UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC.,
Petitioner,

v.

CALIFORNIA INSTITUTE OF TECHNOLOGY,
Patent Owner.

Case IPR2017-00700
Patent 7,421,032

PETITIONER'S REPLY TO PATENT OWNER'S RESPONSE

TABLE OF CONTENTS

I. INTRODUCTION

Caltech's Patent Owner Response ("POR") repeats arguments that the Board has already rejected and fails to rebut Petitioner's showing that the challenged claims are unpatentable. First, Caltech mischaracterizes the teachings of the Ping and MacKay references. Second, Caltech has failed to demonstrate secondary considerations of non-obviousness. Finally, Caltech mischaracterizes the testimony of Petitioner's expert, Prof. Davis.

II. ARGUMENT

A. Caltech Fails to Overcome Petitioner's Showing that the Challenged Claims are Obvious

1. *Claims 11, 12, and 14-16 are Obvious in view of Ping, MacKay, and Divsalar*

The Petition showed that Ping in view of MacKay and Divsalar renders claims 11, 12, and 14-16 obvious. Caltech's arguments about the combination are incorrect for at least the reasons below.

i. Contrary to Caltech's Argument, MacKay teaches that information bits appear in a variable number of subsets

Caltech's suggestion that it is unclear in MacKay whether a column of the parity check matrix corresponds to an information bit or a parity bit is incorrect. (POR, 22.) To even attempt to make this argument, Caltech must ignore MacKay's actual disclosure. MacKay teaches profiles, *e.g.*, 93y, that correspond to parity

check matrices. (Ex. 1002, 1450.) Those matrices have uneven column weights. For example, as shown in MacKay's Figure 2, in 93y matrices, most columns have weight three but some columns have weight nine. MacKay also teaches that codes with such parity check matrices, *i.e.*, matrices with uneven column weights, can outperform their regular counterparts. (Ex. 1065, ¶¶20-24.)[1]

Caltech only attempts to contend that the correspondence between information bits and the columns of a parity check matrix may be unclear in ***some*** of MacKay's parity check matrices (*e.g.*, profile 93y). Caltech does not (and cannot) dispute that this correspondence is perfectly clear in other disclosed matrices (*e.g.*, profile 193y). In particular, in Figures 5 and 6, MacKay states that the first K columns (all columns to the left of the diagonal) correspond to information bits. (Ex. 1002, 1452 ("Bits t_1 … t_K are defined to be source bits.").) As shown in profile 193y, some of these information bits correspond to columns with weight nine and others correspond to columns with weight three, *i.e.*, some information bits appear in nine subsets and others appear in three subsets. MacKay's Figures 5 and 6 thus clearly teach that information bits appear in a variable number of subsets. Using those

[1] After submitting his declaration, Dr. Davis relocated to Europe pursuant to a Fulbright Global Scholar Award. (Ex. 1073, ¶2.) As a result, he was unavailable to work on the Reply. (*Id.*) Petitioner's Reply is instead supported by the Declaration of Dr. Frey.

weightings in Ping results in information bits appearing in variable numbers of subsets (*i.e.*, either nine or three) as claimed. (Ex. 1065, ¶¶20-24.)

ii. Even if MacKay's Irregular Column Weights Could Be Limited As Caltech Contends, Its Argument Would Still Fail

Caltech argues that MacKay's columns with uneven weight could all correspond to parity bits such that the columns corresponding to information bits all had the same weight. (POR, 21.) By Caltech's incorrect logic, that would result in MacKay – standing alone – failing to teach that information bits appear in a variable number of subsets. (*Id.*)

Caltech's argument is false for the reasons demonstrated in Part A(1)(i) above. But even if it were true, Caltech's argument would still fail because it ignores the combination of MacKay's column weight teaching with Ping's unambiguous teaching that all columns in its $\mathbf{H}^{d}$ matrix represent information bits. (Ex. 1065, ¶25.)

The Petition showed, and the Board agreed, that a POSA would have been motivated to use MacKay's uneven column weights in Ping's $\mathbf{H}^{d}$ matrix (or outer coder) to improve the performance of Ping's code. (DI, 13-19.) Doing so would have resulted in information bits appearing in a variable number of subsets, which corresponds exactly to some information bits contributing to more parity bits than others. This is true even if all of MacKay's uneven column weights corresponded to

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.