

controller 36, a cyclic redundancy check (CRC) circuit 38, and a display controller 40. CPU 28 carries out various program operations associated with software loaded in memory 34. Data entered by users via keypad 10 in the manner described above is processed by CPU 28 and stored or buffered in memory 34 for use in program operations associated with software loaded in memory 34. DMA controller allows rapid transfer of data from memory 34 to CRC circuit 38 and display controller 40 as described further below. Display controller 40 may comprise a display controller that is operatively coupled to display 22 as shown in FIG. 1 and FIG. 2 and described above, and provides for operation of display 22. Display controller 40 includes a memory (not shown) for storage of display data.

[0044] The CPU 28, memory 34, DMA controller 36, CRC circuit 38 and display controller 40 are arranged on a motherboard (not shown) in a conventional manner and interconnected thereon by address and data bus 30 and control/processing interface 32. Data processing device 26 may comprise various additional components (also not shown) such as a hard disk drive, floppy disk drive, NIC, CD drive, and/or other conventional hardware elements. Data processing device 26 includes an interface adapter 42 that allows connection of data processor 26 to an external computer 44 via an interface cable or connection 46. Adapter 42 and interface 46 may be in the form of a GPIB, RS-232, PCI, USB, SCSI, ETHERNET®, FIREWIRE® or other IEEE 1394 interface, or other communication interface system for transfer of data to device 26 from external computer 44.

[0045] System memory 34 will generally contain a suitable operating system and software suitable for the operation of the various hardware components, which are operatively coupled to memory 34 and CPU 28 via the address/data bus 30 and control/status signal interface 32. Memory 34 also includes stored programming or software capable of carrying out various operations in accordance with the invention.

[0046] Memory 34 includes programming 48 that is capable of effecting transfer of data streams from memory 34 to CRC circuit 38 via DMA controller 36 by carrying out the operations of seeding the CRC circuit 38 with a desired initial value, setup of DMA controller circuit 36 with source and destination addresses and data stream sizes for data transfer, initiating the transfer of data to CRC circuit 38 by DMA controller circuit 36, and readout of calculated CRC values from CRC circuit 38 back to memory 34.

[0047] Memory 34 additionally includes programming 50, capable of effecting transfer of data from memory 34 to display controller 40 via DMA controller 36, wherein are carried out the operations of setup of display controller 40 with destination address information, setup of DMA controller 36 with source and destination address information and data stream size information, and initiation of data transfer to display controller 40 from memory 34 by DMA controller 36.

[0048] Also included in memory 34 is software or programming 52 capable of comparing compressed input strings to a stored list of compressed strings in memory 34 for authorization, with programming operations for acceptance of a character string input, transfer of the input string to the CRC circuit 38 for compression, searching the stored list of compressed ID strings in memory 34 for a match with

the compressed input string, and validation of the compressed input string. These programming operations are described further below.

[0049] CRC circuit 38 provides for error detection in the transfer of binary data between the various hardware components of data processing device by calculation of check numbers that are used to verify the data stream at a destination. The CRC calculation is carried out by seeding a polynomial with an initial value, and then sequencing through a stream of data into which the polynomial gets divided. The dividend is used at each step as the new seed, with the division algorithm being performed by CRC circuit 38. CRC circuits of this type are well known in the art and need not be described herein.

[0050] Calculation of CRC values or numbers with a CRC circuit has traditionally involved a software loop for incrementing through the data stream and writing of bytes to the CRC circuit carried out by programming operations. CRC calculation in this manner results in a large software overhead and results in delays during calculation of check values for large streams of data. The software overhead consideration is particularly significant for hand held computing devices in which device size imposes limitations on available memory and processing power.

[0051] The subject invention overcomes this drawback by utilizing DMA controller circuit 36 in conjunction with CRC circuit 38 to perform the CRC calculation. Software 48 is used to seed CRC circuit 38 with a desired initial value, and to load the DMA controller circuit 36 with the address of the first byte in the data stream and the number of bytes in the stream for which a CRC calculation is made. The DMA controller circuit 36 then automatically transfers the stream of bytes into the CRC circuit 38 for calculation of a check value. Once the entire data stream has been processed by CRC circuit 38, software is then used to read the resulting calculated check value from CRC circuit 38 to carry out an integrity check for the data stream. Since software 48 is only employed in association with configuring the CRC circuit 38 and DMA controller circuit 36, initiating the transfer of data by DMA controller circuit 36, and readout of the check values, the overall software overhead required for calculation of the check value is small. The time required for CRC calculation using the above procedure can be an order of magnitude shorter than is achievable by transfer of data to CRC circuit 38 via software alone.

[0052] The calculation of check values using DMA controller circuit 36 and CRC circuit 38 in accordance with the invention will be more fully understood by reference to FIG. 4, as well as FIG. 3. At event 100, software 48 loaded in memory 34 is started or initiated which includes programming for carrying out operations associated with seeding the CRC circuit 38, setup of DMA controller circuit 36, initiating the transfer of data to CRC circuit 38 by DMA controller circuit 36, and readout of calculated CRC values from CRC circuit 38.

[0053] At event 110, CRC circuit 38 is seeded with a desired initial value for the data stream for which a CRC value is to be calculated. The initial value will vary depending upon the size of the data stream and CRC value and the degree of confidence required in the integrity of the data stream. The data stream may comprise, for example, data associated with a string of alphanumeric characters entered