(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 October 2023 (12.10.2023)

WIPO | PCT

(10) International Publication Number
WO 2023/196124 A1

(51) International Patent Classification:
G06F 30/30 (2020.01)

(21) International Application Number:
PCT/US2023/016197

(22) International Filing Date:
24 March 2023 (24.03.2023)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
63/329,117 08 April 2022 (08.04.2022) US
18/068,560 20 December 2022 (20.12.2022) US

(71) Applicant: **MICROCHIP TECHNOLOGY INCORPORATED** [US/US]; 2355 West Chandler Blvd., Chandler, Arizona 85224-6199 (US).

(72) Inventor: **TWIGG, Christopher**; 618 Valleyview Drive, Endwell, New York 13760 (US).

(74) Agent: **SLAYDEN, Bruce W., II**; Slayden Grubert Beard PLLC, 401 Congress Ave., Suite 1650, Austin, Texas 78701 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ,

(54) Title: SYSTEM-LEVEL DESIGN TOOL FOR SELECTING AND CONFIRMING COMPATABILITY OF ELECTRICAL COMPONENTS

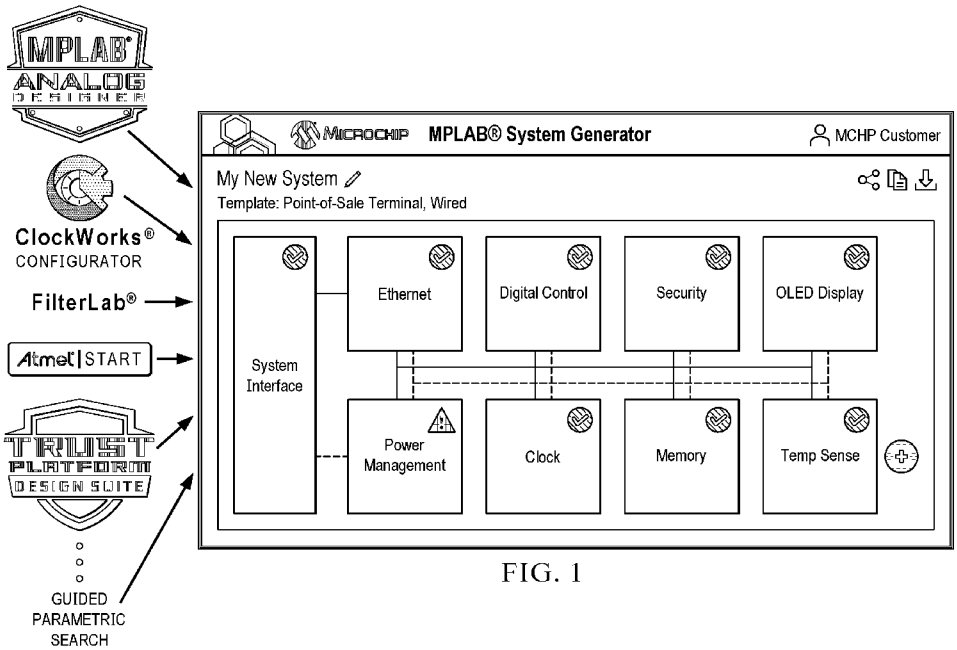


FIG. 1

(57) Abstract: A system-level design generator tool enables users to configure system-level designs made up of electrical components that satisfy design parameters and have corresponding dependencies, e.g., power, communication, control, security, clock and memory. A block diagram representing each of the functions within the system may utilize a guided parametric search or a design tool to create the solution for each individual block. The inputs into that search and/or design tool may come from the system-level choices, constraints, and cross dependencies tracked by the system generator tool.

[Continued on next page]

WO 2023/196124 A1

DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

— *with international search report (Art. 21(3))*

SYSTEM-LEVEL DESIGN TOOL FOR SELECTING AND CONFIRMING COMPATABILITY OF ELECTRICAL COMPONENTS

RELATED APPLICATION

This application claims priority to commonly owned United States Patent Application No. 63/329,117 filed April 8, 2022, the entire contents of which are hereby incorporated by reference for all purposes

TECHNICAL FIELD

The present disclosure relates to systems and methods for selecting electrical components and designing functional blocks that form a total system-level solution with cross-functional-block dependency tracking and constraint management.

BACKGROUND

Semiconductor, microchip, and microprocessor suppliers and manufacturers have software tools that enable users to select individual components having specific performance characteristics and design parameters. Component identification tools are available that provide information about components (e.g., schematics, data sheets, or software, without limitation) given specific design parameters. However, customers need further tools to identify components for complete system solutions formed with the components for particular applications. Further, a customer's specific design parameters rarely match 100% with template designs for complete system architectures or complete system solutions.

Many engineers lack the experience or expertise to design all of the various functions that are involved in a system solution. To aid this, many semiconductor vendors provide reference designs that can be dropped into their customer's application or design tools that guide them on designing a specific block. Reference designs are a good starting point, but they nearly always require some amount of modification for the customer's specific needs. Reference designs are the classic approach to providing customers with a "known good" system solution. The issue is that these are static and not easily modified without manual checking of cross-block dependencies. An example is adding additional functional blocks to an existing reference design that cause a particular supply rail to be overloaded.

Design tools are also often provided to help the customer design a specific block. However, the block is tackled by the design tools independently of the rest of the blocks in the system architecture or complete system solution. The user must know what all of the inherited

constraints are in the system and track the dependencies across blocks. This can lead to bad component/circuit choices when integrated into a full system, leading to iterative cycles of physical board debugging and redesign.

There is a need for a selection and diagnostic tool to configure system-level electrical designs made up of electrical components that satisfy design parameters and have corresponding dependencies.

SUMMARY

A system-level design generator tool enables users to configure system-level designs made up of components that satisfy design parameters and have corresponding dependencies. At the top level, this may be a block diagram representing each of the functions within the system. Each block may utilize a guided parametric search or a design tool to create the solution for that individual block. The inputs into the guided parametric search and/or design tool may come from the system-level choices, constraints, and cross-block dependencies tracked by the system generator tool. This process may start from a template diagram or a blank diagram. A template diagram contains components in a particular configuration and provides a starting point from which modifications to the diagram may be made. A blank diagram contains no components and provides a clean starting point into which components may be added and modified. When a new component block is added to a system configuration, a system generator tool that tracks cross-block dependencies may recognize the additional load placed upon the specific supply rail and flag it as needing attention. The user may then visit the design tool for component block to pick a new solution for that rail that is more appropriate for the current requirements.

According to one aspect, there is provided an apparatus to configure system-level designs made up of electrical components that satisfy design parameters and have corresponding dependencies, comprising a non-transitory, machine-readable medium including instructions wherein the instructions, when loaded and executed by a processor, configure the processor to: present a system diagram of an electronic system having component blocks representing electrical components; receive constraints or parameters for each electrical component represented by the component blocks; present a list of candidate components corresponding to the constraints or parameters for each electrical component represented by the component blocks; receive an instruction to select a component from the list candidate components for each electrical component represented by the component blocks; and check

system dependencies to confirm compatibility of all electrical components selected for inclusion in the system diagram.

Another aspect provides a machine-implemented method to configure system-level designs made up of electrical components that satisfy design parameters and have corresponding dependencies, comprising: presenting a system diagram having component blocks; receiving constraint or parameters for the components of the component blocks; presenting a list of components corresponding to the constraints or parameters; receiving an instruction to select a components from the list; and checking system dependencies to confirm compatibility of all components in the system diagram.

According to another aspect, there is provided a system having: a presentation circuit that presents a system diagram of an electronic system having component blocks representing electrical components; a constraint or parameter input circuit that receives constraint or parameter inputs for each electrical component represented by the component blocks; a component circuit that presents a list of candidate components corresponding to the constraints or parameters for each electrical component represented by the component blocks; a component selection circuit that receives an instruction to select a component from the list candidate components for each electrical component represented by the component blocks; and a system dependency circuit that checks system dependencies to confirm compatibility of all electrical components selected for inclusion in the system diagram.

DESCRIPTION OF THE DRAWINGS

The figures illustrate example methods and systems for a system-level design generator tool that enables users to configure system-level designs made up of components that satisfy design parameters and have corresponding dependencies.

FIGURE 1 illustrates an example system-level design generator tool based on tools for selecting individual components of the system.

FIGURE 2 illustrates, relative to a system generator tool for creating a new electrical control system, an example screen shot of a search screen for finding a template related to a user's application.

FIGURE 3 illustrates a screen shot of an example template.

FIGURE 4 illustrates an example window to modify the System Interface block.

FIGURE 5 illustrates an example of the component block template of FIGURE 3, which illustrates how the system generator tool performs dependency checks between the component

blocks after a component block has been modified via a configuration screen as shown in FIGURE 4.

FIGURE 6 illustrates an example screen for adding component blocks to those already in the system being generated.

FIGURE 7 illustrates an example of the system generated screen of FIGURE 5, where two additional component blocks have been added to the system.

FIGURE 8 illustrates an example of the system generated screen of FIGURE 7, where the two new component blocks have been configured by the user and cross-dependency checks have been performed.

FIGURE 9 illustrates an example screen for updating the Power Management component.

FIGURE 10 illustrates an example screen for viewing rail details, including the connections and inherited constraints.

FIGURE 11 illustrates an example screen for viewing inputs and outputs of the PS1 sub-block of the Power Management component block.

FIGURE 12 illustrates an example window for updating the rail generation for component blocks or sub-blocks.

FIGURE 13 illustrates an example of how users may share and export design files from the system generator tool window.

FIGURE 14 illustrates an example flow chart of a process for generating an electronic system comprised of semiconductor and other components.

FIGURE 15 illustrates an example schematic diagram of a system generator tool.

The reference number for any illustrated element that appears in multiple different figures has the same meaning across the multiple figures, and the mention or discussion herein of any illustrated element in the context of any particular figure also applies to each other figure, if any, in which that same illustrated element is shown.

DETAILED DESCRIPTION

According to examples provided, a system generator tool may guide engineers in choosing system elements and checking cross-block dependencies and cross-function dependencies, provide a working solution for engineers, for example, add a power supply for non-power engineers, and provide an initial starting point for engineers. The system generator tool may be a software tool that helps customers select parts represented by component blocks

and design functional capabilities that work together to form a system solution. The tool may check cross-block and cross-function dependencies and make component recommendations. An engineer may start from a template diagram (reference design) and customize the template diagram to add/remove component blocks, and change functional capabilities of component blocks, such as power, bandwidth, and noise. While power, bandwidth, and noise are illustrated, any parameter or functional capability of component blocks may be checked. Alternatively, an engineer may start from a blank diagram.

The system generator tool may help mass market customers select the right semiconductor parts, components, or products for their specific electronics application and design the parts, components, or products into a total system solution. A feature of the system generator tool is its ability to track system level constraints and dependencies, which provides a solution for the customer's needs, not a static reference design or merely a bucket of parts, components, or products that may or may not work together. The system generator tool helps reduce the chance of gross errors in a new system design. It also helps speed up such designs or modifications of reference designs allowing customers to get their designs to market in fewer iterations of the hardware.

The system generator tool may provide product selection and design guidance for total system solutions to mass market customers. The system generator tool may guide engineers in choosing system elements and checking dependencies. The system generator tool may provide a working solution for engineers, for example, by recommending a power supply for non-power engineers. The system generator tool may provide an initial starting point for engineers, for example by allowing them to work from a template and then customize as needed. The system generator tool may minimize system-level inconsistencies that lead to early prototype failure. The system generator tool may track cross-block and cross-function dependencies so that dependencies are tracked and applied across all the component functional blocks of the entire system.

The system generator tool may be a web-based interface so that users may access the tool via the internet from any remote terminal. The system generator tool may be provided via a website, but it could also be built in a stand-alone, downloadable form. The system generator tool may be an application (App) installed on a computer work station or a device or a web based application accessed from the internet via a computer work station or device. A value of

the online version of the system generator tool application may be that it always reflects the most recent updates in design tools, part selection, or inherited dependencies/constraints.

FIGURE 1 is an example system generator tool illustrating building blocks including: system interface, Ethernet, digital control, security, power management, clock, and memory. Individual systems for identifying microchips and other components for individual component blocks may be incorporated into the system generator tool. For example, such individual systems may include: MPLAB Analog Designer, available from Microchip Technology, Inc., of Chandler Arizona, ClockWorks Configurator available from Microchip Technology, Inc., of Chandler Arizona, FilterLab, and Guided Parametric Search, which are example software programs capable of identifying and configuring electronic parts, components, or products. The system generator tool may control the individual systems to build or design total systems comprising microchips and other components. The system generator tool may incorporate already available software for selecting individual components into a software tool that provides a total system solution.

FIGURE 2 illustrates an example screen shot of a system generator tool for creating a new electrical control system. A customer may use a customer account to log into the web-based interface of the system. A historical listing of systems or projects by the user may be listed under a My Systems catalog. An icon link to development tools or reference design pages may be provided. A list of templates may be provided wherein the name of each template of the list of templates is a respective hyperlink to launch a template upon which the user may click to generate a new system. Alternatively, the user may start from a blank diagram, rather than a template. Templates may include system level templates such as: smart thermostats, BLE-controlled aquarium light, robotic vacuum cleaner, air conditioner compressor motor driver, and Bluetooth LED lamp.

FIGURE 3 illustrates an example screen shot of a template, wherein in this case, it is a template for a wired point-of-sale terminal. Users can modify the name of a “local” copy of the system from the original template. This template provides for system components represented by blocks including: system interface, Ethernet, digital control, security, power management, clock, and memory. Each system component is represented by a component block and interface connections are drawn between the component blocks. A grey circle with a plus sign illustrates an icon for adding more component blocks to the system. Icons may indicate the status of each block, wherein a green circle with a check mark may indicate the status is GOOD and a yellow

triangle with an exclamation point may indicate the status of the component block is NEEDS ATTENTION. If the status is NEEDS ATTENTION, the components identified or the selected parameters of an identified component may not be correct to satisfy cross-block and cross-function dependencies. Parameters may include power, bandwidth, and noise. FIGURE 4 illustrates an example of a window to modify the System Interface block. The System Interface block itself is a hyperlink icon (*see* FIGURE 3) that allows a user to navigate to the System Interface configuration screen shown in FIGURE 4. The System Interface may have input supplies and digital interface characteristics. For the input supplies, a rail name of VinPoE is listed having 37 volt minimum, 57 volt maximum, and the standard is 802.3af. For the digital interfaces, an Ethernet 1 is listed, which is a type of Ethernet having parameters of 1000BASE-T. Users may modify the initial system interface template to match their application, such as changing the input supplies from VinPoE to a local 12 V supply.

FIGURE 5 illustrate and example of the component block template of FIGURE 3, which illustrates how the system generator tool performs dependency checks between the component blocks after a component block has been modified via a configuration screen as shown in FIGURE 4. Each of the blocks having a green circle with a check mark indicates the status of all dependencies between the components is GOOD. Note that in FIGURE 3, the System Interface block has a yellow triangle with an exclamation point, which indicates the status of the block is NEEDS ATTENTION. After the System Interface has been changed via the configuration screen shown in FIGURE 4, the status of the System Interface block has changed to a green circle with a check mark indicating the status of all dependencies of the System Interface are GOOD, as shown in FIGURE 5. However, because the System Interface block was reconfigured (FIGURE 4), the Power Management block status is now indicated as NEEDS ATTENTION, as shown in FIGURE 5. This may indicate the power management parameters are not correct to supply a proper level of power, i.e., the power dependency between the component blocks is not compatible or a better solution may be available as a result of the change of the system input voltage. Typically, incompatible dependencies exist between more than two component blocks. Before selecting a new component for power management, a determination may be made as to whether, or not, all component blocks have been added. If not, additional component blocks may be added before updating the power management block. A grey circle with a plus sign illustrates an icon for adding more component blocks to the system.

FIGURE 6 illustrates an example screen for adding blocks, representing additional components, to those already in the system being generated. A search feature allows the user to search for component blocks. Also, search filters may be implemented to narrow the field of component blocks. A list of component blocks may be provided as searched and filtered results. Component blocks may be added to the system being generated by selecting the green plus icon adjacent to the component description. After selecting all desired component blocks, the user may navigate back to the system generator tool screen.

FIGURE 7 illustrates an example of the system generated screen of FIGURE 5, where two additional new component blocks have been added to the system: OLED Display and Temp Sense. Added component blocks OLED Display and Temp Sense need review to select interfaces and specific solutions. With the added component blocks integrated into the system being generated, the system generator tool again performs dependency checks between the component blocks. For example, blocks having a respective green circle with a check mark indicates the status of all dependencies between the components is GOOD. In this case, all component blocks are indicated as GOOD, except for the newly added component blocks (OLED Display and Temp Sense) and the Power Management block, which have a yellow triangle with an exclamation point indicating the status of the block is NEEDS ATTENTION. The user may select the OLED Display and the Temp Sense component blocks to navigate to block modification screens to identify more compatible components, i.e., components where the cross-block dependencies and cross-function dependencies may check out as GOOD. *See* FIGURE 4. For example, the OLED Display shown in FIGURE 7 may be replaced with an LCD as shown in FIGURE 8.

FIGURE 8 illustrates an example of the system generated screen of FIGURE 7, where the two component blocks have been replaced or reconfigured: LCD and Temp Sense. With the replaced or reconfigured component blocks integrated into the system being generated, the system generator tool again performs dependency checks between the component blocks. The blocks having a respective green circle with a check mark indicates the status of all dependencies between the components is GOOD. In this case, all component blocks are GOOD, except for the Power Management block, which has a yellow triangle with an exclamation point indicating the status of the block is NEEDS ATTENTION. Because the Power Management component provides power to all the other components, it may be more efficient to configure this component after the other components have been identified. The user

may navigate to a screen for updating the Power Management block by selecting the Power Management block.

FIGURE 9 illustrates an example screen for updating the Power Management component. In this case, the power input is Vin12V and the power outputs include: VddIO, VddCore, and VddAna. By hovering over the respective input and output blocks, the user may see summaries or select to see details. A summary of the VddCore block shows it is 1.2 V @ 50 mA. The sub-blocks PS2 and PS3, which provide power for VddCore and VddAna respectively, are indicated with a GOOD status by the green circle with a check mark. However, the sub-block for PS1 indicates the status of the block is NEEDS ATTENTION. The user may select the respective sub-blocks to change the input source or override automatic parameters, such as load estimates. This may also be the interface to the MPLAB Analog Designer (MAD), which allows the user to select a specific solution for this sub-block. After updating the sub-blocks of the Power Management component block, the user may navigate back to the system generator tool system-level screen. The user may also view the rail details by clicking on the lines between the sub-blocks or component blocks.

FIGURE 10 illustrates an example screen for viewing rail details, including the connections to the rail and inherited constraints. In this case, the VddAna rail details are shown. Blocks connected to this rail provide respective requirements, such as load estimates and constraints. The constraints may be broad constraint classes such as restricting to low-dropout regulators, or specific performance criteria, such as tolerances and noise. The digital control of the VddAna sub-block has 1.62 Min (V), 3.63 Max (V), 4mA Load, and no constraints. The temp sense of the VddAna sub-block has 2.3 Min (V), 5.5 Max (V), 0.015 mA Load, and low noise constraints. The user may close the rail detail window to navigate back to the system generator tool system-level screen.

FIGURE 11 illustrates an example screen for viewing inputs and outputs of the PS1 sub-block of the Power Management component block. *See* FIGURE 9. In this case, the user may select the link for the component to view/edit the selection, or select a different component. The Inputs include Vin and En block signals which are both mapped to the system net Vin12V. The user may select these parameters to map them to other system-level signals. The Output is a Vout block signal where the system net is VddIO. The user may select the output rails to view/edit the detailed parameters, as illustrated in FIGURE 10. The user may close the input/output detail window to navigate back to the system generator tool screen.

FIGURE 12 illustrates an example window for updating the rail generation for component blocks or sub-blocks. This window updates sub-block PS1 of the Power Management component block shown in FIGURE 9. The rail generation window allows the user to view or alter design parameters and filters, prepopulated by the PS1 sub-block constraints, for querying the MPLAB Analog Designer (MAD) to identify component blocks and sub-blocks that satisfy the needed design parameters. In particular, the MPLAB Analog Designer (MAD) may provide a list of recommendations (solution selection step) from which a user may select a specific solution to start a design generation phase, also utilizing MAD. When the user selects the specific power sub-blocks, the system generator tool may make calls to the solution database, for example implemented in MAD. The user may select appropriate components that meet the design parameters and navigate back to the system generation window. The system generator tool again performs dependency checks between the component blocks. Component blocks having a respective green circle with a check mark indicates the status of all dependencies between the components is GOOD.

FIGURE 13 illustrates an example of how users may share and export design files from the system generator tool. Once the system generator tool has generated the system design, the system design can be shared with colleagues and/or downloaded. Downloads may include a block diagram, schematics, simulation files, a bill of materials, or software/firmware libraries, a printed circuit board layout, and software, without limitation.

FIGURE 14 illustrates an example flow chart of a process for generating a system comprised of semiconductors and other components. The algorithm starts and prompts the user to select 10 a new system. The new system may be selected from a template diagram or a blank diagram. If a blank diagram is selected, the user builds 20 the system by adding component blocks. The user is then prompted to select 40 individual components for each of the component blocks in the system. The system receives 42 component constraints and parameters to configure an individual component block. The system generator tool then presents 44 a list of component candidates from the database of components. The system receives user selections 46 of a component from the list or the system generator tool selects the component from the list. The subroutine 40, 42, 44, 46 for selecting individual components is repeated for each component block in the system diagram. After all the individual components in the system diagram have been selected, the system generator tool checks 50 for system constraints and dependencies and indicates to the user whether each block is GOOD or NEEDS ATTENTION.

The user may then modify 60 a block by returning to the subroutine 40, 42, 44, 46 for selecting individual components. The user may also add or remove 70 component blocks from the system diagram by returning to the subroutine 40, 42, 44, 46 for selecting individual components. If no blocks are to be modified and no blocks are to be added or removed, the system generator tool allows the user to output 80 system diagrams, schematics, and bills of materials.

FIGURE 15 illustrates an example schematic diagram of a system generator tool 150 having a processor 151 and a non-transitory machine readable medium 152. The system generator tool 150 also has a system diagram presentation circuit 153 presenting a system diagram of an electronic system having component blocks representing electrical components , a constraints or parameters input circuit 154 receiving constraints or parameters for each electrical component represented by the component blocks, a candidate component presentation circuit 155 presenting a list of candidate components corresponding to the constraints or parameters for each electrical component represented by the component blocks, a component selection circuit 156 receiving an instruction to select a component from the list candidate components for each electrical component represented by the component blocks, and a check system dependencies circuit 157 checking system dependencies all electrical components selected for inclusion in the system diagram. System generator tool 150, system diagram presentation circuit 153, constraints or parameters input circuit 154, candidate component presentation circuit 155, component selection circuit 156, and check system dependencies circuit 157 may be implemented by instructions for execution by processor 151, analog circuitry, digital circuitry, control logic, digital logic circuits programmed through hardware description language, application specific integrated circuits (ASIC), field programmable gate arrays (FPGA), programmable logic devices (PLD), or any suitable combination thereof, whether in a unitary device or spread over several devices. System generator tool 150, system diagram presentation circuit 153, constraints or parameters input circuit 154, candidate component presentation circuit 155, component selection circuit 156, and check system dependencies circuit 157 may be implemented by instructions for execution by processor 151 through, for example, a function, application programming interface (API) call, script, program, compiled code, interpreted code, binary, executable, executable file, firmware, object file, container, assembly code, or object. For example, system generator tool 150, system diagram presentation circuit 153, constraints or parameters input circuit 154, candidate component presentation circuit 155, component selection circuit 156, and check

system dependencies circuit 157 may be implemented by instructions stored in the non-transitory medium 152 such as a memory that, when loaded and executed by the processor 151 such as CPU (or any other suitable processor), cause the functionality of system generator tool 150, system diagram presentation circuit 153, constraints or parameters input circuit 154, candidate component presentation circuit 155, component selection circuit 156, and check system dependencies circuit 157 described herein.

In particular examples, the system tool may call other tools via APIs, wherein the system may be a network of processors that perform the tasks.

Although examples have been described above, other variations and examples may be made from this disclosure without departing from the spirit and scope of these disclosed examples.

CLAIMS

1. An apparatus comprising:

a processor

a non-transitory, machine-readable medium including instructions wherein the instructions, when loaded and executed by the processor, cause the processor to build system-level designs made up of electrical components that have cross-block or cross-function dependencies by:

presenting a system diagram of an electronic system having component blocks representing electrical components;

receiving constraints or parameters for each electrical component represented by the component blocks;

presenting a list of candidate components corresponding to the constraints or parameters for each electrical component represented by the component blocks;

receiving an instruction to select a component from the list candidate components for each electrical component represented by the component blocks; and

checking system cross-block or cross-function dependencies for all electrical components selected for inclusion in the system diagram.

2. The apparatus of claim 1, further comprising instructions, when loaded and executed by a processor, cause the processor to build system-level designs by adding a component block to the system diagram.

3. The apparatus of claim 1, further comprising instructions, when loaded and executed by a processor, configure the processor to build system-level designs by removing a component block from the system diagram.

4. The apparatus of claim 1, further comprising instructions, when loaded and executed by a processor, configure the processor to build system-level designs by outputting a description of the electrical components of the system diagram.

5. The apparatus of claim 1, 2, 3, or 4, wherein the presenting of a system diagram of an electronic system comprises presenting a template diagram.

6. The apparatus of claim 1, 2, 3, or 4, wherein the receiving constraints or parameters comprises receiving constraints or parameters selected from power, bandwidth, and noise.

7. The apparatus of claim 1, 2, 3, or 4, wherein the instructions, when loaded and executed by the processor, cause the processor to build the system-level designs by receiving an instruction to modify a component block by selecting a new electrical component.

8. The apparatus of claim 4, wherein the description of the electrical components of the system diagram is selected from a diagram, a schematic, a bill of materials, a printed circuit board layout, and software.

9. A method comprising:

building system-level designs made up of electrical components that satisfy design parameters and have cross-block or cross-function dependencies by:

presenting a system diagram having component blocks;

receiving constraint or parameters for the components of the component blocks;

presenting a list of components corresponding to the constraints or parameters;

receiving an instruction to select a component from the list; and

checking system cross-block or cross-function dependencies of all components in the system diagram.

10. The method of claim 9, further comprising receiving an instruction to modify a component block by selecting a new electrical component.

11. The method of claim 9, further comprising adding a component block to the system diagram.

12. The method of claim 9, further removing a component block from the system diagram.

13. The method of claim 9, further comprising outputting a description of the electrical components of the system diagram.

14. The method of claim 9, 10, 11, 12, or 13, wherein the presenting a system diagram comprises presenting a template diagram.

15. The method of claim 9, 10, 11, 12, or 13, wherein the receiving constraint or parameters comprises receiving a constraint or parameter selected from power, bandwidth, and noise.

16. The method of claim 13, wherein the description of the electrical components of the system diagram is selected from a diagram, a schematic, a bill of materials, a printed circuit board layout, and software.

17. A system comprising:

a presentation circuit that presents a system diagram of an electronic system having component blocks representing electrical components;

a constraint or parameter input circuit that receives constraint or parameter inputs for each electrical component represented by the component blocks;

a component circuit that presents a list of candidate components corresponding to the constraints or parameters for each electrical component represented by the component blocks;

a component selection circuit that receives an instruction to select a component from the list candidate components for each electrical component represented by the component blocks; and

a system dependency circuit that checks system dependencies to confirm compatibility of all electrical components selected for inclusion in the system diagram.

18. The system of claim 17, wherein the constraint or parameters circuit receives constraint or parameter inputs selected from power, bandwidth, and noise.

19. The system of claim 17 or 18, wherein the presentation circuit outputs a description of the electrical components of the system diagram selected from a diagram, a schematic, a bill of materials, a printed circuit board layout, and software.

20. The system of claim 17 or 18, wherein the dependencies checked by the system dependency circuit are selected from power, communication, digital control, security, clock, and memory.

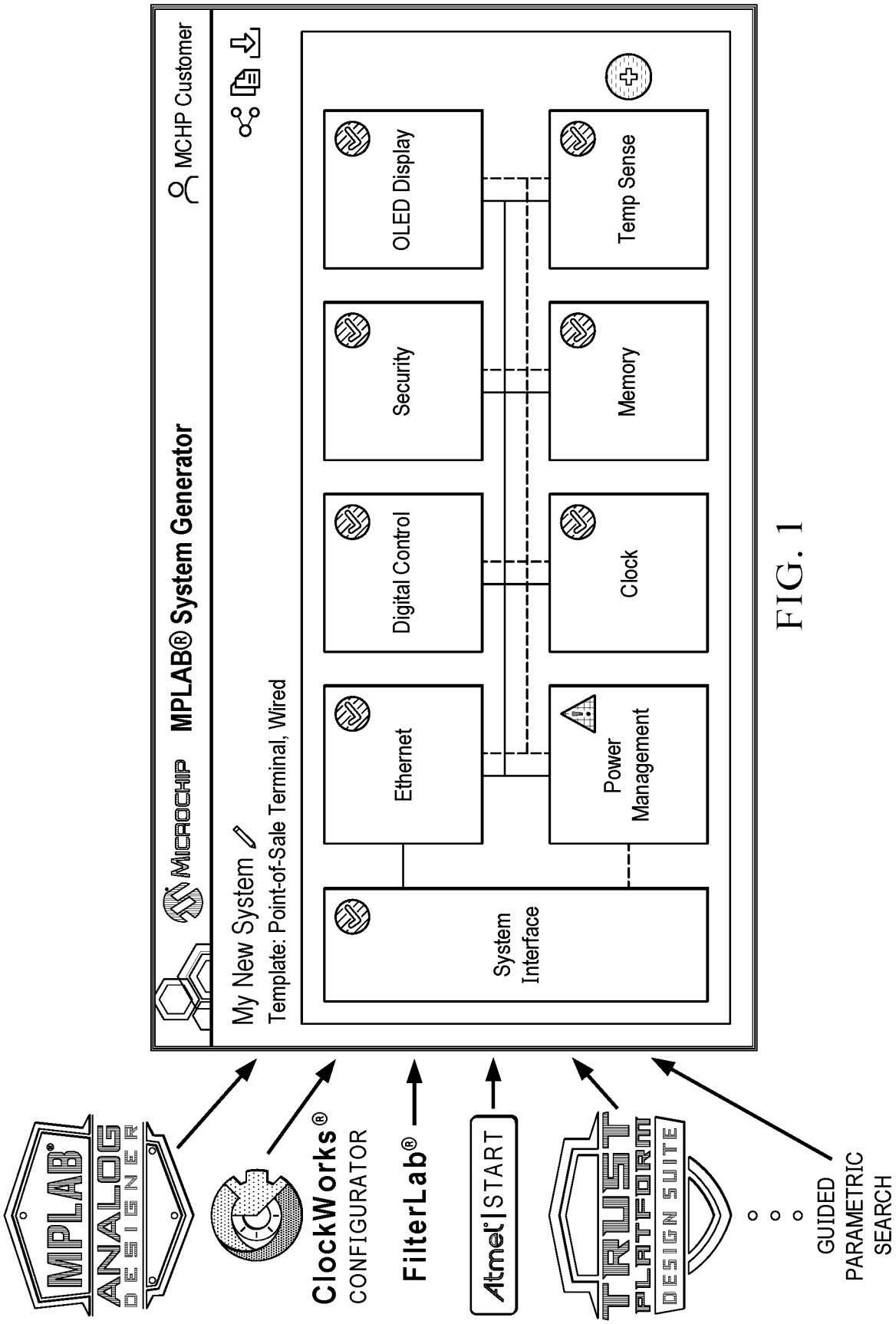

MPLAB® ANALOG DESIGNER
ClockWorks® CONFIGURATOR
FilterLab®
Atmel START
TRUST PLATFORM DESIGN SUITE
GUIDED PARAMETRIC SEARCH
MICROCHIP
MPLAB® System Generator
MCHP Customer
My New System
Template: Point-of-Sale Terminal, Wired
System Interface
Ethernet
Digital Control
Security
OLED Display
Power Management
Clock
Memory
Temp Sense

FIG. 1

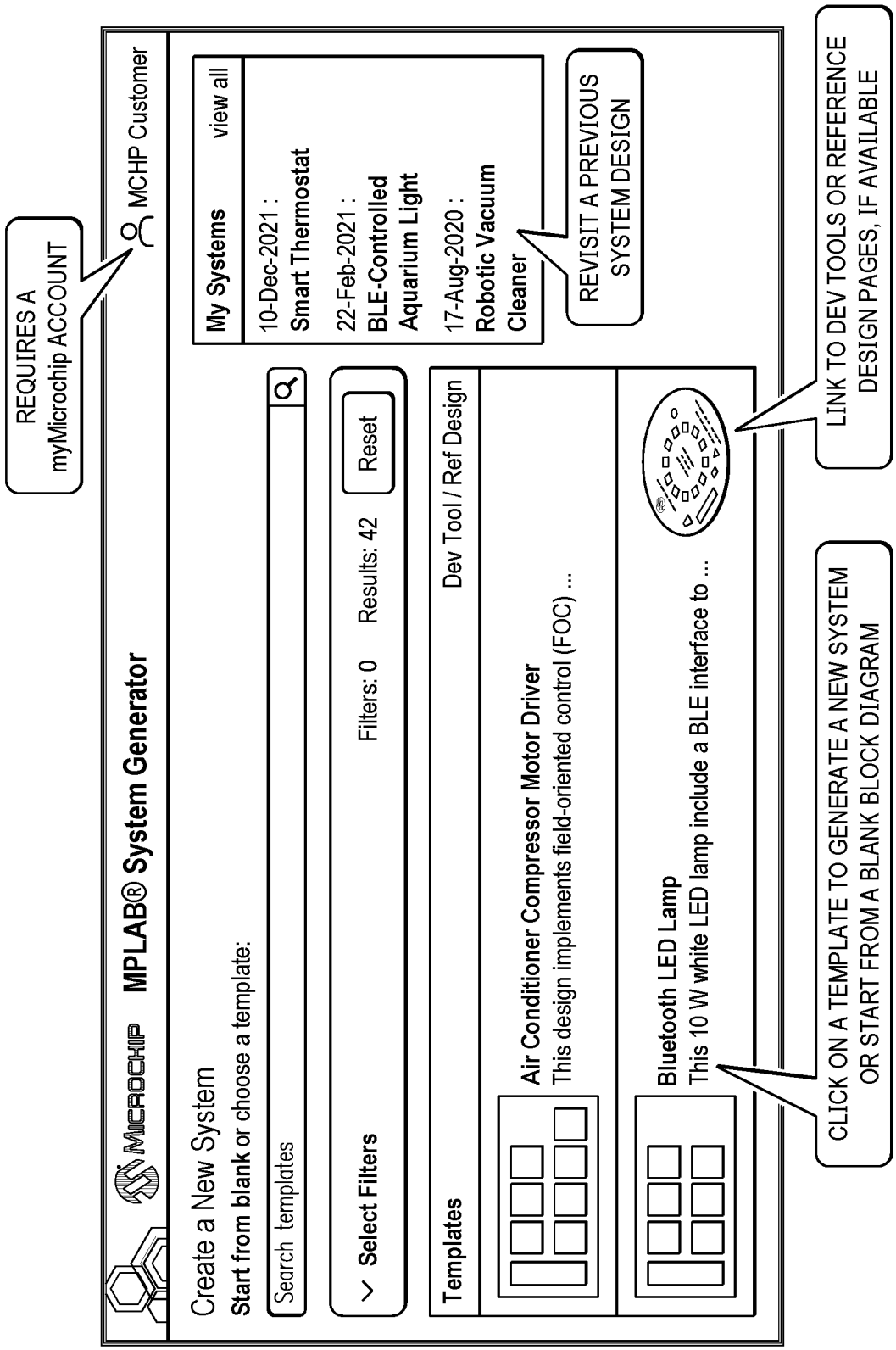

REQUIRES A myMicrochip ACCOUNT
MCHP Customer
MICROCHIP MPLAB® System Generator
Create a New System
Start from blank or choose a template:
Search templates
Select Filters
Filters: 0
Results: 42
Reset
Templates
Dev Tool / Ref Design
Air Conditioner Compressor Motor Driver
This design implements field-oriented control (FOC) ...
Bluetooth LED Lamp
This 10 W white LED lamp include a BLE interface to ...
My Systems
view all
10-Dec-2021 : Smart Thermostat
22-Feb-2021 : BLE-Controlled Aquarium Light
17-Aug-2020 : Robotic Vacuum Cleaner
REVISIT A PREVIOUS SYSTEM DESIGN
LINK TO DEV TOOLS OR REFERENCE DESIGN PAGES, IF AVAILABLE
CLICK ON A TEMPLATE TO GENERATE A NEW SYSTEM OR START FROM A BLANK BLOCK DIAGRAM

FIG. 2

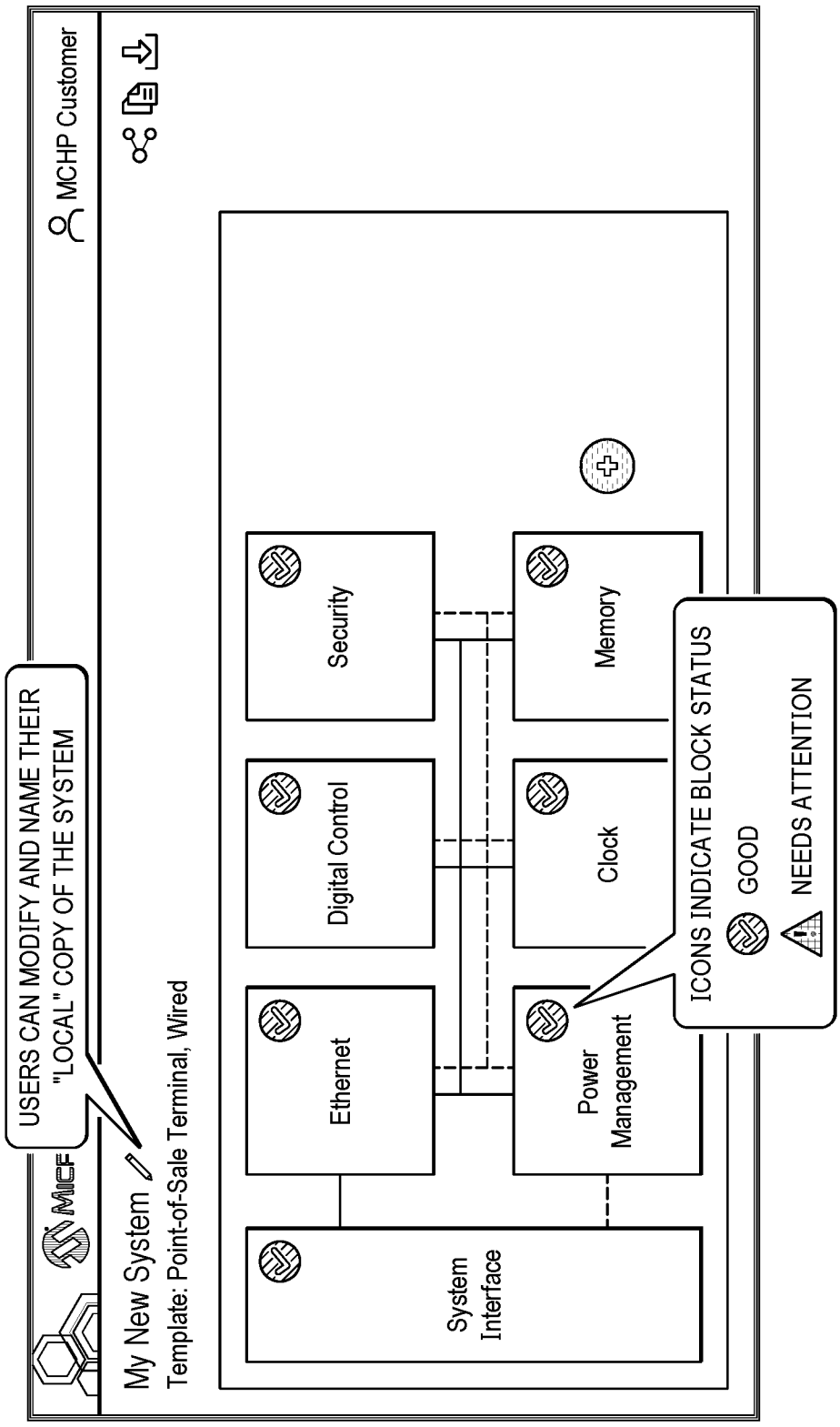

MCHP Customer
MICF
USERS CAN MODIFY AND NAME THEIR "LOCAL" COPY OF THE SYSTEM
My New System
Template: Point-of-Sale Terminal, Wired
Security
Digital Control
Ethernet
Power Management
Memory
Clock
System Interface
ICONS INDICATE BLOCK STATUS
GOOD
NEEDS ATTENTION

FIG. 3

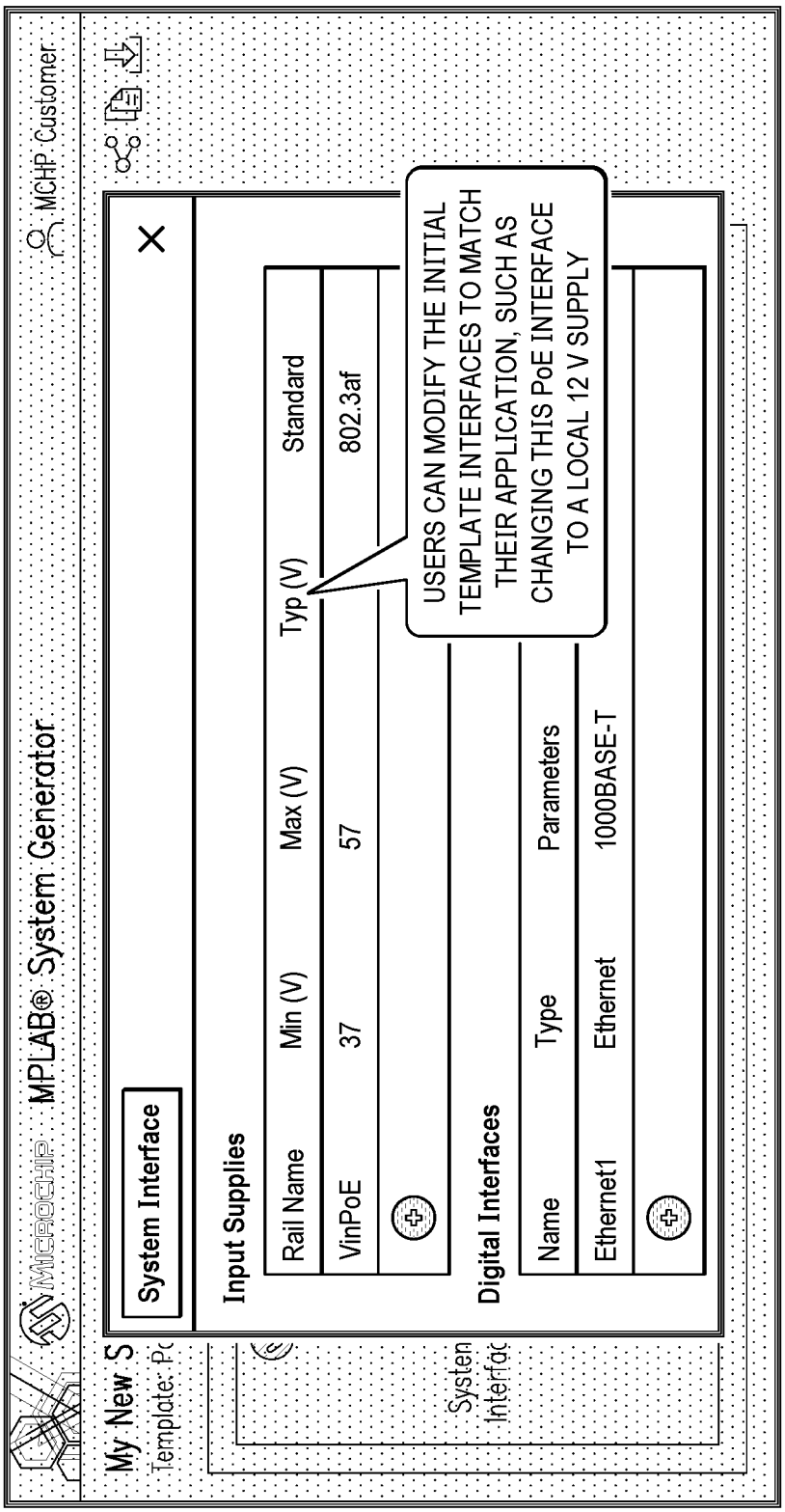

MICROCHIP
MPLAB® System Generator
MCHP Customer
My New S
Template: Pc
System Interface
Input Supplies
Rail Name
Min (V)
Max (V)
Typ (V)
Standard
VinPoE
37
57
802.3af
Digital Interfaces
Name
Type
Parameters
Ethernet1
Ethernet
1000BASE-T
USERS CAN MODIFY THE INITIAL TEMPLATE INTERFACES TO MATCH THEIR APPLICATION, SUCH AS CHANGING THIS PoE INTERFACE TO A LOCAL 12 V SUPPLY

FIG. 4

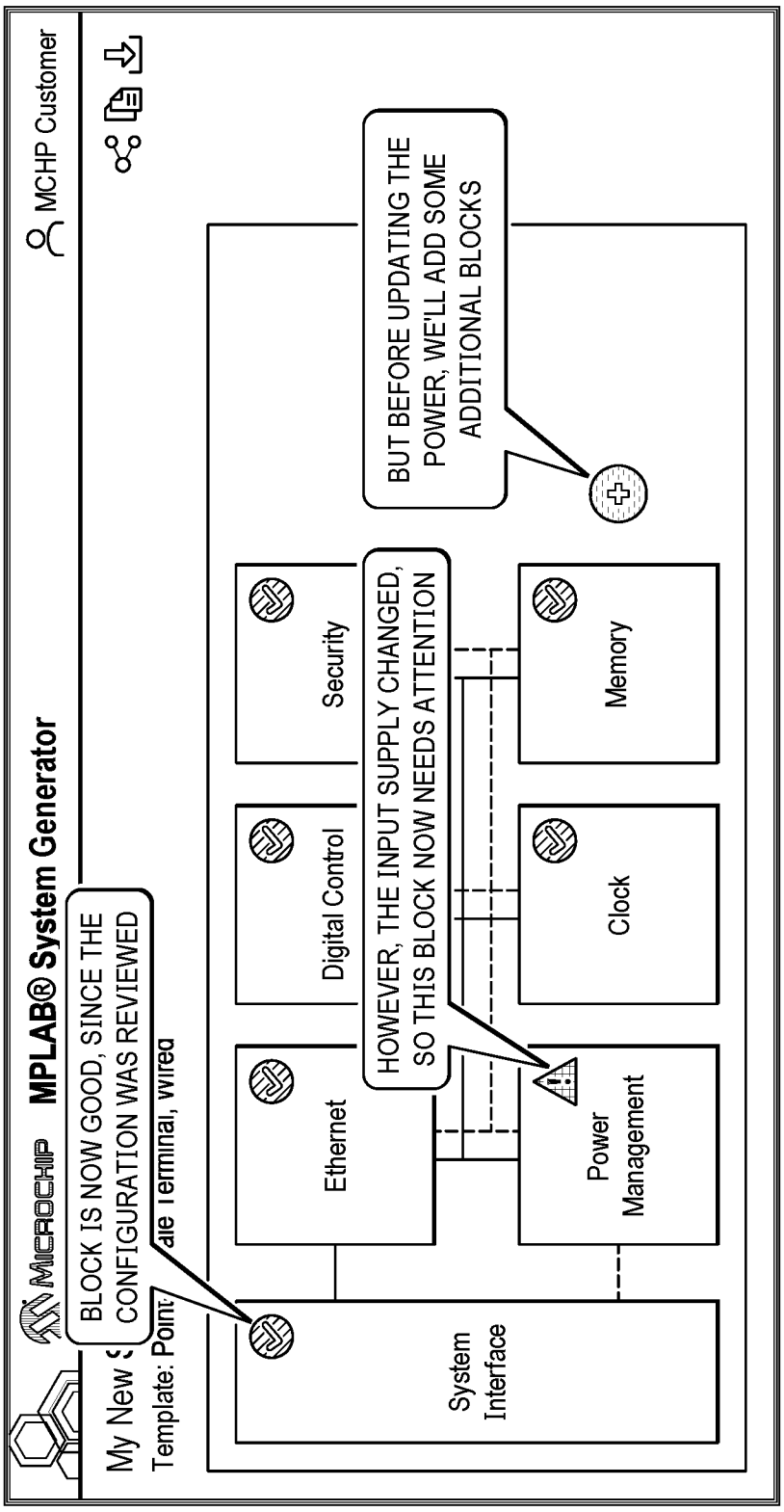

MICROCHIP
MPLAB® System Generator
MCHP Customer
My New
Template: Point
BLOCK IS NOW GOOD, SINCE THE CONFIGURATION WAS REVIEWED
Ethernet
Digital Control
Security
HOWEVER, THE INPUT SUPPLY CHANGED, SO THIS BLOCK NOW NEEDS ATTENTION
BUT BEFORE UPDATING THE POWER, WE'LL ADD SOME ADDITIONAL BLOCKS
System Interface
Power Management
Clock
Memory

FIG. 5

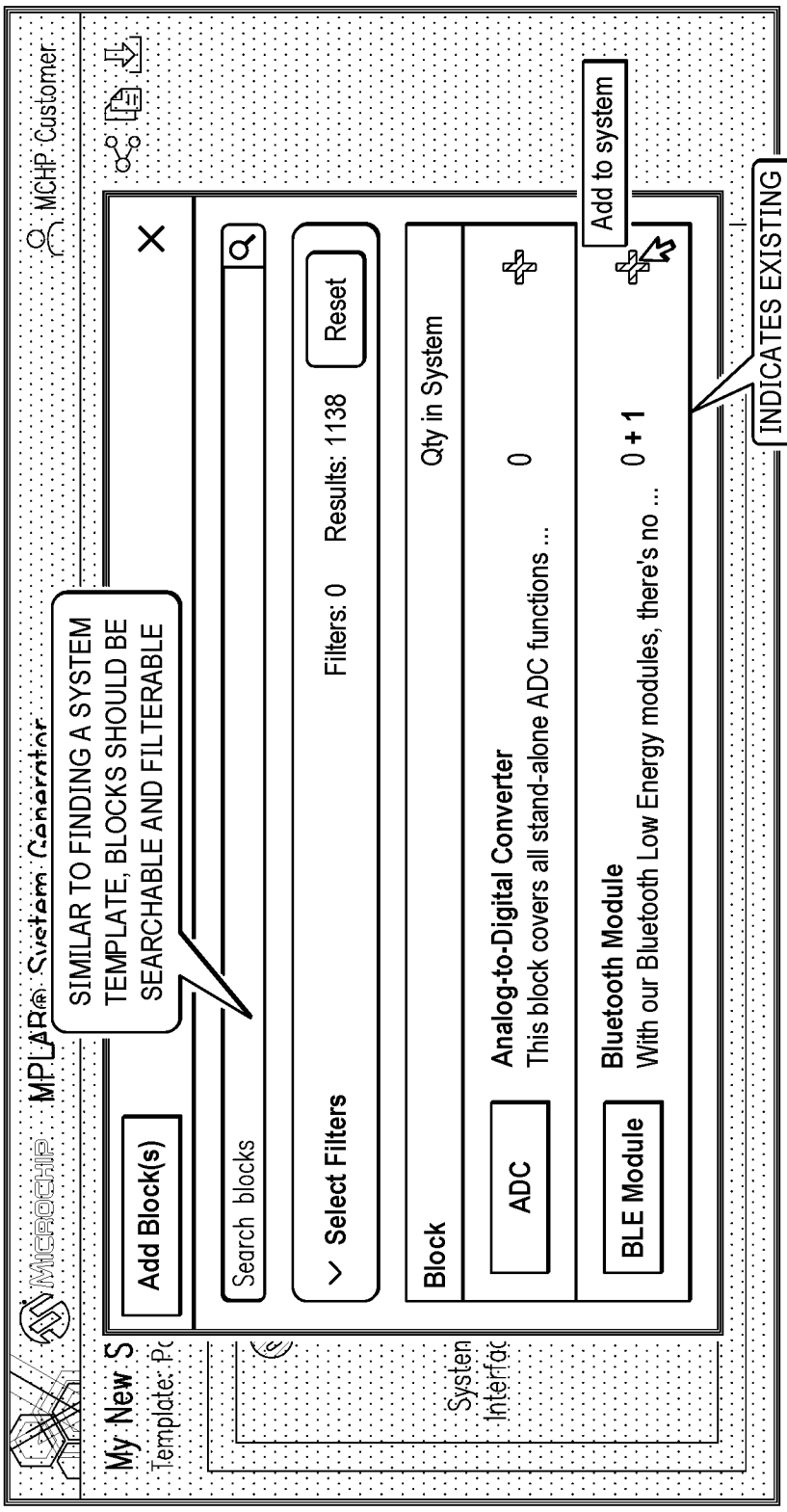

MICROCHIP
MPLAB® System Generator
MCHP Customer
My New S
Template: Pr
Add Block(s)
SIMILAR TO FINDING A SYSTEM TEMPLATE, BLOCKS SHOULD BE SEARCHABLE AND FILTERABLE
Search blocks
Select Filters
Filters: 0
Results: 1138
Reset
Block
Qty in System
ADC
Analog-to-Digital Converter
This block covers all stand-alone ADC functions ...
0
BLE Module
Bluetooth Module
With our Bluetooth Low Energy modules, there's no ...
0 + 1
Add to system
INDICATES EXISTING + NEW BLOCKS

FIG. 6

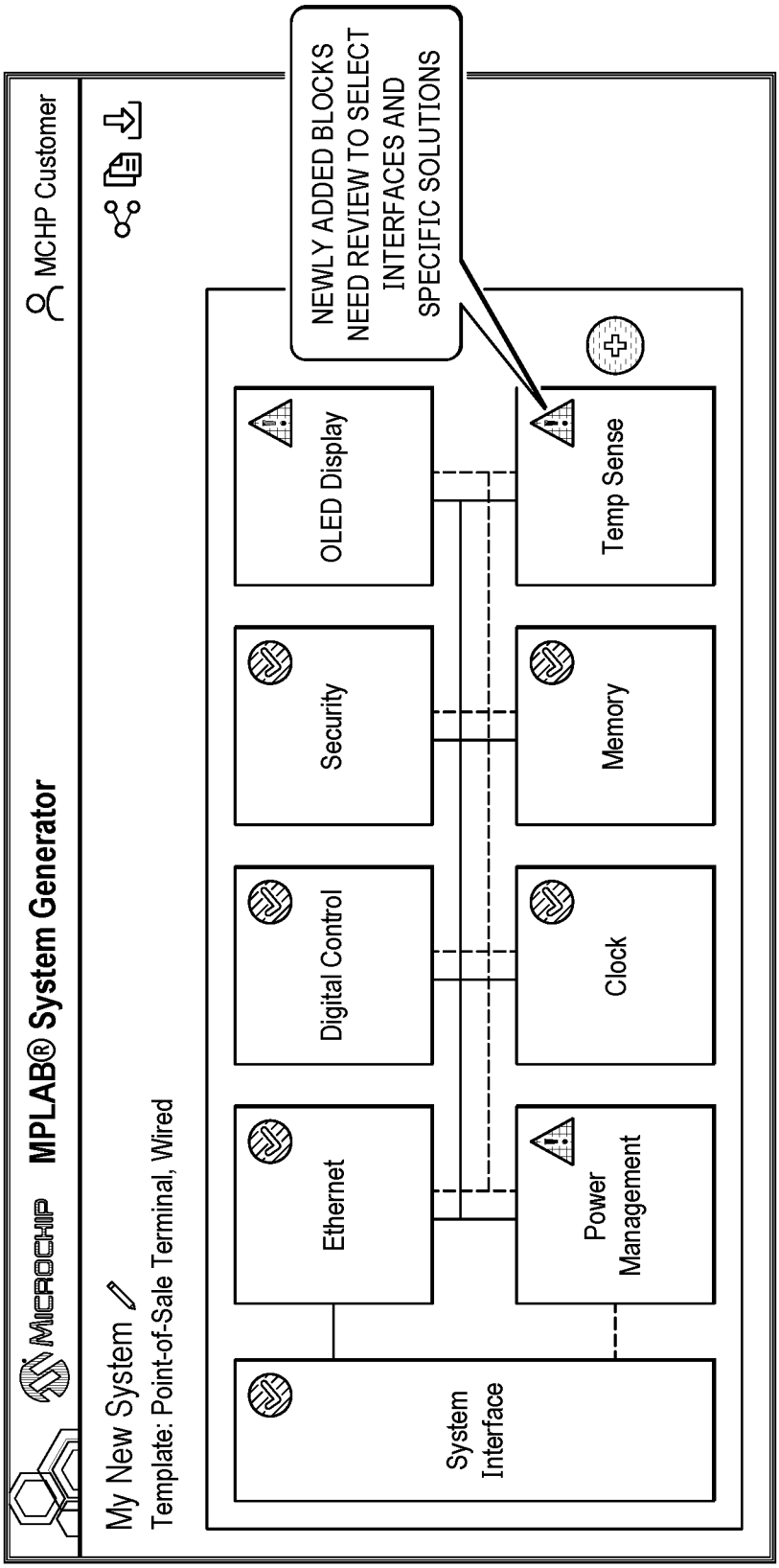

MICROCHIP
MPLAB® System Generator
MCHP Customer
My New System
Template: Point-of-Sale Terminal, Wired
NEWLY ADDED BLOCKS NEED REVIEW TO SELECT INTERFACES AND SPECIFIC SOLUTIONS
System Interface
Ethernet
Digital Control
Security
OLED Display
Power Management
Clock
Memory
Temp Sense

FIG. 7

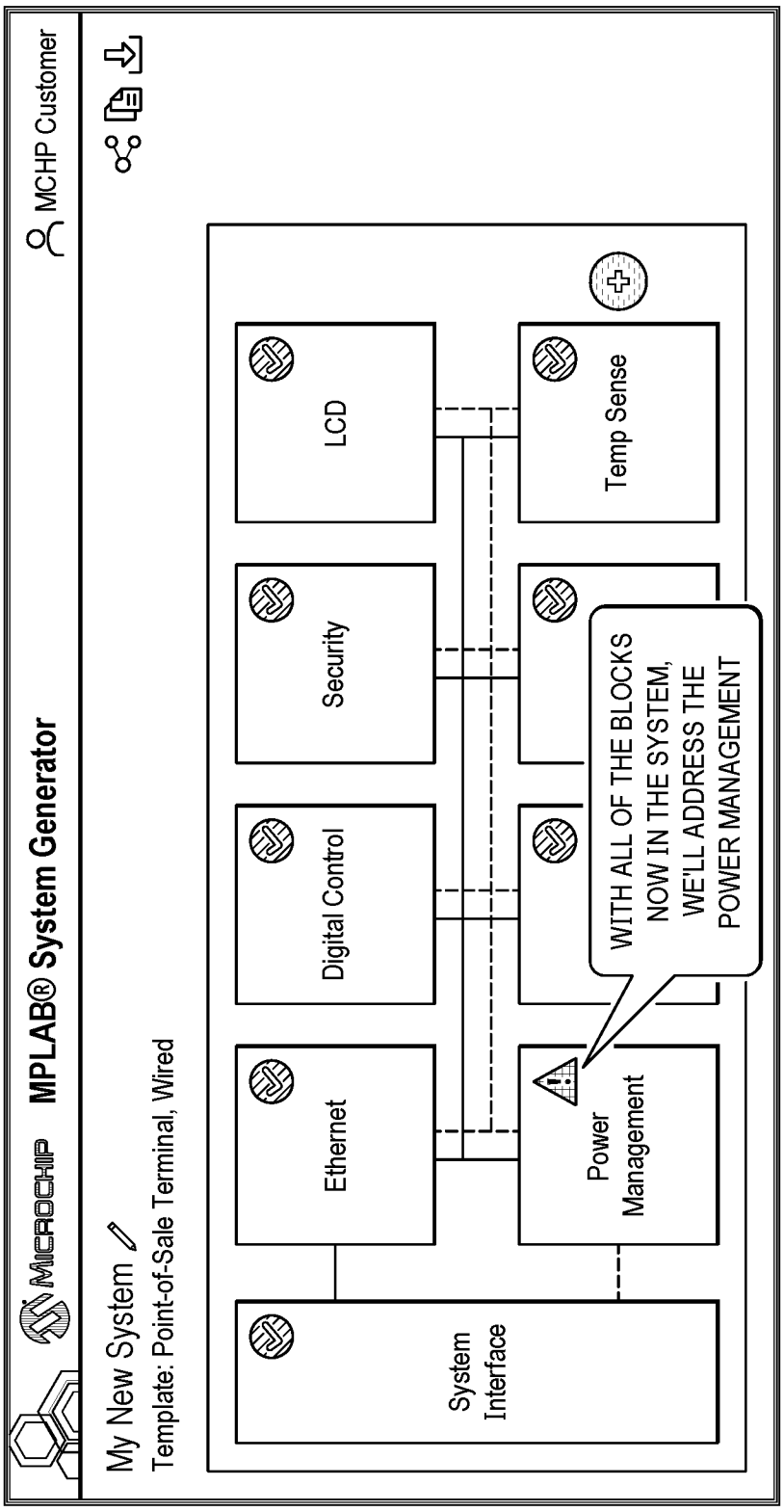

MICROCHIP
MPLAB® System Generator
MCHP Customer
My New System
Template: Point-of-Sale Terminal, Wired
System Interface
Ethernet
Digital Control
Security
LCD
Power Management
Temp Sense
WITH ALL OF THE BLOCKS NOW IN THE SYSTEM, WE'LL ADDRESS THE POWER MANAGEMENT

FIG. 8

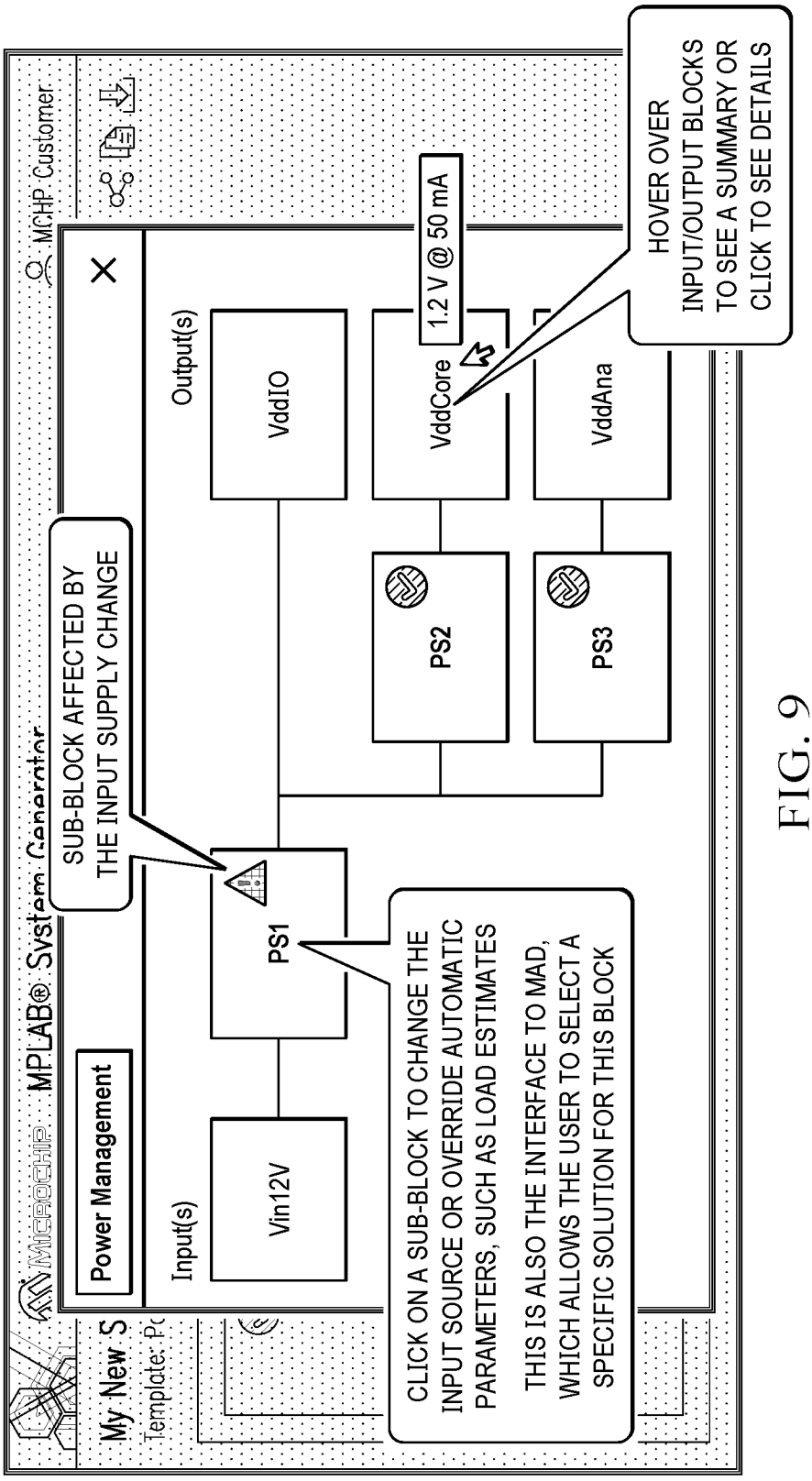

MICROCHIP MPLAB® System Generator
MCHP Customer
My New S
Template: Pc
Power Management
Input(s)
Vin12V
PS1
Output(s)
VddIO
VddCore
1.2 V @ 50 mA
VddAna
PS2
PS3
SUB-BLOCK AFFECTED BY THE INPUT SUPPLY CHANGE
HOVER OVER INPUT/OUTPUT BLOCKS TO SEE A SUMMARY OR CLICK TO SEE DETAILS
CLICK ON A SUB-BLOCK TO CHANGE THE INPUT SOURCE OR OVERRIDE AUTOMATIC PARAMETERS, SUCH AS LOAD ESTIMATES
THIS IS ALSO THE INTERFACE TO MAD, WHICH ALLOWS THE USER TO SELECT A SPECIFIC SOLUTION FOR THIS BLOCK

FIG. 9

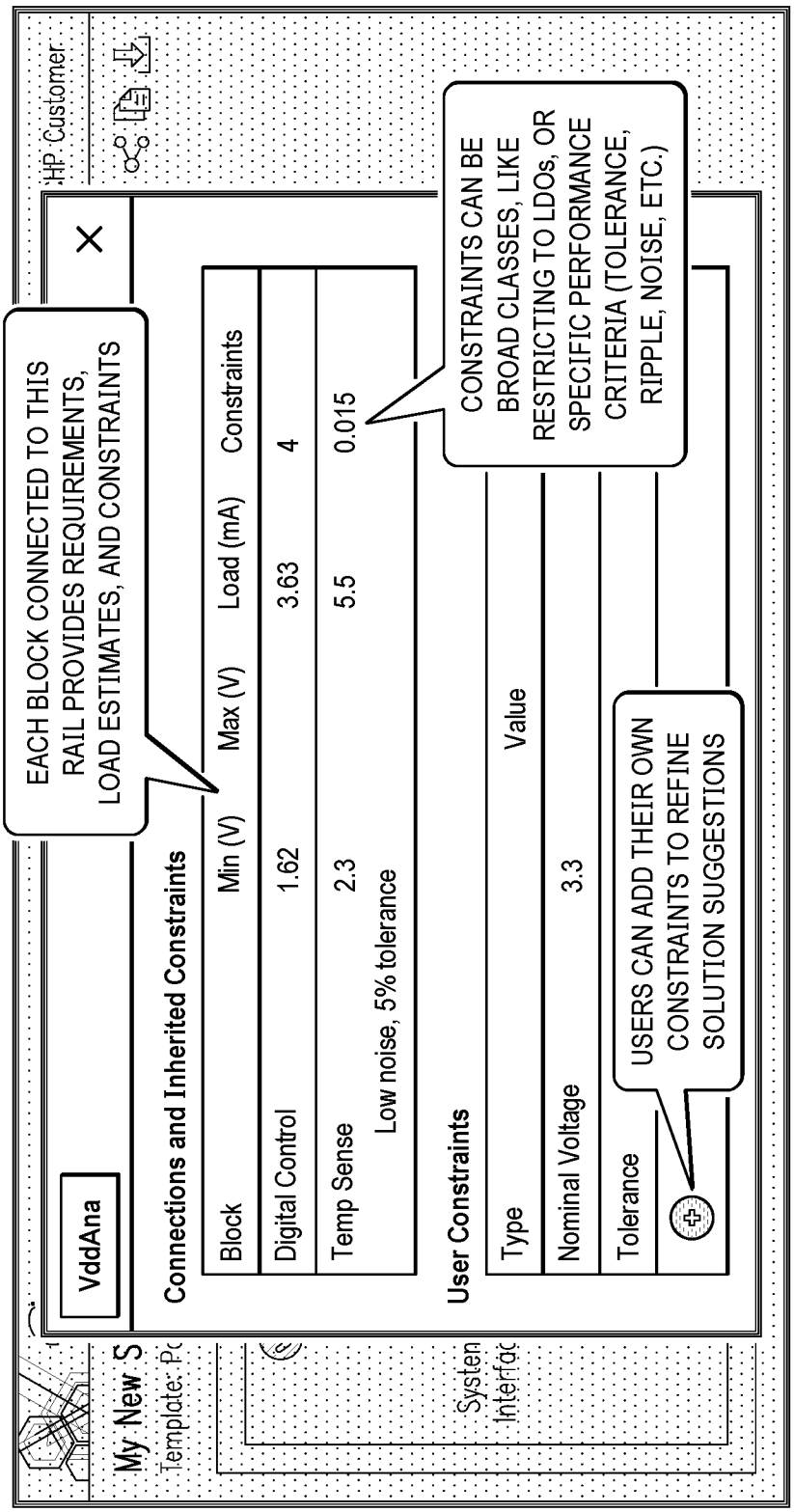

HP Customer
VddAna
EACH BLOCK CONNECTED TO THIS RAIL PROVIDES REQUIREMENTS, LOAD ESTIMATES, AND CONSTRAINTS
Connections and Inherited Constraints
Block
Min (V)
Max (V)
Load (mA)
Constraints
Digital Control
1.62
3.63
4
Temp Sense
2.3
5.5
0.015
Low noise, 5% tolerance
CONSTRAINTS CAN BE BROAD CLASSES, LIKE RESTRICTING TO LDOs, OR SPECIFIC PERFORMANCE CRITERIA (TOLERANCE, RIPPLE, NOISE, ETC.)
User Constraints
Type
Value
Nominal Voltage
3.3
Tolerance
USERS CAN ADD THEIR OWN CONSTRAINTS TO REFINE SOLUTION SUGGESTIONS

FIG. 10

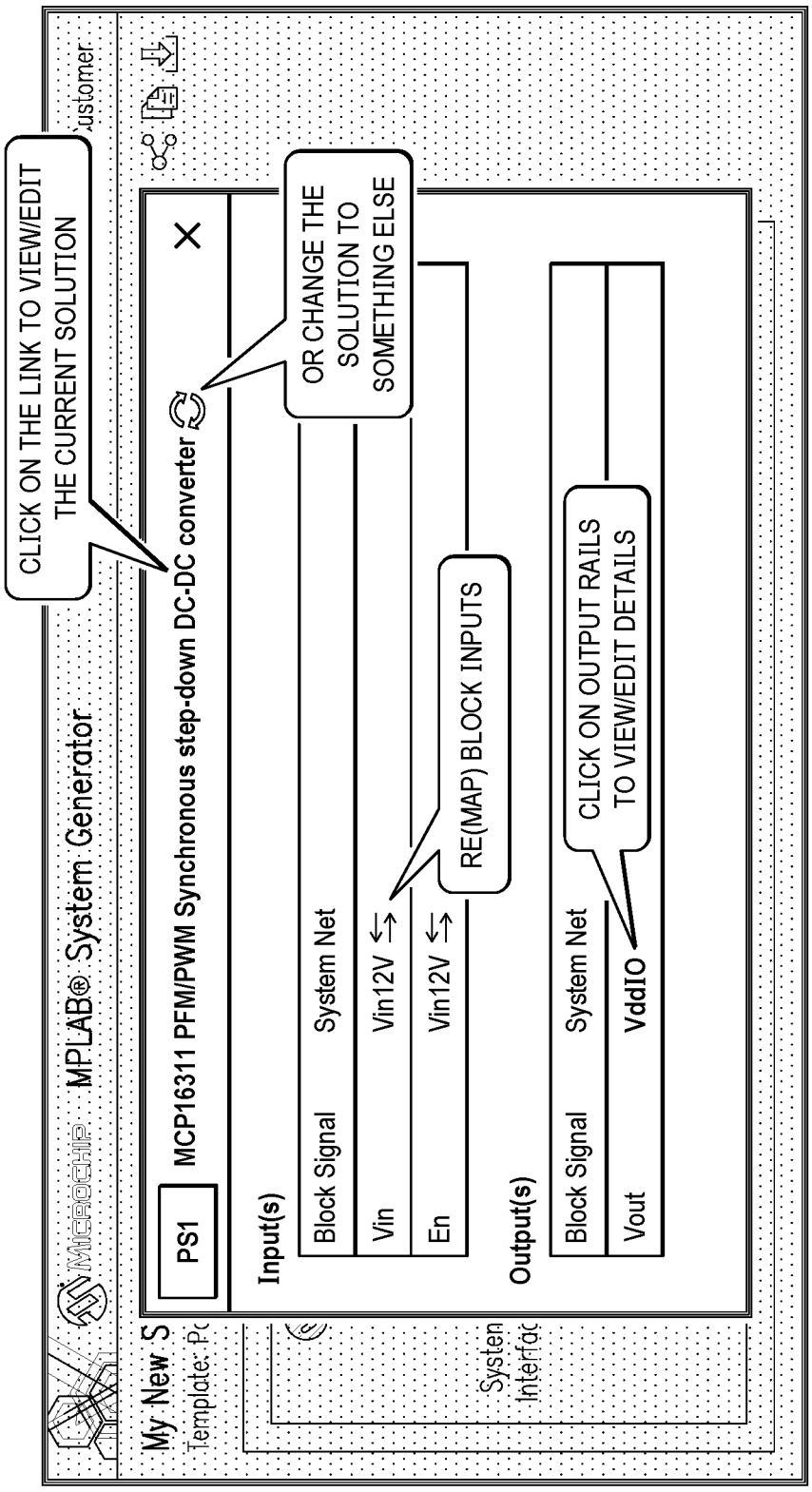

MICROCHIP
MPLAB® System Generator
My New S
Template: Pc
PS1
MCP16311 PFM/PWM Synchronous step-down DC-DC converter
CLICK ON THE LINK TO VIEW/EDIT THE CURRENT SOLUTION
OR CHANGE THE SOLUTION TO SOMETHING ELSE
RE(MAP) BLOCK INPUTS
CLICK ON OUTPUT RAILS TO VIEW/EDIT DETAILS
Input(s)
Block Signal
System Net
Vin
Vin12V
En
Vin12V
Output(s)
Block Signal
System Net
Vout
VddIO
System
Interfac
ustomer

FIG. 11

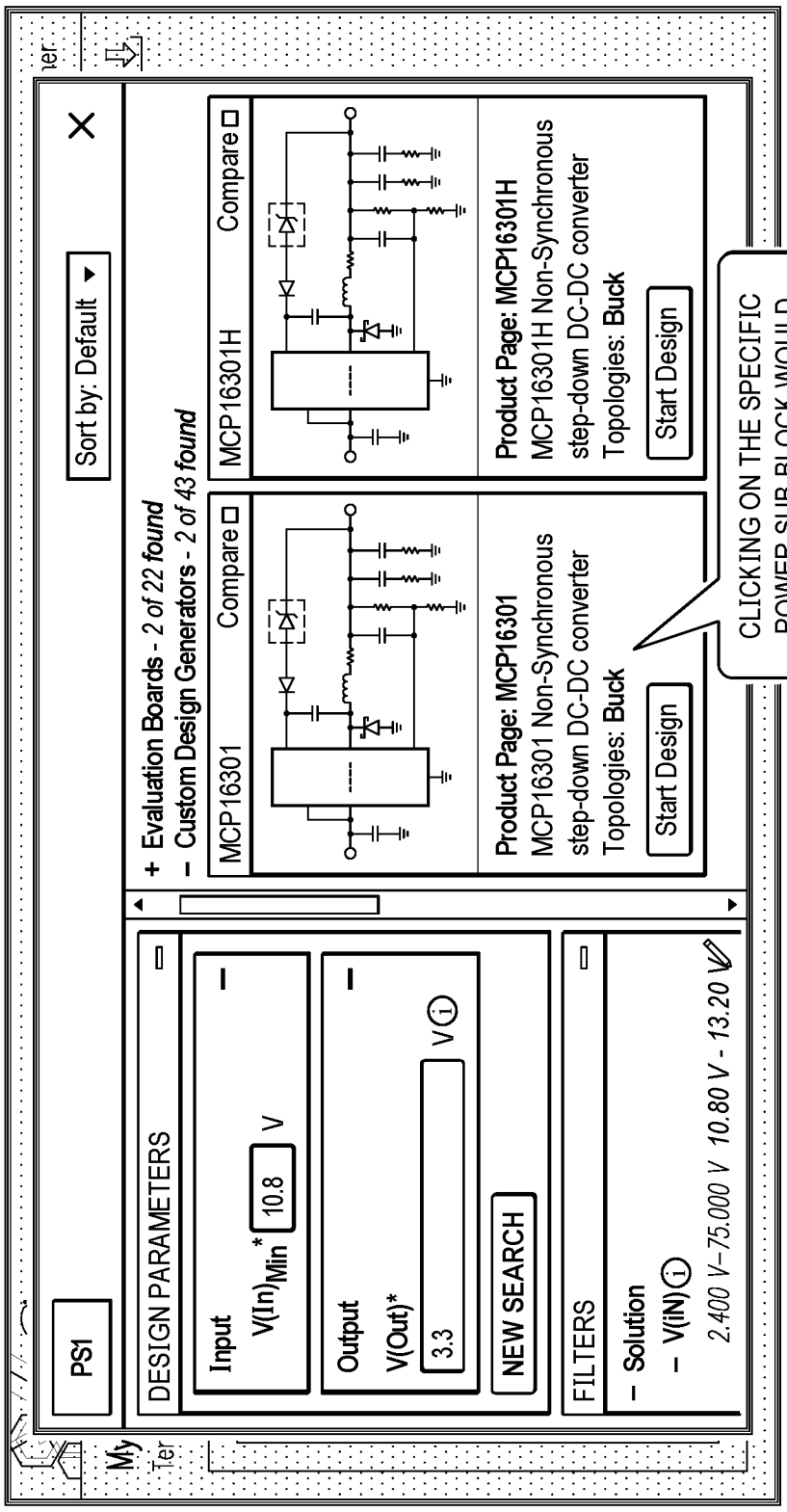

PS1
Sort by: Default
+ Evaluation Boards - 2 of 22 found
– Custom Design Generators - 2 of 43 found
MCP16301
Compare
Product Page: MCP16301
MCP16301 Non-Synchronous step-down DC-DC converter
Topologies: Buck
Start Design
MCP16301H
Compare
Product Page: MCP16301H
MCP16301H Non-Synchronous step-down DC-DC converter
Topologies: Buck
Start Design
DESIGN PARAMETERS
Input
V(In)Min* 10.8 V
Output
V(Out)* 3.3 V
NEW SEARCH
FILTERS
– Solution
– V(IN)
2.400 V–75.000 V 10.80 V - 13.20 V
CLICKING ON THE SPECIFIC POWER SUB-BLOCK WOULD MAKE CALLS TO THE SOLUTION DATABASE CURRENTLY IN MAD

FIG. 12

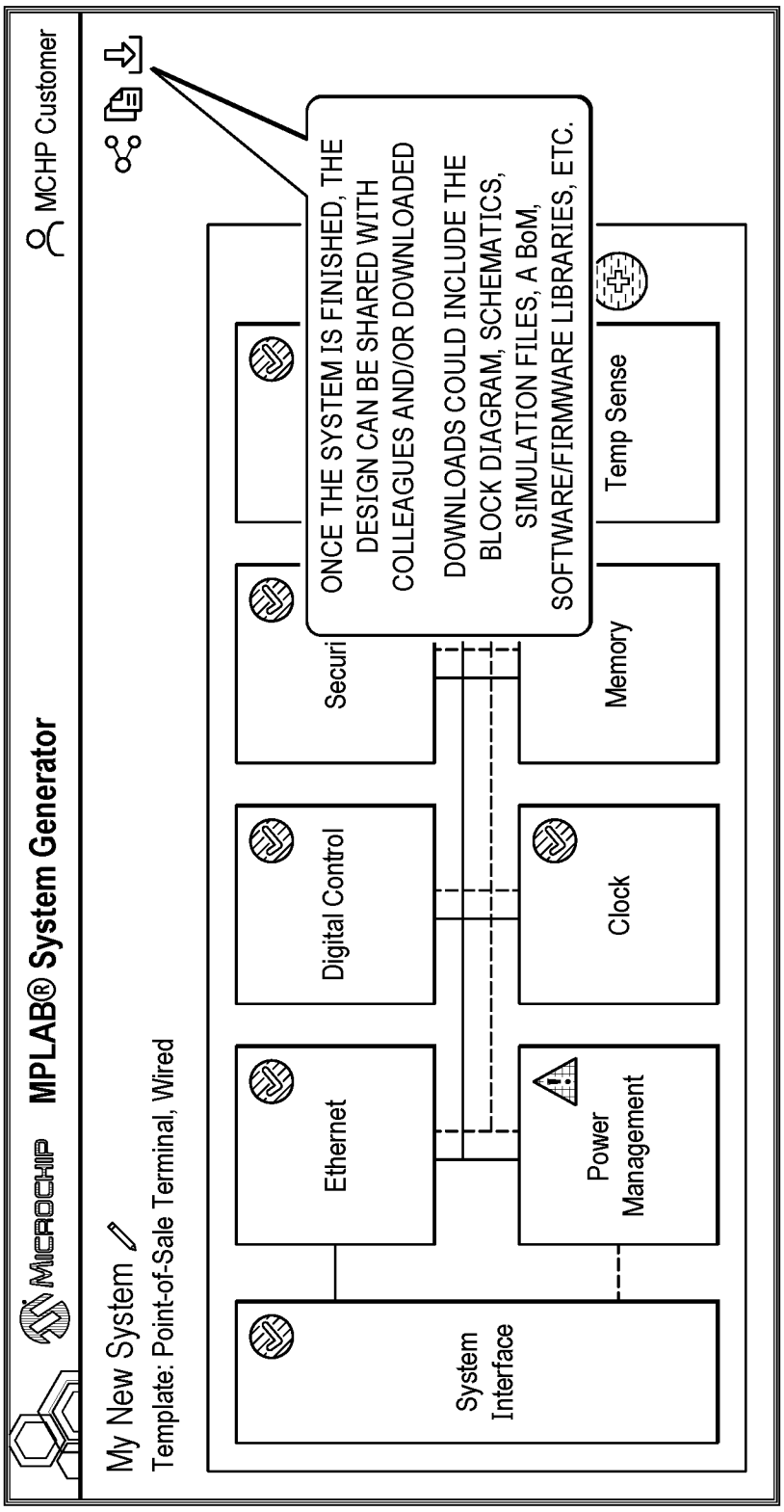

MICROCHIP
MPLAB® System Generator
MCHP Customer
My New System
Template: Point-of-Sale Terminal, Wired
System Interface
Ethernet
Digital Control
Power Management
Clock
Memory
Temp Sense
Securi
ONCE THE SYSTEM IS FINISHED, THE DESIGN CAN BE SHARED WITH COLLEAGUES AND/OR DOWNLOADED
DOWNLOADS COULD INCLUDE THE BLOCK DIAGRAM, SCHEMATICS, SIMULATION FILES, A BoM, SOFTWARE/FIRMWARE LIBRARIES, ETC.

FIG. 13

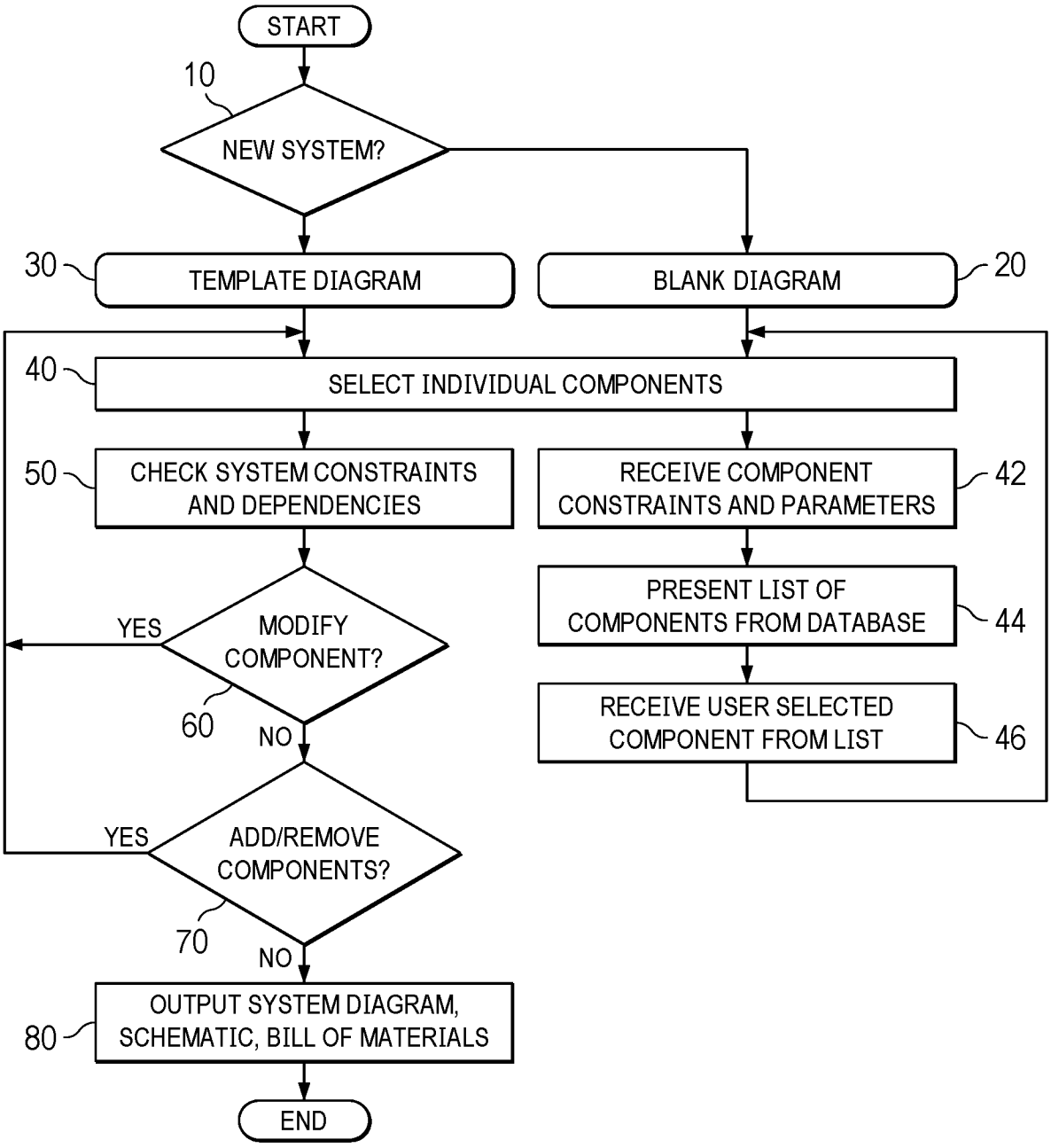

START
10
NEW SYSTEM?
30
TEMPLATE DIAGRAM
BLANK DIAGRAM
20
40
SELECT INDIVIDUAL COMPONENTS
50
CHECK SYSTEM CONSTRAINTS AND DEPENDENCIES
RECEIVE COMPONENT CONSTRAINTS AND PARAMETERS
42
PRESENT LIST OF COMPONENTS FROM DATABASE
44
RECEIVE USER SELECTED COMPONENT FROM LIST
46
YES
MODIFY COMPONENT?
60
NO
YES
ADD/REMOVE COMPONENTS?
70
NO
80
OUTPUT SYSTEM DIAGRAM, SCHEMATIC, BILL OF MATERIALS
END

FIG. 14

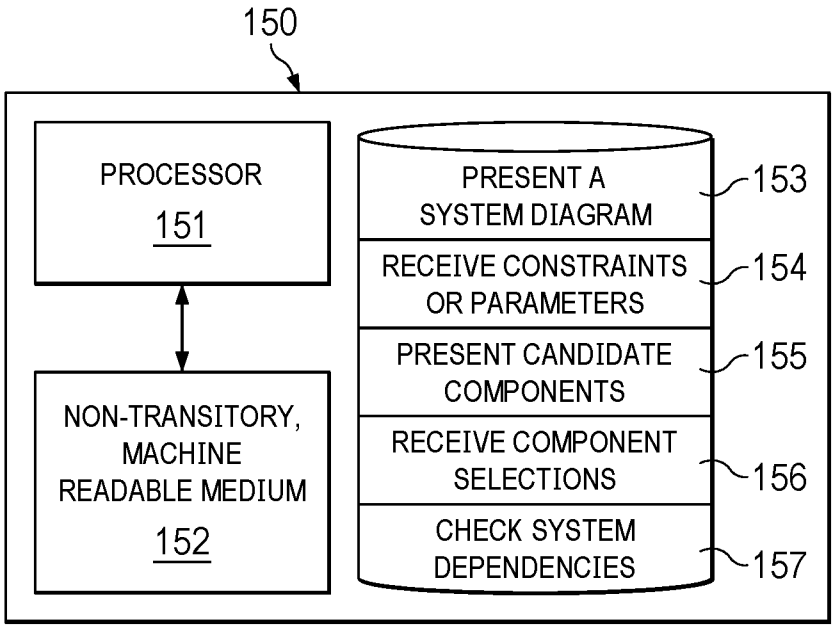

150
PROCESSOR
151
NON-TRANSITORY, MACHINE READABLE MEDIUM
152
PRESENT A SYSTEM DIAGRAM
153
RECEIVE CONSTRAINTS OR PARAMETERS
154
PRESENT CANDIDATE COMPONENTS
155
RECEIVE COMPONENT SELECTIONS
156
CHECK SYSTEM DEPENDENCIES
157

FIG. 15

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2023/016197

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F30/30
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2021/101669 A1 (EFABLESS CORP [US]) 27 May 2021 (2021-05-27) paragraphs [0003], [0005], [0010], [0011], [0052], [0071], [0076]; claims 1-32; figures 1-11E paragraphs [0077], [0108] - [0137], [0147], [0164] - [0168], [0169], [0193] paragraphs [0198], [0232], [0238], [0239], [0258], [0259], [0263] paragraphs [0283], [0285] - [0288], [0289], [0297] - [0300]	1-20
X	US 8 769 477 B1 (GATHOO YOGESH [IN] ET AL) 1 July 2014 (2014-07-01) paragraphs [0077], [0108] - [0137], [0147], [0164] - [0168], [0169], [0193]; claims 1-20; figures 1-11	1-20

-/--

[x] Further documents are listed in the continuation of Box C.

[x] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance;; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance;; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
19 June 2023	27/06/2023

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Schneider, Michael

1

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2003/009730 A1 (CHEN MICHAEL Y [US] ET AL) 9 January 2003 (2003-01-09) claims 1-45; figures 1-20 paragraph [0013] paragraph [0088] - paragraph [0091]	1-20
X	US 2009/172621 A1 (SATHE SANDEEP JAYANT [IN] ET AL) 2 July 2009 (2009-07-02) claims 1-21; figures 1-12 paragraph [0007] - paragraph [0014] paragraph [0018] - paragraph [0020] paragraph [0023] - paragraph [0024] paragraph [0044] - paragraph [0046] paragraph [0079] - paragraph [0087]	1-20
X	US 2013/326457 A1 (MACMUNN NEIL C [CA] ET AL) 5 December 2013 (2013-12-05) claims 1-14; figures 1-5L	1-20
A	US 2020/042664 A1 (LEE YUNSUP [US] ET AL) 6 February 2020 (2020-02-06) claims 1-36; figures 1-13 paragraph [0039] - paragraph [0044] paragraph [0081] - paragraph [0083]	1-20
A	US 2003/229482 A1 (COOK STEPHEN ANTHONY [GB] ET AL) 11 December 2003 (2003-12-11) figures 1-37	1-20
A	US 2005/193353 A1 (MALEKKHOSRAVI BEHNAM [US] ET AL) 1 September 2005 (2005-09-01) paragraph [0006] - paragraph [0016]	1-20
A	US 2008/222589 A1 (BANERJEE SOUMYA [US]) 11 September 2008 (2008-09-11) paragraph [0002] - paragraph [0004] paragraph [0033] - paragraph [0036] paragraph [0041] paragraph [0047] - paragraph [0049]; claims 1-20; figures 1-13 paragraph [0092] paragraph [0094] - paragraph [0097]	1-20
A	US 8 065 653 B1 (BEST ANDREW [US] ET AL) 22 November 2011 (2011-11-22) abstract; figures 1-4	1-20
A	US 2021/081588 A1 (OSBURN EDWARD PAUL [US] ET AL) 18 March 2021 (2021-03-18) paragraph [0041] paragraph [0074]	1-20
8 A	US 2012/017198 A1 (NG PIUS [US] ET AL) 19 January 2012 (2012-01-19) figures 1-6	1-20
1	-/--	

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2023/016197

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Microchip Technology Lnc: "Microchip MPLAB X IDE User's Guide", , 14 September 2015 (2015-09-14), XP055837372, Retrieved from the Internet: URL:http://ww1.microchip.com/downloads/en/devicedoc/50002027d.pdf [retrieved on 2021-09-02] pages 13-30	1-20

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2023/016197

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2021101669 A1	27-05-2021	GB 2603402 A	03-08-2022
		WO 2021101669 A1	27-05-2021
US 8769477 B1	01-07-2014	NONE	
US 2003009730 A1	09-01-2003	NONE	
US 2009172621 A1	02-07-2009	NONE	
US 2013326457 A1	05-12-2013	AU 2013267810 A1	18-12-2014
		AU 2016204456 A1	21-07-2016
		CA 2874095 A1	05-12-2013
		EP 2856336 A2	08-04-2015
		IL 235786 A	30-11-2015
		US 2013326457 A1	05-12-2013
		US 2015242557 A1	27-08-2015
		WO 2013180990 A2	05-12-2013
US 2020042664 A1	06-02-2020	CN 112534436 A	19-03-2021
		EP 3830666 A1	09-06-2021
		KR 20210030475 A	17-03-2021
		US 2020042664 A1	06-02-2020
		US 2021365609 A1	25-11-2021
		WO 2020028628 A1	06-02-2020
US 2003229482 A1	11-12-2003	AU 2003223746 A1	10-11-2003
		CN 1666202 A	07-09-2005
		CN 102902839 A	30-01-2013
		EP 1502214 A1	02-02-2005
		KR 20050010775 A	28-01-2005
		US 2003229482 A1	11-12-2003
		WO 03091914 A1	06-11-2003
US 2005193353 A1	01-09-2005	CN 1934571 A	21-03-2007
		JP 5551127 B2	16-07-2014
		JP 2007525845 A	06-09-2007
		JP 2011249846 A	08-12-2011
		TW I395110 B	01-05-2013
		TW 201322031 A	01-06-2013
		US 2005193353 A1	01-09-2005
		WO 2005086035 A1	15-09-2005
US 2008222589 A1	11-09-2008	NONE	
US 8065653 B1	22-11-2011	US 8065653 B1	22-11-2011
		US 8499270 B1	30-07-2013
US 2021081588 A1	18-03-2021	US 2021081588 A1	18-03-2021
		US 2022207222 A1	30-06-2022
		WO 2021055632 A1	25-03-2021
US 2012017198 A1	19-01-2012	CN 103097986 A	08-05-2013
		EP 2593889 A1	22-05-2013
		JP 5725582 B2	27-05-2015
		JP 2013538386 A	10-10-2013
		KR 20130044321 A	02-05-2013
		TW 201222306 A	01-06-2012
		US 2012017198 A1	19-01-2012

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2023/016197

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		WO 2012009293 A1	19-01-2012