary Fig. 6

Supplementary Fig. 6. A simple path between an erroneous input to a primary output.
Supplementary Fig. 7

Supplementary Fig. 7. Applying error propagation rules for a reconverging path.

<table>
<thead>
<tr>
<th>AND</th>
<th>$P_{1}(\text{out}) = \prod_{i=1}^{n} P_{1}(X_i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$P_{a}(\text{out}) = \prod_{i=1}^{n} [P_{1}(X_i) + P_{a}(X_i)] - P_{1}(\text{out})$</td>
</tr>
<tr>
<td></td>
<td>$P_{a}(\text{out}) = \prod_{i=1}^{n} [P_{1}(X_i) + P_{a}(X_i)] - P_{1}(\text{out})$</td>
</tr>
<tr>
<td></td>
<td>$P_{a}(\text{out}) = 1 - [P_{1}(\text{out}) + P_{a}(\text{out}) + P_{a}(\text{out})]$</td>
</tr>
</tbody>
</table>

| OR | $P_{0}(\text{out}) = \prod_{i=1}^{n} P_{0}(X_i)$ |
|    | $P_{a}(\text{out}) = \prod_{i=1}^{n} [P_{0}(X_i) + P_{a}(X_i)] - P_{0}(\text{out})$ |
|    | $P_{a}(\text{out}) = \prod_{i=1}^{n} [P_{0}(X_i) + P_{a}(X_i)] - P_{0}(\text{out})$ |
|    | $P_{a}(\text{out}) = 1 - [P_{0}(\text{out}) + P_{a}(\text{out}) + P_{a}(\text{out})]$ |

| NOT | $P_{0}(\text{out}) = P_{1}(\text{in}), P_{1}(\text{out}) = P_{0}(\text{in})$ |
|     | $P_{a}(\text{out}) = P_{a}(\text{in}), P_{a}(\text{out}) = P_{a}(\text{in})$ |

Supplementary Table 4. Computing probability at the output of a gate in terms of its inputs.
**Supplementary References**


59. Imagawa, T. *et al.* Purified ryanodine receptor from skeletal muscle sarcoplasmic reticulum is the Ca\textsuperscript{2+}-permeable pore of the calcium release channel. J. Biol. Chem. 262, 16636 (1988).


96. Moeschel, K. et al. Protein kinase C-zeta-induced phosphorylation of Ser318 in insulin receptor substrate-1 (IRS-1) attenuates the interaction with the insulin receptor and the tyrosine phosphorylation of IRS-1. J. Biol. Chem. 279, 25157 (2004).


120. Schuebel, K. E. et al. Phosphorylation-dependent and constitutive activation of Rho proteins by wild-type and oncogenic Vav-2. EMBO. J. 17, 6608 (1999).


164. Yabana, N. & Shibuya, M. Adaptor protein APS binds the NH2-terminal autoinhibitory domain of guanine nucleotide exchange factor Vav3 and augments its activity. Oncogene. 21, 7720 (2002).


175. Yue, Y. et al. Ras GTPase-activating protein binds to Akt and is required for its activation. J. Biol. Chem. 279, 12883 (2004).
