Intel Corporation
v.
Qualcomm Incorporated

IPR2018-01334

U.S. Patent No. 8,838,949

Patent Owner's Demonstrative Exhibits

U.S. Patent No. 8,838,949

(12) **United States Patent**
Gupta et al.

(10) **Patent No.: US 8,838,949 B2**
(45) **Date of Patent: Sep. 16, 2014**

(54) **DIRECT SCATTER LOADING OF EXECUTABLE SOFTWARE IMAGE FROM A PRIMARY PROCESSOR TO ONE OR MORE SECONDARY PROCESSOR IN A MULTI-PROCESSOR SYSTEM**

(75) Inventors: **Nitin Gupta**, San Diego, CA (US); **Daniel H. Kim**, San Diego, CA (US); **Igor Malamant**, San Diego, CA (US); **Steve Haehnichen**, San Diego, CA (US)

(73) Assignee: **QUALCOMM Incorporated**, San Diego, CA (US)

(58) **Field of** CPC ..... USPC .. See appli

(56)

5,978,589
6,079,017
7,447,846

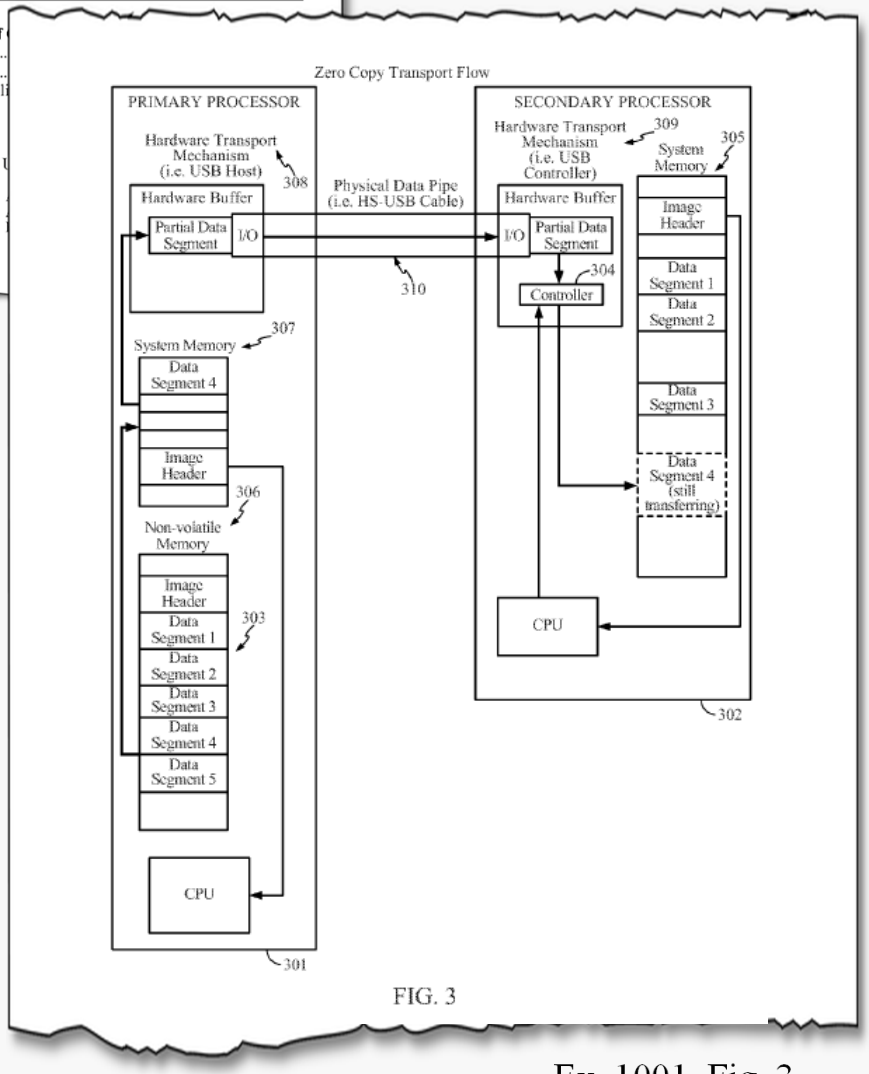
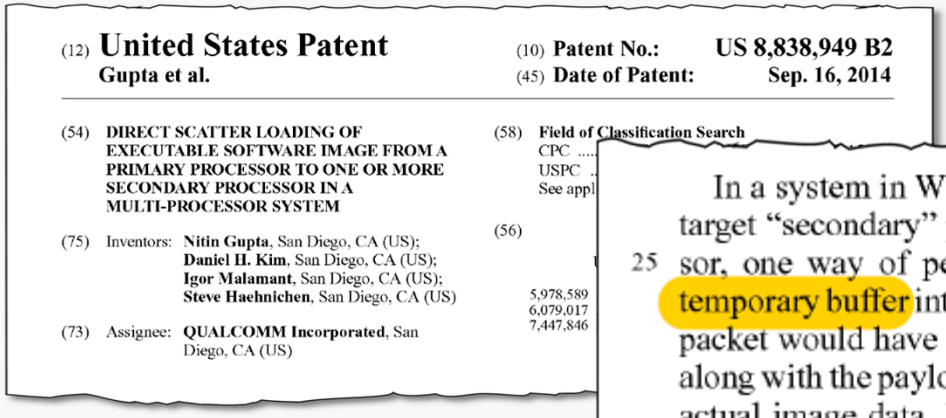


FIG. 3

Ex. 1001, Fig. 3

The Prior Art Used a Temporary Intermediate RAM Buffer at the Secondary Processor



(12) **United States Patent**
Gupta et al.

(10) **Patent No.: US 8,838,949 B2**
(45) **Date of Patent: Sep. 16, 2014**

(54) **DIRECT SCATTER LOADING OF EXECUTABLE SOFTWARE IMAGE FROM A PRIMARY PROCESSOR TO ONE OR MORE SECONDARY PROCESSOR IN A MULTI-PROCESSOR SYSTEM**

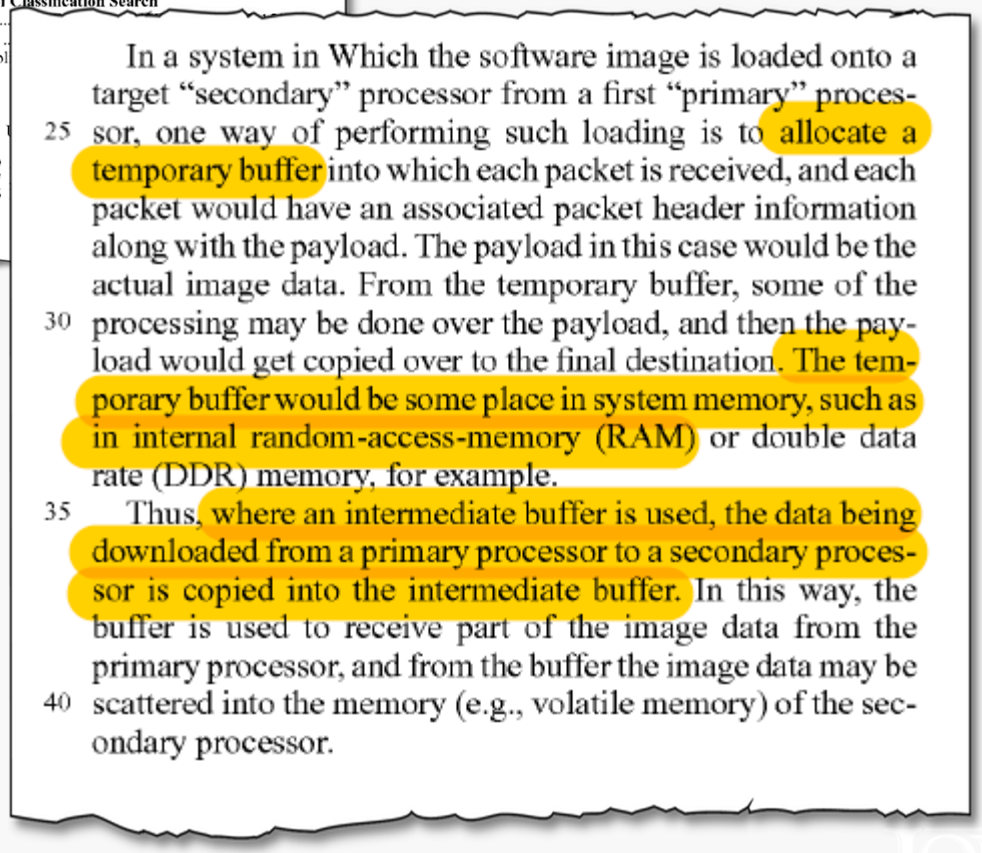
(75) Inventors: **Nitin Gupta**, San Diego, CA (US); **Daniel H. Kim**, San Diego, CA (US); **Igor Malamant**, San Diego, CA (US); **Steve Haehnichen**, San Diego, CA (US)

(73) Assignee: **QUALCOMM Incorporated**, San Diego, CA (US)

(58) **Field of Classification Search**
CPC .....
USPC ..
See appl

(56)

5,978,589
6,079,017
7,447,846



In a system in Which the software image is loaded onto a target "secondary" processor from a first "primary" processor, one way of performing such loading is to allocate a temporary buffer into which each packet is received, and each packet would have an associated packet header information along with the payload. The payload in this case would be the actual image data. From the temporary buffer, some of the processing may be done over the payload, and then the payload would get copied over to the final destination. The temporary buffer would be some place in system memory, such as in internal random-access-memory (RAM) or double data rate (DDR) memory, for example.

Thus, where an intermediate buffer is used, the data being downloaded from a primary processor to a secondary processor is copied into the intermediate buffer. In this way, the buffer is used to receive part of the image data from the primary processor, and from the buffer the image data may be scattered into the memory (e.g., volatile memory) of the secondary processor.

Ex. 1001, 2:24-42 (emphasis added)

The ‘949 Invention Eliminated the Temporary Intermediate RAM Buffer

Zero Copy Transport flow

Aspects of the present disclosure provide techniques for efficiently loading the executable software images from the primary processor’s non-volatile memory to the secondary processor’s volatile memory. As mentioned above, traditional loading processes require an intermediate step where the binary multi-segmented image is buffered (e.g., transferred into the system memory) and then later scattered into target locations (e.g., by a boot loader). Aspects of the present disclosure provide techniques that alleviate the intermediate step of buffering required in traditional loading processes. Thus, aspects of the present disclosure avoid extra memory copy operations, thereby improving performance (e.g., reducing the time required to boot secondary processors in a multi-processor system).

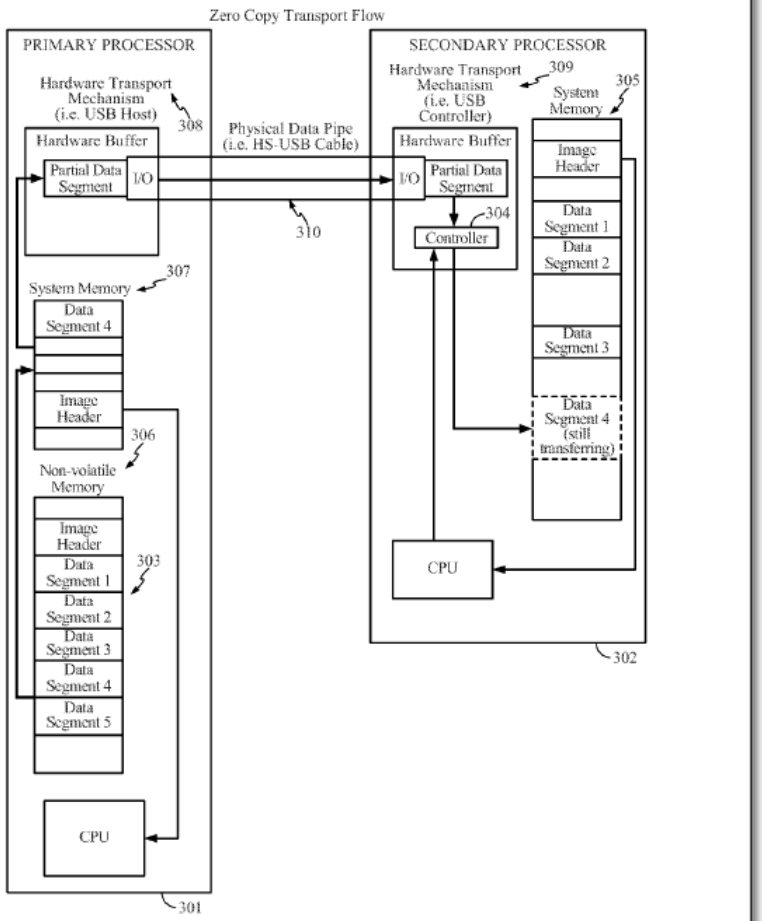


FIG. 3

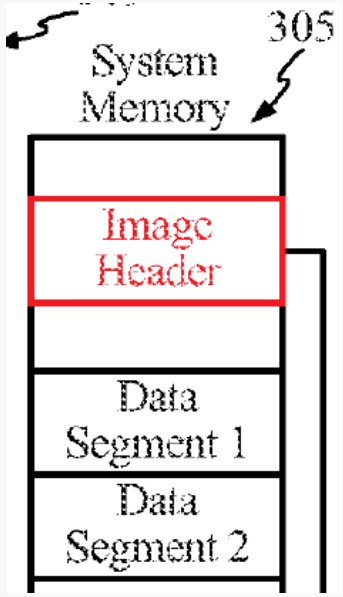
Ex. 1001, 7:15-30; Fig. 3 (emphasis added)

The '949 Invention Uses an Image Header Instead of Segment Headers

In one aspect, the host primary processor does not process or extract any information from the actual image data it simply sends the image data as "raw" data to the target, without any packet header attached to the packet. Because the target secondary processor initiates the data transfer request, it knows exactly how much data to receive. This enables the host to send data without a packet header, and the target to directly receive and store the data. In that aspect, the target requests data from the host as needed. The first data item it requests is the image header for a given image transfer. Once the target has processed the image header, it knows the location and size of each data segment in the image. The image

In accordance with certain aspects of the present invention (e.g., as in the example of FIG. **3**), the raw image data is transported. For instance, rather than transporting each segment of image data with a packet header, the exemplary load process of FIG. **3** determines the needed information about the data from the header associated with the entire image. Thus, the image header may be initially transferred, and all the processing for determining how to store the data to system memory **305** can occur before the transfer of the segments (based on the image header), and then the segments are transferred as raw data, rather than requiring processing of a packet-header for each segment as the segments are transferred. Thus, in the example of FIG. **3**, the raw image data is

processor **307**. The image header includes information used to identify where the modem image executable data is to be eventually placed into the system memory of the secondary processor **305**. The header information is used by the secondary processor **302** to program the scatter loader/direct memory access controller **304** receive address when receiving the actual executable data. Data segments are then sent



Ex. 1001, 7:39-50; 8:17-24; 10:3-14; Fig. 3 (emphasis added)

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.