

(19) **DANMARK**



Patent- og
Varemærkestyrelsen

(12)

Oversættelse af europæisk patentskrift

(10) **DK/EP 3588193 T3**

-
- (51) Int.Cl.: **G 03 G 15/08 (2006.01)** **G 03 G 21/18 (2006.01)**
- (45) Oversættelsen bekendtgjort den: **2022-10-10**
- (80) Dato for Den Europæiske Patentmyndigheds bekendtgørelse om meddelelse af patentet: **2022-07-20**
- (86) Europæisk ansøgning nr.: **19168438.0**
- (86) Europæisk indleveringsdag: **2019-04-10**
- (87) Den europæiske ansøgnings publiceringsdag: **2020-01-01**
- (30) Prioritet: **2018-04-12 US 201815951586**
- (84) Designerede stater: **AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR**
- (73) Patenthaver: **Lexmark International, Inc., 740 West New Circle Road, Lexington, KY 40550, USA**
- (72) Opfinder: **Foley, Nathan Wayne, 840 Glen Abbey Circle, Lexington, KY 40509, USA**
Williams, Jennifer Topmiller, 3108 Roundway Down Lane, Lexington, KY 40509, USA
Woods, Gregory Scott, 4804 Matthew Court, Lexington, KY 40514, USA
Moore, Jr., Jimmy Daniel, 123 Neel Lane, Georgetown, KY 40324, USA
- (74) Fuldmægtig i Danmark: **Valet Patent Services LLP, Siedlungsstr. 4a, 85253 Erdweg, Tyskland**
- (54) Benævnelse: **COMMUNICATION BETWEEN AN IMAGE FORMING DEVICE AND A REPLACEABLE SUPPLY ITEM**
- (56) Fremdragne publikationer:
WO-A1-01/06444
US-A1- 2009 285 594
Jan Axelson.: "USB COMPLETE: THE DEVELOPER' S GUIDE" In: "USB COMPLETE: THE DEVELOPER' S GUIDE", 1 June 2009 (2009-06-01), Lakeview Research LLC, Madison, WI, US, XP055355656, ISBN: 978-1-931448-08-6 pages 1-529, * Section "The Data Toggle"; pages 51, 52 *
AGHDAIE N ET AL: "CLIENT-TRANSPARENT FAULT-TOLERANT WEB SERVICE", CONFERENCE PROCEEDINGS OF THE 2001 IEEE INTERNATIONAL PERFORMANCE, COMPUTING, AND COMMUNICATIONS CONFERENCE. (IPCCC). PHOENIX, AZ, APRIL 4 - 6, 2001; [IEEE INTERNATIONAL PERFORMANCE, COMPUTING AND COMMUNICATIONS CONFERENCE], NEW YORK, NY : IEEE, US, 4 April 2001 (2001-04-04), pages 209-216, XP001049954, DOI: 10.1109/IPCCC.2001.918654 ISBN: 978-0-7803-7001-2

DESCRIPTION

BACKGROUND

1. Field of the Disclosure

[0001] The present disclosure relates generally to image forming devices and more particularly to communication between an image forming device and a replaceable supply item.

2. Description of the Related Art

[0002] An electrophotographic image forming device typically includes one or more customer replaceable supply items that have a shorter lifespan than the image forming device. For example, the image forming device may include replaceable supply items that replenish the image forming device's toner supply and/or that replace worn imaging components, such as the photoconductive drum, fuser, etc. It is desired for the image forming device and the supply items to communicate information with each other for proper operation. For example, it may be desired to communicate such information as authentication or validation information, consumption or usage rates, replacement schedules, etc. Each supply item typically includes a chip configured to communicate with and respond to commands from a controller in the image forming device. Efficient communication between the controller of the image forming device and the chips of the supply items is desired.

[0003] The disclosures of US 2009/285594 A1 and WO 01/06444 A1 may be helpful for understanding the present invention. The same applies to Jan Axelson., "USB COMPLETE: THE DEVELOPER' S GUIDE", ISBN 978-1-931448-08-6 and Aghdaei N. et al., "CLIENT-TRANSPARENT FAULT-TOLERANT WEB SERVICE", Conference Proceedings of the 2001 IEEE International Performance, Computing and Communications, Conference, ISBN 978-0-7803-7001-2.

SUMMARY

[0004] The present invention refers to a method of electrical communication between a slave node of an I2C communication bus positioned on a replaceable supply item installed in an image forming device according to claim 1. Advantageous embodiments may include features of depending claims. The method includes receiving a first write command by the slave node on the replaceable supply item from the controller of the image forming device. The slave node on the replaceable supply item compares a value of a transmission cycle bit contained in a

header of the first write command with a value of a transmission cycle bit contained in a header of a second write command received by the slave node on the replaceable supply item from the controller of the image forming device previous to the first write command. If the value of the transmission cycle bit contained in the header of the first write command is equal to the value of the transmission cycle bit contained in the header of the second write command, the slave node on the replaceable supply item sends to the controller of the image forming device a response to the second write command without executing the first write command.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] The accompanying drawings incorporated in and forming a part of the specification, illustrate several aspects of the present disclosure, and together with the description serve to explain the principles of the present disclosure.

Figure 1 is a schematic view of an image forming device having one or more replaceable supply items according to one example embodiment.

Figure 2 is a block diagram depiction of the image forming device including a communication bus that permits communication between a controller of the image forming device and one or more supply chips according to one example embodiment.

Figure 3 is a flowchart illustrating a method of issuing commands from the controller of the image forming device to a supply chip according to one example embodiment.

Figure 4 is a flowchart illustrating a method of processing and responding to commands received by the supply chip from the controller according to one example embodiment.

Figure 5 is a flowchart illustrating a method of processing and responding to commands received by the supply chip from the controller upon the occurrence of a busy condition according to one example embodiment.

DETAILED DESCRIPTION

[0006] In the following description, reference is made to the accompanying drawings where like numerals represent like elements. The embodiments are described in sufficient detail to enable those skilled in the art to practice the present disclosure. It is to be understood that other embodiments may be utilized and that process, electrical, and mechanical changes, etc., may be made without departing from the scope of the present disclosure. Examples merely typify possible variations. Portions and features of some embodiments may be included in or substituted for those of others. The following description, therefore, is not to be taken in a limiting sense and the scope of the present disclosure is defined only by the appended claims.

[0007] Referring now to the drawings and particularly to Figure 1, there is shown a schematic depiction of an image forming device 10 according to one example embodiment. Image forming device 10 may include, for example, an electrophotographic printer (commonly referred to as a laser printer), which may include scanning, copying or faxing functionality. One or more supply items 100 are removably installed in image forming device 10 allowing a user to replace or repair supply items 100 as needed in order to continue printing. For example, Figure 1 shows an access door 12 on image forming device 10 that opens to allow a user to install or remove supply items 100 from image forming device 10. In the example embodiment illustrated, each supply item 100 includes a reservoir 102 formed within a housing 104 of the supply item 100 for holding a supply of toner for use in print operations by image forming device 10. However, supply item 100 may include any component requiring periodic user replacement such as, for example, a fuser of an electrophotographic printer, a waste toner container that stores unused toner removed from a photoconductive drum or intermediate transfer member of an electrophotographic printer, etc.

[0008] Each supply item 100 includes a supply chip 110 mounted on housing 104 of the supply item 100. Supply chip 110 is electrically connected to electrical contacts on supply item 100 that are positioned to mate with corresponding electrical contacts in image forming device 10 when supply item 100 is installed in image forming device 10 in order to permit communication between supply chip 110 and a controller 20 of image forming device 10. For example, supply chip 110 of supply item 100 and controller 20 of image forming device 10 may exchange authentication or validation information in order to ensure that supply item 100 is authorized and suitable for use in image forming device 10. Supply chip 110 of supply item 100 and controller 20 of image forming device 10 may exchange operating or usage information such as an initial toner fill level, an indication of a toner color contained within reservoir 104, ongoing toner supply levels, consumption or usage rates, replacement schedules, or other information useful for the operation of supply item 100 in image forming device 10.

[0009] Figure 2 shows a communication bus 30 of image forming device 10 that facilitates communication between controller 20 and supply chips 110 according to one example embodiment. Figure 2 shows one supply item 100 and associated supply chip 110 in solid line and an additional supply item 100n and associated supply chip 110n in broken line indicating that additional supply items 100 having supply chips 110 may be used as desired. In the example embodiment illustrated, communication bus 30 utilizes the Inter-Integrated Circuit (I2C) protocol; however, other communication protocols may be used as desired. Controller 20 is configured to serve as the master node of communication bus 30 with each supply chip 110 configured to serve as a respective slave node. In the example embodiment illustrated, a system chip 22 serves as an additional slave node of communication bus 30. In this embodiment, controller 20 and system chip 22 combine to form a control unit 24 of image forming device 10 that controls communication with supply chips 110. In one embodiment, system chip 22 facilitates encrypted communication between controller 20 and supply chips 110 where controller 20 and supply chips 110 are unable to directly interpret messages from each other for security purposes. For example, prior to sending an encrypted command to a supply chip 110, controller 20 may send the command to system chip 22 which may decrypt

and re-encrypt the command according to a format that will be understood by supply chip 110 and return the re-encrypted command to controller 20 to be sent by controller 20 to supply chip 110. Similarly, upon receiving an encrypted response from a supply chip 110, controller 20 may send the response to system chip 22 which may decrypt and re-encrypt the response according to a format that will be understood by controller 20 and return the re-encrypted response to controller 20. System chip 22 may also aid controller 20 in the authentication and validation of supply chips 110.

[0010] In the example embodiment illustrated, communication bus 30 includes a bidirectional data line 32 that carries data back and forth between controller 20 and the slave nodes and a clock line 34 that carries clock signals from controller 20 to the slave nodes. Each slave node includes a respective address on communication bus 30 allowing controller 20 to direct communications to a particular slave node.

[0011] Controller 20, system chip 22 and supply chips 110 may each include a printed circuit board assembly having one or more processors and associated memory. The processor may include one or more integrated circuits in the form of a microprocessor or microcontroller. The memory may be any suitable combination of volatile and non-volatile memory such as, for example, random access memory (RAM), read only memory (ROM), read-write memory, flash memory, EEPROM and/or non-volatile RAM (NVRAM). For example, controller 20 may include non-volatile memory storing program instructions for issuing commands to the slave nodes and for receiving and processing responses from the slave nodes. Similarly, system chip 22 and supply chips 110 may each include non-volatile memory storing program instructions for receiving and processing commands from controller 20 and for responding to commands received from controller 20. It will also be appreciated that various aspects of controller 20, system chip 22 and supply chips 110 may be implemented in special purpose hardware by way of one or more application-specific integrated circuits (ASICs) such that the functions of controller 20, system chip 22 and supply chip(s) 110 may be carried out by hardware, software or a combination thereof.

[0012] Controller 20 as the master node is programmed to issue various write and read commands to the slave nodes over communication bus 30. For example, controller 20 may issue commands to request authentication or validation credentials from the slave nodes or to instruct the slave nodes to perform authentication or validation tests, to request toner capacity or usage information from supply chips 110 or to store toner usage information to supply chips 110, etc. In one embodiment, each write command issued by controller 20 is followed by a corresponding read command from controller 20. Controller 20 initiates write and read commands by sending a start condition over data line 32 of communication bus 30 that includes the address of the intended slave node and a read/write bit set to a value representing either a write command or a read command. After initializing a write command, controller 20 sends a data packet associated with the write command to the receiving slave node. After initializing a read command, controller 20 sends clock pulses on clock line 34 to the receiving slave node. The slave nodes are programmed to receive and process write commands from controller 20 and to send a response to controller 20 upon receiving the

corresponding read command.

[0013] Commands and responses sent over communication bus 30 by the master node and the slave nodes adhere to a particular data packet format allowing the commands and responses to be understood by the receiving master node or slave node. For example, in one embodiment, each data packet includes a header that includes an indication of the length of the data being transmitted, an encryption bit that indicates whether or not the command or response is encrypted and a transmission cycle bit that is toggled for each new write command sent by controller 20 as discussed in greater detail below. Each write command transmitted by the master node also includes a Command ID and each response transmitted by a slave node includes a Response ID that allows the receiving slave node or master node to identify the particular command or response being transmitted. Each data packet also includes the data being transmitted in the body of the data packet. In one embodiment, each data packet also includes a cyclic redundancy check (CRC) value allowing the receiving master node or slave node to check the accuracy of the data transmitted as is known in the art.

[0014] Figure 3 illustrates a method 300 of issuing commands from controller 20 as the master node to supply chip 110 as the slave node according to one example embodiment. At step 301, controller 20 sends a write command over data line 32 of communication bus 30 to supply chip 110 with the transmission cycle bit (tc) at an initial value (tco). At step 302, controller 20 sends a read command over data line 32 of communication bus 30 to supply chip 110 and receives a response over data line 32 of communication bus 30 from supply chip 110. At step 303, controller 20 determines whether the response from supply chip 110 indicates that supply chip 110 calculated a CRC error indicating that an error occurred in the transmission of the write command from controller 20 to supply chip 110. If the response from supply chip 110 indicates that supply chip 110 calculated a CRC error, controller 20 resends the write command to supply chip 110 at step 301 with the transmission cycle bit (tc) at the initial value (tco) and sends a read command to supply chip 110 and receives a response from supply chip 110 at step 302. Resending the write command to supply chip 110 with the transmission cycle bit (tc) at the initial value (tco) signals to supply chip 110 that the write command received is a repeat of the preceding write command rather than a new write command. If the response from supply chip 110 indicates that supply chip 110 did not calculate a CRC error, controller 20 determines whether a CRC error is present in the response received from supply chip 110 at step 304 indicating that an error occurred in the transmission of the response from supply chip 110 to controller 20. If the controller 20 determines that a CRC error is present in the response, controller 20 resends the write command to supply chip 110 at step 301 with the transmission cycle bit (tc) at the initial value (tco) and sends a read command to supply chip 110 and receives a response from supply chip 110 at step 302. Again, resending the write command to supply chip 110 with the transmission cycle bit (tc) at the initial value (tco) signals to supply chip 110 that the write command received is a repeat of the preceding write command rather than a new write command.

[0015] If controller 20 determines that no CRC error is present in the response, controller 20 determines whether the response from supply chip 110 is encrypted at step 305. If the

response from supply chip 110 is encrypted, controller 20 determines that the command was successful at step 306 and toggles the transmission cycle bit (tc) from the initial state (tco) to an opposite state (tc₁), i.e., controller 20 toggles the transmission cycle bit (tc) from a binary zero state to a binary one state or vice versa, at step 307. Toggling the transmission cycle bit (tc) at step 307 allows the next write command sent by controller 20 to include the transmission cycle bit at an initial value that is opposite the initial value (tco) of the write command sent by controller 20 at step 301. In this manner, the transmission cycle bit (tc) toggles back and forth between a binary zero state and a binary one state each time a new write command is sent signaling to supply chip 110 that the write command received is a new write command to be executed.

[0016] If the response from supply chip 110 is not encrypted, controller 20 determines whether the response from supply chip 110 indicates that supply chip 110 is busy at step 308 indicating that controller 20 sent the read command at step 302 before supply chip 110 finished executing the write command sent at step 301. If the response from supply chip 110 indicates that supply chip 110 is not busy, at step 309, controller 20 compares a Command ID value of the write command sent at step 301 with a Response ID value of the response received at step 302. If the Command ID value and the Response ID value match indicating that the response received by controller 20 from supply chip 110 is responsive to the write command issued by controller 20, controller 20 determines that the command was successful at step 306 and toggles the transmission cycle bit (tc) from the initial state (tco) to an opposite state (tc₁) at step 307. If the Command ID value and the Response ID value do not match indicating that the response received by controller 20 from supply chip 110 is not responsive to the write command issued by controller 20, controller 20 resets supply chip 110 at step 310 in order to attempt to resolve the error.

[0017] In one embodiment, if the response from supply chip 110 indicates that supply chip 110 is busy at step 308, controller 20 sends a "Get Response" command over data line 32 of communication bus 30 to supply chip 110 with the transmission cycle bit (tc) at the initial value (tco). In the example embodiment illustrated, the Get Response command is a special purpose command used only after a busy response is received from supply chip 110. The Get Response command instructs supply chip 110 to provide a response to the last write command (other than any preceding Get Response commands) received by supply chip 110. This allows controller 20 to continually poll supply chip 110 for a response while supply chip 110 continues to execute the write command sent at step 301 so that a response may be sent by supply chip 110 once supply chip 110 finishes executing the write command and is no longer in a busy state. The Get Response command allows controller 20 to continually request a response from supply chip 110 without having to resend the full write command each time. The Get Response command includes a smaller data packet and, in this manner, is abbreviated with respect to other write commands sent by controller 20 to supply chip 110. As a result, sending a Get Response command is typically more efficient than resending the original write command.

[0018] Figure 4 illustrates a method 400 of processing and responding to commands received by supply chip 110 as the slave node from controller 20 as the master node according to one

example embodiment. At step 401, supply chip 110 receives a write command (Command_n) from controller 20 over data line 32 of communication bus 30. At step 402, supply chip 110 determines whether a CRC error is present indicating that an error occurred in the transmission of the write command from controller 20 to supply chip 110. If supply chip 110 calculates a CRC error at step 402, upon receiving a read command over data line 32 of communication bus 30 corresponding to the write command (Command_n) received at step 401, supply chip 110 sends an error response over data line 32 of communication bus 30 to controller 20 at step 403 indicating that supply chip 110 calculated a CRC error. If supply chip 110 does not calculate a CRC error at step 402, supply chip 110 determines at step 404 whether the value of the transmission cycle bit (tc_n) contained in the write command (Command_n) received at step 401 is the same as the value of the transmission cycle bit (tc_{n-1}) contained in the preceding write command (Command_{n-1}) received by supply chip 110. If supply chip 110 determines that the transmission cycle bit (tc_n) of the write command (Command_n) received at step 401 is not equal to the transmission cycle bit (tc_{n-1}) of the preceding write command (Command_{n-1}) received by supply chip 110 indicating that the write command (Command_n) received at step 401 is a new write command to be executed, at step 405, supply chip 110 executes the write command (Command_n) received at step 401. At step 406, upon receiving a read command corresponding to the write command (Command_n) received at step 401, supply chip 110 sends a response over data line 32 of communication bus 30 indicating the completion of the write command (Command_n) received at step 401.

[0019] If, on the other hand, supply chip 110 determines that the transmission cycle bit (tc_n) of the write command (Command_n) received at step 401 is equal to the transmission cycle bit (tc_{n-1}) of the preceding write command (Command_{n-1}) received by supply chip 110, supply chip 110 determines at step 407 whether the Command ID value contained in the write command (Command_n) received at step 401 is the same as the Command ID value contained in the preceding write command (Command_{n-1}) received by supply chip 110. If supply chip 110 determines that the Command ID of the write command (Command_n) received at step 401 is not equal to the Command ID of the preceding write command (Command_{n-1}) received by supply chip 110, in the example embodiment illustrated, supply chip determines at step 408 whether the Command ID value contained in the write command (Command_n) received at step 401 indicates that the write command (Command_n) is a Get Response command sent by controller 20 following a busy response from supply chip 110 as discussed above. If supply chip 110 determines that the write command (Command_n) is a Get Response command, supply chip 110 proceeds to the method 500 illustrated in Figure 5 until the busy condition is resolved as discussed in greater detail below. If supply chip 110 determines that the write command (Command_n) is not a Get Response command, upon receiving a read command corresponding to the write command (Command_n) received at step 401, supply chip 110 sends an error response to controller 20 over data line 32 of communication bus 30 at step 403 due to the Command ID inconsistency.

[0020] If supply chip 110 determines that the Command ID of the write command (Command_n) received at step 401 is equal to the Command ID of the preceding write command (Command_{n-1}) received by supply chip 110, supply chip 110 does not execute the write command (Command_n) received at step 401 but instead, upon receiving a read command corresponding to the write command (Command_n) received at step 401, supply chip 110 simply resends the response to the preceding write command (Command_{n-1}) to controller 20 over data line 32 of communication bus 30 at step 409. In this manner, the transmission cycle bit signals to supply chip 110 whether the write command (Command_n) received at step 401 is a new write command to be executed or a repeat of the preceding write command (Command_{n-1}). The recognition by supply chip 110 that the write command (Command_n) received at step 401 is a repeat of the preceding write command (Command_{n-1}) improves the efficiency of supply chip 110 by allowing supply chip 110 to simply resend the response to the preceding write command (Command_{n-1}) without executing the write command (Command_n) again. The recognition by supply chip 110 that the write command (Command_n) received at step 401 is a repeat of the preceding write command (Command_{n-1}) also prevents possible errors that could occur if supply chip 110 were to execute the write commands (Command_{n-1}, n) twice instead of once as intended by controller 20.

[0021] If at any point after the receipt of the write command (Command_n) at step 401 supply chip 110 receives the corresponding read command before supply chip 110 has completed processing the write command (Command_n), for example, prior to or during the execution of steps 402, 404, 405, 407 or 408, supply chip 110 may send a busy response to controller 20 over data line 32 of communication bus 30. As discussed above, the receipt of a busy response from supply chip 110 may cause controller 20 to send a Get Response command to supply chip 110.

[0022] Figure 5 illustrates a method 500 of processing and responding to commands received by supply chip 110 as the slave node from controller 20 as the master node after supply chip 110 issues a busy response to controller 20 according to one example embodiment. At step 501, after previously sending a busy response to controller 20, supply chip 110 receives a Get Response command from controller 20 over data line 32 of communication bus 30. If supply chip 110 is still busy processing the most recent write command (preceding the one or more Get Response commands received) upon receiving a read command from controller 20 corresponding to the Get Response command at step 502, supply chip 110 again sends a busy response to controller 20 over data line 32 of communication bus 30 at step 503. Supply chip 110 may continue to receive additional Get Response commands and send busy responses until supply chip finishes processing the most recent write command from controller 20. Once supply chip 110 completes processing the most recent write command at step 502, upon receiving a read command from controller 20, supply chip 110 sends a response to the most recent write command received from controller 20. In this manner, the Get Response command accounts for the occurrence of a busy condition of supply chip 110 allowing controller 20 to continually request a response from supply chip 110 without having to resend

the full write command each time.

[0023] While the example embodiments discussed above with respect to Figure 3 includes the issuing of commands from controller 20 as the master node to supply chip 110 as the slave node, the commands discussed in Figure 3 may instead to be issued by controller 20 as the master node to system chip 22 as the slave node. Similarly, while the example embodiments discussed above with respect to Figures 4 and 5 include the processing and responding to commands received by supply chip 110 as the slave node from controller 20 as the master node, the commands discussed in Figures 4 and 5 may instead be received and processed by system chip 22 as the slave node from controller 20 as the master node.

[0024] The example embodiment discussed above include communications between a controller 20 of an electrophotographic image forming device 10 and a supply chip 110 of a supply item 100 installed in image forming device 10. The electrophotographic printing process is well known in the art and, therefore, is described briefly herein. During a print operation, a charge roll charges the surface of a photoconductive drum. The charged surface of the photoconductive drum is then selectively exposed to a laser light source to form an electrostatic latent image on the photoconductive drum corresponding to the image being printed. Charged toner from a toner reservoir is transferred by a developer roll to the latent image on the photoconductive drum creating a toned image. The toned image is then transferred from the photoconductive drum to the print media either directly by the photoconductive drum or indirectly by an intermediate transfer member. A fusing unit fuses the toner to the print media. A cleaning blade (or cleaning roll) removes any residual toner adhering to the photoconductive drum after the toner is transferred to the print media. The cleaned surface of the photoconductive drum is then ready to be charged again and exposed to the laser light source to continue the printing cycle.

[0025] The foregoing description illustrates various aspects of the present disclosure. It is not intended to be exhaustive. Rather, it is chosen to illustrate the principles of the present disclosure and its practical application to enable one of ordinary skill in the art to utilize the present disclosure, including its various modifications that naturally follow. All modifications and variations are contemplated within the scope of the present disclosure as determined by the appended claims. Relatively apparent modifications include combining one or more features of various embodiments with features of other embodiments.

REFERENCES CITED IN THE DESCRIPTION

Cited references

This list of references cited by the applicant is for the reader's convenience only. It does not

form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US2009285594A1 [0003]
- WO0106444A1 [0003]

Non-patent literature cited in the description

- **JAN AXELSON**.USB COMPLETE: THE DEVELOPER' S GUIDE, 978-1-931448-08-6 [0003]
- **AGHDAEI N. et al**.CLIENT-TRANSPARENT FAULT-TOLERANT WEB SERVICEConference Proceedings of the 2001 IEEE International Performance, Computing and Communications, Conference, 978-0-7803-7001-2 [0003]

PATENTKRAV

1. En fremgangsmåde til elektrisk kommunikation mellem en slaveknode (110) af en I2C-kommunikationsbus (30) placeret på et udskifteligt forsyningsselement (100) installeret i en billeddannende indretning (10) og en controller (20) af billedannelsesindretningen (10) som en masterknode af I2C-kommunikationsbussen (30), omfattende:

modtagelse (401) af en første skrivekommando af slaveknoten (110) på det udskiftelige forsyningsselement (100) fra styreenheden (20) af billedannende indretningen (10);

sammenligning (404) af slaveknoten (110) på det udskiftelige forsyningsselement (100) en værdi af en transmissionscyklusbit (tc_n) indeholdt i en header af den første skrivekommando med en værdi af en transmissionscyklusbit (tc_{n-1}), indeholdt i en header af en anden skrivekommando modtaget af slaveknoten (110) på det udskiftelige forsyningsselement (100) fra styreenheden (20) af billedannende indretningen (10) forud for den første skrivekommando; og

hvis værdien af transmissionscyklusbitten (tc_n) indeholdt i headeren af den første skrivekommando er lig med værdien af transmissionscyklusbitten (tc_{n-1}) indeholdt i headeren af den anden skrivekommando,

sammenligning (407) af slaveknoten (110) på det udskiftelige forsyningsselement (100) af en værdi af et kommando-ID ($Command_n$) for den første skrivekommando med en værdi af et kommando-ID ($Command_{n-1}$) af den anden skrivekommando; og

hvis værdien af kommando ID ($Command_n$) for den første skrivekommando er lig med værdien af kommando ID ($Command_{n-1}$) af den anden skrivekommando, sendes (409) af slaveknoten (110) på det udskiftelige forsyningsselement (100) til styreenheden (20) af billedannende indretningen (10) et svar på den anden skrivekommando uden at udføre den første skrivekommando.

2. Fremgangsmåden ifølge krav 1,

yderligere omfattende, hvis værdien af transmissionscyklusbitten (tc_n) indeholdt i headeren af den første skrivekommando ikke er lig med værdien af transmissionscyklusbitten (tc_{n-1}) indeholdt i headeren af den anden skrivekommando, udføres (405) af slaveknuden (110) på det udskiftelige forsyningselement (100) den første skrivekommando og afsendelse af slaveknuden (110) på det udskiftelige forsyningselement (100) til controlleren (20) af billeddannende indretningen (10) et svar på den første skrivekommando.

10 3. Fremgangsmåden ifølge krav 1,

yderligere omfattende, hvis værdien af kommando ID ($Command_n$) af den første skrivekommando ikke er lig med værdien af kommando ID ($Command_{n-1}$) af den anden skrivekommando, sende (403) af slaveknuden (110) på det udskiftelige forsyningselement (100) en fejlmeddelelse til styreenheden (20) af billeddannende indretningen (10).

4. Fremgangsmåden ifølge krav 1,

hvor i sammenligning (407) af slaveknuden (110) på det udskiftelige forsyningselement (100) af værdien af kommando-ID'et ($Command_n$) for den første skrivekommando med værdien af kommando-ID'et ($Command_{n-1}$) af den anden skrivekommando kommandoen inkluderer at bestemme (408) af slaveknuden (110) på det udskiftelige forsyningselement (100), om værdien af kommando-ID'en ($Command_n$) for den første skrivekommando indikerer, at den første skrivekommando er en anmodning fra controlleren (20)) af billeddannende indretningen (10) for slaveknuden (110) på det udskiftelige forsyningselement (100) for at sende et svar på en seneste skrivekommando modtaget af slaveknuden (110) på det udskiftelige forsyningselement (100) fra styreenheden (20) af billeddannende indretningen (10) før slaveknuden (110) på det udskiftelige forsyningselement (100), hvilket indikerer en optaget tilstand til billeddannende indretningens (10) styreenhed (20).

DRAWINGS

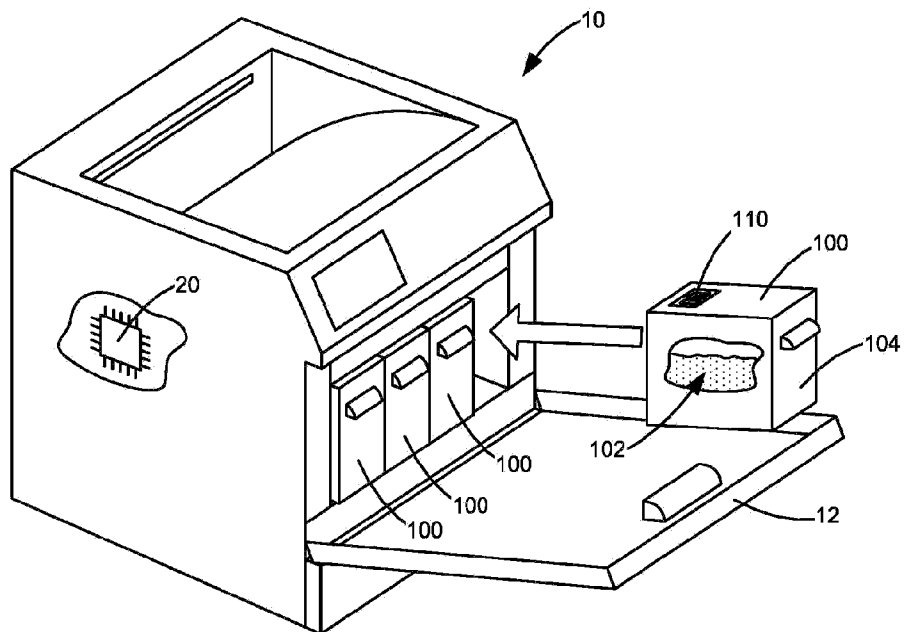


FIGURE 1

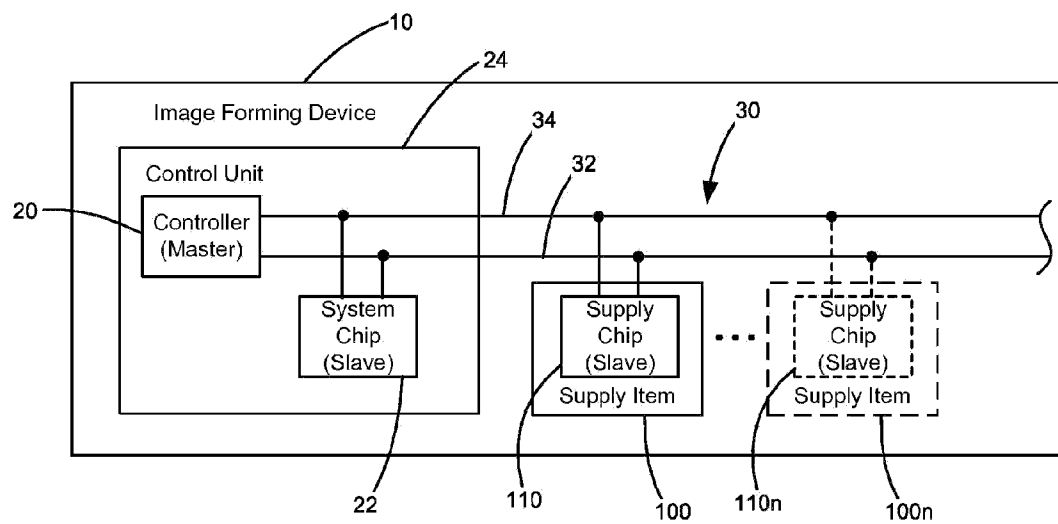


FIGURE 2

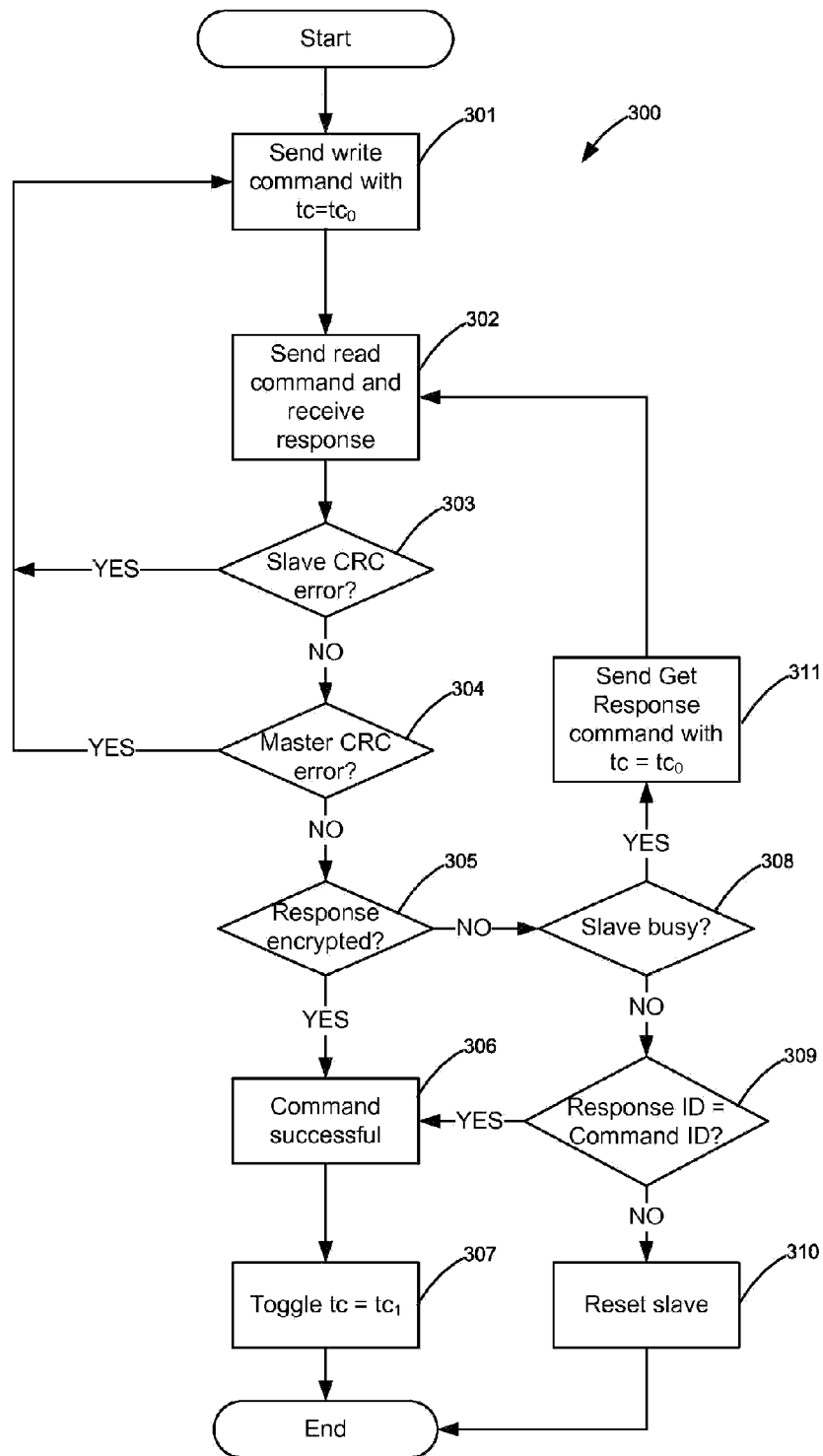


FIGURE 3

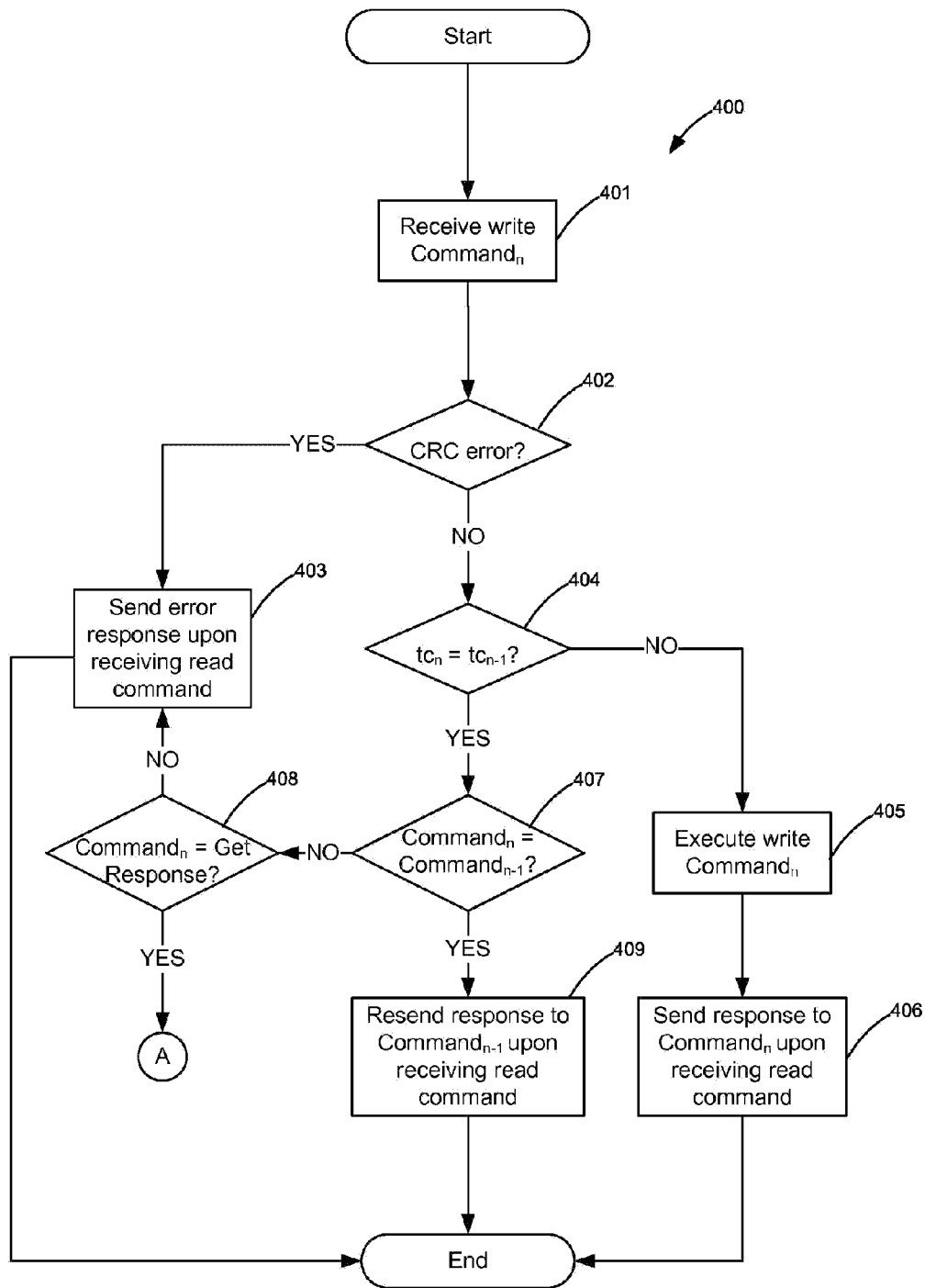


FIGURE 4

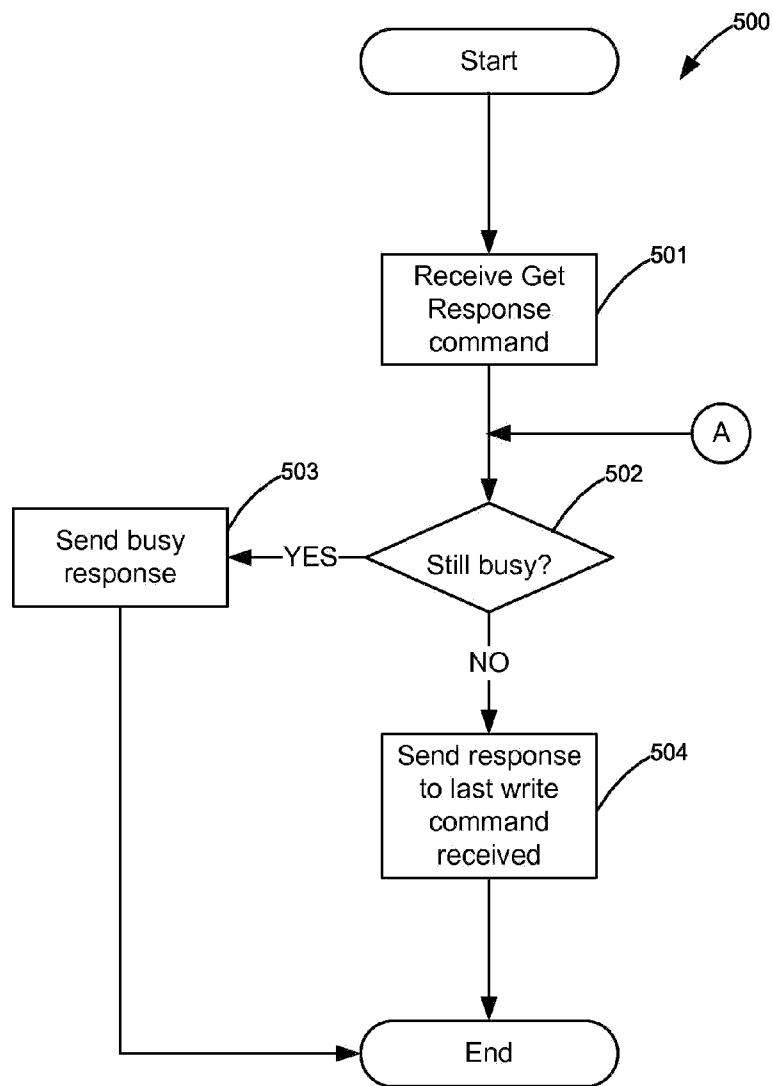


FIGURE 5