IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
TYLER DIVISION

THE PACID GROUP, LLC,	**§**	
	§	
Plaintiff,	**§**	
	§	
v.	**§**	**CIVIL ACTION No. 6:09cv143-LED-JDL**
	§	
APPLE, INC., ET AL.,	**§**	
	§	
Defendants.	**§**	

MEMORANDUM OPINION AND ORDER

This Memorandum Opinion and Order sets forth the Court's constructions for the disputed claim terms in the patents asserted by Plaintiff The PACid Group, LLC ("PACid"). PACid asserts U.S. Patent Nos. 5,963,646 ("the '646 patent") and 6,049,612 ("the '612 patent") and has filed an Opening Claim Construction Brief (Doc. No. 251) ("Opening"), as well as a Reply in support of PACid's proposed constructions (Doc. No. 269) ("Reply"). Defendants Atheros Communications, Inc., Broadcom Corporation, Intel Corporation, and Marvell Semiconductor Inc. (collectively, "Defendants") have filed a Responsive Claim Construction Brief (Doc. No. 265) ("Response"). A *Markman* hearing was held on March 25, 2010 (Doc. No. 309) ("Transcript"), where eleven disputed claim terms were submitted to the Court for construction.[1] (Doc. No. 273-1) ("Joint Claim

[1] Initially the parties submitted thirteen disputed terms. However, as noted in the Provisional Claim Construction Order, since agreement was reached as to a construction for "logic function" and "cryptographic function," those terms will not be discussed herein. The claim term "interrupt control means . . for issuing an interrupt signal upon receipt of said command sequences" in claims 12 and 26 of the '646 patent is the subject of an indefiniteness challenge (Doc. No. 264). This term will not be addressed in the Claim Construction Opinion, but will be addressed in a Report and Recommendation issued contemporaneously.

Chart”).[2] The Court entered a Provisional Claim Construction Order (Doc. No. 297) on April 21, 2010. For the reasons stated herein, the Court adopts the constructions set forth below

OVERVIEW OF THE PATENTS

The technology in this case relates to a system of encryption, and applications thereof, that provide a way to secure the contents of communications that resists attempts to decipher the encoded communications. *See* ‘646 patent at 3:18–3; ‘612 patent at 1:8–4. The technology discussed in the patents generates symmetric encryption keys that are use to encrypt and decrypt files on a computer. ‘646 patent at 1:17–43; ‘612 patent at 1:8-14 and 1:27–49.

The ‘646 Patent

The ‘646 patent relates to a method and system for generating deterministic, symmetric encryption keys used to secure data in computer systems. ‘646 patent at 1:17–21; 1:35–62 and 3:1–23. To protect the inputs of a key generator during the process of generating symmetric encryption keys, the ‘646 patent proposes a key generator as illustrated in Figure 2. *See* ‘646 patent at 3:1–6; 4:55–5:5; and Fig. 2.

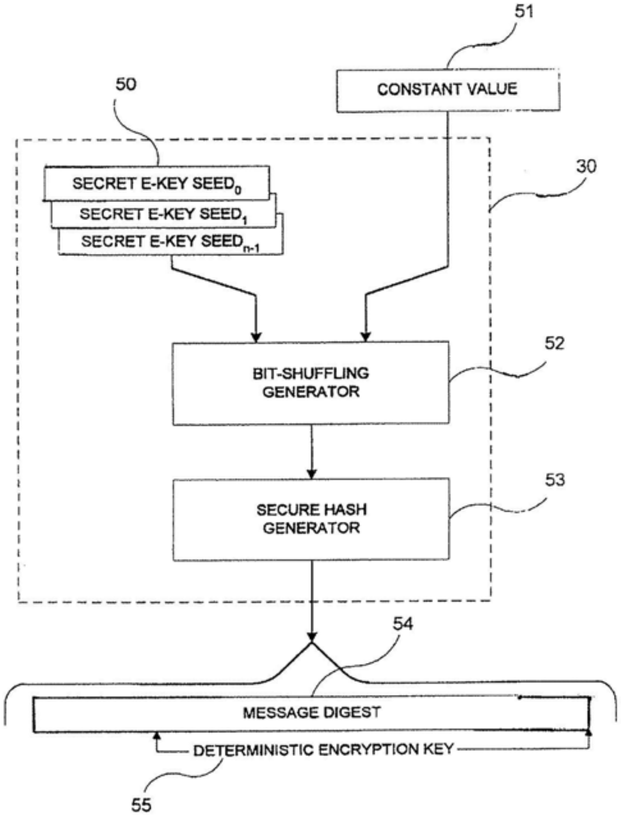

FIG. 2

[2] The parties also provided the Court with a Joint Claim Construction Chart pursuant to P.R. 4-5(d).

Claim 12 of the ‘646 patent is set forth below as a representative claim with disputed claim terms set forth in bold:

> 12. An encryption key generator in electrical communication with a **host system**, which comprises:
>
> an I/O interface means in electrical communication with said **host system** and receiving command sequences from said **host system**;
>
> **interrupt control means** in electrical communication with said I/O interface means for issuing an interrupt signal upon receipt of said command sequences;
>
> a ROM in electrical communication with said I/O interface means and having stored therein operating firmware, a **bit-shuffle computer program**, and a **secure hash computer program**;
>
> a RAM in electrical communication with said I/O interface means and said ROM for storing a current E-Key Seed and a **constant value**;
>
> an EEPROM in electrical communication with said I/O interface means, said ROM, and said RAM, for storing said E-Key Seed and said **constant value**; and
>
> a CPU in electrical communication with said **interrupt control**, said I/O interface means, said ROM, said RAM, and said EEPROM for executing said **bit-shuffle computer program** to combine said **constant value** and said E-Key Seed in a first many-to-few bit mapping, and for extracting a **pseudo-random** symmetric, encryption key from said message digest and storing said encryption key in said EEPROM.

‘ 646 patent at 9:49–10:13 (claim 12).

The ‘612 Patent

The ‘612 patent relates to a method and system for protecting sensitive information files and message from access by unauthorized parties. ‘612 patent at 1:8–14. Figures 3 and 4 illustrate additional security measures undertaken when encrypting an information file with the key.

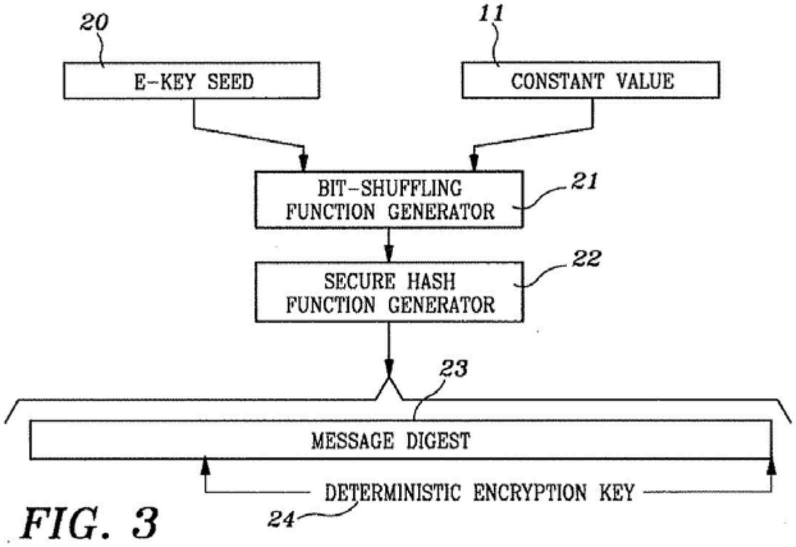
20
11
E-KEY SEED
CONSTANT VALUE
BIT-SHUFFLING FUNCTION GENERATOR
21
SECURE HASH FUNCTION GENERATOR
22
23
MESSAGE DIGEST
DETERMINISTIC ENCRYPTION KEY
24

FIG. 3

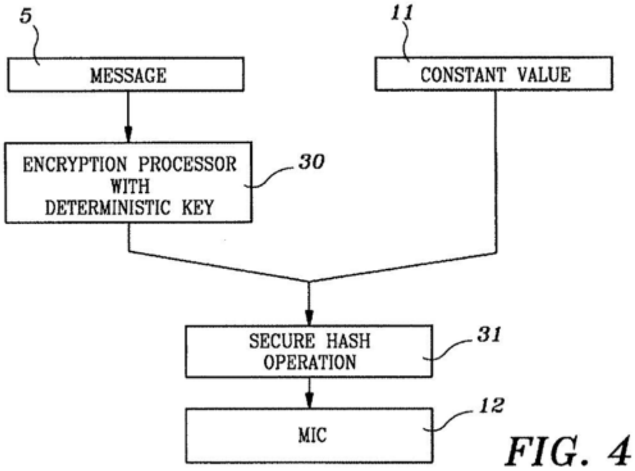

FIG. 4

Claim 1 of the ‘612 patent is set forth below as a representative claim with disputed claim terms set forth in bold:

> 1. A method of protecting an **information file** from unauthorized access, which comprises the following steps:
>
> combining a **constant value** and a secret plural bit sequence in accordance with an **algebraic function to shuffle bits**, perform a first many-to-few bit mapping, and produce a first **pseudo-random** result;

> performing a **secure hash operation** on said first **pseudo-random** result to effect a second many-to-few bit mapping and produce a second **pseudo-random** result;
> extracting a **pseudo-random**, symmetric encryption key from said second **pseudo-random** result;
> encrypting said **information file** in accordance with said **pseudo-random**, symmetric encryption key to form an encrypted **information file**; and
> **concatenating** said **constant value** to a beginning of said encrypted **information file**.

‘612 patent at 7:34–49 (claim 1).

LEGAL STANDARD

“It is a ‘bedrock principle’ of patent law that ‘the claims of a patent define the invention to which the patentee is entitled the right to exclude.’” *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (hereinafter “Phillips”) (quoting *Innova/Pure Water Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)). Under *Markman v. Westview Instruments, Inc.*, the court construes the scope and meaning of disputed patent claims as a matter of law. 517 U.S. 370, 373 (1996). In claim construction, courts examine the patent’s intrinsic evidence to define the patented invention’s scope. *See id.*; *C.R. Bard, Inc. v. U.S. Surgical Corp.*, 388 F.3d 858, 861 (Fed. Cir. 2004); *Bell Atl. Network Servs., Inc. v. Covad Commc’ns Group, Inc.*, 262 F.3d 1258, 1267 (Fed. Cir. 2001). This intrinsic evidence includes the claims themselves, the specification, and the prosecution history. *See Phillips*, 415 F.3d at 1314; *C.R. Bard, Inc.*, 388 F.3d at 861. Courts give claim terms their ordinary and accustomed meaning as understood by one of ordinary skill in the art at the time of the invention in the context of the entire patent. *Phillips*, 415 F.3d at 1312–13; *Alloc, Inc. v. Int’l Trade Comm’n*, 342 F.3d 1361, 1368 (Fed. Cir. 2003).

The claims themselves provide substantial guidance in determining the meaning of particular claim terms. *Phillips*, 415 F.3d at 1314. First, a term's context in the asserted claim can be very instructive. *Id*. Other asserted or unasserted claims can also aid in determining the claim's meaning because claim terms are typically used consistently throughout the patent. *Id*. Differences among the claim terms can also assist in understanding a term's meaning. *Id*. For example, when a dependent claim adds a limitation to an independent claim, it is presumed that the independent claim does not include the limitation. *Id*. at 1314–15.

"[C]laims 'must be read in view of the specification, of which they are a part.'" *Id*. (quoting *Markman v. Westview Instruments, Inc*., 52 F.3d 967, 979 (Fed. Cir. 1995) (en banc)). "[T]he specification 'is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.'" *Id*. (quoting *Vitronics Corp. v. Conceptronic, Inc*., 90 F.3d 1576, 1582 (Fed. Cir. 1996)); *Teleflex, Inc. v. Ficosa N. Am. Corp*., 299 F.3d 1313, 1325 (Fed. Cir. 2002). This is true because a patentee may define his own terms, give a claim term a different meaning than the term would otherwise possess, or disclaim or disavow the claim scope. *Phillips*, 415 F.3d at 1316. In these situations, the inventor's lexicography governs. *Id*. Also, the specification may resolve ambiguous claim terms "where the ordinary and accustomed meaning of the words used in the claims lack sufficient clarity to permit the scope of the claim to be ascertained from the words alone." *Teleflex*, 299 F.3d at 1325. Nonetheless, "'[a]lthough the specification may aid the court in interpreting the meaning of disputed claim language, particular embodiments and examples appearing in the specification will not generally be read into the claims.'" *Comark Commc'ns, Inc. v. Harris Corp*., 156 F.3d 1182, 1187 (Fed. Cir. 1998) (quoting *Constant v. Advanced Micro-Devices, Inc*., 848 F.2d 1560, 1571 (Fed. Cir. 1988)); *see also Phillips*, 415 F.3d at 1323. The prosecution history is another tool to supply the proper context for claim construction because a patent applicant may also define a term in prosecuting the patent. *Home Diagnostics, Inc. v. Lifescan, Inc*., 381 F.3d 1352, 1356 (Fed. Cir. 2004) ("As in the case of the specification, a patent applicant may define a term in prosecuting a patent.").

Although extrinsic evidence can be useful, it is "'less significant than the intrinsic record in determining the legally operative meaning of claim language.'" *Phillips*, 415 F.3d at 1317 (quoting *C.R. Bard, Inc*., 388 F.3d at 862). Technical dictionaries and treatises may help a court understand the underlying technology and the manner in which one skilled in the art might use claim terms, but technical dictionaries and treatises may provide definitions that are too broad or may not be indicative of how the term is used in the patent. *Id*. at 1318. Similarly, expert testimony may aid a court in understanding the underlying technology and determining the particular meaning of a term in the pertinent field, but an expert's conclusory, unsupported assertions as to a term's definition is entirely unhelpful to a court. *Id*. Generally, extrinsic evidence is "less reliable than the patent and its prosecution history in determining how to read claim terms." *Id*.

DISCUSSION

The Court construes the following disputed claim terms:[3] 1) "pseudo-random," 2) "constant value" 3) "shuffled bit result" and its relevant permutations, 4) "secure hash operation" and its relevant permutations, 5) "performing a secure hash operation on said shuffled bit result to produce a message digest," 6) "performing a secure hash operation on said first pseudo-random result to . . . produce a second pseudo-random result," 7) "algebraic function," 8) "host system,"9) "information file" or "message file," and 10) "concatenating."

[3] The parties have also agreed to a number of constructions. *See* JOINT CHART.

I. “pseudo-random”[4]

Plaintiff’s Proposed Construction	**Defendants’ Proposed Construction**
No construction necessary; alternatively, “apparently random, but repeatable and predictable”	“refers to output that is repeatable and predictable to anyone who knows the function’s input but appears to be totally random to those without such knowledge”

The patents-in-suit both contain the claim term “pseudo-random,” and the parties agree that this term has the same meaning in all of the asserted claims.[5] In the specifications, the claim term is recited as “output” “that is repeatable and predictable to anyone who knows the E-Key seed input to the function producing the output.” The specifications further states that “without such knowledge, the output appears to be totally random.” ‘646 patent at 5:14–18;‘612 patent at 4:1–4.

PACid contends that this term does not need to be construed because the jury may apply its plain meaning. OPENING at 3. Alternatively, PACid proposes a construction that incorporates the “relevant properties of a pseudo-random output.” In particular, PACid suggests a construction for pseudo-random that it is “repeatable and predictable, yet apparently random.” *Id*. at 4. Defendants contend that a lay juror will not be familiar with the claim term and the Court should adopt the explicit definition in the specifications. RESPONSE at 3–4.

A review of the disputed claim term suggests that even while “pseudo-random” should be accorded its “ordinary and customary meaning,” in this case, determining the plain and ordinary meaning of “pseudo-random” goes beyond the application of widely accepted meaning of

[4] The term “pseudo-random” is contained in claims 1, 12, 17, and 26 of the ‘646 patent and claim 1 of the ‘612 patent.

[5] The presumption that the same claim term retains the same meaning throughout related patents will be applied throughout the instant Memorandum Opinion and Order. *See Omega Eng’g, Inc. v. Raytek Corp*., 334 F.3d 1314, 1334 (Fed. Cir. 2003) (“we presume, unless otherwise compelled, that the same claim term in . . . related patents carries the same construed meaning”).

commonly understood words. *See Phillips*, 415 F.3d at 1314. Accordingly, it is appropriate to provide a construction that would assist a lay jury in understanding what a person of ordinary skill in the art would understand "pseudo-random" to mean in light of the '646 and '612 patents. *See O2 Micro International Ltd. v. Beyond Innovation Tech. Co.*, 521 F.3d 1351, 1362–63 (Fed. Cir. 2008).

The Court finds that the inventor provided an express definition for the disputed term:

> The term 'pseudo-random' as used in this specification means that the output referred to is repeatable and predictable to anyone who knows they E-Key seed input to the function producing the output. Without such knowledge, the output appears to be totally random.

'646 patent at 5:14–18; '612 patent at 4:1–4. The inventor's express definition offers a construction that includes three elements that the patentee considered important at the time of the invention: (1) "without knowledge of the function's input," the output "appears to be totally random"; (2) the function is repeatable and predictable "to anyone who knows the function's input; and (3) the output appears to be "totally" random. *See* '646 patent at 5:14–18; '612 patent at 4:1–5. Since these elements were clearly set forth, a construction not including all of the inventor's requirements— as provided in the specification— would impermissibly broaden the term beyond its express definition. PACid considers the explicit requirements of the specification to be "surplus language," but by explicitly defining "pseudo-random" in the specification, the patentee instructed those skilled in the art that all three requirements would be present in the claim term. *See Cook Biotech Inc. v. Acell, Inc.*, 460 F.3d 1365, 1372 (Fed. Cir. 2006) (recognizing that the person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification).Where, as here, the inventor provided an explicit definition for "pseudo-random," that definition will govern. *Phillips*, 415 F.3d at 1313; *Martek Biosciences Corp. v. Nutrinova, Inc.*, 579 F.3d 1363, 1380 (Fed. Cir. 2009).

Accordingly, the Court finds that a construction is needed to resolve the parties' dispute and that the claim scope is best defined by combining passages in the specification. The construction for the term "pseudo-random" is "refers to output that is repeatable and predictable to anyone who knows the function's input but appears to be totally random to those without such knowledge."

II. "constant value"[6]

Plaintiff's Proposed Construction	**Defendants' Proposed Construction**
No construction necessary; alternatively, "a value that does not change for any given instance of generating an encryption key"	"a value that does not change"

The patents-in-suit both contain the claim term "constant value." PACid advocates that a plain reading of the claims and patents-in-suit illustrates that a "constant value" is "not a universal and immutable value that does not change." OPENING at 5. Should a construction be necessary, PACid suggests: "a value that does not change for any given instance of generating an encryption key." PACid supports its alternative construction with language in the '612 specification describing examples of a constant value as having "several different components that can be changed." *Id*. (citing '612 patent at 6:6–21). Using the E-Key Seed ID as an example, PACid contends that the constant value can differ amongst host systems and that "different values can have different lengths." OPENING at 5.

[6] The term "constant value" is contained in claims 1, 6, and 12–16 of the '646 patent and claims 1 and 7 of the '612 patent.

Defendants dispute this approach, arguing for the dictionary definition of "constant value," meaning "[a] value that does not change." RESPONSE at 26. Defendants contend that the patentee never choose to be his own lexicographer and their proposed construction ascribes a meaning to "constant" that aligns with the commonly accepted meaning of the term. In short, Defendants maintain that the value does not deviate based on circumstances. *Id*.

As an initial matter, the differences articulated between PACid and Defendants' proposed constructions indicate uncertainty as to the claim term's scope, which is properly resolved through claim construction. *O2 Micro*, 521 F.3d at 1361 ("A determination that a claim term 'needs no construction' or has the 'plain and ordinary meaning' may be inadequate when a term has more than one 'ordinary' meaning or when reliance on a term's 'ordinary' meaning does not resolve the parties' dispute."). The application of "constant" throughout the patents does not indicate that the patentee chose to ascribe a special meaning to this term. Therefore, in the absence of any "special definition," one of ordinary skill in the art would understand "constant" to mean what it means in a dictionary: "unchanging, permanent, or fixed." *See Comaper Corp. v. Antec, Inc.*, 596 F.3d 1343, 1348 (Fed. Cir. 2010) (where the patent specification does not assign or suggest a particular meaning to the claim term, the claim is to be construed as it would be understood by a person of ordinary skill in the art in light of a general dictionary definition) (internal citation omitted); *see also* AM. HERITAGE COLLEGE DICTIONARY 298 (3d ed. 1993). As such, the Court finds the Defendants' proposed construction uncontroverted by the intrinsic record and affirmatively supported by the extrinsic record.

Although clearly presented, Defendants' proposed construction does not account for the particular context in which an encryption key is generated. In order to decipher the scope of the claim term, it is proper to evaluate the particular circumstances required to generate the key. *See Therasense, Inc. v. Becton, Dickinson and Co.*, 593 F.3d 1289, 1323 (Fed. Cir. 2010) (recognizing that "patentees frequently use terms idiosyncratically," which is why persons of ordinary skill must look to the full context in which a term is used) (citing *Phillips*, 415 F.3d at 1314). In other words, while "a value that does not change" is an accurate depiction of how "constant value" is used throughout the patents, it is also appropriate to clarify how this general term is understood in light of the process disclosed in the claims. To that end, the Court finds that the patents-in-suit only require that the "constant value" be constant for the purpose of generating a given encryption key (regardless of how many times the key is produced), and once that key is generated, another constant may be used to generate a different encryption key.

The meaning of "constant" in the asserted claims is only fully understood through the patents' description of the larger encryption/ decryption process.[7] This is unambiguously borne out in the '612 patent specification at column 6, lines 6 through 21, as well as Figure 6. The described examples, using references to the E-Key Seed ID, shows that a "constant value" has different components that can be changed, and thus, disclose a different "constant value" for each encryption key. In particular, a "constant value" can include an E-Key Seed ID, and the user may choose to assign another E-Key Seed ID. *See* '612 patent at 6:13–18 ("When the constant value 11 is first being formed, the E-Key Seed ID is automatically entered as that of the host system. A user is

[7] The importance of the circumstances underlying this term can be aptly analogized to familiar "constant values" like Pi or the acceleration of gravity. While Pi is an immutable constant that does not vary with context, the acceleration of gravity depends upon location because gravity varies with contextual facts such as altitude and local properties of the Earth (e.g., density and shape). Yet, both Pi and the gravitational pull are true constant values. Like gravity, the constant value in these patents depends upon circumstances such as the length byte and E-Key Seed ID.

prompted, however, to either accept the ID or assign another. In this manner, files may be shared between PCs, workstations, and workgroups that normally use different E-Key Seeds.").

The specification language supports both Defendants' position that "constant value" does not change, but it also supports PACid's position that the "constant" of a particular encryption is determined by the circumstances underlying its generation. Therefore, to ease juror understanding of the claim term at trial, the Court adopts a construction for "constant value" that means "a value that does not change," but this definition implicitly incorporates the teaching that the "constant value" can be different for each instance of generating an encryption key and the same "constant value" is employed when repeatedly producing a particular encryption key.

III. "shuffled bit result" and its relevant permutations

Permutation of Claim Term	**Plaintiff's Proposed Construction**	**Defendants' Proposed Construction**
"shuffled bit result"[8]	"the result of an operation that mixes the bits of its inputs"	"the result of an operation that mixes and maps the bits of its inputs"
"bit shuffling operations"[9]	No separate construction (in light of the others); alternatively, "operations that mix the bits of inputs"	"operations that mixes and maps the bits of their inputs"
"bit shuffling function"[10]	No separate construction (in light of the others); alternatively, "a function that mixes the bits of its inputs"	"a function that mixes and maps the bits of its inputs"

[8] The term "shuffled bit result" is contained in claims 1, 3, and 18 of the '646 patent.

[9] The term "bit shuffling operations" is contained in claim 18 of the '646 patent.

[10] The term "bit shuffling function" is contained in claim 19 of the '646 patent.

“function to shuffle bits”[11]	No separate construction (in light of the others); alternatively, “a function that mixes the bits of its inputs”	“a function that mixes and maps the bits of its inputs”
“bit shuffle computer program”[12]	No separate construction (in light of the others); alternatively, “a computer program that mixes the bits of its inputs”	“computer program that performs a bit shuffle operation”

The patents-in-suit both contain permutations of claim terms that will be collectively referred to as the “shuffle bit terms.” The shuffle bit terms include “shuffled bit result,” “bit shuffling operations,” “bit shuffling function,” “function to shuffle bits,” and “bit shuffle computer program.” In all instances, the parties agree that shuffling requires mixing bits, but they disagree as to whether shuffling also requires mapping bits. The primary dispute for these claims terms is whether mapping should be included in the definition of “shuffling”.

PACid argues that the ‘646 specification supports an understanding of the shuffle bit terms that singularly includes the mixing of bits. OPENING at 6 (citing ‘646 patent at 4:60–65; and 5:1–2). PACid contends that the specification distinguishes between shuffling and mapping, and that mixing is the result of a specific algebraic, cryptographic, and/or logic function that is not random. *Id*. Arguing that the term “shuffled” in the context of the ‘646 and ‘612 patents means mixed, PACid also suggests that the meanings of the shuffle bit terms is “readily understood by a lay jury” and does not require separate construction. REPLY at 6.

[11] The term “function to shuffle bits” is contained in claim 1 of the ‘612 patent.

[12] The term “bit shuffle computer program” is contained in claim 12 of the ‘646 patent.

Defendants argue that "bit-shuffle" is not a term that has a well-known definition in the field of encryption and must be construed in light of the patent specifications. RESPONSE at 16. Defendants then cite to relevant portions of the specifications to argue that the Summary of the Invention section requires both mixing and mapping and that the disclosed embodiments require a bit shuffling generator "to mix the input bits and to map the input bits to a result." *Id*. at 16–17 (citing '646 patent at 3:22–28 (Summary of the Invention); '646 patent at 4:60–5:1; '612 patent at 4:47–53). Defendants also maintain that the specification supports the suggested mixing and mapping to occur "randomly."

"When the parties present a fundamental dispute regarding the scope of a claim term, it is the court's duty to resolve it." *O2 Micro*, 521 F.3d at 1362. Appreciating that the shuffle bit terms present both disputed meanings and divergent depiction as to the terms' technical scope, the instant terms require construction. *See U.S. Surgical Corp. v. Ehticon, Inc.*, 103 F.3d 1554, 1568 (Fed. Cir. 1997).

1. Bit Shuffling does not require "mapping"

While the patents provide no definition for the word "mapping," the context of its usage is clear, particularly in view of the parties' briefings and arguments. The intrinsic record coupled with the common meaning of the term instruct that a mapping of an input value into an output value requires a correlation between particular input bits and particular output bits. In effect, the non-technical common meaning of the word "map" would also imply input-output correlation in that it would suggest a path between an input bit and an output bit. However, unlike the common meaning of "map," the patents clearly discuss many-to-few bit mapping, and in doing so, clearly eschew any notion that mapping requires that each input bit maps to a unique output bit.

The intrinsic record presents that mapping and shuffling are distinct concepts, and therefore, mapping should not be included in the definition of shuffling. Starting with the language of the claims, claim 1 of the ‘612 patent approaches “shuffle bits” and “many-to-few bit mapping” as separate actions: “combining a constant value and secret plural bit sequence in accordance with an algebraic function to **shuffle bits**, perform a first many-to-few **bit mapping**, and produce a first pseudo-random result.” ‘612 patent at 7:36–39 (claim 1) (emphasis added). A plain reading of this claim language supports the conclusion that mapping and shuffling are not one and the same. Defendants suggest there is a double requirement to mix and map because certain portions of the specification use the words “mixed” and “mapped” when describing the same bit shuffling process. This conjunctive language, however, does not support the conclusion that mapping necessarily occurs when shuffling takes place. Instead, the specification distinguishes between the two concepts by using “mixing” and “mapping” in alternative forms. In particular, the ‘646 patent recites: “employs a many-to-few bit mapping and a combination bit shuffle.” ‘646 patent at 1–4. The patentee is clearly requiring that the bits be mapped, but this mapping is not required to be one and the same as the alternative mixing process.

This point was further illustrated at the *Markman* hearing, where the meaning of “mapping” was flushed out in the context of a logical XOR function.[13] The ‘646 specification provides, as an example of a preferred embodiment, a bit-shuffling generator that employs the XOR function as expressed in the equation $A \oplus B = C$. In this example embodiment, the XOR function operates on the two input values “A” and “B,” and results in the output “C.” The parties agreed at the hearing that the XOR function effects a mapping. TRANSCRIPT at 76. The Court agrees because the

[13] A two-input logical XOR function yields a true (“1”) result if one or the other of two inputs is true (“1”), but not if both inputs are true (“1”).

application of the XOR function correlates or maps the particular input bits ("A" and "B") to one particular output bit ("C"). Thus, through use of the logical XOR function as a bit-shuffling example, the patent is clear that mapping may occur during the process of bit-shuffling. There is equal clarity, however, that bit-shuffling need not employ a logical function such as "XOR." The patent clearly states that bit-shuffling may employ algebraic, cryptographic and/or logic function. *See* '646 patent at 4:60–65. Unlike the logical functions—such as the XOR function— algebraic and cryptographic functions do not necessarily yield output bits that specifically correlate to input bits. Thus, mapping is not a requirement of bit shuffling.

Lastly, during the hearing, Defendants argued that bit-shuffling requires mapping (correlation), but not a one-to-one correlation between input and output bits. This argument offers only a limited explanation as to how a larger number of input bits may be mapped into a smaller number of output bits– i.e., that mapping is nevertheless mapping even if two or more input bits map to fewer output bits. Although this argument is well-taken, the premise does not bear on whether mapping itself is a requirement of bit-shuffling. Furthermore, while the Court finds that "mixing" is a requirement of shuffling, the Court does interpret mixing so confined as Defendants' implied during the hearing. *See* TRANSCRIPT at 69.

2. Bit Shuffling is not random

Additionally, the proposed constructions dispute whether to include a limitation that the mixing occurs "randomly." PACid argues to exclude this language because it imposes a requirement that is contrary to the pseudo-random result in the specification. REPLY at 6–7; '646 patent at 3:22–28 (reciting in the '646 patent: "a bit-shuffling which results in the mapping of a large number of bits into a first pseudo-random number"); '612 patent at 3:18–24 (reciting in the '612 patent:

"shuffles the bits and provides a pseudo-random result"). Defendants, however, maintain that the specification expressly requires that the bits are "randomly mixed and mapped." RESPONSE at 17 (citing '612 patent at 4:47–53) ("The bits. . . thereby are randomly mixed and mapped from a large binary length to a smaller binary length.").

While acknowledging the specification's use of "randomly" in its description of the claimed invention, the Court concludes that rather than limiting the claims, this language is an express example that uses the world "random" in a broad capacity that includes "pseudo-random." This is apparent because the word "random" is expressly applied to the XOR operation discussed above. As indicated above, an XOR operation is completely deterministic and not random. As such, a skilled artisan would understand that only a pseudo-random result may apply to an XOR function.

This result is also supported by the express deterministic requirements of the claims and specifications. The patents are directed to deterministic keys and the specifications explain what that means: "Encryption keys may in addition be classified as deterministic or non-deterministic. A deterministic encryption key is one which is repeatable each time a specific input is applied to the encryption key generator." '612 patent at 2:19–23. Given these teachings, the result of the key creation operation could not employ randomness in the manner suggested by Defendants and yet remain deterministic. Thus, the isolated citation to the word "random" in the '612 patent at column 4, lines 47–53 must be interpreted broadly in including a pseudo-random operation.

Accordingly, finding that (1) shuffling does not include both mixing and mapping, and (2) that mixing, as used in the shuffled bit terms, is not "random," the Court provides the following constructions:

- The term “shuffled bit result” means “the result of an operation that mixes the bits of its inputs.”
- The term “bit shuffling operations” means “operations that mix the bits of inputs.”
- The terms “bit shuffling function” and “function to shuffle bits” mean “a function that mixes the bits of its inputs.”
- The term “bit shuffle computer program” means “a computer program that mixes the bits of its inputs.”

IV. “secure hash operation” and its relevant permutations

Permutation of Claim Term	**Plaintiff’s Proposed Construction**	**Defendants’ Proposed Construction**
“secure hash operation”[14]	“algorithm that produces a deterministic output having no known relationship with the input that may be used to recover the input from the output”	“an operation that accepts an input that can be of variable bit length, but always produces an output having the same bit length such that it is computationally infeasible to determine (a) the input from the output and (b) two inputs that produce the same output, and where if a single bit of the input is changed, on average approximately 50% of the output bits are changed”

[14] The term “secure hash operation” is contained in claim 1 of the ‘646 patent and claim 1 of the ‘612 patent.

“secure hash algorithm”[15]	“algorithm that produces a deterministic output having no known relationship with the input that may be used to recover the input from the output”	“an algorithm that accepts an input that can be of variable bit length, but always produces an output having the same bit length such that it is computationally infeasible to determine (a) the input from the output and (b) two inputs that produce the same output, and where if a single bit of the input is changed, on average approximately 50% of the output bits are changed”
“secure hash computer program”[16]	No separate construction (in light of the others); alternatively, “a computer program that produces a deterministic output having no known relationship with the input that may be used to recover the input from the output”	“computer program that uses a secure hash algorithm” (as defined above)

The patents-in-suit both contain permutations of claim terms that will be collectively referred to as the “secure hash terms” or “secure hash function.” The secure hash terms include “secure hash operation,” “secure hash algorithm,” and “secure hash computer program.”

PACid argues for a construction of the secure hash terms that includes two important properties of secure hash operations: (1) that the output is deterministic (i.e., repeatable in that the same inputs always produce the same outputs), and (2) that the output has no known relationship with the input that may be used to recover the input from the output. REPLY at 6. PACid’s proposed construction is premised on language in the specifications reciting, “There is no known relationship

[15] The term “secure hash algorithm” is contained in claim 12 of the ‘646 patent.

[16] The term “secure hash computer program” is contained in claim 12 of the ‘646 patent.

between the input and output of a hash algorithm which may be used to recover the input from the output." OPENING at 7 (citing '646 patent at 2:8–10; '612 patent at 2:15–17).

Defendants argue that the specifications and extrinsic evidence known to a person of skill in the art require a secure hash function to include the four key properties of a secure hash, a "well-known tool used in cryptography": (1) that the output is always the same binary length regardless of the size of the input, (2) that it is computationally infeasible to determine the input from the output, (3) that it is computationally infeasible to determine two inputs that produce the same output, and (4) that on average, approximately 50 percent of the secure hash output bits are changed when only one single bit is changed. RESPONSE at 6–10; *see also id.* at 5, n.2 (citing cryptography handbooks and other technical resources)). Defendants present specific arguments contending that these four requirements of a secure hash were embraced by the patentee at the time of the invention and are further supported through intrinsic evidence and technical standards established in the prior art. *Id*. at 5–6. Given the disagreement surrounding the four criteria in Defendants' proposal for "secure has operation," each requirement will be evaluated in turn.

1. Accepting the proposal that the output is always the same binary length regardless of the size of the input

The first requirement proposed by Defendants is well-supported by the intrinsic language of the '646 and '612 patent specifications. Both patents clearly state that a "characteristic of a hash algorithm is that the output is always the same binary length regardless of the size of the input." '646 patent at 2:2–4; '612 patent at 5:5–11. This is further clarified in the '612 patent in describing "the present invention" by stating that a secure hash function produces a message digest of constant binary length, regardless of the length of the input. '612 patent at 2:54–65. Importantly, while this specification language appears to describe the properties of secure hash functions generally, the

Court notes that the use of "present invention" does not automatically trigger a narrowing of the claim language. *See Fenner Investments, Inc. v. 3Com Corp.*, No. 6:08-cv-61, 2009 WL 1505407, at *11–12 (E.D. Tex. May 26, 2009) (stating that the Court will not read a specification discussing "present invention" to be magic words that automatically triggers a narrowing of the claim language) (citing *Verizon Services Corp. v. Vonage Holdings Corp.*, 503 F.3d 1295, 1308 (Fed. Cir. 2007)).

Instead, as the language is used here, the description of "the present invention" in the '612 patent offers additional insight as to the patentee's understanding of the secure hash terms in a manner that comports with relevant teachings cited in the prior art.[17] Defendants have put forward a comprehensive showing that the definition of "secure hash" was known at the time of the invention through a consistent collection of prior art representing that when there was a variable length input, there was always a fixed length output. *See* RESPONSE at 7 (citing technical resources). In particular, the patentee's knowledge of this property was characterized through the disclosure of the SHA-1 reference, as well as through the of the Schneier Book instructing one of ordinary skill in the art as to "general information related to file encryption techniques." '612 patent at 2:49–51. Recognizing that cited prior art "can have particular value as a guide to the proper construction of the term," the body of prior art identified by Defendants effectively demonstrates the basic requirements of a "secure hash" known at the time of the invention. *Arthur A. Collins, Inc. v. N. Telecom Ltd.*, 216 F.3d 1042, 1045 (Fed. Cir. 2000). Therefore, based on the consistent representation of this property in the intrinsic record, the Court accepts as a key requirement of a

[17] For example, both patents-in-suit disclose the Secure Hash Standard, FIPS PUB 180-1 ("SHA-1") as an example of "secure hash." '646 patent at 2:26–28; 5:27–28; '612 patent at 2:2–5; and 5:2–4. SHA-1 demonstrates that an input can be of variable length, but always produces an output having the same bit length.

secure hash function that an input of variable bit length will always produce an output having a fixed length.

2. Rejecting proposals (a) that it is computationally infeasible to determine the input from the output and (b) that it is computationally infeasible to determine two inputs that produce the same output

It was clarified at the hearing that the "computationally infeasible" language in Defendants' proposal is not present in the PACid patents, but Defendants suggest that it is nonetheless intrinsic because it derives from the SHA-1 reference[18] disclosed in the '646 and '612 patent specifications. *See* '646 patent at 2:26–28; 5:27–28;'612 patent at 2:2–5 and 5:2–4. While it has previously been noted that cited prior art can guide the construction of a claim term because it may indicate the meaning of the term to a person of skill in the art at the time of the invention, Defendants' incorporation of "computationally infeasible" improperly imports an unclaimed limitation into the proposed construction.

Even with the support of the SHA-1 or Schneier Book prior art references, the Federal Circuit's claim construction teachings strongly weigh against interjecting limitations that are not present in the claims, the specification, or the prosecution history of the patents. *Dayco Products, Inc. v. Total Containment, Inc.*, 258 F.3d 1317, 1325 (Fed. Cir. 2001) (declining to read an unclaimed and potentially undisclosed limitation into the claims). It is a basic canon of claim construction that one may not read a limitation into a claim from the written description, *Simmons, Inc. v. Bombardier, Inc.*, 73 Fed. Appx. 421, 423 (Fed. Cir. 2003) (citing *Renishaw PLC v. Marposs Societa' per Azioni*, 158 F.3d 1243, 1248 (Fed. Cir. 1998)), and the same holds true for limitations

[18] Defendants seek to include these requirements based on language in the SHA-1 standard reciting that "[t]he SHA-1 is called secure because it is computationally infeasible to find a message [i.e., input] which corresponds to a given message digest [i.e, output] . . . ." RESPONSE at 8, n.7 (citing SHA-1 standard).

imported from intrinsically cited prior art. In evaluating the prosecution history and cited prior art, a court should consider how the evidence demonstrates the inventor understood the invention and whether the inventor intended to limit the scope of the invention. *Trading Technologies Intern., Inc. v. eSpeed, Inc*., 595 F.3d 1340, 1352 (Fed. Cir. 2010) (internal citations omitted). Here, there is no indication that the inventor was seeking to include the meanings presented in the SHA-1 reference, and in the absence of clear intent, the proposed limitation unnecessarily restricts the scope of the claim terms. *See Saunders Group, Inc. v. Comfortrac, Inc*., 492 F.3d 1326, 1332 (Fed. Cir. 2007) (holding that claim scope is not limited to the disclosed embodiments "unless the patentee has demonstrated a clear intention to [do so]").

Despite related language cited from the written description that purportedly confirms the "computational infeasibility" of the secure has function, the Court finds that the patentee failed to identify Defendants' proposed requirement when defining the invention or otherwise negotiating with the PTO during the prosecution history. Accordingly, the definition provided for these claim terms will not include the "computationally infeasible" restriction, but will encompass the underlying substantive teachings as presented in the patents.

3. Altering the proposal that on average, approximately 50 percent of the secure hash output bits are changed when only one single bit is changed

Lastly, Defendants propose the written description requires that if a single bit of the input is changed, on average approximately 50% of the output bits are changed. RESPONSE at 11. PACid, however, maintains that this improperly imports a limitation from the specification in to the claims and unnecessarily complicates the construction of the "secure hash" terms. REPLY at 5. As a preliminary matter, this limitation, unlike the "computationally infeasible" language, finds support in the intrinsic record. At the hearing both parties cited to the Background of the Invention section, stating: "if only one bit in a message or file is changed, approximately 50% of the bits in the output change." '646 patent at 2:2–8;'612 patent at 2:13–15. Similarly, the '646 specification states that, "[t]he irreversibility of the encryption key was made even more difficult by using a secure has algorithm, which has the property of changing on average approximately 50 percent of its output bits when only a single bit in the input is changed." '646 patent at 6:5–9. In general, the effect of Defendants' argument and proposal regarding these specification passages could confuse the jury into applying an approximate floor and ceiling to the bits that may change as a result of the claimed hash operations. The Court finds that the application of such a floor or ceiling is at odds with the specification and the statistical underpinning of a secure hash operation.

Looking first at the cited passages, it is determined that, in context, neither passage imposes a strict floor or ceiling to the percentage of bit changes that may occur in a secure hash operation. Rather, both passages were reciting a desirable characteristic of secure hash function and an apparently true characteristic of the preferred embodiment SHA algorithm. For example, in both instances, the SHA algorithm is identified within just a few lines of the passage. Furthermore, even referring to the SHA algorithm or strict avalanche criteria cited by Defendants, there may only be a ½ probability that each bit will change. *See* RESPONSE at 10 (citing ALFRED J. MENEZES, PAUL C. VAN OORSCHOT & SCOTT A. VANSTONE, HANDBOOK OF APPLIED CRYPTOGRAPHY 277 (1997) (defining "strict avalanche criterion" as "whenever one input bit is changed, every output bit must change with probability of 1/2")). Since there is simply a statistical ½ chance of each bit changing, on rare occasions, far greater than 50% of the bits will change. Thus, if the jury misunderstood that any strict ceiling or floor applied, then the preferred embodiment would fall outside the claims in certain instances.

Of course, while the fact finder may be confused into applying a floor or ceiling, Defendants do not technically argue for literal bit-change limits, but rather propose a statistical average limit of approximately 50%. A general notion of an average 50% change is indeed supported in the intrinsic evidence quoted above. Defendants further cite extrinsic references to bolster and enforce the necessity of its proposed 50% average limit. In particular, Defendants point to extrinsic definition for the term "strict avalanche theory." However, Defendants do not sufficiently support that strict avalanche theory applies to all secure hash algorithms such as the non-SHA algorithms listed in the specifications. *See* '646 patent at 5:30–43; '612 patent at 5:5–16. Nevertheless, the Court is persuaded that the specifications reference to a 50% average bears on the definition of the secure hash terms at least to require the "dramatic change" to the input, as referenced in the specifications. *See* '646 patent at 5:21–24; '612 patent at 4:64–67. Further, the specifications' repetitive recitation of "dramatic change" supports Defendants' proposal with respect to the notion that a high percentage of bits must change on average. Thus, the 50% language should not be completely ignored, but should be applied in a way that respects the specifications' teachings regarding dramatic change and does not set a strict ceiling.

Finally, at the hearing, Plaintiff suggested that, if the Court were to adopt the 50% principle, then there would be agreement in applying the words "at least," thus implying a floor on average, but not a ceiling. Defendants agreed that they had proposed that language. *See* TRANSCRIPT at 60–61. Accordingly, the Court provides the following constructions:

- The term "secure hash operation" means "a deterministic operation that produces a fixed output bit length regardless of input bit length such that it is practically impossible to determine (a) the input from the output, and (b) two inputs that produce the same output and where if a single bit of the input is changed, on average at least approximately 50% of the output bits are changed."

- The term "secure hash algorithm" means "an algorithm that implements a secure hash operation."
- The term "secure has computer program" means "a computer program that implements a secure hash operation."

V. "performing a secure hash operation on said shuffled bit result to produce a message digest"[19]

Plaintiff's Proposed Construction	Defendants' Proposed Construction
No separate construction (in light of the others)	"the input to the secure hash operation is the shuffled bit result from step (a), and the output of the secure hash operation is a message digest"

The parties dispute the meaning of the term "performing a secure hash operation on said shuffled bit result to produce a message digest" in claim 1 of the '646 patent. PACid proposes that no separate construction is necessary for the phrase because it has a plain meaning in light of the construction provided for "secure hash operation" and "shuffled-bit result." REPLY at 6. Defendants contend that an actual construction is needed to reflect that the claims require that the secure hash operation is "performed on" the "shuffled-bit result" and "pseudo-random result," respectively. RESPONSE at 12. Defendants further clarify that "these results therefore are inputs to the secure hash operation. The thing 'produce[d]— the output— is the message digest and second pseudo-random result." *Id*.

[19] The term "performing a secure hash operation on said shuffled bit result to produce a message digest" is contained in claim 1 of the '646 patent.

The Court finds that the term "performing a secure hash operation on said shuffled bit result to produce a message digest" does not need additional construction in light of the other constructions provided herein, but offers additional clarification as to understanding this term in light of the Figures provided in the '646 patent specification. The downward arrow in Figure 2 of the '646 patent teaches that the shuffled-bit result is the input used for the secure hash operation, and the second arrow indicates that the output of the secure hash operation is the message digest. '646 patent, Fig. 2; *see also* '612 patent Fig. 3; '612 patent at 1:67–2:18; 3:21–24; 4:57–67; 6:55–59; and 7:20–22. Similarly, Figure 6 of the '646 illustrates this same relationship through a logic flow diagram where (1) the output of the shuffle-bit generator is the input into the bit-shuffling generator, and (2) the output of the secure hash generator is the message digest. '646 patent at Fig. 6.

VI. "performing a secure hash operation on said first pseudo-random result to . . . produce a second pseudo-random result"[20]

Plaintiff's Proposed Construction	**Defendants' Proposed Construction**
No separate construction (in light of the others); alternatively, "performing an algorithm that produces a deterministic output having no known relationship with the input that may be used to recover the input from the output"	"the input to the secure hash operation . . . is the first pseudo-random result from step (a) and the output of the secure hash operation is a second pseudo-random result"

The parties dispute the meaning of the term "performing a secure hash operation on said first pseudo-random result to. . . produce a second pseudo-random result" in claim 1 of the '612 patent. The arguments are largely the same as those presented in the previous term. Again the Court does not offer additional constructions in light of other constructions provided herein, but clarifies that

[20] The term "performing a secure hash operation on said first pseudo-random result to. . . produce a second pseudorandom result" is contained in claim 1 of the '612 patent.

the ‘612 patent specification teachings. The specification presents that the output of the shuffle-bit generator is the input to the secure hash function, and that the output of the secure hash function is a “second many-to-few bit mapping” called “pseudo-random message digest.” *See* ‘612 patent at 3:18–23. Figure 3 of the ‘612 patent provides a flow diagram illustrating the input and output of the secure hash function. ‘612 patent, Fig. 3.

VII. “algebraic function”[21]

Plaintiff’s Proposed Construction	**Defendants’ Proposed Construction**
No construction necessary; alternatively, “any operation used in mathematics”	“any operation used in mathematics or logic”

The patents-in-suit both contain the claim term “algebraic function.” The parties generally agree that a construction for this term should include “any operation in mathematics,” but dispute whether the claim term includes “logic.” PACid argues for a narrow claim scope under a theory of claim differentiation. OPENING at 9. PACid contends that an algebraic function is separate from a logic function because they are discussed separately in the claims and the specification for the patents-in-suit. *Id*. Defendants’ contend that the patentee gave the claim term an explicit definition and the patent specifically explains that “algebraic function” “is to be understood” to include logic functions. RESPONSE at 23–24.

Starting with the language of the claims themselves, there is nothing that points towards a restrictive reading that would exclude logic, and the claims of the patent will not be read restrictively unless the patentee has demonstrated a clear intention to limit the claim scope using “words or

[21] The term “algebraic function” is contained in claims 3, 13, and 16 of the ‘646 patent and claims 1 and 3 of the ‘612 patent.

expressions of manifest exclusion or restriction." *Teleflex, Inc. v. Ficosa N. Am. Corp.*, 299 F.3d 1313, 1327 (Fed. Cir. 2002). In all five claims where this term appears, a plain reading does not present language excluding logic functions. In particular, while the claims occasionally juxtapose the terms "logic function" and "algebraic function," there is no claim usage requiring that these terms refer to mutually exclusive realms in math. In every claim instance, logic functions may be partially or entirely subsumed in algebraic functions. The same is true with respect to the specification where a form of the term "algebra" is repetitively used to describe an operation of the bit shuffling generator (Item 52, Figure 2). '646 patent at 3:22–26; 3:38–48; 3:49–53; 4:60–65; 5:44–50; and 8:52–55. Further, like the claims, while some of the specification references juxtapose the two terms, none of the references serve to re-define or constrain the term "algebra" to a realm of mathematics that would not include logic. One definitional specification reference, in fact, provides an express inclusion of logic within the realm of algebra:

> It is to be understood that *the algebraic function executed by the function generator* 52, where two inputs. . . are subjected to a bit-shuffling mapping. . . can be any of numerous *other logic, cryptographic, or algebraic functions* that would protect the E-Key Seed from being discovered.

'646 patent at 5:44–50 (emphasis added). Thus, the specification states that the "algebra" performed by bit shuffling generator 52 is not confined to the XOR logic operation used in the example embodiment, '646 patent at 4:60–65, but may include "any of numerous other logic, cryptographic, or algebraic functions." This is further consistent with a commonly accepted dictionary meaning that is inclusive of the logic functions associated with "Boolean algebra." *See* CHAMBERS 21ST CENTURY DICTIONARY 805 (1996); WEBSTER'S NEW WORLD DICTIONARY OF COMPUTER TERMS 298 (6th ed. 1997). Therefore, the patentee is entitled to the full scope of the claim language, and based on this record, the claims properly include "logic." In accepting the teaching of the '646 patent specification, the Court also rejects PACid's arguments that "algebraic function" is differentiated in the '612 patent. PACid maintains that based on claims 1 and 4 of the '612 patent, if the claim term included logic functions, it would "render claim 4 superfluous" because the dependent claim separately claims "algebraic, logic, and cryptographic functions." OPENING at 9; REPLY at 9. Contrary to this assertion, however, claim 4 does not alter the broad definition of "algebra." While claim 1 refers to "algebraic function" without limitation and is later narrowed by the step disclosed in dependent claim 4, nothing about claims 1 and 4 indicates that the term "algebraic function" would not remain inclusive of Boolean algebra. Even with the added step of "combining a constant value and a secret plural bit sequence in accordance with an algebraic function," a person of ordinary skill in the art would still read "algebra" to include Boolean concepts based in logic. *Phillips,* 415 F.3d at 1312 (Claim terms are "generally given their ordinary and customary meaning," the meaning that the term would have to "a person of ordinary skill in the art. . . at the time of the invention.").

Accordingly, the Court finds that the proper construction of the term "algebraic function" is "any operation used in mathematics or logic."

VIII. "host system"[22]

Plaintiff's Proposed Construction	**Defendants' Proposed Construction**
No construction necessary; alternatively, "a system for providing command sequences"	"computer that inputs command sequences to an encryption key generator"

[22] The term "host system" is contained in claim 12 of the '646 patent.

The parties dispute the meaning of the term "host system" in claim 12 of the '646 patent. The parties first dispute whether a "host system" is a computer. The parties agree that a host system provides command sequences, but they further dispute whether the construction should specify the recipient of the "command sequence" produced by the host system.

PACid contends that claim 12 offers a broad meaning for "host system" that identifies the "I/O interface" as receiving the command sequence. REPLY at 7. PACid disputes any attempt to limit the definition of "host" to a computer, arguing at the *Markman* hearing that one of skill in the art would know that the encryption key generator could be used in "any system needing an encryption key," (e.g., server, state machine, computer, smart phone, printer, ATM, kiosk, or access point).

Defendants respond that claim 12 requires a "host system" to identify the recipient of the command sequences and further argues that the patent is directed towards protecting information stored on a *computer system*. Referencing intrinsic and extrinsic evidence, Defendants contend that a person of ordinary skill in the art would read "host" to mean "a computer on a network." RESPONSE at 19 (citing '646 patent at 1:41–42).

First, the plain language of claim 12 does not clarify the underlying meaning of "host," but the term's use throughout the patent as a whole supports Defendants' position that "host" is referring to a computer. The Federal Circuit has clearly stated that the written description "can provide guidance as to the meaning of the claims, thereby dictating the manner in which the claims are to be construed, even if the guidance is not provided in explicit definitional format." *SciMed Life Sys., Inc. v. Advanced Cardiovascular Sys., Inc.*, 242 F.3d 1337, 1344 (Fed. Cir. 2001). Thus, where as here, the patentee uses a claim term throughout the entire patent specification, in a manner consistent with only a single meaning, he has defined that term "by implication." *Vitronics*, 90 F.3d at 1582;

see also Hockerson-Halberstadt, Inc. v. Avia Group Intern., Inc., 222 F.3d 951, 955 (Fed. Cir. 2000). A definition by implication is provided for the claim term "host" in the context of the '646 and '612 patents. In particular, the Backgrounds of the Invention in each patent support a context in which "host" is used to explain the intricacies of a computer.

The '646 specification states that the patented technology protects information stored on a computer system or communicated over networks, '646 patent at 1:41–42, and the '612 specification particularly mentions that "with the introduction of the personal computer (PC), a migration to local computing through the use of centralized host/server systems began. Again, the conventional wisdom was that sensitive information could be protected by guarding against unauthorized access to the host/server system." '612 patent at 1:30–35. The '646 patent then links "host system" to a "computing system" that truncates the message digest and ultimately produces a symmetric and deterministic encryption key. '646 patent at 8:39–44. These passages all evidence an intrinsic representation of "host system" that aligns with a broad understanding of "computer." Since the patents-in-suit do not provide a special definition for the computer functioning as the "host" in claim 12, the Court sets forth its understanding of "computer" in light of the intrinsic record, as well as analysis provided by other courts as to a general definition of "computer." Claim 12 discloses that the I/O interface receives command sequences from the host system. Therefore, the patentee's instructions as to command sequences should be included in the construction of this term. Where the command sequences go, however, seems to be an ancillary to question of what is understood to be a computing system.

Defendants request that "host" be defined according to a dictionary definition that equates "host" to "the main computer in a system of computers" or "a computer containing data or programs that another computer can access by means of a modem or network." RESPONSE at 19 (citing AM. HERITAGE DICTIONARY 849 (4th ed. 2000) & MICROSOFT PRESS COMPUTER DICTIONARY 201 (2d ed. 1994)). Nonetheless, the meaning of computer to one of ordinary skill in the art is a recurring issue in patent infringement litigation, and therefore, it is useful to consider what other courts have emphasized in understanding computer functionality. The Federal Circuit has instructed that this claim term, like any other, should first be construed according to intrinsic evidence, but where such a record is not conclusive, the court understood "computer" to encompass "peripherals that are within a reasonable proximity to the CPU and its main memory and directly connected to the CPU or the CPU circuit board." *Pickholtz v. Rainbow Techs*., 284 F.3d 1365, 1374 (Fed. Cir. 2002). In *Pickholtz,* the Federal Circuit implicitly disagreed with PACid's understanding of "host" or "computer" as being the broadest possible system, such as "any system needing an encryption key." Judge Lourie specifically noted, "[T]he term 'computer' cannot be so unbounded as to include all devices connected in any way to the CPU." *Pickholtz,* 284 F.3d at 1374.

Without a precise definition for "host system," in the '646 and '612 patents, the Court finds that the intrinsic record supports the implication that "host system" is synonymous with "computer." Extrinsic evidence is necessary to clarify the scope of the term and the Court adopts the dictionary definition of computer used previously in this district. *See Sovereign Software LLC v. Amazon.com Inc*., No. 6:04-CV-14, 2005 WL 6225276, at *9 (E.D. Tex. Apr. 7, 2005). Judge Davis consulted a technical dictionary to broadly define "computer" to mean "a functional unit that can perform substantial computation, including numerous arithmetic operations, or logic operations without human intervention." *Id*. (relying on IEEE STANDARD DICTIONARY OF ELEC. & ELECS. TERMS 192 (6th ed. 1996)). This understanding of "computer" is incorporated in the Court's construction of "host system" in this case. Accordingly, the disputed claim term is defined as "a computer that provides command sequences through an I/O interface."

IX. "information file" or "message file"[23]

Plaintiff's Proposed Construction	Defendants' Proposed Construction
"message or file"	"a collection of information stored as a unit and identified by a unique name"

The parties dispute the meaning of the term "information file" in claim 1 of the '612 patent. The parties generally dispute the meaning of "file" as it is used in the '612 patent specification: "a method and system for protecting an **information file** from unauthorized access, and more specifically to the encryption of a message or file. . ." '612 patent at 1:8–10 (emphasis added). The same passage additionally states that "[a] method and system is disclosed for protecting sensitive **information files** and messages from access by unauthorized parties. . . ." '612 patent at 3:12–14 (emphasis added).

PACid contends that this specification language supports a generalized definition of "information file." PACid argues that the limitations imposed in Defendants' proposed construction find no support in the specification and narrows the claim term to a meaning that is inconsistent with what would be understood in the context of the patents-in-suit and the relevant art. OPENING at 16.

Defendants respond that the '612 patent specification "consistently distinguishes" between an "information file" and a "message" by referring to the two terms in conjunctive form. RESPONSE at 20. Defendants further contend that "message" should not be a part of the construction because during the prosecution history, the inventor amended claim 1 to differentiate between "information

[23] The term "information file" is contained in claim 1 of the '612 patent.

file" and "message." *Id*. (citing RESPONSE, EXH. 10, '612 PROS. HIST. (12/28/98 AMENDMENT)). Arguing that this claim term was not provided with a "special definition" in the patent, Defendants suggest that a contemporaneous dictionary definition should be used to understand the meaning of "file." *Id*. at 21.

As the claim term is used in the specification, the patentee did not ascribe any special meaning to "information file," nor is there any explicit disclaimer in the file history that limits the meaning to the specific requirements suggested by Defendants. *Cordis Corp. v. Boston Scientific Corp.*, 561 F.3d 1319, 1329 (Fed. Cir. 2009) (holding that unclear prosecution history cannot be used to limit the plain language of the claims). Defendants must overcome the "heavy presumption" in favor of giving the claim terms their ordinary meaning, and the patentee's intention, as expressed in the '612 specification, fails to confine an "information file" to a collection of information with "a unique name." *Phillips*, 415 F.3d at 1312–13, 1323; *Liebel-Flarsheim*, 358 F.3d at 913 (finding that claim terms must be given their ordinary meaning unless the claim language is overcome by statements of "clear disclaimer" expressly indicating "manifest exclusion or restriction"). To the contrary, the requirement of a "unique name" is wholly absent from the intrinsic record, and therefore, the patentee is entitled to the full scope of the claim language. *Home Diagnostics*, 381 F.3d at 1358.

To that end, when determining the scope of the claims, there is indication that the '612 patent uses "file" in a broad sense, but PACid's proposed construction does not define the claim term with sufficient particularity to be readily applied by a jury. For the sake of clarity, the dual terms "information file" and "message" are defined separately to reflect differences in the type of data being accessed, as well as the conjunctive form in which they appear in the specification.

Accordingly, in the absence of an intrinsic definition for these terms, the Court adapts the dictionary definition provided by Defendants to include the ordinary meaning that a file may be accessed and manipulated as a single unit. This definition determines the ordinary meaning of the terms according to how one of ordinary skill in the art would read the claim language, but without the unsupported limitation that the data be "identified by a unique name."

The proper construction for "information file" is "an organized collection of information that can be accessed and manipulated as a single named unit." Similarly, the proper construction for "message file" is "a message that can be accessed and manipulated as a single named unit."

X. "concatenating"[24]

Plaintiff's Proposed Construction	Defendants' Proposed Construction
No construction necessary; alternatively, "linking units together"	"placing one bit field directly next to another"

The parties dispute the meaning of the term "concatenating" in claim 1 of the '612 patent. The '612 patent specification explicitly defines "concatenation" to mean "that one bit field is juxtaposed to another." '612 patent at 4:6–7. Based on the definition provided in the specification, the parties generally agree that the claim term requires that the bit fields be linked together, the parties dispute, however, whether "concatenating" requires that the items be directly adjacent.

PACid contends that this term has a clear meaning, but if construed, should reflect the full scope of the claim term as it is commonly used in computing. OPENING at 18. PACid maintains that the Defendants' proposed construction improperly limits the claim to the manner it is used in a preferred embodiment. *Id.*

[24] The term "concatenating" is contained in claim 1 of the '612 patent.

Defendants offer that in requiring that one bit field be placed "directly next to" another one, they are proposing a construction that comports with the patent's explicit definition. RESPONSE at 22. Referring to a dictionary definition, Defendants explain that "juxtapose" means "to place things side by side." *Id*. (citing CHAMBERS 21ST CENTURY DICTIONARY 737 (1996)). Defendants further cite language in the specification that implicitly describes the constant value being placed next to the encrypted information file. *Id*. (citing '612 patent at 2:67–3:3; 6:61–64).

For this term, the express definition provided in the specification unquestionably governs. *Phillips*, 415 F.3d at 1316 (explaining that in instances where the inventor has clearly defined his own terms, the inventor's lexicography will govern during claim construction); *CCS Fitness Inc. v. Brunswick Corp*., 288 F.3d 1359, 1366 (Fed. Cir. 2002). Identifying the inventor's definition for "concatenating" to be "one bit field juxtaposed to another," the remaining issue is whether the accepted meaning of "juxtapose" requires the bit units to be directly next to one another.

Defendants identify language in the specification describing the constant value being concatenated "at the beginning of the encrypted information file," and thus conclude that "the constant value must be *directly next to* the encrypted information file." RESPONSE at 23 (citing '612 patent at 6:61–64 (emphasis added)). This conclusion, however, restricts the scope of the claim to the manner in which it is used in a preferred embodiment. Even where a patent describes only a single embodiment, claims will not be read restrictively unless the patentee has demonstrated a clear intention to limit claim scope." *Innova/Pure Water*, 381 F.3d at 1117 (internal citation omitted). Failing to identify any language of manifest exclusion, the Court finds Defendants' restrictive reading of claim 1 to be incorrect.

The patentee does not offer a clear meaning for “juxtapose,” therefore the specification term will be given its ordinary meaning as established through extrinsic evidence. *See Phillips*, 415 F.3d at 1318 (“We have especially noted the help that technical dictionaries may provide to a court “to better understand the underlying technology” and the way in which one of skill in the art might use the claim terms.”) (citing *Vitronics*, 90 F.3d at 1584, n.6). Defendants offer a dictionary definition that understands “juxtapose” to mean “side by side,” but even if afforded this definition, common use of either “juxtapose” or “concatenate” does not preclude intervening bit fields. Instead, “concatenating” is understood broadly to account for bit fields that are “side by side,” but not necessarily directly next to each other in a group of side by side bit fields.[25] Incorporating this understanding of “juxtaposed,” the proper construction of the term “concatenating” is “placing one bit field side-by-side with another.” The Court notes, however, that side by side does not require that such bit fields be directly next to one another.

CONCLUSION

For all the foregoing reasons, the Court construes the disputed claim language in this case in the manner set forth above. For the ease of reference, the Court’s claim interpretations are set forth in a table attached to this Memorandum Opinion and Order as an Appendix.

So ORDERED and SIGNED this 15th day of July, 2010.

John D Love

JOHN D. LOVE
UNITED STATES MAGISTRATE JUDGE

[25] This concept is fully illustrated through an example. In other words, if “1,” “2,” and “3” are “concatenated” as “123,” then “1” and “3” are still “juxtaposed” within the group of numbers.

IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
TYLER DIVISION

THE PACID GROUP, LLC,	**§**	
	§	
Plaintiff,	**§**	
	§	
v.	**§**	**CIVIL ACTION No. 6:09cv143-LED-JDL**
	§	
APPLE, INC., ET AL.,	**§**	
	§	
Defendants.	**§**	

APPENDIX

U.S. PATENT Nos. 5,963,646 and 6,049,612

Claim Language	Patent	Claims	Plaintiff's Proposed Construction	Defendants' Proposed Construction	Court's Construction
"pseudo-random"	'646 patent '612 patent	claims 1, 12, 17, and 26 claim 1	No construction necessary; alternatively, "apparently random, but repeatable and predictable"	"refers to output that is repeatable and predictable to anyone who knows the function's input but appears to be totally random to those without such knowledge"	"refers to output that is repeatable and predictable to anyone who knows the function's input but appears to be totally random to those without such knowledge"
"constant value"	'646 patent '612 patent	claims 1, 6, and 12–16 claims 1 and 7	No construction necessary; alternatively, "a value that does not change for any given instance of generating an encryption key"	"a value that does not change"	"a value that does not change"
"shuffled bit result"	'646 patent	claims 1, 3, and 18	"the result of an operation that mixes the bits of its inputs"	"the result of an operation that mixes and maps the bits of its inputs"	"the result of an operation that mixes the bits of its inputs"
"bit shuffling operations"	'646 patent	claim 18	No separate construction (in light of the others); alternatively, "operations that mix the bits of inputs"	"operations that mixes and maps the bits of their inputs"	"operations that mix the bits of inputs"

Claim Language	Patent	Claims	Plaintiff's Proposed Construction	Defendants' Proposed Construction	Court's Construction
"bit shuffling function"	'646 patent	claim 19	No separate construction (in light of the others); alternatively, "a function that mixes the bits of its inputs"	"a function that mixes and maps the bits of its inputs"	"a function that mixes the bits of its inputs"
"function to shuffle bits"	'612 patent	claim 1	No separate construction (in light of the others); alternatively, "a function that mixes the bits of its inputs"	"a function that mixes and maps the bits of its inputs"	"a function that mixes the bits of its inputs"
"bit shuffle computer program"	'646 patent	claim 12	No separate construction (in light of the others); alternatively, "a computer program that mixes the bits of its inputs"	"computer program that performs a bit shuffle operation"	"a computer program that mixes the bits of its inputs"
"secure hash operation"	'646 patent '612 patent	claim 1 claim 1	"algorithm that produces a deterministic output having no known relationship with the input that may be used to recover the input from the output"	"an operation that accepts an input that can be of variable bit length, but always produces an output having the same bit length such that it is computationally infeasible to determine (a) the input from the output and (b) two inputs that produce the same output, and where if a single bit of the input is changed, on average approximately 50% of the output bits are changed"	"a deterministic operation that produces a fixed output bit length regardless of input bit length such that it is practically impossible to determine (a) the input from the output, and (b) two inputs that produce the same output and where if a single bit of the input is changed, on average at least approximately 50% of the output bits are changed"

Claim Language	Patent	Claims	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction	Court’s Construction
“secure hash algorithm”	‘646 patent	claim 12	“algorithm that produces a deterministic output having no known relationship with the input that may be used to recover the input from the output”	“an algorithm that accepts an input that can be of variable bit length, but always produces an output having the same bit length such that it is computationally infeasible to determine (a) the input from the output and (b) two inputs that produce the same output, and where if a single bit of the input is changed, on average approximately 50% of the output bits are changed”	“an algorithm that implements a secure hash operation”
“secure hash computer program”	‘646 patent	claim 12	No separate construction (in light of the others); alternatively, “a computer program that produces a deterministic output having no known relationship with the input that may be used to recover the input from the output”	“computer program that uses a secure hash algorithm” (as defined above)	“a computer program that implements a secure hash operation”
“performing a secure hash operation on said shuffled bit result to produce a message digest”	‘646 patent	claim 1	No separate construction (in light of the others)	“the input to the secure hash operation is the shuffled bit result from step (a), and the output of the secure hash operation is a message digest”	No separate construction.

Claim Language	Patent	Claims	Plaintiff's Proposed Construction	Defendants' Proposed Construction	Court's Construction
"performing a secure hash operation on said first pseudo-random result to . . . produce a second pseudo-random result"	'612 patent	claim 1	No separate construction (in light of the others); alternatively, "performing an algorithm that produces a deterministic output having no known relationship with the input that may be used to recover the input from the output"	"the input to the secure hash operation . . . is the first pseudo-random result from step (a) and the output of the secure hash operation is a second pseudo-random result"	No separate construction.
"algebraic function"	'646 patent '612 patent	claims 3, 13, and 16 claims 1 and 3	No construction necessary; alternatively, "any operation used in mathematics"	"any operation used in mathematics or logic"	"any operation used in mathematics or logic"
"host system"	'646 patent	claim 12	No construction necessary; alternatively, "a system for providing command sequences"	"computer that inputs command sequences to an encryption key generator"	"a computer that provides command sequences through an I/O interface"
"logic function"	'646 patent '612 patent	claims 14 and 16 claim 4	No construction necessary; alternatively, "a function involving operations on variables that may only take a finite number of possible values or states"	"a function that involves yes-no decisions"	"a function involving operations on variables that may only take a finite number of possible values or states"
"cryptographic function"	'646 patent '612 patent	claim 15 claim 5	No construction necessary; alternatively, "a function used in encoding or decoding"	"a function used in encryption or decryption"	"a function used in encoding and decoding, including encryption or decryption."

Claim Language	Patent	Claims	Plaintiff's Proposed Construction	Defendants' Proposed Construction	Court's Construction
"information file" or "message file"	'612 patent	claim 1	"message or file"	"a collection of information stored as a unit and identified by a unique name"	Information file means "an organized collection of information that can be accessed and manipulated as a single named unit" Message file means "a message that can be accessed and manipulated as a single named unit"
"concatenating"	'612 patent	claim 1	No construction necessary; alternatively, "linking units together"	"placing one bit field directly next to another"	"placing one bit field side-by-side with another"
"interrupt control means"	'646 patent	claims 12 and 26	Section 112(6) does not apply and no construction is necessary; alternatively: "hardware or software that issues a signal to interrupt the operation of a processor" 112(6) Function: issuing an interrupt signal upon receipt of command sequences Corresponding Structure: interrupt control unit 104	Section 112(6) applies. 112(6) Function: issuing an interrupt signal upon receipt of said command sequences Corresponding Structure: None; claim is indefinite	Section 112(6) applies. 112(6) Function: issuing an interrupt signal upon receipt of command sequences Corresponding Structure: interrupt control unit 104