UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

THE HILLMAN GROUP, INC.,

Petitioner,

v.

HY-KO PRODUCTS COMPANY LLC,

Patent Owner.

Case No. IPR2023-00076
U.S. Patent No. 11,227,455

DECLARATION OF DR. PAUL C. CLARK

I. INTRODUCTION 1

II. QUALIFICATIONS AND BACKGROUND 1

III. MATERIALS CONSIDERED 5

IV. LEGAL STANDARDS 7

A. Claim Construction 8

B. Anticipation 8

C. Obviousness 9

V. THE '455 PATENT 10

A. Effective Filing Date of the Patent 10

B. Disclosure and Claims of the '455 Patent 11

C. Prosecution History of the '455 Patent 14

VI. LEVEL OF SKILL IN THE ART 14

VII. CLAIM CONSTRUCTION 16

VIII. THE PRIOR ART DISCLOSES OR RENDERS OBVIOUS CLAIMS 1-6 OF THE '455 PATENT 16

A. Prior Art to the '455 Patent 17

1. Chies 17

2. Marsh 19

B. Grounds 1 and 2: Chies Anticipates and/or Renders Obvious Claims 1 and 3. 21

1. Independent Claim 1 21

i. 1(pre): A method for duplicating an access device, comprising: 21

ii. 1(a): providing a first computing device configured to communicate with a

transmitter/receiver device (t/r device), the t/r device including an antenna for wirelessly communicating with a master key for a vehicle; 22

iii. 1(b): running a cloning application associated with said t/r device on the first computing device, the cloning application configured to electronically communicate with the t/r device; 24

iv. 1(c): retrieving, by the antenna of the t/r device, security information digitally stored by said master key; 25

v. 1(d): communicating, by the cloning application, the security information for said master key to said network server; 27

vi. 1(e): generating, by the network server, a cloning security information; 29

vii. 1(f): communicating, by the network server, the cloning security information to the cloning application; and 30

viii. 1(g): transmitting, by the cloning application, the cloning security information to the t/r device to program an electronic access device of a generally programmable duplicate key with the cloning security information. 34

2. Claim 3: The method of claim 1, further comprising: in response to communicating, by the cloning application, the security information for said master key to said network server, determining, by the network server, whether the security information is encrypted; and generating the cloning security information. 37

C. Ground 3: Chies and Marsh Render Claims 2-6 Obvious. 39

1. Claim 2: 39

a. 2(pre): "The method of claim 1, further comprising:" 39

b. 2(a): "in response to retrieving, by the t/r device, security information from said master key," 40

c. 2(b): "determining, by the cloning application, whether the security information is encrypted and" 40

d. 2(c): "generating the cloning security information" 44

2. Claim 3: The method of claim 1, further comprising: in response to communicating, by the cloning application, the security information for said master key to said network server, determining, by the network server, whether the security information is encrypted; and generating the cloning security information. 48

3. Claim 4: 51

4(pre) The method of claim 1, further comprising: 51

4(a) capturing at least one image of a blade of said master key, ... 51

4(a) communicating said at least one image of said blade to said network server, ... 54

4(c) cutting a duplicate blade of a generally programmable duplicate key, and 56

4(d) providing said generally programmable duplicate key to said customer. 57

4. Claim 5: The method of claim 1, further comprising: placing said generally programmable duplicate key in proximity with said vehicle, sampling communications between said generically programmable duplicate key and said vehicle to generate authentication sample data. 59

5. Independent Claim 6 61

i. 6(pre): A vehicle access device duplication system comprising:.....61

ii. 6(a): a non-transitory computer-readable storage medium storing executing instructions that, when executed, cause a cloning application to perform steps comprising:.....62

iii. 6(b): determining that a first device of a customer is communicating with a transmitter/receiver device (t/r device), the t/r device including an antenna ring for wirelessly communicating with an access device for a vehicle; 64

iv. 6(c): retrieving security information digitally stored by said access device;.....70

v. 6(d): communicating the security information for said access device to a network server;.....71

vi. 6(e): generating a cloning security information; and 73

vii. 6(f): transmitting the cloning security information to the t/r device to program a duplicate access device with the cloning security information; and 74

viii. 6(g): a processor configured to execute the instructions. 75

IX. Conclusion 78

I. INTRODUCTION

1. I, Paul C. Clark, submit this declaration to state my opinions on the matter described below.

2. I have been retained by Petitioner The Hillman Group, Inc. ("Hillman" or "Petitioner") as an independent expert in this proceeding before the United States Patent and Trademark Office. I am being compensated at my usual and customary hourly consulting rate, no part of my compensation depends on the outcome of this proceeding, and I have no other financial interest in the parties involved in this proceeding.

3. I understand that this proceeding involves U.S. Patent No. 11,227,455 (the "'455 patent"), and I have been asked to consider the patentability of the claims of the '455 patent in view of certain prior art references. I have also been asked to consider the state of the art and prior art available as of March 18, 2018. Based on the prior art discussed in this declaration, it is my opinion that claims 1-6 of the '455 patent are unpatentable in view of the prior art for the reasons provided below.

II. QUALIFICATIONS AND BACKGROUND

4. I believe that I am qualified to serve as a technical expert in this matter based upon my educational and professional experience, including my approximately 30 years of relevant educational and technical experience prior to

the earliest filing date of the ’455 patent in the fields of electrical engineering, computer science, data security/cryptography, and information systems. My curriculum vitae is attached to this declaration as Appendix A.

5. I received a Bachelor of Science in Mathematics from the University of California Irvine, a Master of Science in Electrical Engineering and Computer Science from the University of Southern California, and a Doctor of Science in Electrical Engineering and Computer Science from The George Washington University with concentrations in Computer and Network Security, Graphics and Intellectual Property Law. My doctoral dissertation and professional work for the last 35 years included advanced secure solutions to the distribution of key data.

6. From the late 1980s to the mid-1990s, I was a Senior Security Engineer at Trusted Information Systems. In that role, I participated in the design and implementation of several networked systems providing integrity, authentication, and encryption services.

7. Between January 1994 and July 1999, I was Chief Scientist for DynCorp Network Solutions. In that role, I designed and directed the implementation of several cryptographically enabled network systems including the IRS’s Secure Submission and Retrieval System. That secure system received three Al Gore Hammer Awards for important contributions to improving the U.S. government.

8. In the mid-1990s, I served as a member of the Federal Advisory Committee for Key Management Infrastructure and as Chairman of the Interoperability Working Group for Cryptographic Key Recovery.

9. Also in the mid-1990s, I served as a Cooperative Research and Development Agreements partner in a joint effort between the National Institute of Standards and Technology and several companies to begin deployment of the key elements of a public key infrastructure.

10. Since 1999, I have been President and Chief Technology Officer of SecureMethods, Inc. and Paul C. Clark LLC. SecureMethods specializes in the design, implementation, and deployment of advanced network applications for commercial and government clients, including the United States Department of Defense ("DoD"). SecureMethods provides a comprehensive scalable, COTS-based secure network architecture, implemented through the use of the SM Gateway. The SM Gateway is a next-generation security appliance developed by SecureMethods that is available on UNIX-based platforms using commercial, government, and Type I cryptography, implemented in both hardware and software. In my capacity as President and Chief Technology Officer of SecureMethods, I have technical and operational oversight of all projects and corporate technical operations. I provide guidance to senior technical personnel relating to design, implementation, and troubleshooting for a wide range of

systems, both internal and external. My work includes network systems and security, cryptographic applications, certification, key management, authentication, and integrity strategies for network applications. My firm specializes in complex software and hardware systems for commercial and DoD clients.

11. I have published several articles on network security and encryption, including "BITS – A Smartcard Protected Operating System," with Lance Hoffman, Communications of the ACM, November 1994; "Service Layering Promotes Secure Data Exchange in Diverse Environments," Computer News, October 23, 1995; and "Threats Posed to Cryptographic Applications by Random Numbers," presented to the RSA Data Security Conference, January 1996.

12. I have also given several presentations at technical conferences relating to security architecture, including wireless access and key distribution protocol topics relevant to the subject matter of the '455 patent.

13. I have also been called to provide expert testimony before Congress on issues related to encryption, authentication and secure wireless technology; have spoken at numerous conferences in this technical field; and regularly consult on topics related to the subject matter of the '455 patent long before its 2018 filing date.

14. My academic and professional backgrounds are closely related to the subject matter of the '455 patent, and I have extensive experience with methods of

computer security, including the management of cryptographic keys and associated data. I have served as an adjunct professor in the Electrical Engineering and Computer Science Department at The George Washington University, teaching doctoral-level cryptography and security courses. Since receiving my doctorate in 1994, I have worked in the computer and networking field specializing in the design, implementation, and deployment of advanced secure network applications for commercial, Department of Defense, and government clients.

15. I am the named inventor on four computer security-related U.S. patents:

- U.S. Pat. No. 5,448,045, titled "System for Protecting Computers via Intelligent Tokens or Smart Cards";
- U.S. Pat. No. 5,892,902, titled "Intelligent Token Protected System with Network Authentication";
- U.S. Pat. No. 8,695,066, titled "System and Method for Secure Communication Between Domains"; and
- U.S. Pat. No. 10,129,214, titled "System and Method for Secure Communication Between Domains."

III. MATERIALS CONSIDERED

16. In reaching the conclusions described in this declaration, I have relied on the documents and materials cited herein as well as those identified in this

declaration, including the '455 patent, the prosecution history of the '455 patent, and prior art references cited herein. These materials comprise patents, related documents, or printed publications. Each of these materials is a type of document that experts in my field would have reasonably relied upon when forming their opinions. In quoting the cited references, all emphasis is added unless otherwise noted.

17. I have also relied on my education, training, research, knowledge, and personal and professional experience in the relevant technologies and systems that were already in use prior to the timeframe of the earliest possible filing date of the '455 patent, which is March 18, 2018.

18. In forming my opinions, I have reviewed the following documents:

Exhibit	Description
Ex. 1001	U.S. Patent No. 11,227,455 to Mutch et al.
Ex. 1003	Prosecution History of U.S. Pat. No. 11,227,455
Ex. 1004	WO 2008/145199A1 to Chies et al. ("Chies")
Ex. 1005	U.S. Patent No. 9,563,885 to Marsh et al. ("Marsh")
Ex. 1006	U.S. Patent No. 10,013,833 to Blalock et al. ("Blalock")
Ex. 1007	U.S Pat. App. Pub. No. 2015/0050094 to Gerlings ("Gerlings")

IV. LEGAL STANDARDS

19. I am not a lawyer. My understanding of the legal standards to apply in this declaration is based on discussions with counsel for Petitioner, my experience applying similar standards in other patent-related matters, and my reading of the documents submitted in this proceeding. In preparing this declaration, I have tried to faithfully apply these standards to the challenged claims. My understanding of these concepts is summarized below.

20. I understand that the claims define the invention. I also understand that an unpatentability analysis is a two-step process. First, the claims of the challenged patent are construed to determine their meaning and scope. Second, once the claims have been construed, the content of the prior art is compared to the construed claims.

21. I understand there are two ways in which prior art may render a patent claim unpatentable. First, the prior art can "anticipate" the claim. Second, the prior art can make the claim "obvious" to a person of ordinary skill in the art ("POSITA"). I understand that for an invention claimed in a patent to be patentable, it must not be anticipated and must not be obvious based on what was known before the invention was made.

22. For this declaration, I have been asked to opine on issues regarding the technology at issue, the level of ordinary skill in the art at the time of invention,

and obviousness. I have been informed of the following standards, which I have applied in forming my opinions.

A. Claim Construction

23. I have been instructed that the terms appearing in the ’455 patent should be interpreted primarily in view of the claim language itself, the specification, and the prosecution history of the patent, and in some cases, in view of any relevant extrinsic evidence (e.g., dictionary definitions). The words of a claim are generally given their ordinary and customary meaning to a POSITA at the time of the invention, which I have been asked to assume here is March 18, 2018. While I understand that claim limitations cannot ordinarily be read into the claims from the specification, the specification is the single best guide to the meaning of a disputed term. I have followed these principles in reviewing the claims of the ’455 patent and forming the opinions set forth in this declaration.

B. Anticipation

24. I have been told that under 35 U.S.C. § 102, a patent claim is invalid if its subject matter was patented or described in a printed publication before the effective filing date of the claimed invention. I have been told that this is referred to as invalidity by anticipation. I have been told that a patent claim is anticipated under § 102 if a single prior art reference discloses all limitations of the claimed invention.

C. Obviousness

25. I understand that a claim may be unpatentable as obvious if the subject matter of the claim as a whole would have been obvious to a POSITA at the time the invention was made in view of any differences between the patented subject matter and the prior art. I further understand that the existence of each and every element of the claimed invention in the prior art does not necessarily prove obviousness and that most, if not all, inventions rely on building blocks of prior art. I have been advised that several factual inquiries underlie an obviousness determination, including (1) the scope and content of prior art; (2) the level of ordinary skill; (3) the differences between the claimed invention and the prior art; and (4) any objective evidence of nonobviousness.

26. I understand that a person of ordinary skill in the art is presumed to know all of the relevant prior art, and that an obviousness analysis may take into account the inferences and creative steps that a person of ordinary skill in the art would employ. I understand a prior art reference may be modified or combined with other references, or with the knowledge of a person of ordinary skill in the art, if the person of ordinary skill in the art would have found the modification or the combination obvious. Where a party alleges obviousness, I understand that party must identify a reason why a person skilled in the art would have been motivated to modify or combine references in the manner recited in the claims, and explain

why the person skilled in the art would have a reasonable expectation of success in making such modifications or combinations.

27. I understand that there is no rigid rule that a reference or combination of references must contain a "teaching, suggestion, or motivation" to combine the references. But I also understand such a "teaching, suggestion, or motivation" can be used in establishing a rationale for combining prior art elements. I understand that the law permits the application of "common sense" in examining whether a claimed invention would have been obvious to the person skilled in the art. For example, I understand combining familiar elements according to known methods in a predictable way suggests obviousness.

V. THE '455 PATENT

A. Effective Filing Date of the Patent

28. I understand that the application for the '455 patent was filed on March 18, 2019, and claims priority to a provisional patent application having a filing date of March 18, 2018. Ex. 1001 at 1. I have been asked to use March 18, 2018, as the "effective" filing date and priority date for purposes of my analysis, and I have done so, but I have not analyzed whether the claims of the '455 patent are supported by the provisional application filed on March 18, 2018, or whether any claim of the '455 patent is entitled to that priority date.

B. Disclosure and Claims of the '455 Patent

29. The '455 patent relates to "a distributed cloning system, and method for replacement, generation, or reprogramming of vehicle access devices, such as transponder keys or remotes." Ex. 1001, 1:18:21. The key duplication system uses a "transmitter/receiver device ('t/r device')" that can receive information from a vehicle key. Ex. 1001, 2:25-45. The t/r device may transmit the key information to a network server to generate "cloning security information" that the t/r device then transmits to a "duplicate access device" (i.e., blank transponder key) to program a duplicate key. *Id.*

30. Independent claim 1 of the '455 patent recites:

> 1. A method for duplicating an access device, comprising:
>
> providing a **first computing device** configured to communicate with a **transmitter/receiver device (t/r device)**, the **t/r device** including an **antenna** for wirelessly communicating with a **master key** for a vehicle;
>
> running a cloning application associated with said **t/r device** on the **first computing device**, the cloning application configured to electronically communicate with the **t/r device**;
>
> retrieving, by the **antenna** of the **t/r device**, security information digitally stored by said **master key**;

communicating, by the cloning application, the security information for said **master key** to said **network server**;

generating, by the **network server**, a cloning security information;

communicating, by the **network server**, the cloning security information to the cloning application; and

transmitting, by the cloning application, the cloning security information to the **t/r device** to program an electronic access device of a **generally programmable duplicate key** with the cloning security information.

Ex. 1001, 9:58-10:19.

31. As shown in annotated Figure 4 below, the '455 patent discloses a first computing device (computing device 100, **green**), a t/r device (t/r device 10, **blue**), an antenna of the t/r device (antenna 40, **orange**), a master key for a vehicle (master key 20, **purple**), a network server (network server 130, **red**), and a generally[1] programmable duplicate key (generically programmed duplicate electronic access device 22', **yellow**). Ex. 1001, 7:30-67.

[1] The '455 patent refers to the duplicate key of element 50 as a "generically programmable duplicate key," "generically programmed duplicate key," "generally programmed duplicate key," and generally programmable duplicate key." *See, e.g.*,

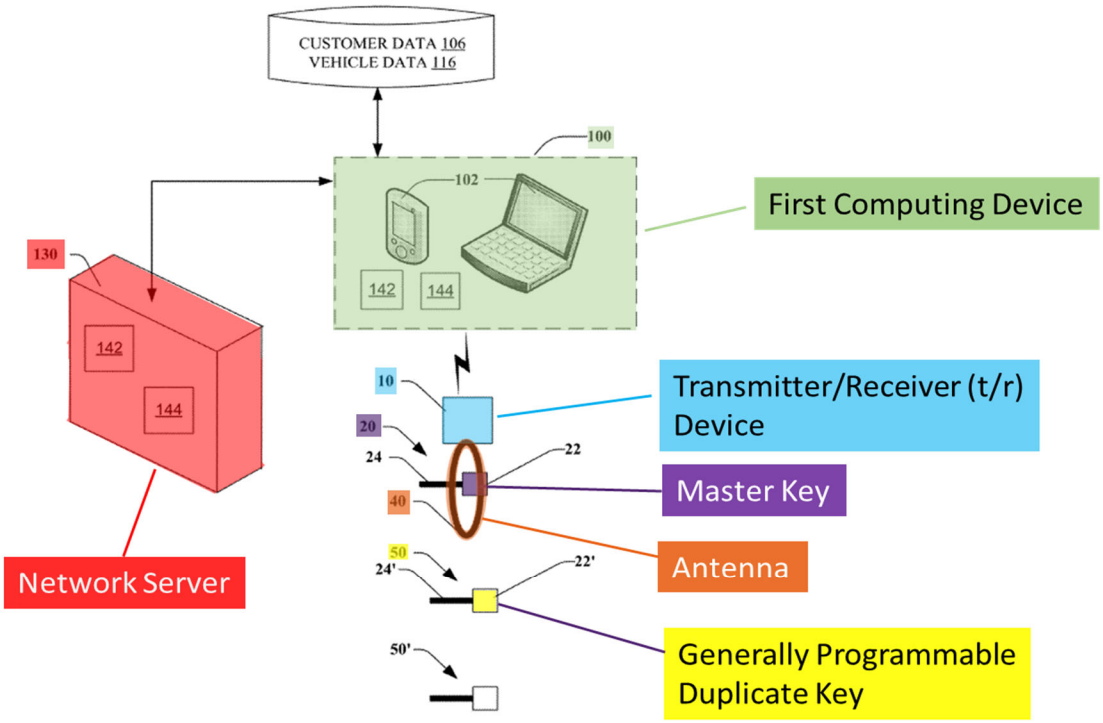


FIG. 4

32. The t/r device reads the security information on the master key and transmits the information to the network server via the cloning application of the computing device. Ex. 1001, 7:30-66. The network server then processes the security information from the master key, e.g. by decrypting the security information, and generates cloning security information that is transmitted to the

---

Ex. 1001, 6:7-16, 6:61-67, claim 1. For purposes of this declaration only, I treat these terms as interchangeable and as generally designating a programmable blank transponder key.

generally programmable duplicate key via the antenna of the t/r device to create a duplicate of the master key. *Id.*, 7:14-67.

C. Prosecution History of the '455 Patent

33. I have reviewed the prosecution history of the application that led to the '455 patent. Ex. 1003. During prosecution, the Examiner failed to find prior art disclosing the elements of independent claims 1 and 6, including the claimed "t/r device including an antenna ring for wirelessly communicating with an access device for a vehicle," as well as "retrieving security information digitally stored by said access device." Ex. 1003, 157 (emphasis in original).

34. I understand that the Examiner did not consider the prior art presented in this Declaration, including the Chies and Marsh references. As both of those references demonstrate, it was well known before the '455 patent was filed to duplicate a master key for a motor vehicle (e.g., a transponder key) by using wireless communications to retrieve information digitally stored on the master key and using that information to program a blank transponder key.

VI. LEVEL OF SKILL IN THE ART

35. I have been informed that unpatentability must be analyzed from the perspective of "one of ordinary skill in the art" in the same field as the patent-in-suit at the time of the alleged invention. I have also been informed that several factors are considered in assessing the level of ordinary skill in the art, including:

the (1) types of problems encountered in the prior art; (2) prior-art solutions to those problems; (3) rate at which innovations are made; (4) sophistication of the technology; and (5) educational level of active workers in the field.

36. I am familiar with the technology at issue and the state of the art as of March 18, 2018. A POSITA at the time the application leading to the ’455 patent was filed would have been someone with a degree in engineering and three years of experience in key duplication equipment. This level of skill is approximate, and more experience can compensate for less formal education, and vice versa. For example, an individual with no engineering degree, but ten years of key duplication experience, would qualify as an ordinarily skilled artisan. Further, a person having no key duplication experience, but a masters or a doctorate in engineering would also qualify as an ordinarily skilled artisan. Experience in the field of key duplication equipment may include experience with various types of key duplication equipment like vending machines. More education can substitute for professional experience, and vice versa.

37. I consider myself to have such a “level of ordinary skill in the art” with respect to the subject matter of the ’455 patent at least because I earned Masters and Doctorate degrees in the fields of Electrical Engineering and Computer Science decades prior to the earliest filing date of the patent. I also have over 35 years of professional experience in the fields of electrical engineering,

computer science, data security/key management, and information systems. Thus, I am able to opine on how a POSITA would have understood the disclosure and claims of the '455 patent, disclosures of the prior art, motivations to combine the prior art, and what combinations would have been obvious to one of ordinary skill.

VII. CLAIM CONSTRUCTION

38. I have considered and applied the above-mentioned legal standards of claim construction in my analysis. In my opinion, no claim terms require construction.

VIII. THE PRIOR ART DISCLOSES OR RENDERS OBVIOUS CLAIMS 1-6 OF THE '455 PATENT

39. Each asserted reference identified in the table below was filed before (and issued before) March 18, 2018, the earliest possible priority date of the '455 patent. Thus, I have been instructed to assume that each reference is prior art to the '455 patent.

Prior Art References
WO 2008/145199A1 to Chies et al. ("Chies," Ex. 1004), published December 4, 2008.
U.S. Patent No. 9,563,885 to Marsh et al. ("Marsh," Ex. 1005), filed July 6, 2015, and issued February 7, 2017.

40. In the analysis that follows, I identify the following prior art, alone or in combination, that in my opinion either anticipates or renders obvious the '455 patent's claims:

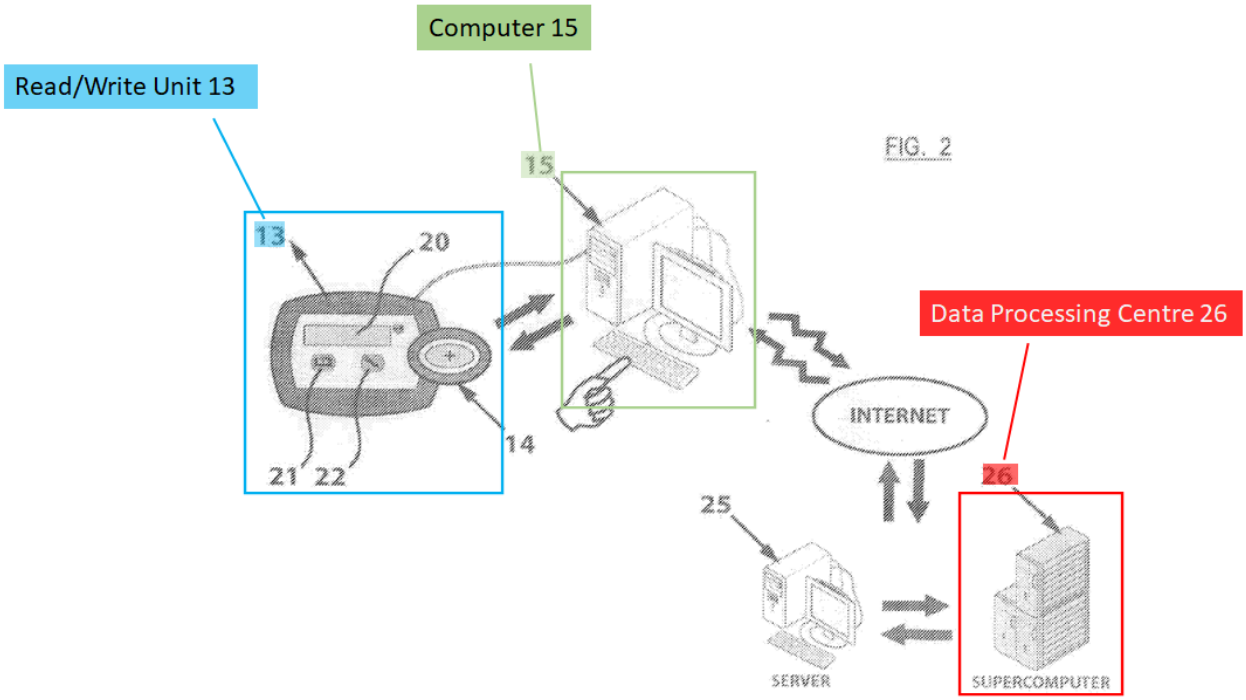
Grounds of Unpatentability	
1	Chies anticipates claims 1 and 3.
2	Chies renders obvious claims 1 and 3.
3	Chies in combination with Marsh renders obvious claims 2-6.

A. Prior Art to the '455 Patent

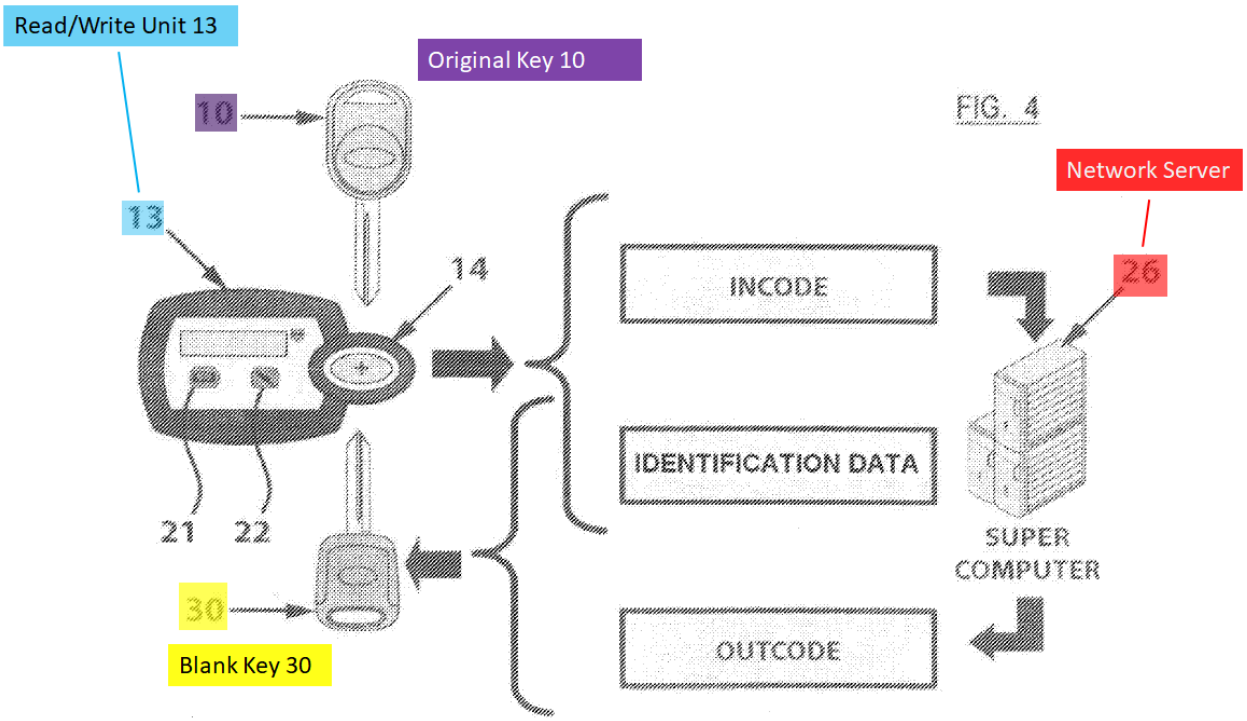
1. Chies

41. Chies discloses a method for duplicating "electronic-code keys" with "code-based electronic recognition means," such as keys used in motor vehicle starting systems. Ex. 1004, 1:10-17. For example, Chies discloses duplicating key 10 by inserting it into a read/write unit 13 (**blue** in annotated Figure 2 below) and establishing wireless communication with the key via the antenna of the read/write unit 13. *Id.*, 4:26-5:2, Fig. 2. The read/write unit 13 is connected to a computer 15 (**green** below), which runs an "application program that is adapted to handle the flow of data coming in from the read/write unit 13." *Id.*, 5:33-6:4. The computer 15 is connected to a network (e.g., the Internet) and transmits key information to

remote servers including server 25 and data processing centre 26 (**red** below). *Id.*, 5:33-6:28.



42. After receiving the master key security information transmitted from read/write unit 13 in the form of "INCODE" data, data processing centre 26 processes the key information, such as by decrypting the INCODE data, to prepare "OUTCODE" data for programming a blank key. Ex. 1004, 6:30-8:34. The data processing centre 26 sends the OUTCODE data to read/write unit 13, which writes the data to the microchip of blank key 30 (**yellow** in annotated Figure 4 below) to create a duplicate of original key 10 (**purple** below). 8:11-34, Fig. 4.



43. Chies is within the same field of endeavor as the '455 patent, transponder key duplication, and is analogous prior art.

2. Marsh

44. Marsh discloses systems and methods for duplicating transponder keys and managing associated key information. Ex. 1005, 1:1-16. For example, Marsh discloses a system including key duplication kiosk 200 (**green** in annotated Figs. 2 and 13 below), which includes a transponder antenna 1302 that reads information from a transponder chip 1306 of transponder key 1304 inserted into the kiosk (**purple** below). Ex. 1005, 1:54-3:14, 16:23-42, Figs. 2, 13. The kiosk may transmit the key information to server 1310 (**red** below) over a

communication network, and server 1310 then processes the information, e.g., to create a duplicate key. Ex. 1005, 17:51-18:9, 25:61-65.

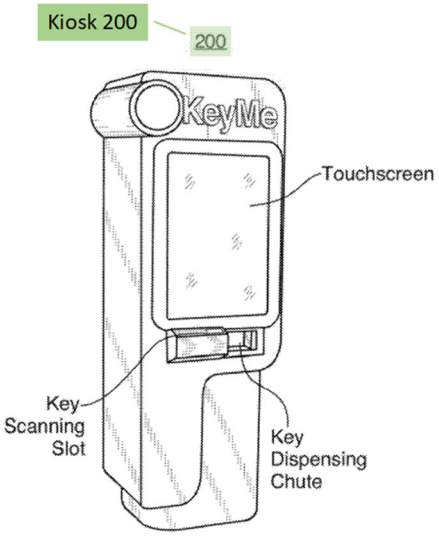


FIG. 2

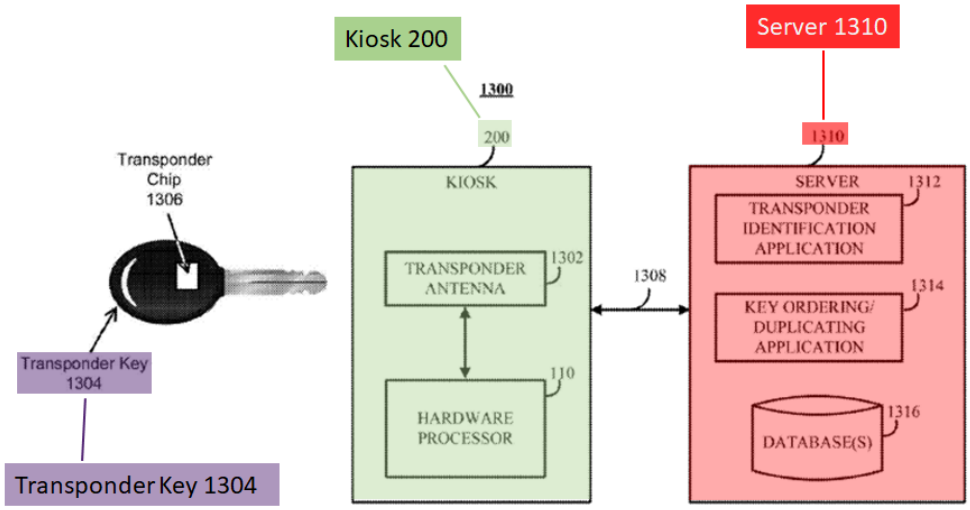
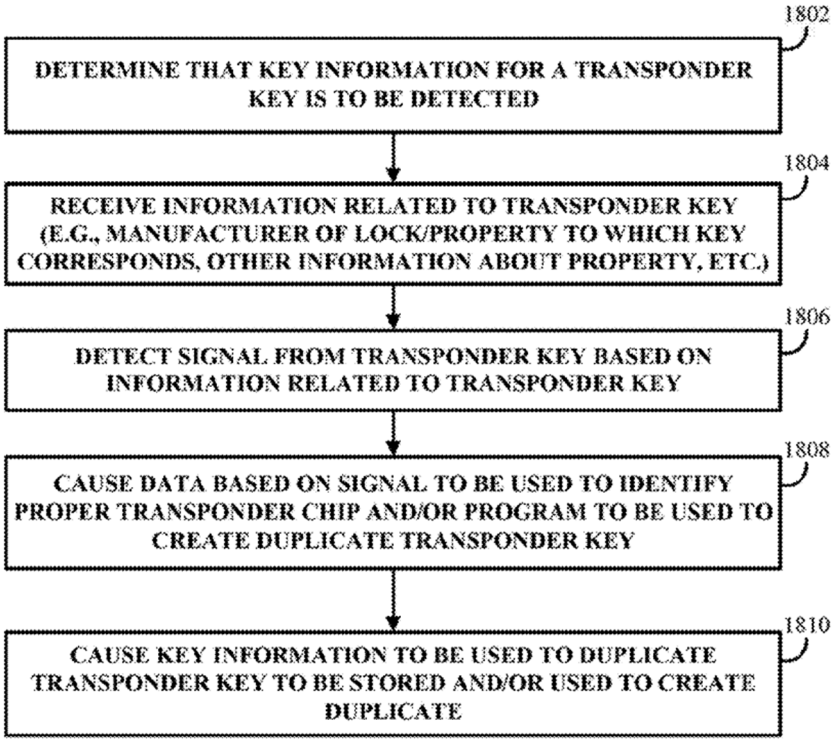


FIG. 13

45. Marsh also discloses a process of reading information from a transponder key to duplicate the key, as shown in the flow chart of Fig. 18 below.

Ex. 1005, 22:25-25:65, Fig. 18. The process includes detecting a signal from the transponder key, using that signal to identify a proper transponder chip for use in a key blank, and using the key information to duplicate the transponder key. *Id.*



46. Marsh is within the same field of endeavor as the ’455 patent, transponder key duplication, and is analogous prior art.

B. Grounds 1 and 2: Chies Anticipates and/or Renders Obvious Claims 1 and 3.

1. Independent Claim 1

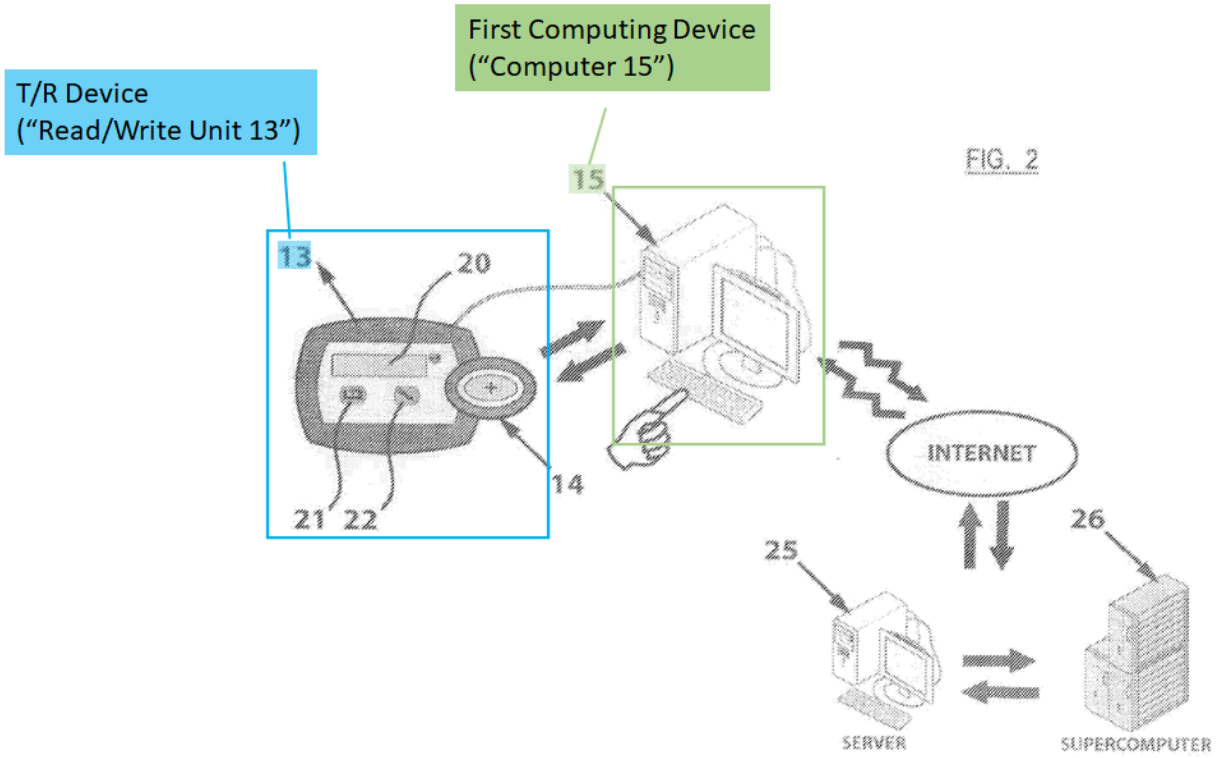
i. 1(pre): A method for duplicating an access device, comprising:

47. Chies discloses a method for duplicating an access device. The ’455 patent describes “vehicle access devices” as including “transponder keys or

remotes." Ex. 1001, 1:18-21, 2:12-14. Chies similarly describes systems and methods for duplicating "keys provided with code-based electronic recognition means," including transponder keys. Ex. 1004, 1:10-17, 4:26-29.

ii. 1(a): providing a first computing device configured to communicate with a transmitter/receiver device (t/r device), the t/r device including an antenna for wirelessly communicating with a master key for a vehicle;

48. Chies discloses the use of a first computing device (e.g., computer 15) that is configured to communicate with a t/r device (e.g., read/write unit 13). Ex. 1004, 5:33-34 ("The peripheral read/write unit 13 is connected to a computer 15, e.g. via a serial interface RS232 or a Universal Serial Bus (USB)."), 6:2-4 ("Installed in the same computer 15 there is an application program that is adapted to handle the flow of data coming in from the read/write unit 13."), Fig. 2.



49. A POSITA would have understood that Chies' read/write unit 13 is a "t/r device" because it is described as both transmitting information to (i.e., writing) and receiving information from (i.e., reading) devices such as transponder keys. Ex. 1004, 4:31-5:2, 6:6-13 ("The read/write unit 13 performs data reading and writing operations and is capable of identifying the type of key"), 6:30-7:19, 8:11-34.

50. Chies further discloses that the read/write unit 13 includes an antenna for wirelessly communicating with a master key for a vehicle (e.g., key 10). Ex. 1004, 4:31-5:2 ("The code-based electronic recognition means is energized via an electromagnetic coupling established through an antenna of a read/write unit 13 adapted to receive a key 10 or 30 in a pit or hole 14 (see Figure 4)."), 6:6-13

("[T]he read/write unit 13 is provided with . . . an antenna (not shown) for receiving and transmitting data"), 7:4-5 ("[A] wireless communication is then established between the read/write unit 13 and the code-based electronic recognition means 11 of the key 10.").

51. Chies further discloses that the read/write unit 13 communicates wirelessly with "original key 10 to be duplicated," and that Chies' method is applicable to motor vehicle keys such as those used "in connection with motor-vehicle starting systems." Ex. 1004, 1:10-17, 6:30-7:5, 8:21-34. A POSITA would have understood that Chies' "original key 10 to be duplicated" is the claimed "master key" because that term refers to an original key that is duplicated. *See* Ex. 1001, 4:27-29 ("Such vehicles typically include an original key that is a suitable match for the vehicle, commonly referred to as the master key."), 4:47-49, 6:7-16.

52. Chies thus discloses providing a first computing device configured to communicate with a transmitter/receiver device (t/r device), the t/r device including an antenna for wirelessly communicating with a master key for a vehicle.

iii. 1(b): running a cloning application associated with said t/r device on the first computing device, the cloning application configured to electronically communicate with the t/r device;

53. Chies discloses running a cloning application (e.g., the "application program") that is associated with the t/r device (e.g., read/write unit 13) on the first computing device (e.g., computer 15), where the cloning application is configured

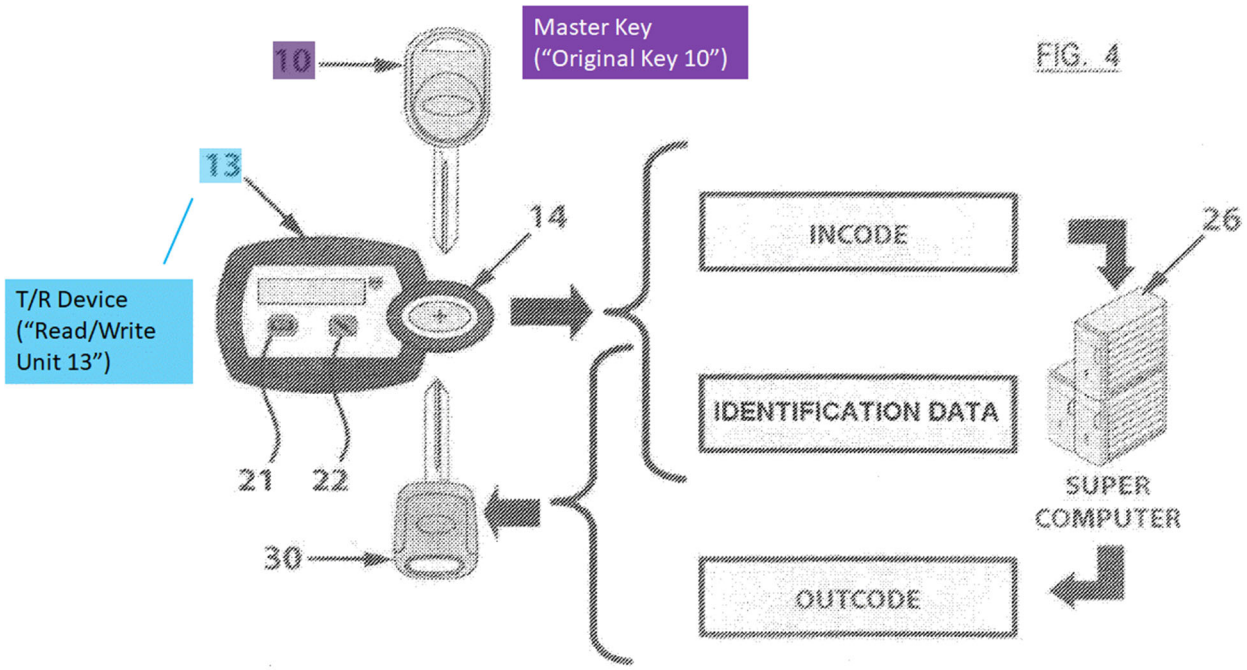
to electronically communicate with the t/r device. Ex. 1004, 5:33-6:4 ("Installed in the same computer 15 there is an application program that is adapted to handle the flow of data coming in from the read/write unit 13."), 7:12-19 ("The read/write unit 13 transmits the INCODE . . . to the computer connected thereto. The computer 15, as duly provided with the application program, receives such information and sends it to the serving computer or server 25.").

54. Chies thus discloses running a cloning application associated with said t/r device on the first computing device, the cloning application configured to electronically communicate with the t/r device.

iv. 1(c): retrieving, by the antenna of the t/r device, security information digitally stored by said master key;

55. As discussed above for Claim 1(a), Chies discloses a t/r device (e.g., read/write unit 13) including an antenna for wirelessly communicating with a master key for a vehicle. Ex. 1004, 4:31-5:2, 6:6-13 ("[T]he read/write unit 13 is provided with . . . an antenna (not shown) for receiving and transmitting data"), 7:4-5 ("[W]ireless communication is then established between the read/write unit 13 and the code-based electronic recognition means 11 of the key 10."). Chies further discloses retrieving security information digitally stored by the master key through those wireless communications. Ex. 1004, 5:4-14 (describing storing information digitally in memory on the key, including "identification data and a

secret code of the key"), 6:30-7:19 ("[A] key 10 to be duplicated is introduced in the appropriate receptacle 14 of a read/write unit 13. Via the interactive read button 21, the read/write unit 13 reads the identification data of the key 10, thereby recognizing the type of key being handled, and temporarily stores such data in the internal memory thereof."), Fig. 4.



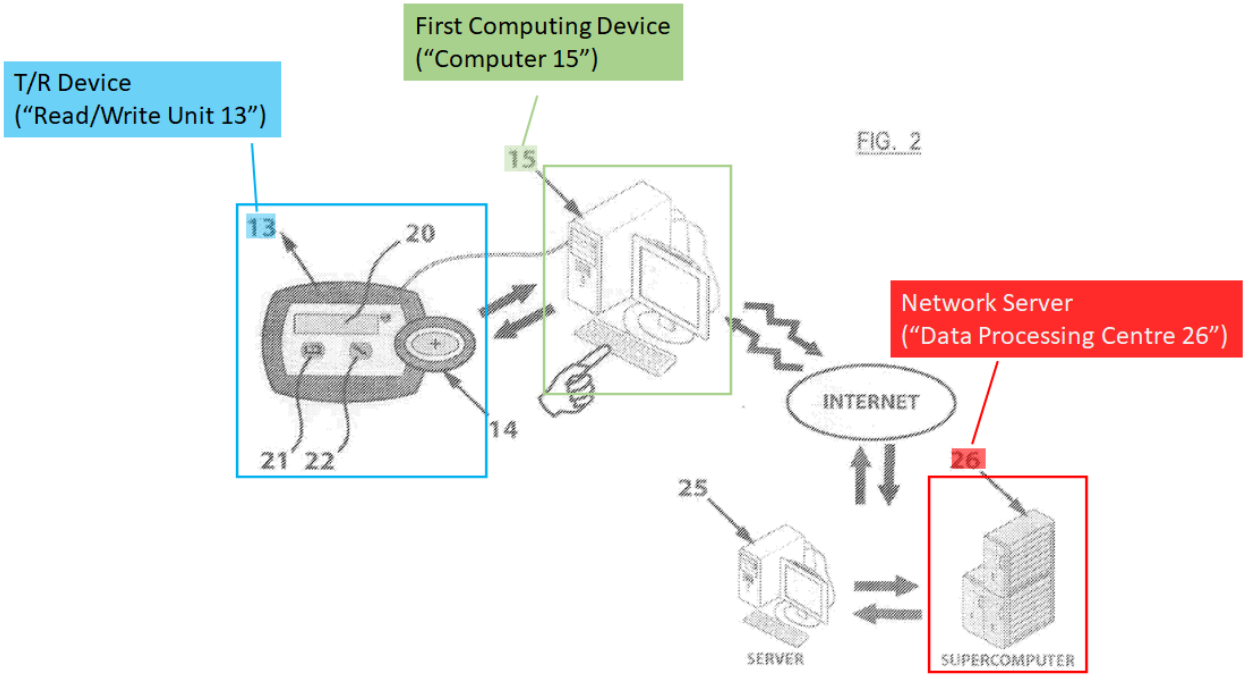
56. The information read from key 10 may be encrypted security information digitally stored on the key. *Id*. ("In the assumption that the key 10 is of the type with a secret (encrypted) code, the read/write unit 13 concurrently performs an inquiry operation to query the key 10. Such query occurs through a code that is implemented in the read/write unit 13 and is solely known by the manufacturer."). Chies refers to the security information received from the master key by read/write unit 13 as "INCODE." Ex. 1004, 7:8-10 ("What the read/write

unit 13 acquires is therefore a data-based information that is encrypted at random in a first encryption form, which shall be defined as INCODE hereinafter."), 7:29-8:9 (describing a process wherein "the INCODE is decrypted," "the secret code is recognized and authenticated" such that "the single and sole secret code corresponding to the one contained in the microchip of the key 10 to be duplicated is acquired," and "[t]he data processing centre 26 encrypts the corresponding secret code . . . into a second encryption form, which shall be defined as OUTCODE hereinafter.").

57. Chies thus discloses retrieving, by the antenna of the t/r device, security information digitally stored by said master key.

v. 1(d): communicating, by the cloning application, the security information for said master key to said network server;

58. Chies discloses a cloning application (e.g., the application program of computer 15) communicating the security information (e.g., the INCODE data) of a master key (e.g., key 10) to a network server (e.g., data processing centre 26). Ex. 1004, 6:15-7:31, Fig. 2.

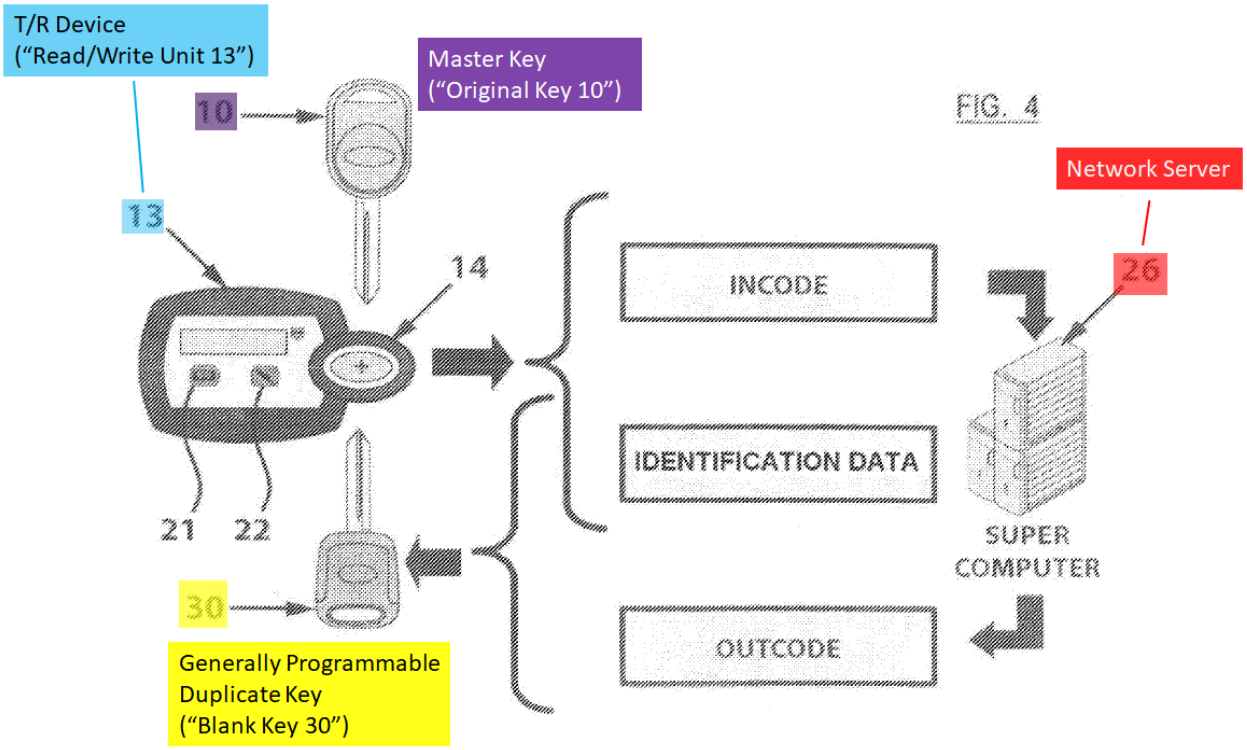


59. In particular, Chies discloses that read/write unit 13 transmits the INCODE data that it read from original key 10 to computer 15. Ex. 1004, 7:12-16. The "computer 15, as duly provided with the application program, receives such information and sends it to the serving computer or server 25." *Id.*, 7:16-18. Server 25 then transmits the INCODE data to data processing centre 26 (to which server 25 is connected over the Internet), where the INCODE data is processed. *Id.*, 6:15-28, 7:21-31.

60. Chies thus discloses communicating, by the cloning application, the security information for said master key to said network server.

vi. 1(e): generating, by the network server, a cloning security information;

61. Chies discloses a network server (e.g., data processing centre 26) generating cloning security information (e.g., "OUTCODE" data). Ex. 1004, 7:29-8:9. The '455 patent does not expressly define the meaning of "cloning security information," but describes it as information that is used to program a key blank to prepare a duplicate of the master key. *See, e.g.*, Ex. 1001, 7:57-63 ("The cloning application 102 may communicate with the t/r device 10 to allow the antenna 40 to transmit the cloning security information 148 to the generically programmed duplicate key 50 to create a cloned duplicate key 50'."). Chies similarly describes processing the security information from the master key (e.g., the INCODE data) to prepare "OUTCODE" data used to program a blank transponder key. Ex. 1004, 7:29-8:9 (describing a process wherein "the INCODE is decrypted," "the secret code is recognized and authenticated" such that "the single and sole secret code corresponding to the one contained in the microchip of the key 10 to be duplicated is acquired," and "[t]he data processing centre 26 encrypts the corresponding secret code . . . into a second encryption form, which shall be defined as OUTCODE hereinafter."), 8:21-34, Fig. 4.

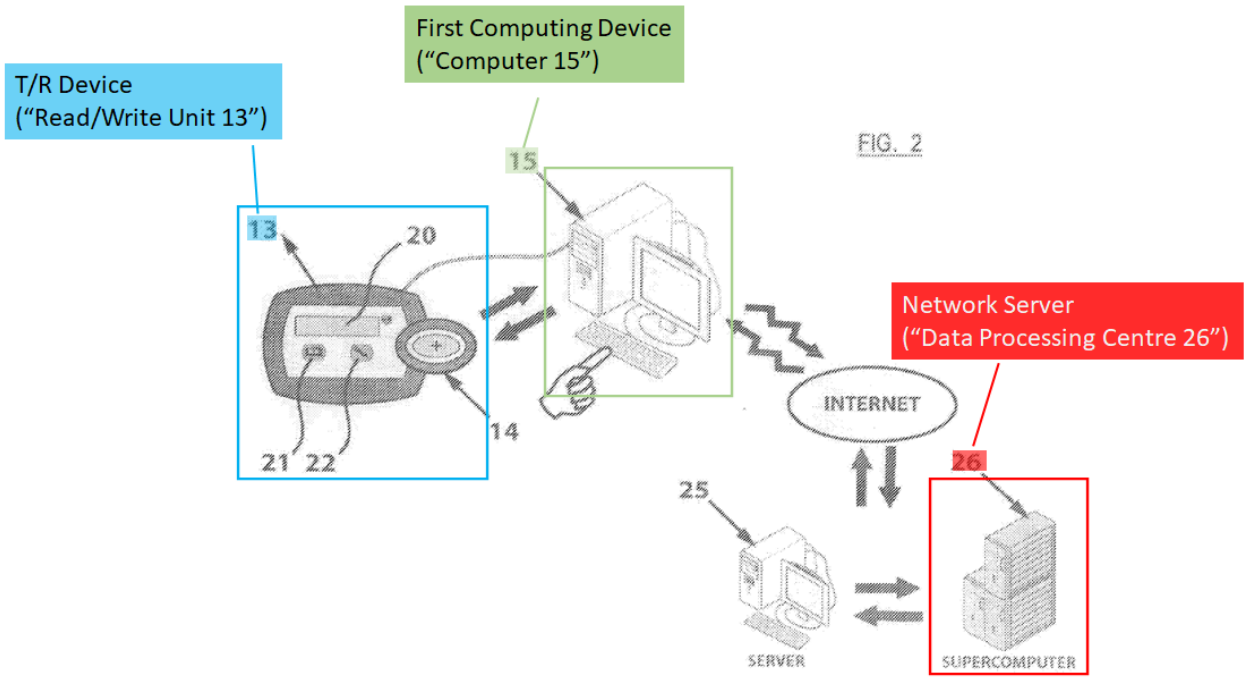


62. Chies thus discloses generating, by the network server, a cloning security information.

vii. 1(f): communicating, by the network server, the cloning security information to the cloning application; and

63. Chies discloses a network server (e.g., data processing centre 26) communicating the cloning security information (e.g., OUTCODE data) to the cloning application (e.g., the application program of computer 15). Ex. 1004, 8:11-34. For example, data processing centre 26 sends the OUTCODE data to the read/write unit 13 "[v]ia the same telematic network" that was used to transmit the key security information (e.g., INCODE) upstream to the data processing centre 26. *Id.*, 8:11-12 ("Via the same telematic network, the data processing centre 26

sends the OUTCODE to the read/write unit 13 from which the related request had originated."). This "telematic network" would have included each component of the network described in more detail for the upstream key information transmission process, including server 25, computer 15 (running the application program), and read/write unit 13. Ex. 1004, 6:15-7:27, Fig. 2.



64. Because the transmission from the data processing centre 26 to the read/write unit 13 uses "the same telematic network," a POSITA would have understood from Chies' disclosure that data processing centre 26 would have communicated the OUTCODE data to the cloning application of computer 15, which would have further communicated that data to the read/write unit 13. Ex. 1004, 7:12-8:24. This would have been the case at least because (as shown in Figure 2 above) computer 15 is connected to the peripheral read/write unit 13 via,

e.g., USB, and is thus the network component enabling read/write unit 13 to transmit information over the network, including to/from data processing centre 26. Ex. 1004, 5:33-6:2.

65. To the extent that Chies does not explicitly state that data processing centre 26 transmits the OUTCODE data to the application program of computer 15, a POSITA would have understood this to be the case (or at least would have found this obvious) based on Chies' disclosure of using computer 15 to provide network (e.g., Internet) access for peripheral read/write unit 13:

> The peripheral read/write unit 13 is connected to a computer 15, e.g. via a serial interface RS232 or a Universal Serial Bus (USB). The computer 15 is interconnected with the Internet network and is located in proximity of the read/write unit 13. Installed in the same computer 15 there is an application program that is adapted to handle the flow of data coming in from the read/write unit 13.

Ex. 1004, 5:33-6:4. Computer 15 would have therefore been an intermediate network component that receives the OUTCODE data from data processing centre 26 and transmits that OUTCODE data to read/write unit 13 using the computer's application program. The use of computers, such as computer 15, to transmit and receive information over the Internet for connected "peripheral" devices such as

read/write unit 13 was also routine in the field and would have been well known in the art for decades, rendering this claim feature obvious in any event.

66. It would have also been known to a POSITA that Chies' "application program" of computer 15 would have been the software of computer 15 receiving the cloning security information from the network server, as opposed to another program running of computer 15. As noted above, Chies explains that the application program is computer 15's software interface with read/write unit 13 because the application program "is adapted to handle the flow of data coming in from the read/write unit 13." Ex. 1004, 5:33-6:4. Chies further states that the downstream information subsequently sent from data processing centre 26 to read/write unit 13 is sent "[v]ia the same telematic network" as the upstream transmissions, and a POSITA would have understood that the cloning security information would thus be sent to read/write unit 13 from data processing centre 26 via the application program of computer 15. Ex. 1004, 8:11-19.

67. To the extent that Chies does not expressly state that the application program of computer 15 is used for downstream transmissions to read/write unit 13 as well as upstream transmissions, a POSITA would have found it obvious to use the same program to send Chies' transponder key data in two directions (instead of one) for numerous reasons, including (1) improving the efficiency of computer 15 by only requiring the use of a single program for bidirectional transmissions, rather

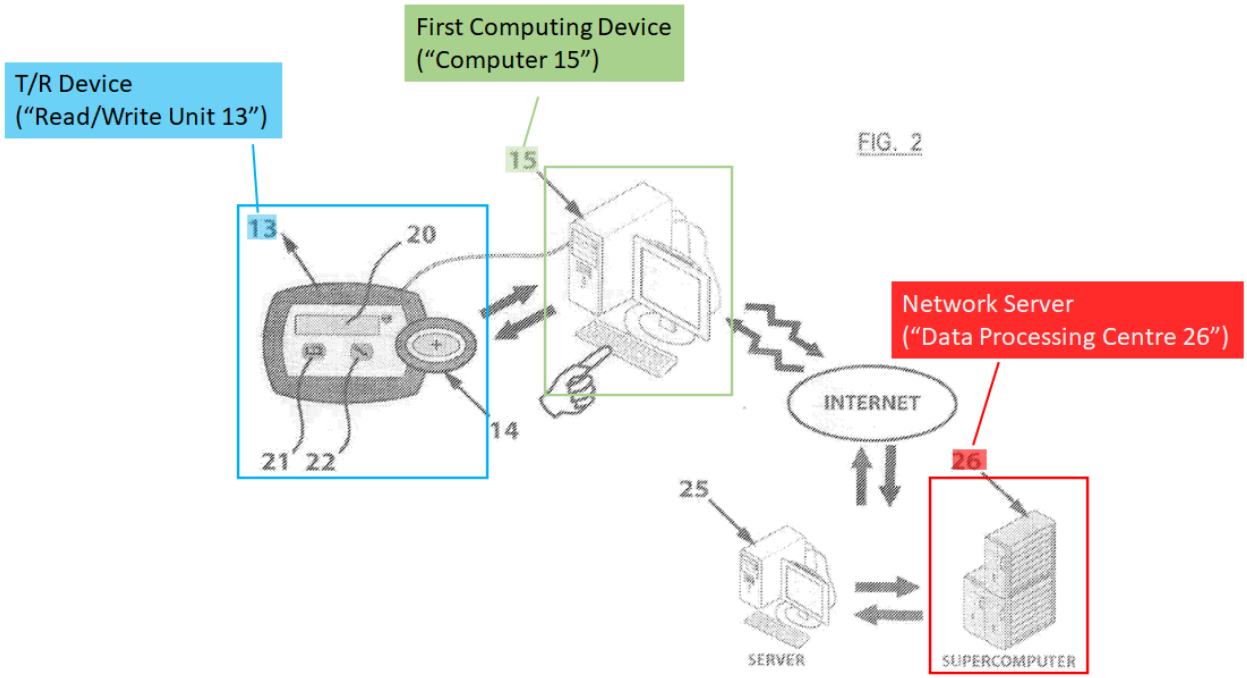
than using multiple programs; (2) reducing the overall complexity of the system, and thus the likelihood of errors in the transmission process; and (3) making the system as a whole easier to develop and deploy by reducing the number of programs of computer 15 that read/write unit 13 and data processing centre 26 would respectively interface with. While no modification to Chies is necessary for the application program of computer 15 to perform this functionality, modifying Chies in this manner would in any event have required only routine skill, knowledge, and standard computer programming techniques, would have been a simple design choice for a POSITA (especially in view of the above-noted benefits of using the same program for this functionality), and would have yielded predictable results with a reasonable expectation of success, i.e., using a single application program for downstream transmissions over a data network in addition to managing upstream data transmissions.

68. Chies therefore discloses, or at least renders obvious, communicating, by the network server, the cloning security information to the cloning application.

viii. 1(g): transmitting, by the cloning application, the cloning security information to the t/r device to program an electronic access device of a generally programmable duplicate key with the cloning security information.

69. As discussed above for Claim 1(f), Chies discloses (or at least renders obvious) a network server (e.g., data processing centre 26) transmitting cloning

security information (e.g., "OUTCODE" data) to a cloning application (e.g., the "application program" of computer 15). Chies similarly discloses (or at least renders obvious) the application program of computer 15 transmitting the OUTCODE data to a t/r device (e.g., read/write unit 13) to program an electronic access device (e.g., "the microchip of the new blank key 30") of a generally programmable duplicate key (e.g., blank key 30) with the OUTCODE data. Ex. 1004, 8:21-34. The network transmission path from the data processing centre 26 to the read/write unit 13 is "the same telematic network" used for upstream transmissions from the read/write unit 13 to the data processing centre 26, and that network includes computer 15 (running the application program), which transmits information to and from read/write unit 13. Ex. 1004, 5:33-6:2, Fig. 2.



70. As with Claim 1(f), to the extent that Chies does not explicitly state that the application program of computer 15 transmits the OUTCODE data to the read/write unit 13, a POSITA would have understood this to be the case (or would have at least found this obvious) based on Chies' disclosure of using computer 15 to provide network (e.g., Internet) access for peripheral read/write unit 13. Ex. 1004, 5:33-6:2 ("The peripheral read/write unit 13 is connected to a computer 15, e.g. via a serial interface RS232 or a Universal Serial Bus (USB). The computer 15 is interconnected with the Internet network and is located in proximity of the read/write unit 13."). Computer 15 would have therefore been an intermediate network component that receives the OUTCODE data from data processing centre 26 and transmits that OUTCODE data to read/write unit 13 using the computer's application program. The use of computers, such as computer 15, to transmit and receive information over the Internet for connected "peripheral" devices such as read/write unit 13 was also routine in the field and has been well known in the art for decades, rendering this claim feature obvious. Moreover, as discussed above for Claim 1(f), a POSITA would at least have found it obvious that the application program of computer 15 (as opposed to another program on the computer) would have managed transmitting the cloning security information to read/write unit 13.

71. After receiving the OUTCODE data, read/write unit 13 "transfers all identification data of [original key 10] into the microchip of the new blank key 30,

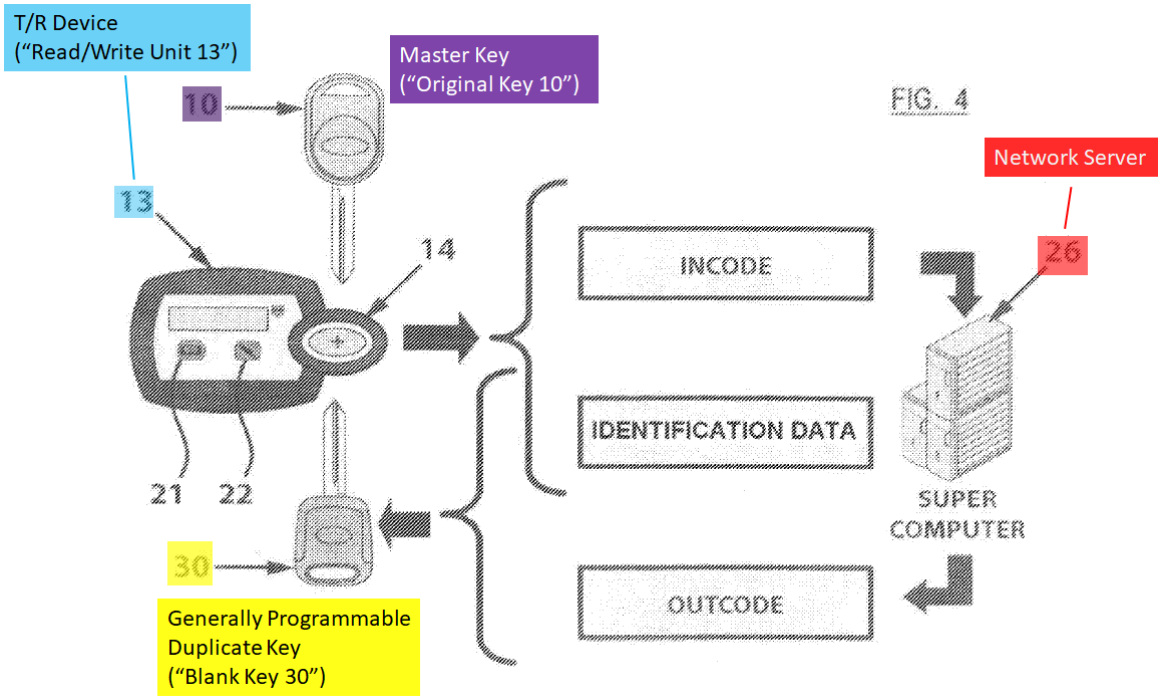
wherein this operation is performed with the aid of the interactive write button 22." Ex. 1004, 8:21-29. The OUTCODE data is then "decrypted and definitively stored – along with the other identification data – solely inside the microchip of the blank key 30." Ex. 1004, 8:29-31. The blank key 30 is then "able to exactly emulate the functions of the configuration of the original key." Ex. 1004, 8:31-34.

72. Chies therefore discloses, or at least renders obvious, transmitting, by the cloning application, the cloning security information to the t/r device to program an electronic access device of a generally programmable duplicate key with the cloning security information.

2. Claim 3: The method of claim 1, further comprising: in response to communicating, by the cloning application, the security information for said master key to said network server, determining, by the network server, whether the security information is encrypted; and generating the cloning security information.

73. As discussed above, Chies anticipates and/or renders obvious claim 1. For Claim 1(d), Chies discloses communicating, by a cloning application, security information for a master key to a network server. Chies further discloses (or at least renders obvious), in response to this process, determining, by the network server (e.g., data processing centre 26) whether the security information (e.g., INCODE data) is encrypted, and generating the cloning security information (e.g., OUTCODE data). Ex. 1004, 7:29-8:9, Fig. 2. For example, data processing centre 26 decrypts the INCODE data (and thus determines whether it is encrypted when it

does so) before generating the cloning security information (e.g., preparing the OUTCODE data). *Id.*



74. To the extent that Chies discloses decrypting the INCODE data but does not expressly state that it first determines whether that data is encrypted, a POSITA would have found it obvious to first determine whether data is encrypted at Chies' data processing centre 26 before performing decryption processing. The process of characterizing whether information (such as transponder key information) is encrypted, known in the art as a form of "data typing," is a fundamental and routine aspect of information processing in the field of computer science that would have been known to a POSITA. Determining whether data is encrypted before performing decryption processing—or before transmitting the

data elsewhere for such processing—would have been beneficial to ensure the data is processed correctly. For example, a POSITA would have known that decrypting data that is not encrypted could corrupt that data, including potentially making the corrupted data unrecoverable. Determining the data type prior to performing decryption processing on non-encrypted data would also have been expected to conserve computing resources, since decryption processing would not be necessary in those circumstances. A POSITA would have had a reasonable expectation of success in determining the data type (i.e., whether the data is encrypted) before performing decryption processing, and doing so would have required only routine skill, knowledge, and standard computer programming techniques.

75. Chies thus discloses, or at least renders obvious, in response to communicating, by the cloning application, the security information for said master key to said network server, determining, by the network server, whether the security information is encrypted, and generating the cloning security information.

C. Ground 3: Chies and Marsh Render Claims 2-6 Obvious.

1. Claim 2:

a. 2(pre): "The method of claim 1, further comprising:"

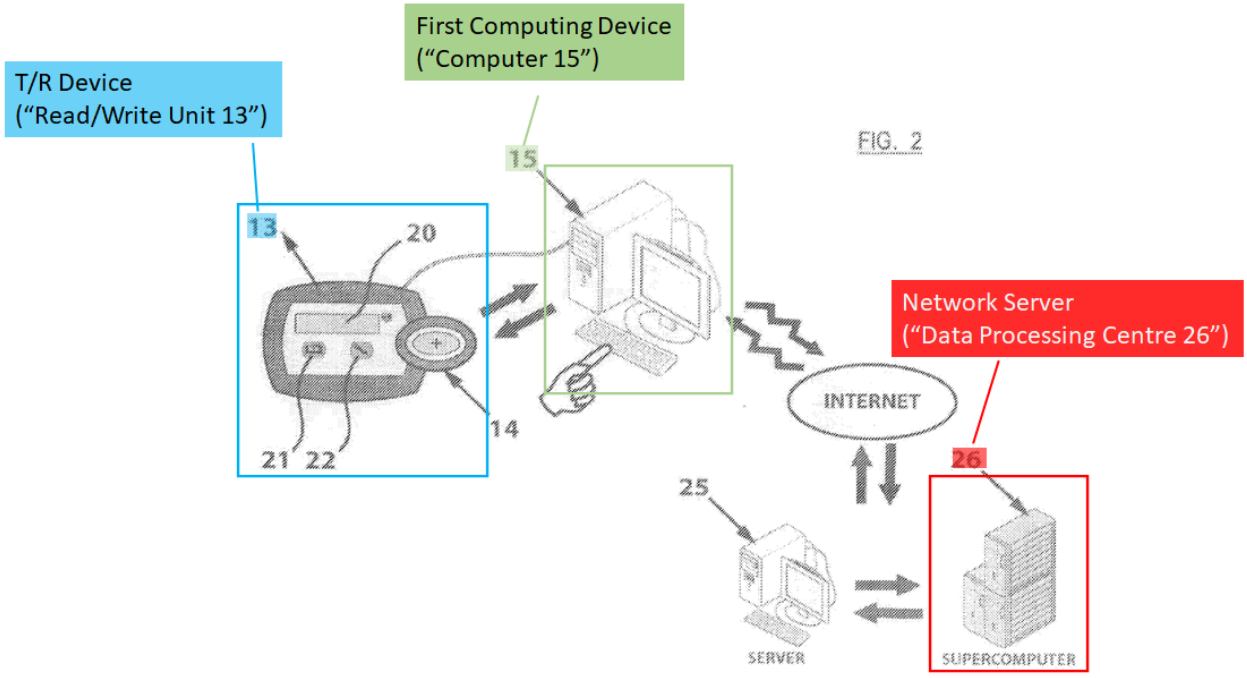
76. As I explain above, Chies anticipates and/or renders obvious claim 1.

b. 2(a): "in response to retrieving, by the t/r device, security information from said master key,"

77. As discussed above for Claim 1(c), Chies discloses (or at least renders obvious) using a t/r device to retrieve security information from a master key.

c. 2(b): "determining, by the cloning application, whether the security information is encrypted and"

78. A POSITA would have found it obvious, in view of the disclosures of Chies and Marsh, to determine whether the security information (e.g., the INCODE data) is encrypted by Chies' cloning application (e.g., the application program of computer 15) before performing decryption processing. For example, Chies' application program communicates the INCODE data of a master key (e.g., key 10) to a network server (e.g., data processing centre 26). Ex. 1004, 6:15-7:31, Fig. 2.



79. Marsh discloses a network server (e.g., server 1310) that is remote from a computer running a cloning application (e.g., software on kiosk 200 that transmits the key information to server 1310). *See* Ex. 1005, 17:51-55 (kiosk 200 transmits key information to server 1310), 19:54-58 (kiosks 200 "can include any of a general purpose device such as a computer"), 24:34-43 (noting that kiosk 200 is an example of "a computing device executing process 1800," as shown in Fig. 18), 26:30-53 (describing computer programs "such as . . . a computer application"), Fig. 13. Marsh further discloses that server 1310 may, upon receiving the information related to the transponder key, "*determine whether the information received from the transponder key is encrypted* and/or can decrypt the information received from the transponder key." *Id.*, 25:26-32 (emphasis added).

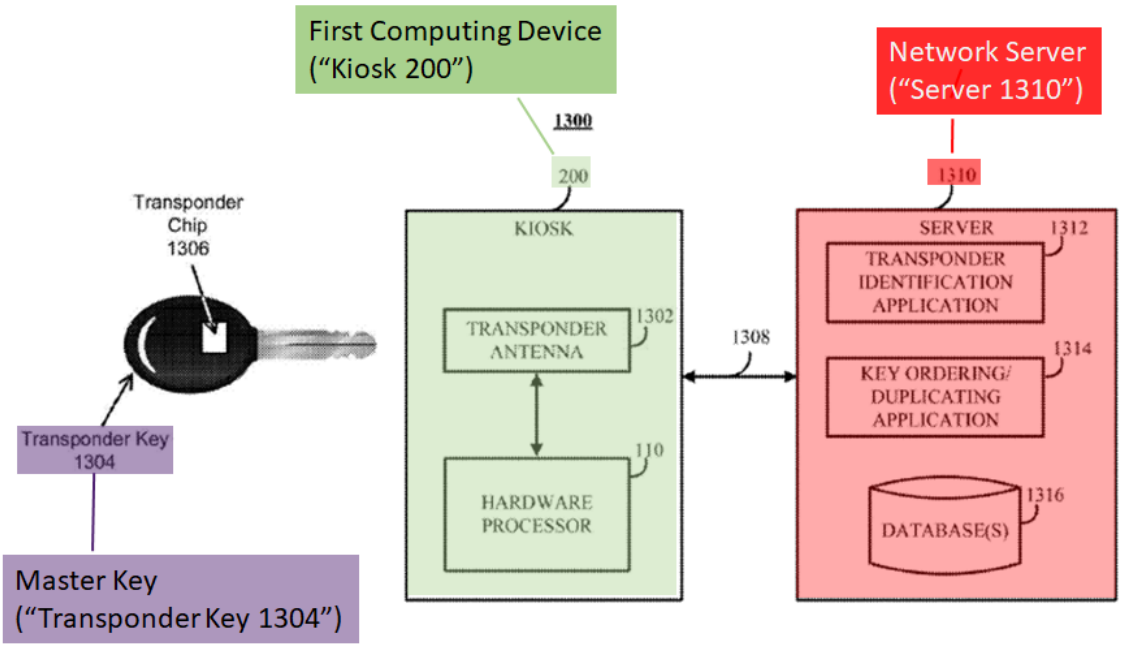


FIG. 13

80. A POSITA considering Chies in view of Marsh's disclosure of "determin[ing] whether the information received from the transponder key is encrypted" would have found it obvious to similarly determine whether Chies' transponder key information is encrypted at Chies' application program before performing decryption processing (or before transmitting the transponder key information to a remote server to perform such processing). For example, a POSITA would have found it obvious to first determine whether the security information read from the master key is encrypted at Chies' application program prior to transmitting that information to a remote server (e.g., data processing centre 26) for decryption processing.

81. As I explain above, the process of characterizing whether information (such as transponder key information) is encrypted, known in the art as a form of "data typing," is a fundamental and routine aspect of information processing in the field of computer science that would have been known to a POSITA. Determining whether data is encrypted before performing decryption processing—or before transmitting the data elsewhere for such processing—would have been beneficial to ensure the data is processed correctly. For example, a POSITA would have known that decrypting data that is not encrypted could corrupt that data, including potentially making the corrupted data unrecoverable. Determining the data type prior to performing decryption processing on non-encrypted data would also have

been expected to conserve computing resources, since decryption processing would not be necessary in those circumstances. A POSITA would have had a reasonable expectation of success in determining the data type (i.e., whether the data is encrypted) before performing decryption processing, and doing so would have required only routine skill, knowledge, and standard computer programming techniques.

82. A POSITA having this fundamental knowledge of decryption processing would have understood that it may not be necessary to perform such processing in Chies' system, depending on whether the key information received by read/write unit 13 is encrypted. Chies states that it assumes, for purposes of its disclosure, that the transponder key information is encrypted and does not explicitly discuss an alternative where the key information is not encrypted. Ex. 1004, 6:30-7:10 ("*In the assumption that the key 10 is of the type with a secret (encrypted) code*, the read/write unit concurrently performs an inquiry operation to query the key 10 . . . .") (emphasis added). But a POSITA would have understood that, if Chies' application program of computer 15 determined that the received transponder key information was not encrypted (as it would have been obvious to do), then it would not be necessary for the application program to transmit the transponder key data to data processing centre 26 for decryption processing. Rather, a POSITA would have understood that it would be beneficial for the

application program to instead send the non-encrypted key information back to read/write unit 13 (after performing this encryption check) to copy the non-encrypted data to a transponder key blank in a simple and routine data copying operation without consuming computing resources to unnecessarily transmit the data to data processing centre 26, thus reducing the application program's use of network bandwidth. Modifying Chies' system to operate in this manner would have required only routine skill, knowledge, and standard computer programming techniques, including routine read/write operations for duplicating non-encrypted data, and a POSITA would have had a reasonable expectation of success when doing so in view of the POSITA's aforementioned knowledge of decryption processing.

d. 2(c): "generating the cloning security information"

83. If this clause of the claim term is interpreted such that the "generating the cloning security information" occurs at the claimed network server, then as discussed above for Claims 1(e) and 3, Chies alone discloses that interpretation. For example, Chies describes performing decryption processing on the encrypted INCODE data at data processing centre 26, thus generating cloning security information (e.g., OUTCODE data). Ex. 1004, 7:29-8:9.

84. If this clause is alternatively interpreted to require that the cloning application (running on the first computing device, per Claim 1(b)) generates the

cloning security information, a POSITA would have found that obvious based on the disclosures of Chies and Marsh.

85. As discussed above, Chies discloses a cloning application (e.g., the application program of computer 15). Ex. 1004, 5:33-6:4, 7:12-19. Marsh discloses a key duplication system including a network server (e.g., server 1310) that determines whether the master transponder key's security information is encrypted and that can decrypt that information. Ex. 1005, 25:26-32. Marsh further explains that its key duplication hardware and software may be deployed in a wide variety of locations and configurations, including as "client-side software" and "client-side hardware" for "the mechanisms described herein," which include its server-side decryption processing systems. Ex. 1005, 26:30-34. A POSITA would understand, from Marsh's broad disclosure regarding the placement of its various computing devices and processes as either "client-side" (i.e., near the master key being duplicated) or "server-side" (i.e., in a remote server such as server 1310), that the exemplary decryption processing systems of Marsh's server 1310 could alternatively be deployed client side, e.g., in kiosk 200.

86. Indeed, client-side key duplication systems were known in the art. For example, Gerlings describes a system for duplicating transponder keys locally at a kiosk (i.e., without an express use of a remote database/server) as shown in annotated Figure 4 below, wherein key duplication machine 100 (**green** below)

scans a transponder key at transponder sensor 124 (**orange** below) to program a duplicate transponder key at cloning pocket 207 (**blue** below). *See* Ex. 1007 ¶¶ 50, 82, 88, 97-98.

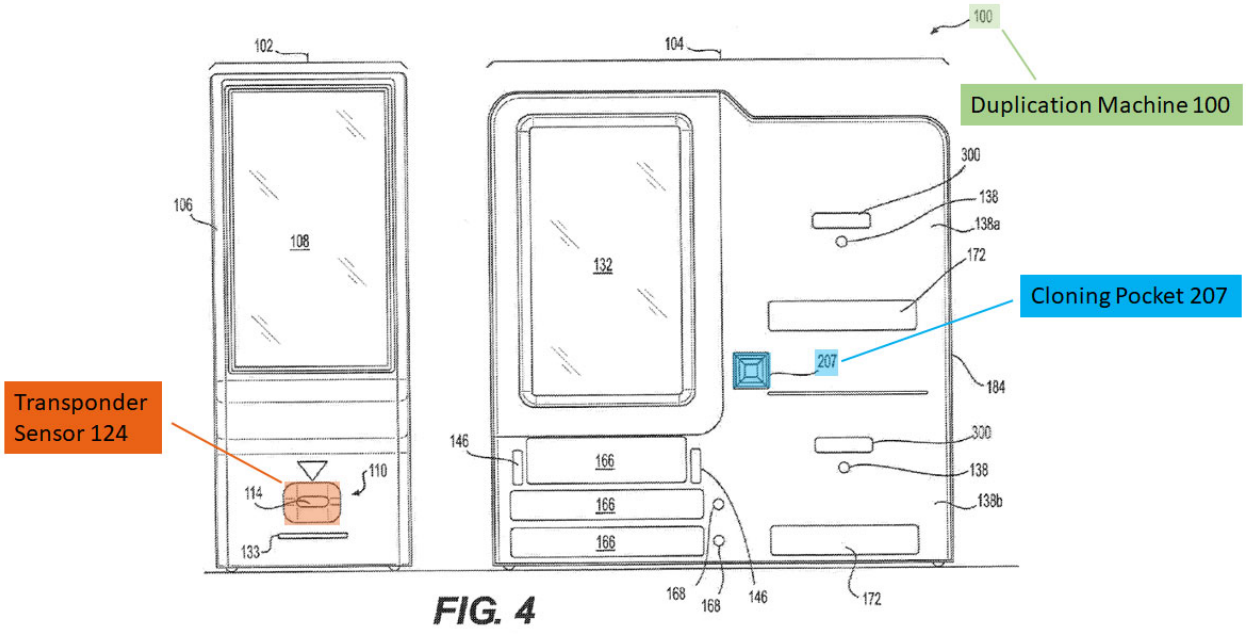


FIG. 4

87. A POSITA would have understood several benefits of performing decryption processing (and generating the information for programming into a blank transponder key) in a client-side application like the application program of computer 15, such as to (1) improve the security of the system by maintaining sensitive key data locally (i.e., in proximity to a user and the key being duplicated); (2) reduce the use of network resources by eliminating the need to transmit encrypted information from kiosk 200 to server 1310 in situations where kiosk 200 is able to decrypt and duplicate the key information; (3) reduce the complexity of the key duplication system by consolidating processing resources in one place

(e.g., within kiosk 200), rather than using a distributed system (including remote servers) with multiple potential points of processing delay or failure; and (4) improve the response speed of the system (and the speed of overall key duplication for a customer) by eliminating unnecessary network components from the key duplication process, especially when duplicating transponder keys where the kiosk 200 is able to decrypt and duplicate the key information.

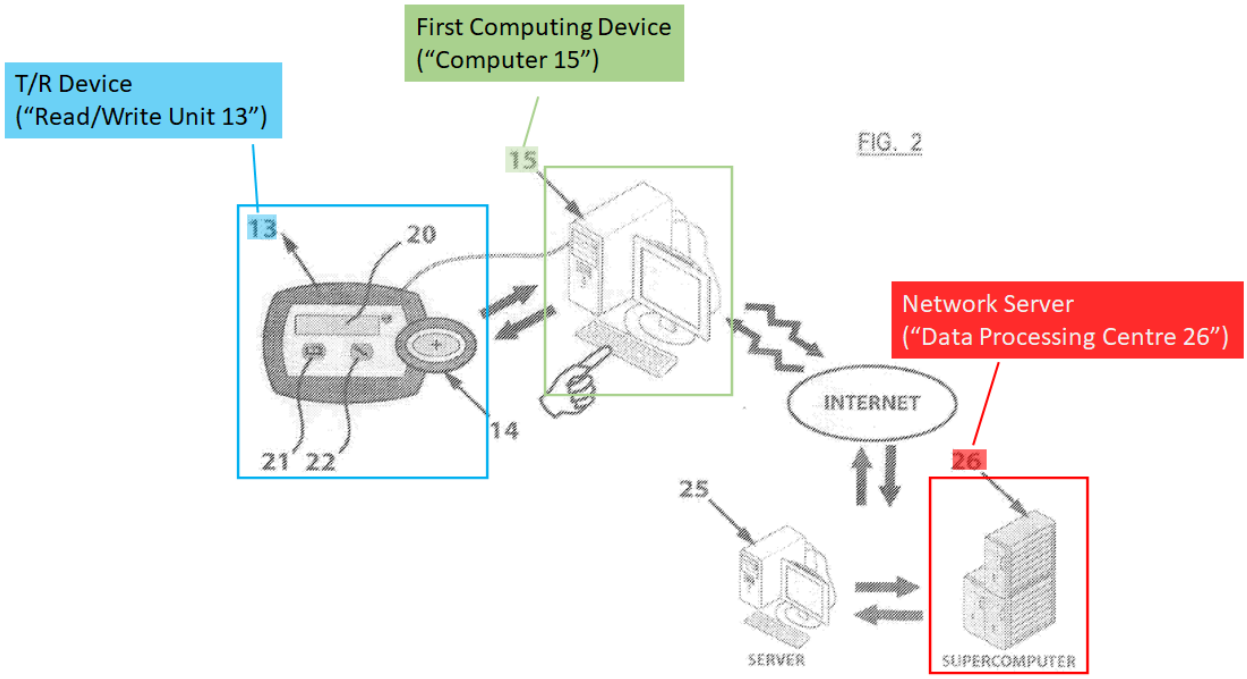
88. In view of Marsh's disclosure, a POSITA would have found it obvious to modify Chies to incorporate a client-side decryption engine (e.g., the decryption system of data processing centre 26) in the application program of computer 15 to obtain the aforementioned benefits of client-side data decryption and generation of blank transponder key information. A POSITA would have had a reasonable expectation of success in doing so because this modification would have required only routine skill, knowledge, and standard computer programming and manufacturing techniques, and would have been a simple design choice within the skill of a POSITA between two known alternatives (i.e., either performing the transponder data decryption/processing in a client-side system or a server-side system).

89. Chies in combination with Marsh thus renders obvious, in response to retrieving, by the t/r device, security information from said master key,

determining, by the cloning application, whether the security information is encrypted and generating the cloning security information.

2. Claim 3: The method of claim 1, further comprising: in response to communicating, by the cloning application, the security information for said master key to said network server, determining, by the network server, whether the security information is encrypted; and generating the cloning security information.

90. Chies anticipates and/or renders obvious claim 1. As discussed above in for Claim 1(d), Chies discloses communicating, by a cloning application (running on computer 15), security information for a master key to a network server. Ex. 1004, 6:15-7:31, Fig. 2.



91. Chies further discloses, in response to this process, determining, by the network server (e.g., data processing centre 26) whether the security

information (e.g., INCODE data) is encrypted, and generating the cloning security information (e.g., OUTCODE data), as I discuss above for Claim 1(e). Ex. 1004, 7:29-8:9. For example, data processing centre 26 decrypts the INCODE data (and thus determines whether it is encrypted when it does so) before preparing the OUTCODE data. *Id.*

92. As discussed above for Claim 2, to the extent that Chies discloses decrypting the INCODE data but does not expressly state that data processing centre 26 first determines whether that data is encrypted, a POSITA would have found it obvious to first determine whether data is encrypted before performing decryption processing in view of the disclosure of Marsh. In particular, Marsh discloses a network server (e.g., server 1310) that is remote from a computer running a cloning application (e.g., software on kiosk 200 that transmits the key information to server 1310). *See* Ex. 1005, 17:51-55 (kiosk 200 transmits key information to server 1310), 19:54-58 (kiosks 200 "can include any of a general purpose device such as a computer"), 24:34-43 (noting that kiosk 200 is an example of "a computing device executing process 1800," as shown in Fig. 18), 26:30-53 (describing computer programs "such as . . . a computer application"). Marsh further discloses that server 1310 may, upon receiving the information related to the transponder key, "determine whether the information received from

the transponder key is encrypted and/or can decrypt the information received from the transponder key." Ex. 1005, 25:26-32.

93. A POSITA considering Chies in view of Marsh's disclosure of "determin[ing] whether the information received from the transponder key is encrypted" would have found it obvious to similarly determine whether Chies' transponder key is encrypted at the network server (e.g., Chies' data processing centre 26) before performing decryption processing. Determining whether data is encrypted before performing decryption processing—or before transmitting the data elsewhere for such processing—would have been beneficial to ensure the data is processed correctly. For example, a POSITA would have known that decrypting data that is not encrypted could corrupt that data, including potentially making the corrupted data unrecoverable. Determining the data type prior to performing decryption processing on non-encrypted data would also have been expected to conserve computing resources, since decryption processing would not be necessary in those circumstances. A POSITA would have had a reasonable expectation of success in determining the data type (i.e., whether the data is encrypted) before performing decryption processing, and doing so would have required only routine skill, knowledge, and standard computer programming techniques.

94. Chies in combination with Marsh thus renders obvious, in response to communicating, by the cloning application, the security information for said

master key to said network server, determining, by the network server, whether the security information is encrypted, and generating the cloning security information.

3. Claim 4:

4(pre) The method of claim 1, further comprising:

95. As I explain above, Chies anticipates and/or renders obvious claim 1.

4(a) capturing at least one image of a blade of said master key,

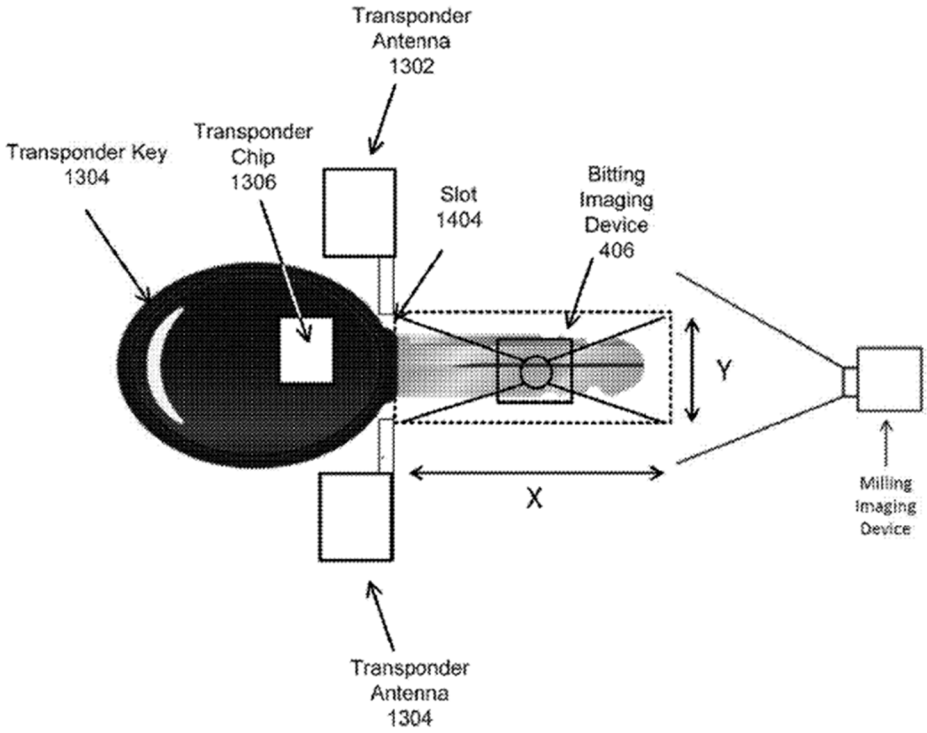
96. Chies does not explicitly disclose capturing at least one image of a blade of said master key.

97. Marsh discloses systems and methods of duplicating keys, including several embodiments relating to imaging the blade of a key in order to duplicate it. For example, Marsh discloses key detector 106, which "can be any suitable mechanism for detecting the bitting pattern and/or the blank type of a key," including optical technologies (e.g., cameras). Ex. 1005, 5:32-33. Marsh also describes "imaging devices 406 and 408," which "can be used to optically detect a bitting pattern and a key blank type of the key," including by capturing images of the key. Ex. 1005, 13:41-50. For transponder keys, in particular, Marsh discloses obtaining images of the key where the:

> image data can include a shape of the key head, one or more logos associated with the transponder key, a shape and/or position of the shaft of the transponder key (if the transponder key includes a shaft), a bitting pattern of the

> transponder key (if the transponder key includes a bitting pattern), an overall shape of the transponder key, textual information associated with the transponder key and/or any other suitable information.

Ex. 1005, 22:64-23:10; *see also* Ex. 1005, 2:30-37 (describing "receiving an image of the transponder key captured by an image sensor included in the key scanner; determining geometric information about the transponder key based on an image of the transponder key captured by the image sensor; and causing the geometric information about the transponder key to be used to create the duplicate transponder key."), 16:59-65, 18:39-67 (describing obtaining the "geometric properties of the key (e.g., using detector(s) 106)"), Fig. 14.



98. A POSITA would have been motivated to modify Chies to incorporate the key blade imaging systems and methods disclosed by Marsh to expand the capabilities of the Chies system, such as by enabling Chies' system to additionally duplicate the blade of a master key in addition to the transponder data of the key. A POSITA would have understood that doing so would allow for a duplicate key to include the characteristics of the master key's blade in addition to the transponder data, thereby allowing the duplicate transponder key to be a more complete duplicate of the master transponder key. Duplicating the blade of the master key in this manner would enable the duplicate key to provide access to systems or devices that may not otherwise be accessible using only the transponder data, such as an ignition to a car engine requiring insertion of the key blade for activation. In my experience, the concept of imaging and duplicating key blades has been known in the key duplication field for decades, and modifying Chies in this manner would have required only routine skill, knowledge, and standard manufacturing techniques, and would have yielded predictable results with a reasonable expectation of success, i.e., having a system capable of duplicating both the transponder data of a transponder key as well as the blade of the transponder key.

99. The combination of Chies and Marsh thus renders obvious capturing at least one image of a blade of said master key.

4(b) communicating said at least one image of said blade to said network server,

100. Chies does not explicitly disclose communicating at least one image of a key blade to a network server.

101. Marsh discloses communicating the captured images of a transponder key's blade to a network server. For example, in addition to obtaining images of a transponder key as part of process 1800 for duplicating the transponder key (e.g., Ex. 1005, 22:64-23:10), Marsh further discloses sending "geometric information about a shaft portion of the transponder key" to a network server (e.g., server 1310) from kiosk 200 to duplicate the transponder key. *Id.*, 2:30-37, 16:59-65, 18:2-67, 20:59-67, 25:33-47, Fig. 13.

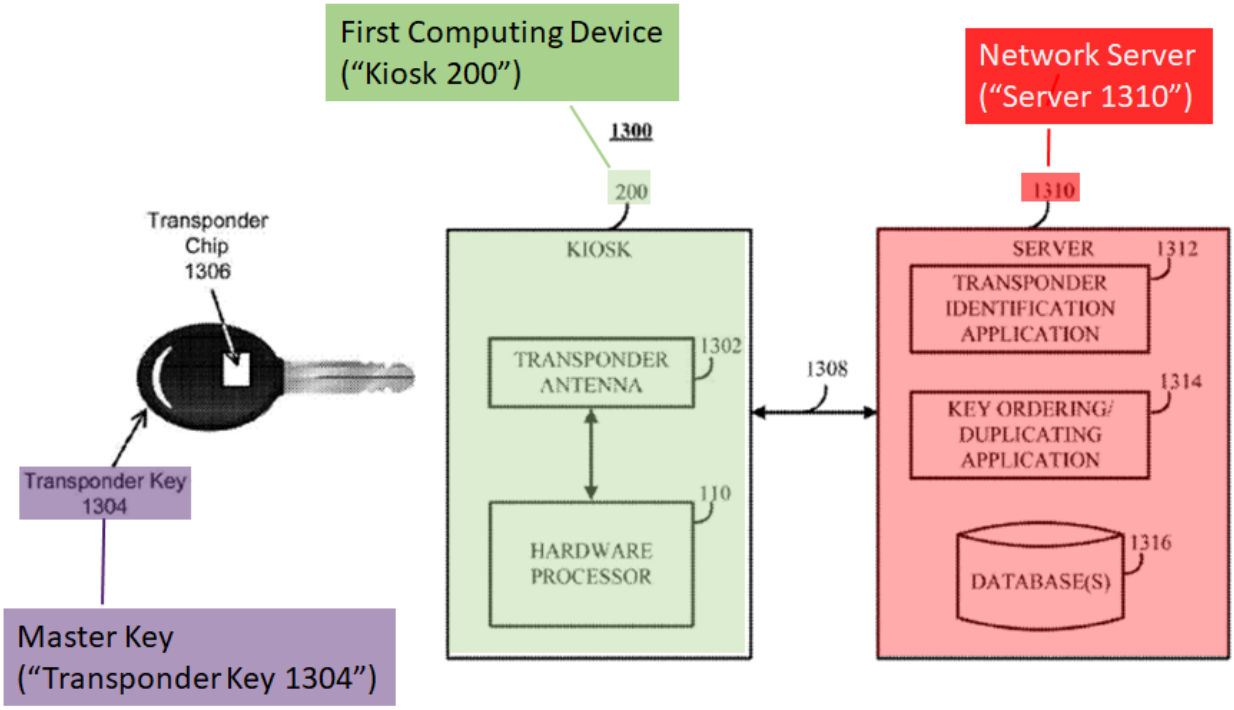


FIG. 13

102. A POSITA would have understood that Marsh's disclosure of sending the image data (including the key's "geometric information") to a network server would have included transmitting (i.e., communicating) the images of the key blade used to duplicate the key to server 1310 for further processing. A POSITA would have been motivated to implement Marsh's capabilities of communicating at least one image of the key blade to the network server into Chies' system for the same reasons described above for Claim 4(a). A POSITA would have had a reasonable expectation of success in doing so, as this would have required only routine skill, knowledge, and standard computer programming techniques.

103. The combination of Chies and Marsh thus renders obvious communicating said at least one image of said blade to said network server.

4(c) cutting a duplicate blade of a generally programmable duplicate key, and

104. Chies does not disclose cutting a duplicate blade of a generally programmable duplicate key.

105. Marsh discloses cutting a blade of a generally programmable duplicate key (e.g., a transponder key blank) to create a duplicate of a master transponder key. For example, Marsh describes key cutting and cleaning mechanisms used to cut a key blank. Ex. 1005, 8:6-43, 10:31-40 (describing using a key cutting mechanism "to cut the key according to the detected bitting pattern and then clean the key to remove burrs, etc."), 12:32-41. Marsh further describes cutting a key blank for a transponder key "based on geometric information included in the key information." Ex. 1005, 18:27-30. Marsh also describes providing the duplicate key to a customer from a remote location by allowing the customer to "request a physical copy via mail order," as well as to "make a physical copy with any suitable key duplication hardware" (i.e., cutting a duplicate blade of a generally programmable duplicate key at a remote location before mailing it to a customer). Ex. 1005, 15:10-27.

106. A POSITA would have been motivated to implement Marsh's capabilities of cutting a duplicate blade of a generally programmable duplicate key

into Chies' system for the same reasons described above for Claim 4(a). A POSITA would have had a reasonable expectation of success in doing so, as this would have required only routine skill, knowledge, and standard computer programming and/or manufacturing techniques.

107. The combination of Chies and Marsh thus renders obvious cutting a duplicate blade of a generally programmable duplicate key.

4(d) providing said generally programmable duplicate key to said customer.

108. Chies does not disclose providing a generally programmable duplicate key to a customer after cutting the blade of the key blank.

109. Marsh describes, after cutting the blade of a generally programmable duplicate key (e.g., transponder key blank) and programming a transponder chip within the key blank, causing the "the key to be dispensed to a user. For example, the hardware processor can control the key cutting and cleaning mechanism 116 to drop the key in the key dispensing chute." Ex. 1005, 10:41-44, 13:1-4; 25:48-52 (describing conducting process 1800 for duplicating a transponder key in conjunction with process 300 for duplicating a key, including cutting the blade of the key blank), Fig. 2 (*see* "Key Dispensing Chute" below). In addition to dispensing a physical copy of the duplicate key at a kiosk, Marsh also describes providing the duplicate key to a customer from a remote location by allowing the

customer to "request a physical copy via mail order," as well as to "make a physical copy with any suitable key duplication hardware." Ex. 1005, 15:10-27.

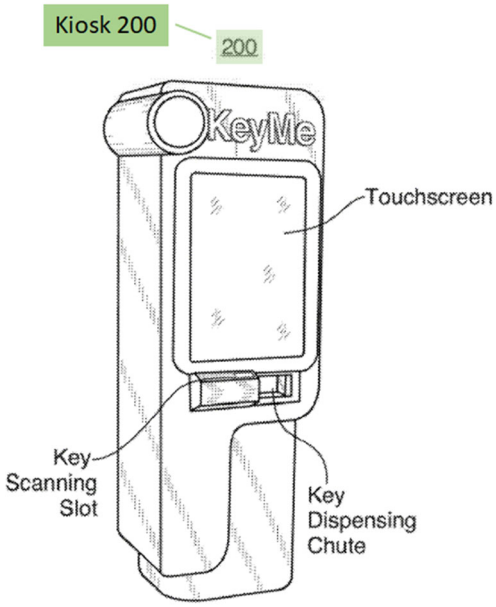


FIG. 2

110. A POSITA would have been motivated to implement Marsh's capabilities of providing the generally programmable duplicate key to a customer (e.g., a user of Marsh's kiosk 200) after cutting the blade of the key blank into Chies' system for the same reasons described above for Claim 4(a). A POSITA would have had a reasonable expectation of success in doing so, as this would have required only routine skill, knowledge, and standard computer programming and/or manufacturing techniques. The combination of Chies and Marsh thus renders obvious providing the generally programmable duplicate key to a customer.

4. Claim 5: The method of claim 1, further comprising: placing said generally programmable duplicate key in proximity with said vehicle, sampling communications between said generically programmable duplicate key and said vehicle to generate authentication sample data.

111. Chies anticipates and/or renders obvious claim 1. Chies does not disclose placing a generally programmable duplicate key in proximity with a vehicle and sampling communications between the generally programmable duplicate key and the vehicle to generate authentication sample data.

112. Marsh discloses placing a transponder key in proximity with a vehicle in order to obtain a responsive signal from the transponder key. For example, Marsh discloses the use of a "portable scanning device" associated with kiosk 200 to be used with a "device that emits the signal that includes the particular information (e.g., the automobile corresponding to the transponder key)." Ex. 1005, 24:34-43. The portable scanning device "can capture one or more signals emitted by the transponder by the transponder key responsive to the signal emitted by, for example, the automobile." *Id.*, 24:44-47. The portable scanning device "can include an input device (e.g., a button), and process 1800 can instruct a user to operate the input device (e.g., by pressing the button) when the user has brought the transponder key into the automobile and/or performed any other suitable actions (such as operating an ignition of the automobile)." *Id.*, 24:47-53, Fig. 18.

113. A POSITA would have understood that Marsh's disclosure describes placing a transponder key in proximity with a vehicle, for example, to sample communications between the transponder key and the vehicle to generate authentication sample data, such as when a transponder key must first receive a signal emitted by the vehicle before emitting information that can be used to create a duplicate transponder key:

> [I]n the case of an automobile key using a passive transponder chip, the mechanisms described herein for duplicating transponder keys and managing key information thereof can cause a signal to be emitted that, in turn, causes the passive transponder chip of the automobile key to emit a responsive signal that is required by the automobile in order to allow the automobile to be started and/or turned on. This responsive signal can be used to identify the properties and/or programming of a transponder chip to be used to create a duplicate of the automobile key.

Ex. 1005, 16:8-19, 24:7-33. Using the same rationale taught by Marsh, a POSITA would have found it obvious to place a generally programmable duplicate key (e.g., a transponder key blank) in proximity with a vehicle to generate similar authentication sample data, such as to determine whether the key blank is an appropriate model and type for the vehicle at issue and to obtain the sample data emitted from the key blank. Sampling communications between a transponder key

and a vehicle in this manner was routine and well-known in the art, and a POSITA would have found it obvious to practice the method of sampling communications between a transponder key and a vehicle (as taught by Marsh) in the key duplication process taught by Chies in order to obtain the same benefits taught by Marsh, i.e., to generate responsive data from the transponder in proximity to the vehicle that is useful to the key duplication process. A POSITA would have had a reasonable expectation of success in doing so with the blank transponder key, as this would have required only routine skill, knowledge, and techniques already taught by the prior art.

114. The combination of Chies and Marsh thus renders obvious placing said generally programmable duplicate key in proximity with said vehicle, sampling communications between said generically programmable duplicate key and said vehicle to generate authentication sample data.

5. Independent Claim 6

i. 6(pre): A vehicle access device duplication system comprising:

115. Chies discloses a vehicle access device duplication system. Ex. 1004, 1:10-17, 4:26-29.

ii. 6(a): a non-transitory computer-readable storage medium storing executing instructions that, when executed, cause a cloning application to perform steps comprising:

116. Chies discloses a computer (e.g., computer 15) that includes a cloning application (e.g., the "application program") performing steps for duplicating a transponder key. Ex. 1004, 5:33-6:28 ("Installed in the same computer 15 there is an application program that is adapted to handle the flow of data coming in from the read/write unit 13."), 7:12-19 ("The read/write unit 13 transmits the INCODE . . . to the computer connected thereto. The computer 15, as duly provided with the application program, receives such information and sends it to the serving computer or server 25."). While Chies does not expressly describe a storage medium within computer 15, a POSITA would have understood that computer 15 would include a readable storage medium for permanent (i.e., non-transitory) storage of programs and data, such as a hard disk drive storing the "application program" and any associated computer code allowing the computer 15 to execute instructions associated with the application program. Such storage devices are ubiquitous in computers and have been used for decades, and a POSITA would have interpreted Chies' computer 15 as including such a storage medium for storing and executing the computer code for the application program.

117. To the extent that Chies lacks express details regarding the storage of computer 15, the use of a non-transitory computer-readable storage medium that

stores instructions for causing an application to perform certain steps would have been obvious in view of Marsh. Marsh's kiosk 200 includes a computer containing a non-transitory computer-readable storage medium (e.g., the computer memory within the kiosk such as a hard disk or other computer-readable media). Ex. 1005, 19:54-20:18 (kiosks 200 "can include any of a general purpose device such as a computer," which may include "any suitable components such as a hardware processor" and "memory"), 20:47-58, 26:54-67 (describing "computer readable media [that] can be used for storing instructions for performing the processes described herein" as including "non-transitory computer-readable media" such as "magnetic media (such as hard disks)," "optical media," and "semiconductor media"). This storage medium in Marsh's kiosk 200 stores executing instructions that, when executed, cause a cloning application to perform certain steps. Ex. 1005, 20:47-67 (describing "a non-transitory computer-readable medium . . . for storing a computer program for controlling hardware processor 1702," wherein "[h]ardware processor 1702 can use the computer program to execute the mechanisms described herein for duplicating keys and/or duplicating transponder keys and managing the key information thereof"), 24:34-61 (kiosk 200 is a "computing device executing process 1800" for duplicating a transponder key), 26:30-53 (describing the use of a computer program "that causes a processor (such as hardware processor 110, hardware processor 1702 and/or hardware processor

1722) to execute the mechanisms described herein," such as a computer program "written in a programming language recognizable by kiosk 200"), Fig. 13.

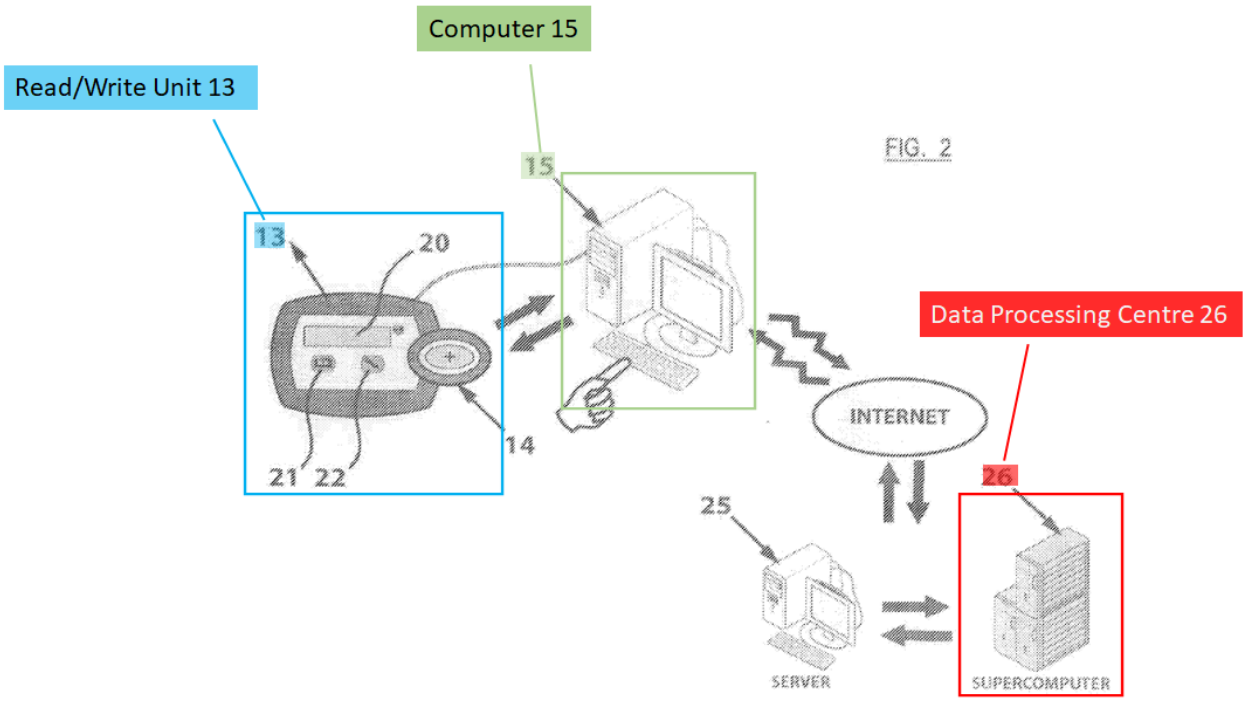
118. To the extent it would have been necessary to modify Chies' computer 15 to include a non-transitory computer-readable storage medium such as Marsh's, a POSITA would have been motivated to do so to allow computer 15 to store the application program in permanent storage, thereby allowing repeated use of the application program, and would have had a reasonable likelihood of success in doing so given that the use of such storage mediums (and storing programs on such storage mediums) was routine and well-known in the field.

119. Chies alone, or in combination with Marsh, thus renders obvious a non-transitory computer-readable storage medium storing executing instructions that, when executed, cause a cloning application to perform steps.

iii. 6(b): determining that a first device of a customer is communicating with a transmitter/receiver device (t/r device), the t/r device including an antenna ring for wirelessly communicating with an access device for a vehicle;

120. Chies discloses a t/r device (e.g., read/write unit 13) that wirelessly communicates with an access device for a vehicle (e.g., key 10). For example, Chies discloses duplicating original key 10 by inserting it into a read/write unit 13 (**blue** in annotated Figure 2 below) and establishing wireless communication via the antenna of the read/write unit 13. Ex. 1004, 4:26-5:2, Fig. 2. The read/write

unit 13 is connected to a computer 15 (**green** below), which runs an "application program that is adapted to handle the flow of data coming in from the read/write unit 13." *Id.*, 5:33-6:4. The computer 15 is connected to the Internet and transmits key information to remote servers including server 25 and data processing centre 26 (**red** below). *Id.*, 5:33-6:28.



121. Chies further discloses determining that a first device of a customer (e.g., a master transponder key such as original key 10) is communicating with a t/r device (e.g., read/write unit 13). Ex. 1004, 6:30-7:10 (describing, after a key 10 is introduced into read/write unit 13, using an "interactive read button 21" to "read[] the identification data of the key 10, thereby recognizing the type of key being

handled" and optionally "the read/write unit 13 concurrently performs an inquiry operation to query the key 10" such that "a wireless communication is then established between the read/write unit 13 and the code-based electronic recognition means 11 of the key 10."). To the extent this claim requires that the cloning application (e.g., Chies' application program) performs this "determining" step, Chies discloses that its application program does so when it receives the key information transmitted to it by read/write unit 13. Ex. 1004, 7:13-18 ("The read/write unit 13 transmits the INCODE, which contains also the information that identifies the 15 key 10 to be duplicated in a univocal manner, owing to its having been assigned a log, to the computer connected thereto. The computer 15, as duly provided with the application program, receives such information and sends it to the serving computer or server 25."). By successfully "receiving such information" (i.e., the INCODE data from the transponder key sent by read/write unit 13), the application program determines that the read/write unit 13 is communicating with a first device of a customer (e.g., key 10).

122. If it is asserted that the "determining" step requires more than receiving data from the t/r device at the cloning application, then Marsh's disclosure further underscores that this claim term would have been obvious to a POSITA. In particular, Marsh describes kiosk 200 automatically detecting the presence of a transponder key and establishing communication between the key

and the transponder antenna 1302 within the kiosk. Ex. 1005, 22:38-44 ("[P]rocess 1800 can use transponder antenna 1302 and/or any other suitable device or combination of devices to automatically detect the presence of a transponder key (e.g., a transponder key inserted in slot 1404, a transponder key held near transponder antenna 1302, etc.) using any suitable technique or combination of techniques."). Because Marsh's kiosk 200 is a computing device running software (i.e., a cloning application) that reads information from a transponder key via transponder antenna 1302 and transmits that information to a remote server (Ex. 1005, 17:51-18:9, 26:30-53), Marsh's disclosure of its kiosk 200 automatically detecting the presence of the key and then reading information from that key is the a manner of determining that the transponder key is communicating with transponder antenna 1302. A POSITA would have found it obvious to similarly implement automatic recognition in Chies' system such that the application program of computer 15 automatically determines when a transponder key is inserted into read/write unit 13 and communication between the two devices is established, for example, to eliminate the need for a user to take additional steps to allow the application program to begin reading security information from the transponder key via read/write unit 13. Doing so would have been within the capability of a POSITA, and a POSITA would have had a reasonable expectation

of success in doing so because it would require only routine and conventional computer programming techniques known in the art.

123. While Chies does not describe the shape of the antenna of read/write unit 13, a POSITA would have at least found it obvious to implement an antenna with a ring shape into read/write unit 13 because such shapes were ubiquitous and well-known in the art for devices reading information from transponders. For example, as shown in annotated Figure 15 below, Marsh's transponder antenna 1302 (**orange** below) within kiosk 200 may be a ring-shaped "loop antenna" that reads security information from a transponder key. Ex. 1005, 16:23-58, Fig. 15.

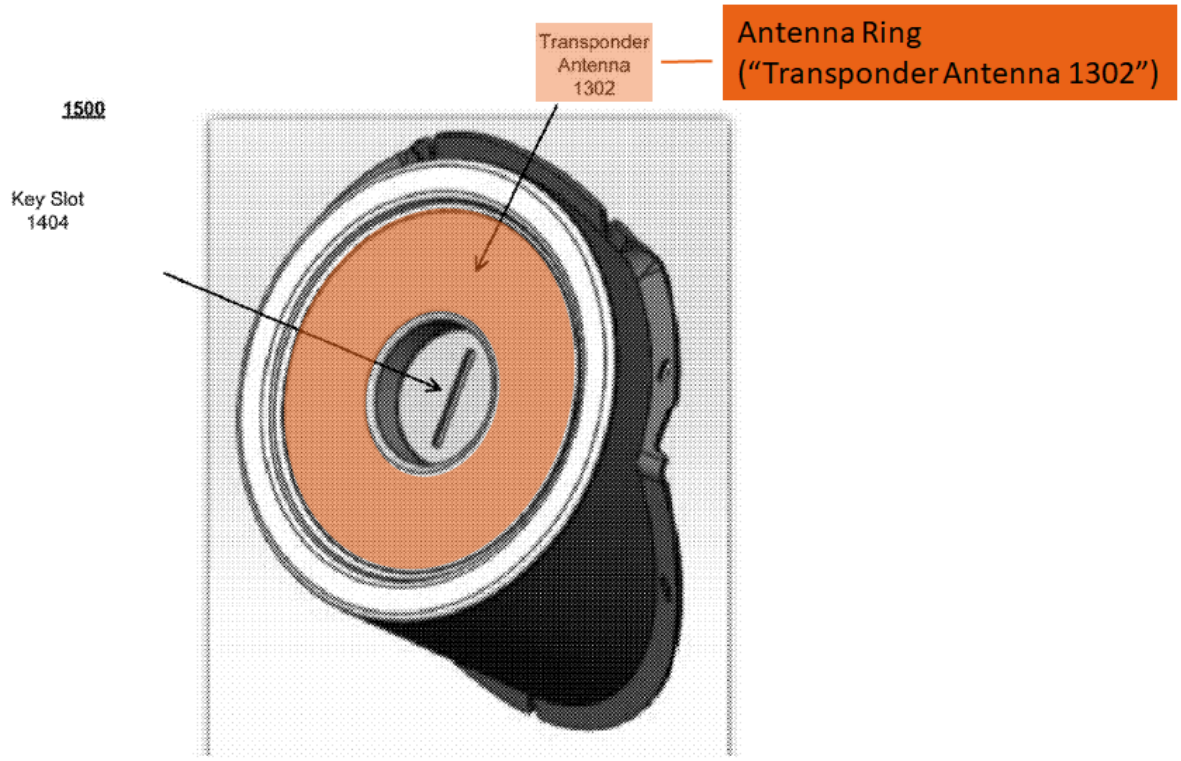


FIG. 15

124. Further demonstrating that such antennas were well-known in the art, Blalock discloses a ring-shaped antenna for reading information from a transponder key for a vehicle. Ex. 1006, 2:32-33 (describing "an antenna 215 which surrounds the slot 210" containing the transponder key), 3:21-37 ("The antenna 415 reads the transponder code off the transponder key 405 . . . ."), Fig. 4.

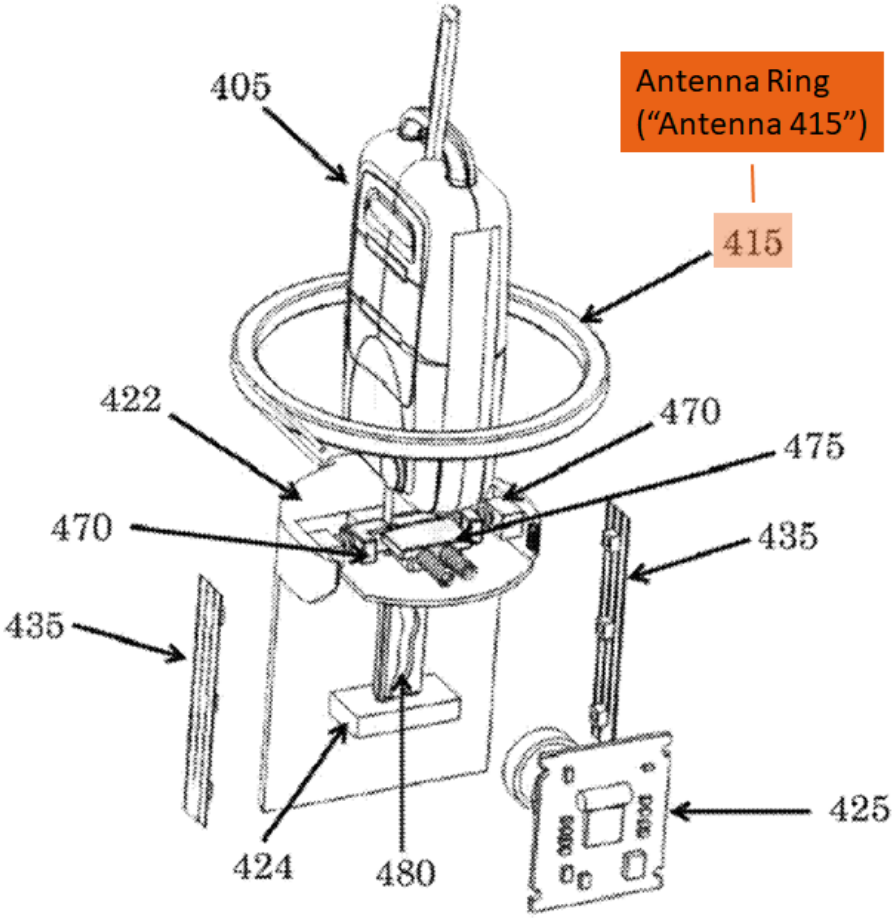


Figure 4

125. A POSITA would have implemented a ring antenna in Chies' read/write unit 13 to fully surround the transponder key to more reliably read

information from the transponder key, including by reading the transponder information regardless of the key's spatial orientation (a known benefit of such antenna designs). In any event, implementing a ring antenna in Chies' read/write unit 13 would have been a matter of simple design choice well within the skill of a POSITA, especially in view of the known benefits of using a ring-shaped antenna. A POSITA would have had a reasonable expectation of success for implementing a ring antenna into Chies' read/write unit 13 since such antennas are routinely used in the field for communicating with a transponder, and implementing such an antenna in Chies' read/write unit 13 would have required only routine skill and knowledge in the field.

126. Chies alone, or in combination with Marsh, thus renders obvious determining that a first device of a customer is communicating with a t/r device, the t/r device including an antenna ring for wirelessly communicating with an access device for a vehicle.

iv. 6(c): retrieving security information digitally stored by said access device;

127. As discussed above, Chies discloses a t/r device (e.g., read/write unit 13) that includes an antenna for wirelessly communicating with an access device (e.g., original key 10). Ex. 1004, 4:31-5:2, 6:6-13 ("[T]he read/write unit 13 is provided with . . . an antenna (not shown) for receiving and transmitting data"), 7:4-5 ("[A] wireless communication is then established between the read/write unit

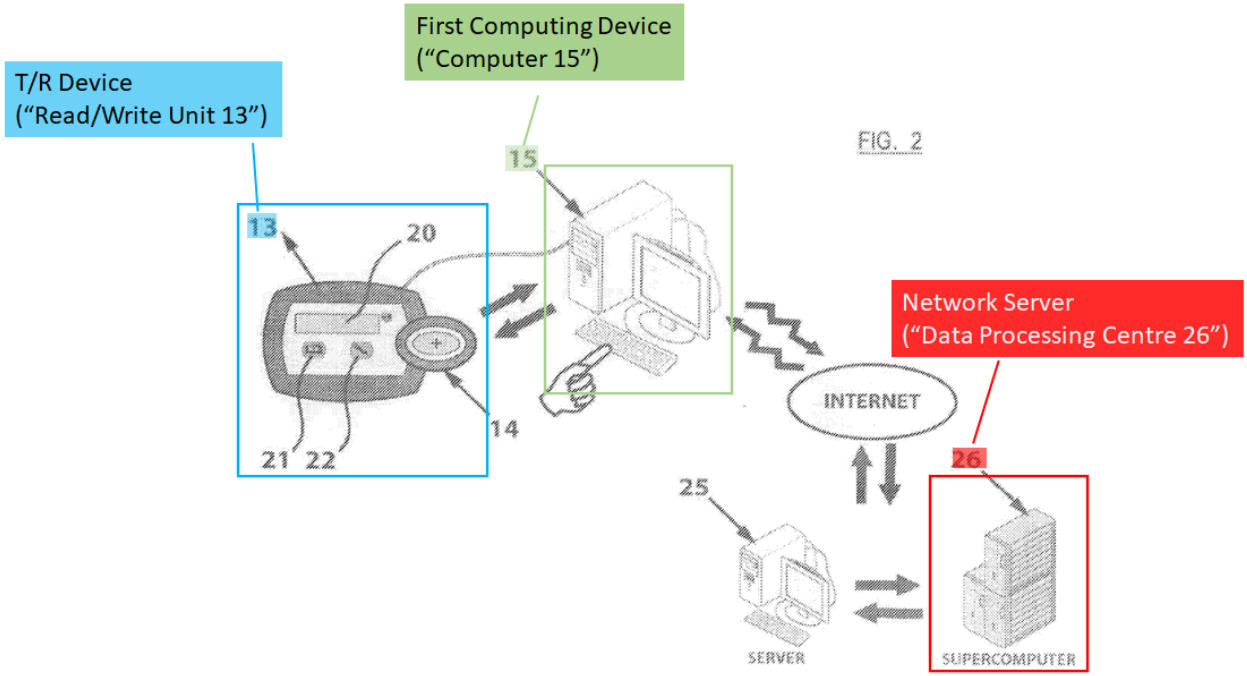
13 and the code-based electronic recognition means 11 of the key 10."). Chies further discloses retrieving security information digitally stored by the access device. Ex. 1004, 4:26-5:27 (describing storing security information in the memory of transponder key 10), 6:30-7:19 ("[A] key 10 to be duplicated is introduced in the appropriate receptacle 14 of a read/write unit 13. Via the interactive read button 21, the read/write unit 13 reads the identification data of the key 10, thereby recognizing the type of key being handled, and temporarily stores such data in the internal memory thereof.").

128. The information read from key 10 may be encrypted security information stored on the key. Ex. 1004, 6:30-7:10 ("In the assumption that the key 10 is of the type with a secret (encrypted) code, the read/write unit 13 concurrently performs an inquiry operation to query the key 10. Such query occurs through a code that is implemented in the read/write unit 13 and is solely known by the manufacturer."). Chies refers to the security information received from the master key by read/write unit 13 as "INCODE." Ex. 1004, 7:8-10 ("What the read/write unit 13 acquires is therefore a data-based information that is encrypted at random in a first encryption form, which shall be defined as INCODE hereinafter.").

v. 6(d): communicating the security information for said access device to a network server;

129. Chies discloses communicating the security information (e.g., the INCODE data from read/write unit 13) for said access device (e.g., original key

10) to a network server (e.g., data processing centre 26). Ex. 1004, 6:15-7:31. In particular, Chies discloses that read/write unit 13 transmits the INCODE data read from the original key to computer 15. Ex. 1004, 7:12-18 (The "computer 15, as duly provided with the application program, receives such information and sends it to the serving computer or server 25."). Server 25 then transmits the INCODE data to data processing centre 26 (to which server 25 is connected over the Internet), where the INCODE data is processed. *Id*., 6:15-28, 7:21-31, Fig. 2.



130. Marsh similarly discloses receiving security information from a transponder key 1304 at kiosk 200 and transmitting that information over a communication link 1308 to a network server (e.g., server 1310) for processing the transponder information. Ex. 1005, 17:51-18:9 ("[K]iosk 200 can cause information received by transponder antenna 1302 to be transmitted to a server

1310 over a communication link 1308 and/or a communication network"), 19:10-24 (describing features of Marsh's communication network 1606), 20:19-46 (same). Server 1310 may be "a transponder key information server 1310 that facilitates identification of a proper transponder chip and/or transponder chip programming for a particular transponder key." Ex. 1005, 20:36-46, Fig. 13.

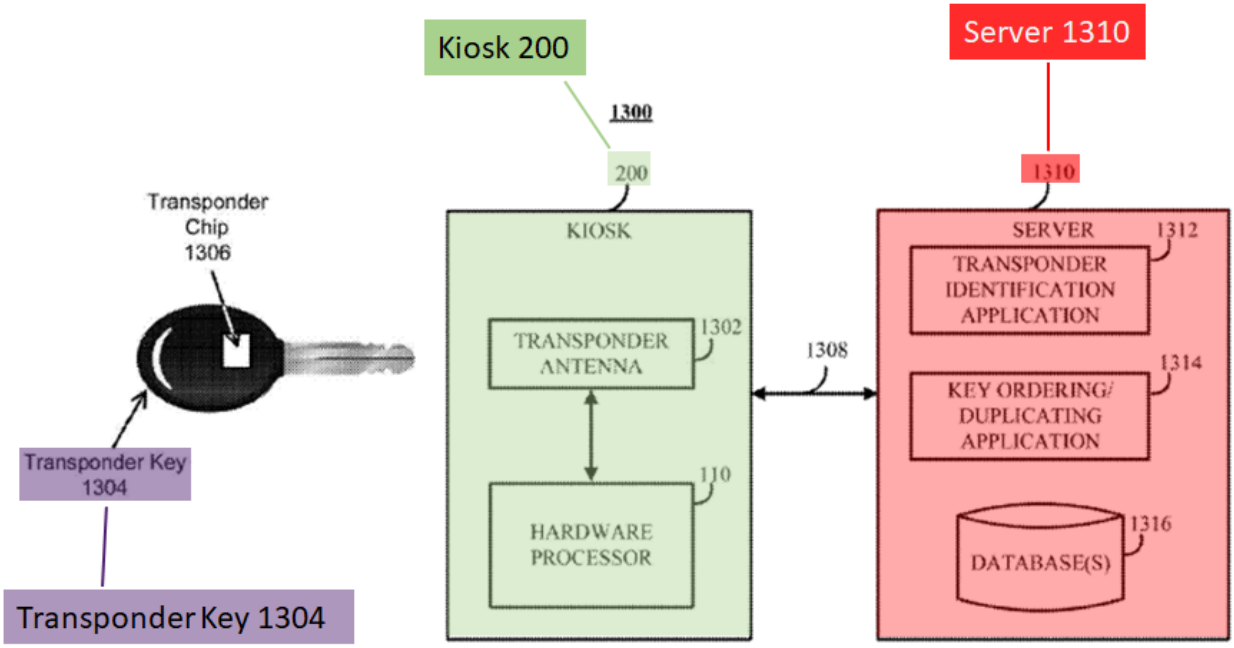


FIG. 13

vi. 6(e): generating a cloning security information; and

131. Chies discloses generating, by the network server (e.g., data processing centre 26), a cloning security information (e.g., "OUTCODE" data). Ex. 1004, 7:29-8:9. In particular, Chies describes processing the security information from the master key (e.g., the INCODE data) to prepare "OUTCODE"

data used to program a blank transponder key. Ex. 7:29-8:9 (describing a process wherein "the INCODE is decrypted," "the secret code is recognized and authenticated" such that "the single and sole secret code corresponding to the one contained in the microchip of the key 10 to be duplicated is acquired," and "[t]he data processing centre 26 encrypts the corresponding secret code . . . into a second encryption form, which shall be defined as OUTCODE hereinafter."), 8:21-34.

132. As discussed above for Claim 2, it would also have been obvious to generate a cloning security information using the cloning application (e.g., the application program of Chies' computer 15) in view of the disclosures of Chies and Marsh. Chies in combination with Marsh thus renders obvious generating a cloning security information.

vii. 6(f): transmitting the cloning security information to the t/r device to program a duplicate access device with the cloning security information; and

133. Chies discloses transmitting the cloning security information (e.g., "OUTCODE" data) to the t/r device (e.g., read/write unit 13) to program a duplicate access device (e.g., blank key 30) with the cloning security information. Ex. 1004, 8:11-34. After receiving the OUTCODE data, read/write unit 13 "transfers all identification data of [original key 10] into the microchip of the new blank key 30, wherein this operation is performed with the aid of the interactive write button 22." Ex. 1004, 8:21-29. The OUTCODE data is then "decrypted and

definitively stored – along with the other identification data – solely inside the microchip of the blank key 30." Ex. 1004, 8:29-31. The blank key 30 is then "able to exactly emulate the functions of the configuration of the original key." Ex. 1004, 8:31-34.

viii. 6(g): a processor configured to execute the instructions.

134. As discussed above for Claim 6(a), Chies discloses the use of a computer 15 that includes the application program that reads key information from the master transponder key via the read/write unit 13. Ex. 1004, 5:33-6:28 ("Installed in the same computer 15 there is an application program that is adapted to handle the flow of data coming in from the read/write unit 13."), 7:12-19 ("The read/write unit 13 transmits the INCODE . . . to the computer connected thereto. The computer 15, as duly provided with the application program, receives such information and sends it to the serving computer or server 25."). While Chies does not expressly describe a processor within computer 15, a POSITA would have understood that computer 15 would include a processor that executes instructions associated with the application program. Such processors are ubiquitous in computers and have been used for decades, and a POSITA would have interpreted Chies' computer 15 as including such a processor.

135. To the extent that Chies lacks express details regarding a processor in computer 15, the use of such a processor would have been obvious in view of

Marsh. Marsh discloses a processor configured to execute instructions stored on a non-transitory computer-readable storage medium, such as the computer processor within kiosk 200. Ex. 1005, 19:54-20:18 (kiosks 200 "can include any of a general purpose device such as a computer," which may include "any suitable components such as a hardware processor" and "memory"), 20:47-62 (describing "a non-transitory computer-readable medium . .. for storing a computer program for controlling hardware processor 1702," wherein "[h]ardware processor 1702 can use the computer program to execute the mechanisms described herein for duplicating keys and/or duplicating transponder keys and managing the key information thereof"), 24:34-61 (kiosk 200 is a "computing device executing process 1800" for duplicating a transponder key), 26:30-53 (describing the use of a computer program "that causes a processor (such as hardware processor 110, hardware processor 1702 and/or hardware processor 1722) to execute the mechanisms described herein," such as a computer program "written in a programming language recognizable by kiosk 200").

136. To the extent it would have been necessary to modify Chies' computer 15 to include a processor to execute the instructions stored on a non-transitory computer-readable storage medium such as Marsh's, a POSITA would have been motivated to do so to allow computer 15 to execute the instructions, thus allowing Chies' application program to function, and would have had a reasonable

likelihood of success in doing so given that the use of such processors is routine and well-known in the field.

137. Chies alone, or in combination with Marsh, thus renders obvious a processor configured to execute instructions.

IX. Conclusion

138. For the reasons set forth above, I believe claims 1-6 of the ’455 patent are unpatentable in view of the prior art.

139. In signing this declaration, I understand that the declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I acknowledge that I may be subject to cross-examination in this case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

140. I declare that all statements made herein of my knowledge are true, that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: 11/2/22 By: 

Dr. Paul C. Clark

APPENDIX A

PAUL C. CLARK
4705 Broad Brook Drive, Bethesda, Maryland, 20814
703.628.9500
paul@securemethods.com

SKILLS SUMMARY

A senior executive with extensive IT design, development and deployment experience, Dr. Clark has twenty five years of direct technical and management experience in the computer and systems engineering environment. He is the President of Paul C. Clark LLC. His specialties include complex commercial development and deployment of scalable secure network and data processing systems. He has served as keynote speaker, expert witness for high profile commercial clients and before Congress. He has served on federal advisory committees and as an adjunct professor at The George Washington University.

EMPLOYMENT

President and CTO July 1999 – Present
SecureMethods Inc. / Paul C. Clark LLC
Bethesda, MD

- Serves as Managing Director
- Manages operations and sales staff.
- Manages commercial product development staff.
- Provides product design, development and deployment guidance.
- Provides sales engineering support and collected customer feedback.
- Defines and communicates strategic technical vision.
- Successfully directed the development and deployment of commercial products on multiple Windows and Unix Platforms

Chief Scientist Jan. 1995 – July 1999
DynCorp Network Solutions
Fairfax, VA

- Managed technical staff and deliverables for multiple large projects.
- Provided project design, development and troubleshooting guidance.
- Provided customer technical interface and problem resolution.
- Designed and deployed next generation architecture for high volume network database and storage systems.
- Created a suite of secure products marketed and sold to DoD and the Federal Government.
- Provided corporate-wide technical consultation and support.

Senior Security Engineer Sept. 1990 – Jan. 1995
Trusted Information Systems
Glenwood, Maryland

- Participated in the design and implementation of the reference implementation of Privacy Enhanced Mail (PEM) with public and secret key encryption to provide security services for electronic transmissions.
- Designed NIST's Smartcard API (SCAPI). Implemented the SCAPI for the NIST 250 and utilized it to perform cryptographic operations for PEM.
- Implemented X.500 Certification and Distinguished Naming support for PEM. Functions supported included: CA and user registration, revocation, and high speed database for certificate storage and retrieval.
- Systems design and programming, including the TCB, for the Trusted Xenix and Trusted Mach MLS operating systems. Tasks included MAC labeling and audit strategies, as well as application development.
- Designed and implemented multilevel electronic mail and HTTP proxy for Trusted Mach. Task included cryptographic support for the TCB as well as application specific validation and semantic checking for up/downgrade.
- Inventor of the Boot Integrity Token System (BITS), which provides hardware, enforced authentication within the boot sequence and guarantees operating system integrity using smart tokens. U.S. Patent Nos. 5,448,045; 5,892,902 and 8,695,066

Technical Lead Nov 1989 – Sept 1990
GTE Government Systems
Rockville, MD

- Managed the SAFE91 testbed effort with responsibilities including: budgets, schedules, customer negotiations, briefings, training, and tasking of technical staff. The testbed was created to simulate the client environment for the purpose of evaluating software packages and platforms for use by the client.
- Developed network load simulations for OS/2 LAN Manager to determine performance characteristics during periods of heavy traffic. This task included the design and development of a multi-threaded connection server under OS/2 version 1.0
- Designed and implemented an X Windows interface for the Minstrel System. This included the individual development of 20,000 lines of code in less than two months.
- Developed and taught DEC Windows and X Windows classes for GTE technical personnel. Responsible for instruction and problem solving throughout the subsequent development cycle.

Systems Engineer May 1985 – Nov. 1989
Ultrasystems Defense and Space

- Redesigned the Morse Mission Trainer while coordinating and tasking a team of programmers. During this time, received the President's quarterly award for outstanding performance. Task included systems and network and kernel level programming on SCO Xenix.
- Designed and implemented a database file server, benchmarked at over 100 retrievals per second on a million entry database. System implemented B* trees in C and ran on a Sun 3/260 workstation.
- Designed and implemented a satellite mission scheduler for multiple vehicles with multiple resources. Task included defining areas of interest (AOIs), flight paths, and optimal time allocation of resources.
- Designed and implemented the Ultraplot graphics tool to produce line and scatter plots, as well as bar graphs and histograms from large datafiles. This utility was implemented utilizing the DI3000 (device independent) graphics package.

HARDWARE Mainframe, Workstation, PC, including: IBM, DEC, Sun, HP, SGI, Intel

SOFTWARE UNIX (Linux, System V, BSD, Solaris, AIX, IRIX, Xenix), MS DOS/Windows, VMS, OS/2, X Windows, DEC Windows, Presentation Manger, TCP/IP, X25, Xenix Net, IBM LAN, DEC Net, Ethernet, Token Ring

EDUCATION

DSc. in Computer Science
Concentration in Security, Graphics, Intellectual Property Law
The George Washington University, 1994

M.S. in Electrical Engineering and Computer Science
University of Southern California, 1988

B.S. in Mathematics
University of California Irvine, 1986

Graduate level study in all major areas of computer science

REPRESENTATIVE PUBLICATIONS

"BITS – A Smartcard Protected Operating System," with Lance Hoffman, Communications of the ACM, November 1994.

"Service Layering Promotes Secure Data Exchange in Diverse Environments," Computer News, October 23, 1995

"Threats Posed to Cryptographic Applications by Random Numbers," presented to the RSA Data Security Conference, January 1996.

"A Reference Model for Electronic Commerce," with Daniel J. Blum and John Jauregui, Messaging Magazine, December 1996, Volume 2, Number 7.

"Secure Compartmented Data Access over an Untrusted Network Using a COTS-based Architecture," with Marion C. Meissner, and Karen O. Vance, Presented to the Annual Computer Security Applications Conference (ACSAC'00), New Orleans, December, 2000. Later published in "Statistical Methods in Computer Security," Marcel Dekker, ISBN 0-8247-5939-7, edited by William W. S. Chen, 2005

ADDITIONAL INFORMATION

(1) Dr. Clark was a member of the Federal Advisory Committee for Key Management Infrastructure (KMI); he was Chairman of the Interoperability Working Group for Cryptographic Key Recovery.

(2) Dr. Clark served as a Cooperative Research and Development Agreements (CRADA) partner, which is a joint effort between the National Institute of Standards and Technology (NIST) and several companies formed to begin development of the elements of a Public Key Infrastructure (PKI). A core element of this effort is the development of a Minimum Interoperability Specification for PKI components MISPC.

(3) Dr. Clark serves as an adjunct professor in the Electrical Engineering and Computer Science Department at The George Washington University. He teaches doctoral level cryptography and computer security courses.

(4) Speaker at The Federal Information Security Conference; presented "Embedded Security Deployment," Colorado Springs, CO, March 31, 2006

(5) Keynote Speaker for the Washington DC Bar Association; presented "Security for the Networked Computing Environment," August 8, 2005

(6) Appeared before Congressional committee to provide expert testimony; presented "Advanced Technology for Border Control," July 23, 1998.

(7) Keynote speaker at Mass Storage Conference; presented "Secure Data Access Over Public Networks," New Orleans, LA, October, 1996

(8) Keynote speaker at health care convention, presented "Security for Health Care Records," Nashville, TN, May, 1997.

(9) Keynote speaker at the USDA Plant and Genome Conference; presented "A Secure Architecture for Data and Systems," Nimes, France, October, 1997

(10) Speaker at IEEE Technical Meeting; presented "A Comprehensive Security Architecture," Virginia, June, 1996.

LEGAL CASES

Testifying expert for Google et al – ContentGuard v Google – Verdict for Google

Testifying expert for TransPerfect v MotionPoint – Verdict for Transperfect

Testifying expert for Cisco – VirNetX v Cisco – Verdict for Cisco

Testifying expert for CME – RealTime v CME – Summary judgment for CME

Testifying expert for IBM et al – Tecsec v IBM – Summary judgment for IBM

Testifying expert for Netflix et al – Parallel Networks v Netflix – Case dismissed

Testifying expert for Oracle - EpicRealm v Oracle – Summary judgment for Oracle

Testifying expert for Lucent - Microsoft v Lucent –Verdict for Lucent

Testifying expert for Oracle - Mangosoft v Oracle – Summary judgment for Oracle

Testifying expert for RSA Data Security – Digital Privacy v RSA – Summary judgment for RSA at the Markman