FILE HISTORY

US 5,949,880

PATENT:	5,949,880
INVENTORS:	Curry, Stephen M. Loomis, Donald W. Bolan, Michael L.
TITLE:	Transfer of valuable information between a secure module and another module
APPLICATION NO:	US1997978798A
FILED:	26 NOV 1997
ISSUED:	07 SEP 1999
COMPILED:	12 JAN 2012

70647 U.S. PTO
08/978798
11/26/97

ISSUE CLASSIFICATION	
Class	Subclass

SCANNED 4 cms

[redacted]

BEST COPY 5949880

UTILITY SERIAL NUMBER	PATENT DATE SEP 07 1999	PATENT NUMBER 5949880

SERIAL NUMBER	FILING DATE	CLASS	SUBCLASS	GROUP ART UNIT	EXAMINER
08/978,798	11/26/97 RULE 60	380	24	~~3642~~ 2766	White

APPLICANTS

STEPHEN M. CURRY, DALLAS, TX; DONALD W. LOOMIS, COPPELL, TX; MICHAEL L. BOLAN, DALLAS, TX.

white

CONTINUING DATA***********************
VERIFIED yes THIS APPLN IS A DIV OF 08/594,975 01/31/96
CW

FOREIGN APPLICATIONS**********
VERIFIED none
CW

CERTIFICATE
APR 25 2000
OF CORRECTION

FOREIGN FILING LICENSE GRANTED 03/04/98

		AS FILED →	STATE OR COUNTRY	SHEETS DRWGS.	TOTAL CLAIMS	INDEP. CLAIMS	FILING FEE RECEIVED	ATTORNEY'S DOCKET NO.
Foreign priority claimed 35 USC 119 conditions met Verified and Acknowledged	□ yes ☒ no □ yes ☒ no CW Examiner's initials		TX	8	6	1	$790.00	20661-429D1

ADDRESS

STEVEN R GREENFIELD
JENKENS & GILCHRIST
1445 ROSS AVENUE
SUITE 3200
DALLAS TX 75202

TITLE

TRANFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

U.S. DEPT. OF COMM./PAT. & TM—PTO-436L (Rev.12-94)

PARTS OF APPLICATION FILED SEPARATELY

10.16.99
Applications Examiner

NOTICE OF ALLOWANCE MAILED		CLAIMS ALLOWED	
10/16/98	Carmen D. White Assistant Examiner	Total Claims 6	Print Claim 16

ISSUE FEE			DRAWING		
Amount Due 1320.00	Date Paid 1-19-99	THOMAS H. TARCZA SUPERVISORY PATENT EXAMINER GROUP ~~2200~~ 3640 Thomas Tarcza Primary Examiner	Sheets Drwg. 8	Figs. Drwg. 7	Print Fig. 1
			ISSUE BATCH NUMBER K51		

Label Area

PREPARED FOR ISSUE

WARNING: The information disclosed herein may be restricted. Unauthorized disclosure may be prohibited by the United States Code Title 35, Sections 122, 181 and 368. Possession outside the U.S. Patent & Trademark Office is restricted to authorized employees and contractors only.

Form PTO-436A
(Rev. 8/92)

~~Formal Drawings (~~ shts) set

SCAN 4
OC

ISSUE FEE IN FILE

(FACE)

5,949,880

TRANFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

Transaction History

Date	Transaction Description
11/26/1997	Preliminary Amendment
11/26/1997	Information Disclosure Statement (IDS) Filed
11/26/1997	Information Disclosure Statement (IDS) Filed
1/21/1998	Initial Exam Team nn
3/3/1998	IFW Scan & PACR Auto Security Review
3/19/1998	Case Docketed to Examiner in GAU
8/10/1998	Notice Mailed--Application Incomplete--Filing Date Assigned
8/10/1998	Preexamination Location Change
9/30/1998	Case Docketed to Examiner in GAU
10/16/1998	Mail Examiner's Amendment
10/16/1998	Examiner's Amendment Communication
10/16/1998	Mail Notice of Allowance
10/16/1998	Notice of Allowance Data Verification Completed
1/19/1999	Workflow - Drawings Finished
1/19/1999	Workflow - Drawings Matched with File at Contractor
1/19/1999	Workflow - Drawings Received at Contractor
1/19/1999	Issue Fee Payment Verified
1/19/1999	Mailroom Date of Drawing(s)
1/28/1999	Drawing(s) Received at Publications
2/5/1999	Drawing(s) Processing Completed
2/5/1999	Drawing(s) Matched to Application
2/24/1999	Workflow - File Sent to Contractor
4/28/1999	Application Is Considered Ready for Issue
8/30/1999	Issue Notification Mailed
9/7/1999	Recordation of Patent Grant Mailed
10/1/1999	Workflow - Complete WF Records for Drawings
3/28/2000	Post Issue Communication - Certificate of Correction

PATENT APPLICATION

08978798

9-7

APPROVED FOR LICENSE □

INITIALS ____________

ISSUE

CONTENTS

Date Entered or Counted		CONTENTS	Date Received or Mailed
	1.	Application ____________ papers.	
	2.	[redacted] VOID	
	3.	IDS w/att	11/26/97
	4.	Pre A	11-26-97
5/26	5.	notice of allowability/Exmr's Amdt/B	10/16/98
2-2-99	6.	Formal Drawings (8 sht(s)) set 1	1-19-99
	7.		11-16-99
	8.		
	9.		
	10.		
	11.		
	12.		
	13.		
	14.		
	15.		
	16.		
	17.		
	18.		
	19.		
	20.		
	21.		
	22.		
	23.		
	24.		
	25.		
	26.		
	27.		
	28.		
	29.		
	30.		
	31.		
	32.		

(FRONT)

PATENT NUMBER

☆U.S.GOVERNMENT PRINTING OFFICE 1997-422-215

ORIGINAL CLASSIFICATION

CLASS	SUBCLASS
380	24

APPLICATION SERIAL NUMBER
08/978,798

APPLICANT'S NAME (PLEASE PRINT)
Stephen M. Curry et al

IF REISSUE, ORIGINAL PATENT NUMBER

CROSS REFERENCE(S)

CLASS	SUBCLASS (ONE SUBCLASS PER BLOCK)			
380	25			
705	39	42		

INTERNATIONAL CLASSIFICATION

H	0	4	L		9/00
					/
					/
					/

GROUP ART UNIT
3642

ASSISTANT EXAMINER (PLEASE STAMP OR PRINT FULL NAME)
Carmen White

PRIMARY EXAMINER (PLEASE STAMP OR PRINT FULL NAME)
Thomas A. Tarcza

PTO 270
(REV. 5-91)

ISSUE CLASSIFICATION SLIP

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE

Claim		Date							
Final	Original	5/25/98							
	~~1~~								
	~~2~~								
	~~3~~								
	~~4~~								
	~~5~~								
	~~6~~								
	~~7~~								
	~~8~~								
	~~9~~								
	~~10~~								
	~~11~~								
	~~12~~								
	~~13~~								
	~~14~~								
	~~15~~								
1	16	=							
2	17	=							
3	18	=							
4	19	=							
5	20	=							
6	21	=							
	22								
	23								
	24								
	25								
	26								
	27								
	28								
	29								
	30								
	31								
	32								
	33								
	34								
	35								
	36								
	37								
	38								
	39								
	40								
	41								
	42								
	43								
	44								
	45								
	46								
	47								
	48								
	49								
	50								

SYMBOLS

✓	Rejected
=	Allowed
- (Through numberal)	Canceled
+	Restricted
N	Non-elected
I	Interference
A	Appeal
O	Objected

Claim		Date							
Final	Original								
	51								
	52								
	53								
	54								
	55								
	56								
	57								
	58								
	59								
	60								
	61								
	62								
	63								
	64								
	65								
	66								
	67								
	68								
	69								
	70								
	71								
	72								
	73								
	74								
	75								
	76								
	77								
	78								
	79								
	80								
	81								
	82								
	83								
	84								
	85								
	86								
	87								
	88								
	89								
	90								
	91								
	92								
	93								
	94								
	95								
	96								
	97								
	98								
	99								
	100								

(LEFT INSIDE)

SEARCHED

Class	Sub.	Date	Exmr.
380	23	5/21/98	cAw
	24	↓	↓
	25		
705	39		
	40		
	42	✓	✓

INTERFERENCE SEARCHED

Class	Sub.	Date	Exmr.
380	24	5/20/98	Cw

SEARCH NOTES

	Date	Exmr.
APS Text Search	5/21/98	cAw

(RIGHT OUTSIDE)

US005949880A

United States Patent [19]

Curry et al.

[11] **Patent Number: 5,949,880**

[45] **Date of Patent: Sep. 7, 1999**

[54] **TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE**

[75] Inventors: **Stephen M. Curry**, Dallas; **Donald W. Loomis**, Coppell; **Michael L. Bolan**, Dallas, all of Tex.

[73] Assignee: **Dallas Semiconductor Corporation**, Dallas, Tex.

[21] Appl. No.: **08/978,798**

[22] Filed: **Nov. 26, 1997**

Related U.S. Application Data

[62] Division of application No. 08/594,975, Jan. 31, 1996.

[51] **Int. Cl.**^{6} **H04L 9/00**

[52] **U.S. Cl.** **380/24**; 380/25; 705/39; 705/42

[58] **Field of Search** 380/23, 24, 25; 705/39, 40, 42

[56] **References Cited**

U.S. PATENT DOCUMENTS

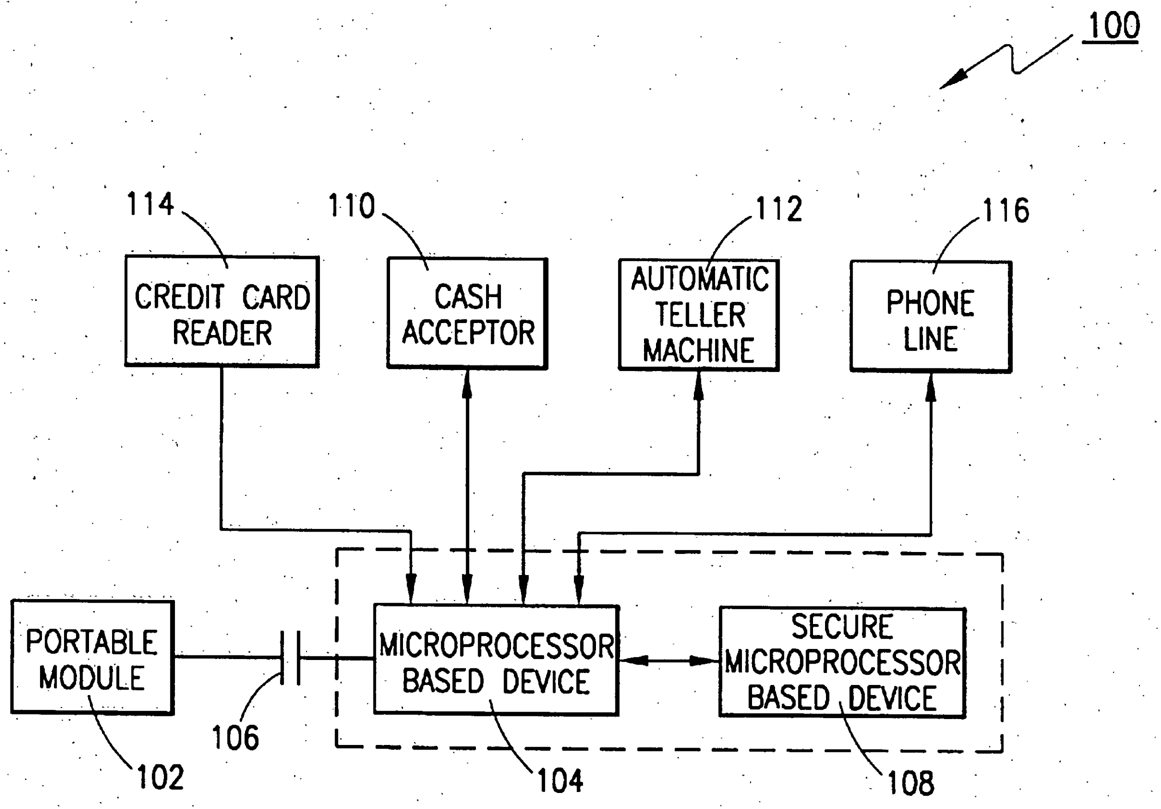
5,003,594	3/1991	Shinagawa	380/24
5,539,825	7/1996	Akiyama et al.	380/24
5,546,463	8/1996	Caputo et al.	380/25
5,577,121	11/1996	Davis et al.	380/24
5,621,796	4/1997	Davis et al.	380/24
5,642,419	6/1997	Rosen	380/23
5,671,280	9/1997	Rosen	380/24

Primary Examiner—Thomas H. Tarcza
Assistant Examiner—Carmen D. White
Attorney, Agent, or Firm—Jenkens & Gilchrist

[57] **ABSTRACT**

The present invention relates to system, apparatus and method for communicating valuable data from a portable module to another module via an electronic device. More specifically, the disclosed system, apparatus and method are useful for enabling a user to fill a portable module with a cash equivalent and to spend the cash equivalent at a variety of locations. The disclosed system incorporates an encryption/decryption method.

6 Claims, 8 Drawing Sheets



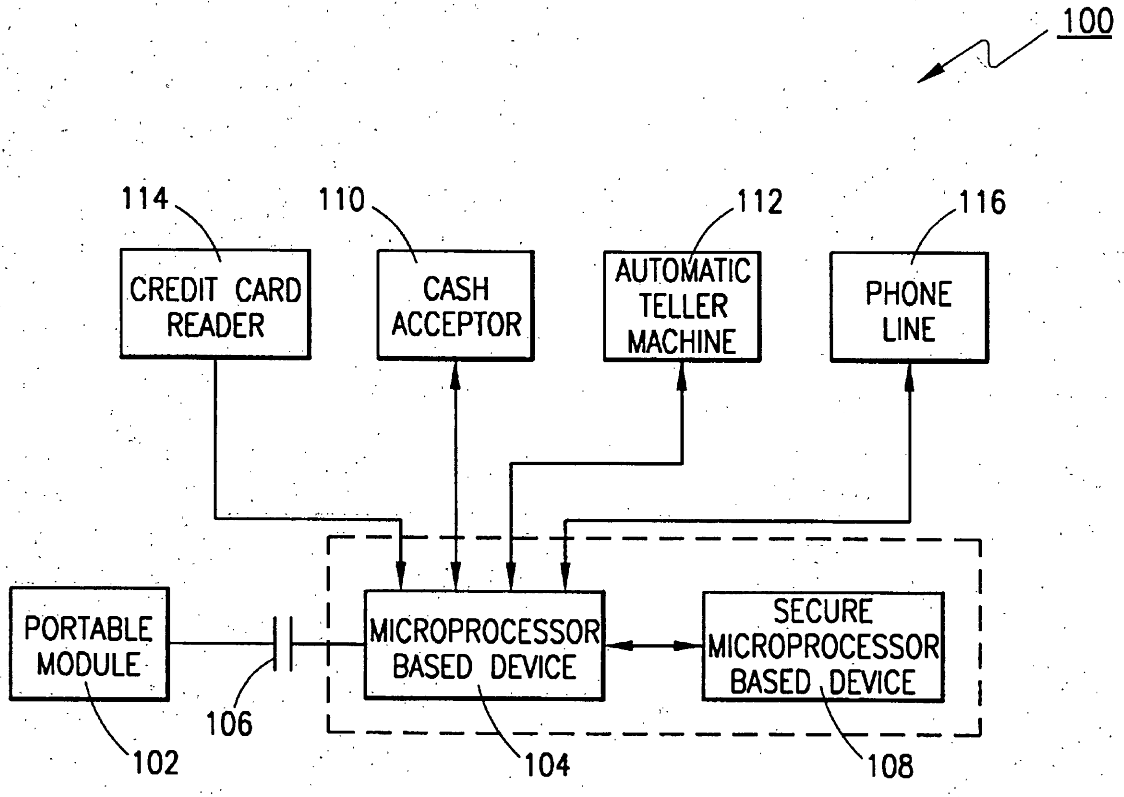

100
114
110
112
116
CREDIT CARD READER
CASH ACCEPTOR
AUTOMATIC TELLER MACHINE
PHONE LINE
PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MICROPROCESSOR BASED DEVICE
106
102
104
108

FIG. 1

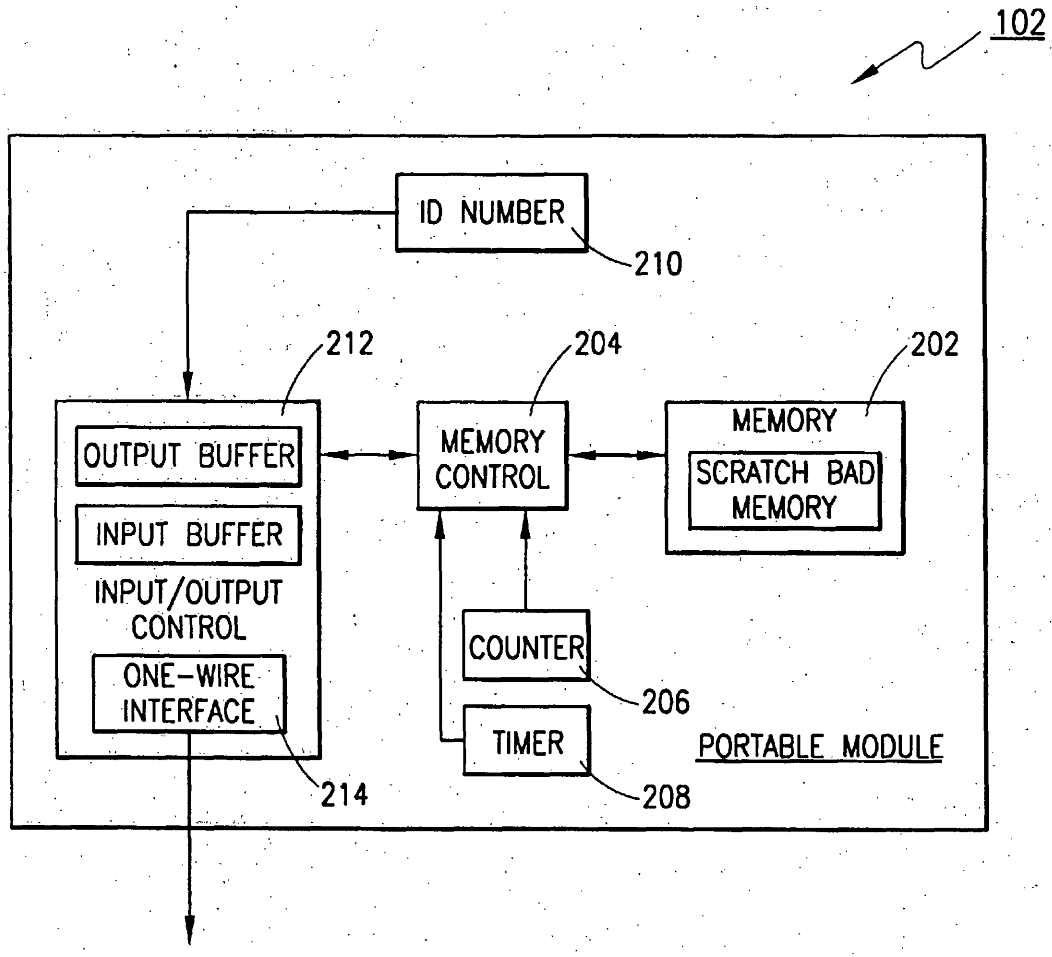

102
ID NUMBER
210
212
204
202
OUTPUT BUFFER
INPUT BUFFER
INPUT/OUTPUT CONTROL
ONE-WIRE INTERFACE
MEMORY CONTROL
MEMORY
SCRATCH BAD MEMORY
COUNTER
206
TIMER
208
214
PORTABLE MODULE

FIG. 2

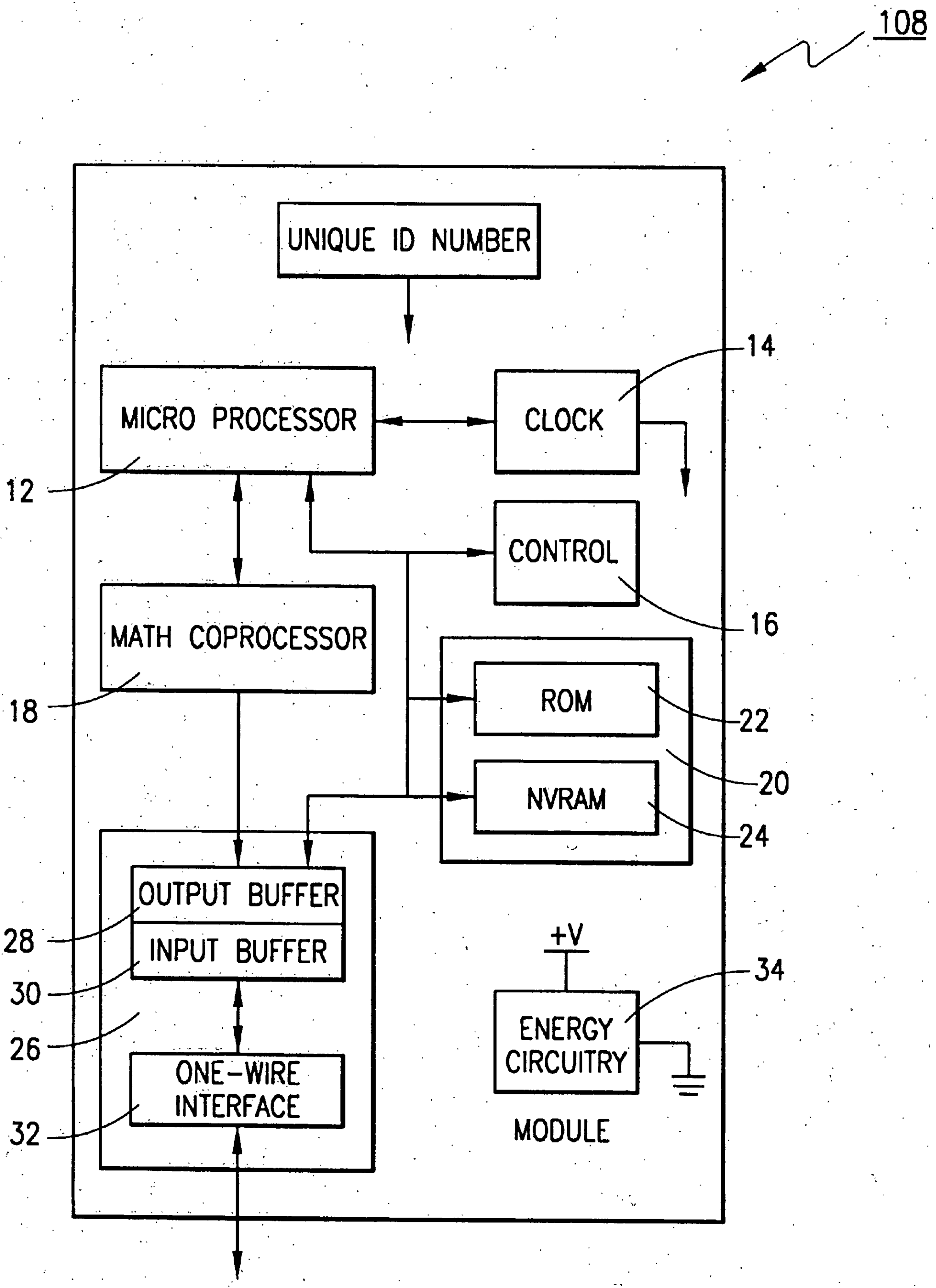

108
UNIQUE ID NUMBER
MICRO PROCESSOR
CLOCK
14
12
CONTROL
16
MATH COPROCESSOR
18
ROM
22
20
NVRAM
24
OUTPUT BUFFER
INPUT BUFFER
28
30
26
+V
34
ENERGY CIRCUITRY
ONE-WIRE INTERFACE
32
MODULE

FIG. 3

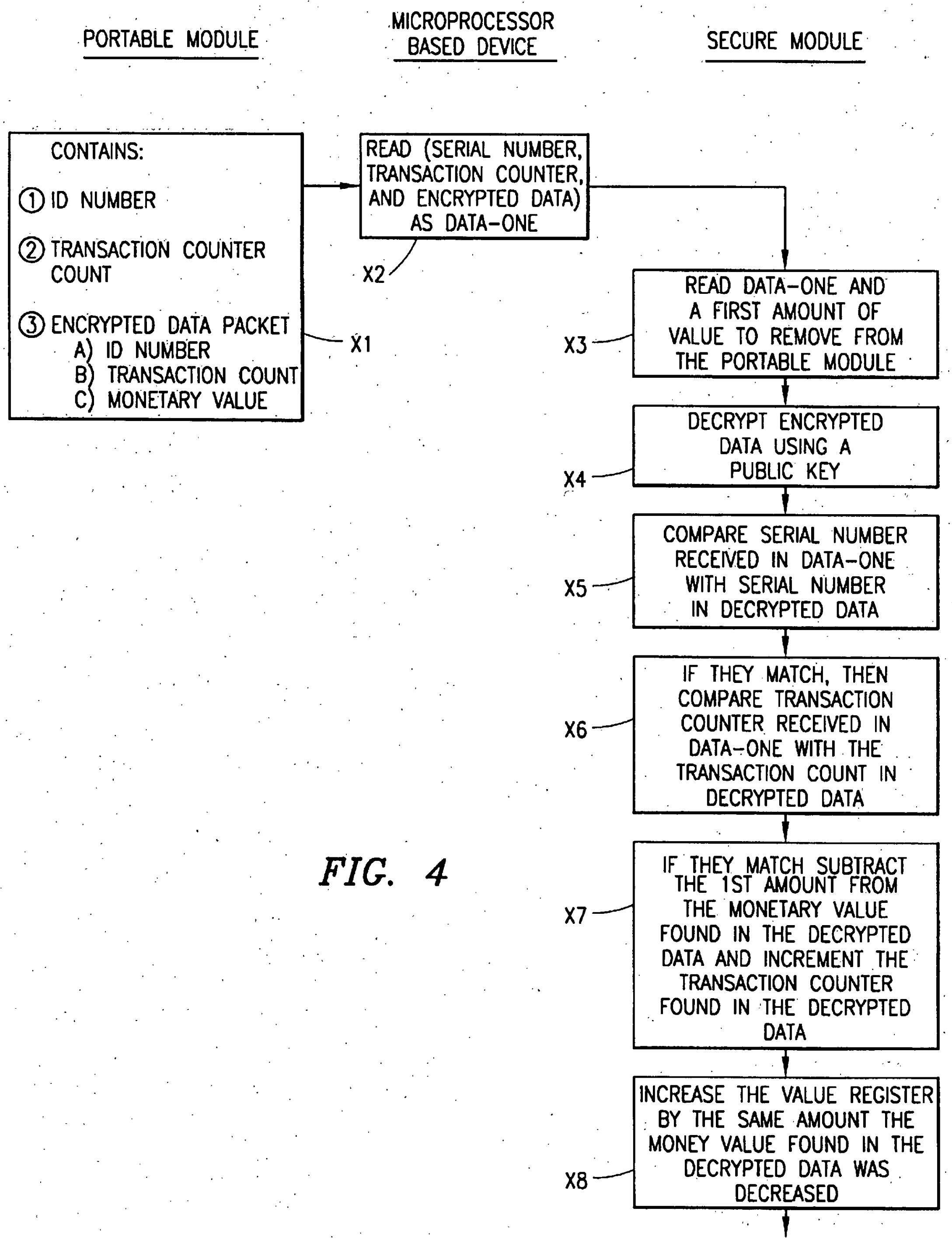

PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CONTAINS:
① ID NUMBER
② TRANSACTION COUNTER COUNT
③ ENCRYPTED DATA PACKET
A) ID NUMBER
B) TRANSACTION COUNT
C) MONETARY VALUE
X1
READ (SERIAL NUMBER, TRANSACTION COUNTER, AND ENCRYPTED DATA) AS DATA-ONE
X2
X3
READ DATA-ONE AND A FIRST AMOUNT OF VALUE TO REMOVE FROM THE PORTABLE MODULE
X4
DECRYPT ENCRYPTED DATA USING A PUBLIC KEY
X5
COMPARE SERIAL NUMBER RECEIVED IN DATA-ONE WITH SERIAL NUMBER IN DECRYPTED DATA
X6
IF THEY MATCH, THEN COMPARE TRANSACTION COUNTER RECEIVED IN DATA-ONE WITH THE TRANSACTION COUNT IN DECRYPTED DATA
X7
IF THEY MATCH SUBTRACT THE 1ST AMOUNT FROM THE MONETARY VALUE FOUND IN THE DECRYPTED DATA AND INCREMENT THE TRANSACTION COUNTER FOUND IN THE DECRYPTED DATA
X8
INCREASE THE VALUE REGISTER BY THE SAME AMOUNT THE MONEY VALUE FOUND IN THE DECRYPTED DATA WAS DECREASED

FIG. 4

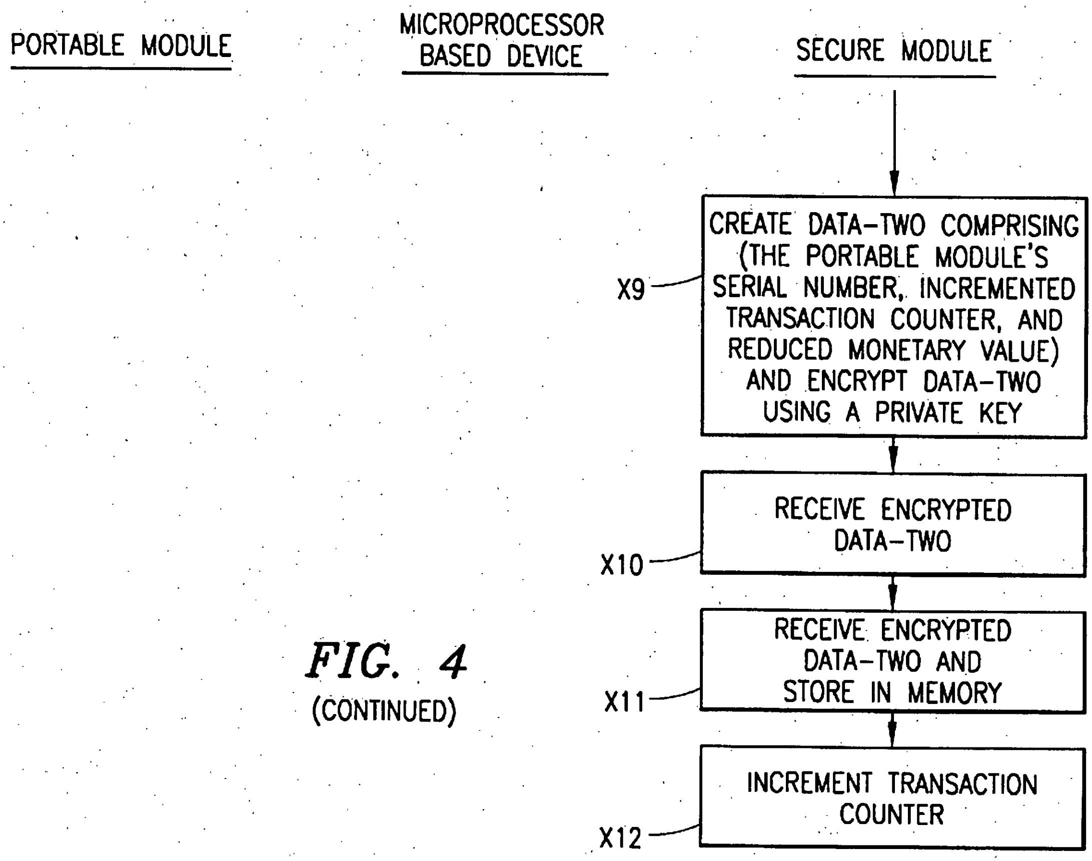

PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CREATE DATA-TWO COMPRISING (THE PORTABLE MODULE'S SERIAL NUMBER, INCREMENTED TRANSACTION COUNTER, AND REDUCED MONETARY VALUE) AND ENCRYPT DATA-TWO USING A PRIVATE KEY
X9
RECEIVE ENCRYPTED DATA-TWO
X10
RECEIVE ENCRYPTED DATA-TWO AND STORE IN MEMORY
X11
INCREMENT TRANSACTION COUNTER
X12

FIG. 4
(CONTINUED)

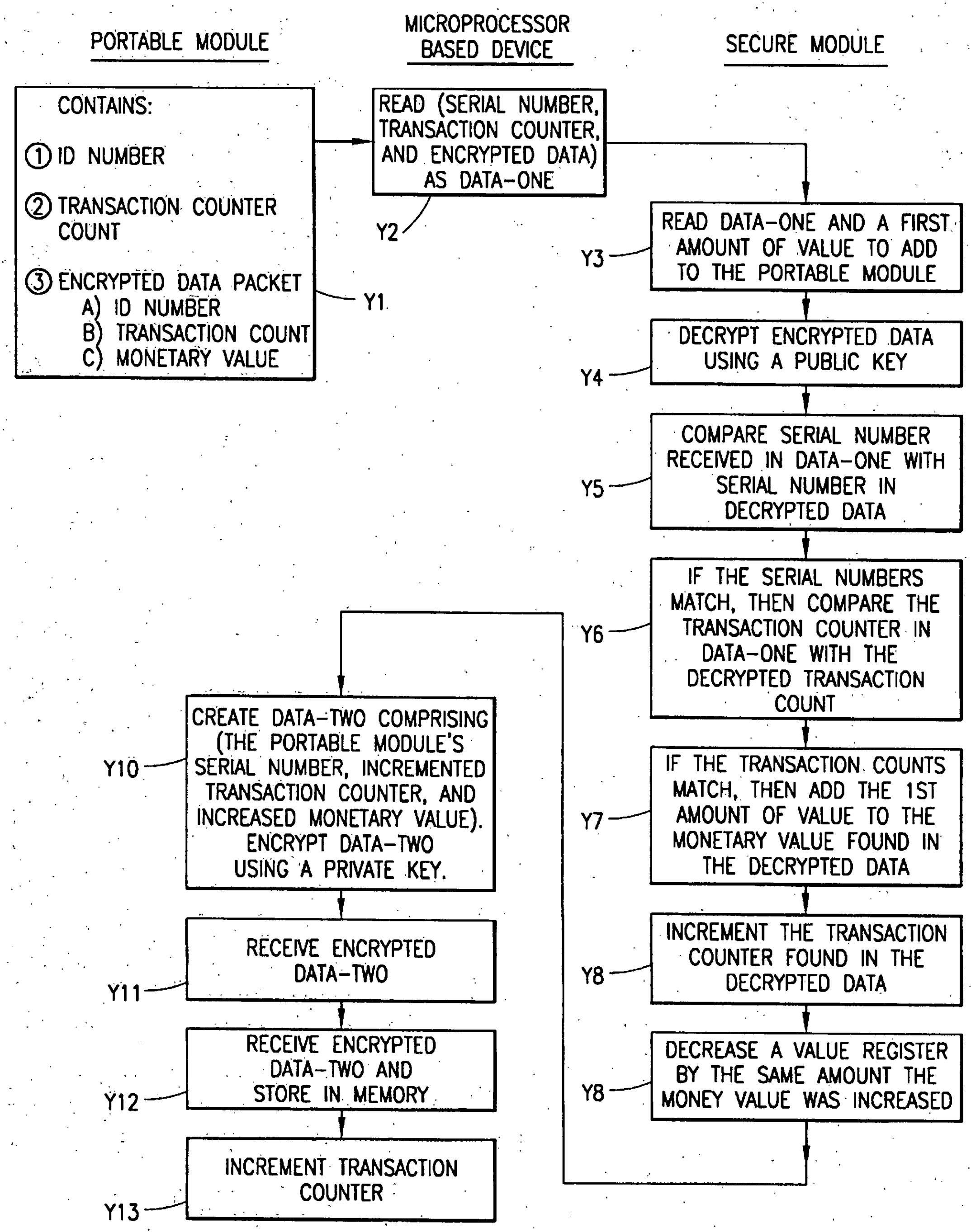

PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CONTAINS:
① ID NUMBER
② TRANSACTION COUNTER COUNT
③ ENCRYPTED DATA PACKET
A) ID NUMBER
B) TRANSACTION COUNT
C) MONETARY VALUE
Y1
READ (SERIAL NUMBER, TRANSACTION COUNTER, AND ENCRYPTED DATA) AS DATA-ONE
Y2
READ DATA-ONE AND A FIRST AMOUNT OF VALUE TO ADD TO THE PORTABLE MODULE
Y3
DECRYPT ENCRYPTED DATA USING A PUBLIC KEY
Y4
COMPARE SERIAL NUMBER RECEIVED IN DATA-ONE WITH SERIAL NUMBER IN DECRYPTED DATA
Y5
IF THE SERIAL NUMBERS MATCH, THEN COMPARE THE TRANSACTION COUNTER IN DATA-ONE WITH THE DECRYPTED TRANSACTION COUNT
Y6
IF THE TRANSACTION COUNTS MATCH, THEN ADD THE 1ST AMOUNT OF VALUE TO THE MONETARY VALUE FOUND IN THE DECRYPTED DATA
Y7
INCREMENT THE TRANSACTION COUNTER FOUND IN THE DECRYPTED DATA
Y8
DECREASE A VALUE REGISTER BY THE SAME AMOUNT THE MONEY VALUE WAS INCREASED
Y8
CREATE DATA-TWO COMPRISING (THE PORTABLE MODULE'S SERIAL NUMBER, INCREMENTED TRANSACTION COUNTER, AND INCREASED MONETARY VALUE). ENCRYPT DATA-TWO USING A PRIVATE KEY.
Y10
RECEIVE ENCRYPTED DATA-TWO
Y11
RECEIVE ENCRYPTED DATA-TWO AND STORE IN MEMORY
Y12
INCREMENT TRANSACTION COUNTER
Y13

FIG. 5

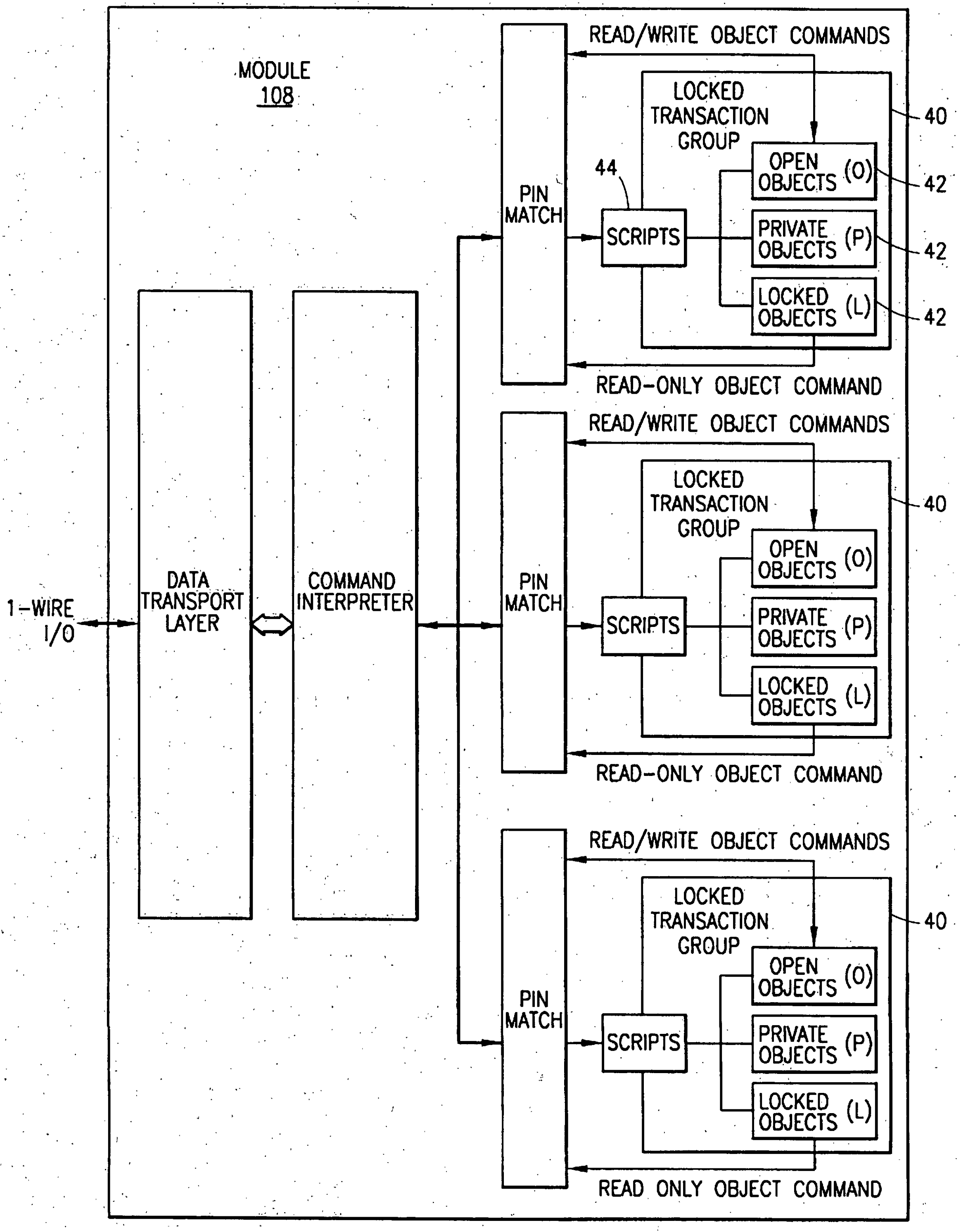

MODULE
108
1-WIRE
I/O
DATA
TRANSPORT
LAYER
COMMAND
INTERPRETER
PIN
MATCH
READ/WRITE OBJECT COMMANDS
LOCKED
TRANSACTION
GROUP
40
44
OPEN
OBJECTS
(O)
42
SCRIPTS
PRIVATE
OBJECTS
(P)
42
LOCKED
OBJECTS
(L)
42
READ-ONLY OBJECT COMMAND
PIN
MATCH
READ/WRITE OBJECT COMMANDS
LOCKED
TRANSACTION
GROUP
40
OPEN
OBJECTS
(O)
SCRIPTS
PRIVATE
OBJECTS
(P)
LOCKED
OBJECTS
(L)
READ-ONLY OBJECT COMMAND
PIN
MATCH
READ/WRITE OBJECT COMMANDS
LOCKED
TRANSACTION
GROUP
40
OPEN
OBJECTS
(O)
SCRIPTS
PRIVATE
OBJECTS
(P)
LOCKED
OBJECTS
(L)
READ ONLY OBJECT COMMAND

FIG. 6

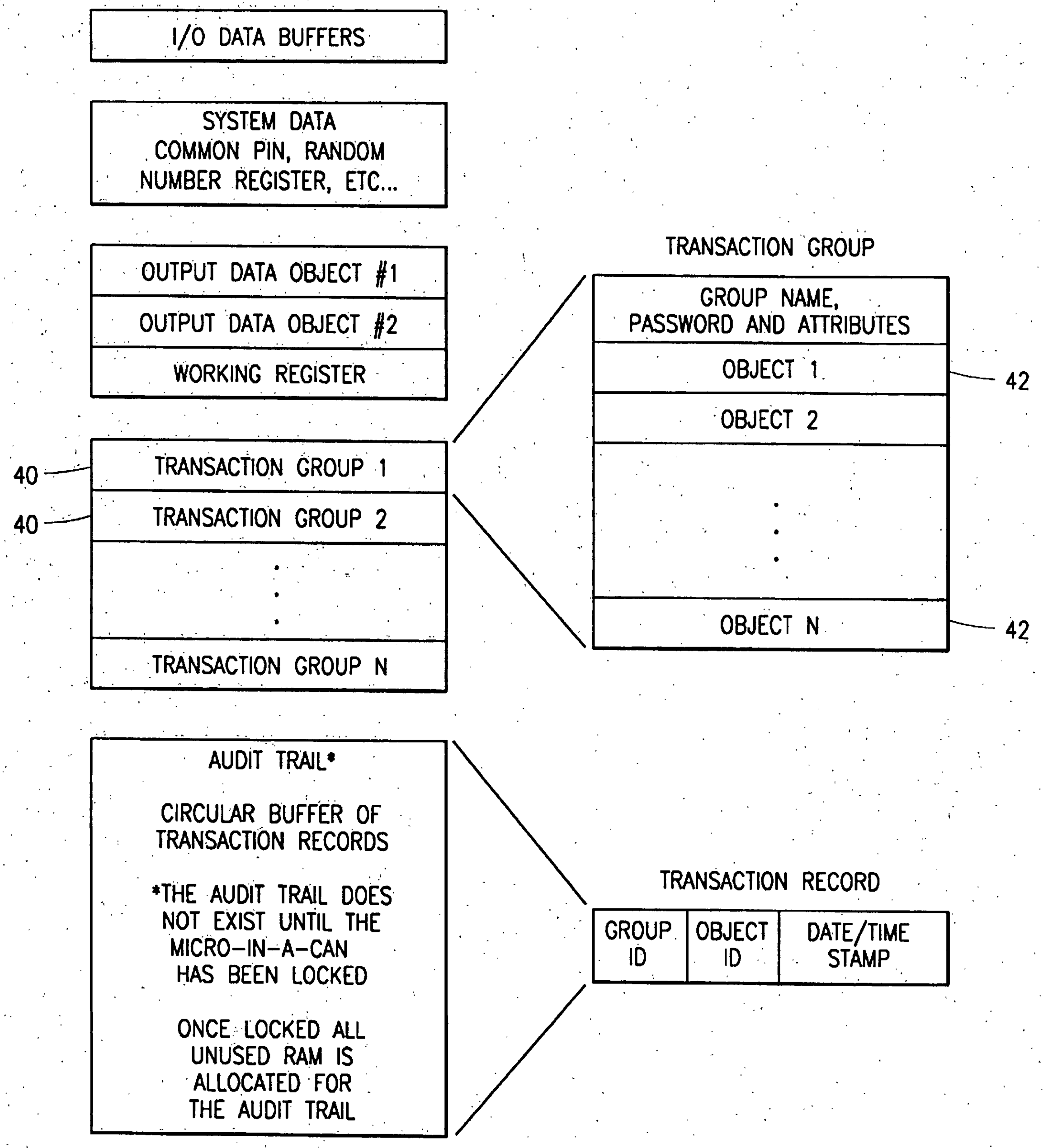

I/O DATA BUFFERS
SYSTEM DATA
COMMON PIN, RANDOM
NUMBER REGISTER, ETC...
OUTPUT DATA OBJECT #1
OUTPUT DATA OBJECT #2
WORKING REGISTER
40
40
TRANSACTION GROUP 1
TRANSACTION GROUP 2
TRANSACTION GROUP N
TRANSACTION GROUP
GROUP NAME,
PASSWORD AND ATTRIBUTES
OBJECT 1
OBJECT 2
OBJECT N
42
42
AUDIT TRAIL*
CIRCULAR BUFFER OF
TRANSACTION RECORDS
*THE AUDIT TRAIL DOES
NOT EXIST UNTIL THE
MICRO-IN-A-CAN
HAS BEEN LOCKED
ONCE LOCKED ALL
UNUSED RAM IS
ALLOCATED FOR
THE AUDIT TRAIL
TRANSACTION RECORD
GROUP ID
OBJECT ID
DATE/TIME STAMP

FIG. 7

TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

This application is a Divisional of application Ser. No. 08/594,975 filed on Jan. 31, 1996.

CROSS REFERENCE TO OTHER APPLICATIONS

The following applications of common assignee contains related subject matter and is hereby incorporated by reference:

Ser. No. UNKNOWN, filed Jan. 31, 1996, entitled METHOD, APPARATUS, SYSTEM AND FIRMWARE FOR SECURE TRANSACTIONS; and

Ser. No. UNKNOWN, filed Jan. 31, 1996, entitled METHOD, APPARATUS AND SYSTEM FOR TRANSFERRING UNITS OF VALUE.

BACKGROUND OF THE INVENTION

1. Technical Field of the Invention

The present invention relates to a method and system for transferring valuable information securely between a secure module and another module. More particularly, the present invention relates to transferring units of value between a microprocessor based secure module and another module used for carrying a monetary equivalent.

2. Description of Related Art

In the past the preferred means for paying for an item was cash. As our society has become more advanced, credit cards have become an accepted way to pay for merchandise or services. The payment is not a payment to the merchant, but instead is a credit given by a bank to the user that the merchant accepts as payment. The merchant collects money from the bank based on the credit. As time goes on, cash is used less and less, and money transfers between parties are becoming purely electronic.

Present credit cards have magnetic strips to identify the owner of the card and the credit provider. Some credit cards have electronic circuitry installed that identifies the credit card owner and the credit or service provider (the bank).

The magnetic strips installed in present credit cards do not enable the card to be used as cash. That is the modern credit card does not allow the consumer to buy something with the credit card and the merchant to receive cash at the time of the transaction. Instead, when the consumer buys something on credit, the merchant must later request that the bank pay for the item that the consumer bought. The bank then bills the consumer for the item that was bought.

Thus, there is a need for an electronic system that allows a consumer to fill an electronic module with a cash equivalent in the same way a consumer fills his wallet with cash. When the consumer buys a product or service from a merchant, the consumer's module can be debited and the merchant's cash drawer can be credited without any further transactions with a bank or service provider.

SUMMARY OF THE INVENTION

The present invention is an apparatus, system and method for communicating a cash equivalent electronically to and from a portable module. The portable module can be used as a cash equivalent when buying products and services in the market place.

The present invention comprises a portable module that can communicate to a secure module via a microprocessor based device. The portable module can be carried by a consumer, filled with electronic money at an add-money station, and be debited by a merchant when a product or service is purchased by the consumer. As a result of a purchase, the merchant's cash drawer will indicate an increase in cash value.

BRIEF DESCRIPTION OF THE DRAWINGS

A more complete understanding of the method and apparatus of the present invention may be had by reference to the following Detailed Description when taken in conjunction with the accompanying Drawings wherein:

FIG. **1** depicts an exemplary system for transferring valuable information between a module and a secure device;

FIG. **2** is a block diagram of an embodiment of a portable module;

FIG. **3** is a block diagram of an embodiment of a microprocessor based module;

FIG. **4** is an exemplary technique for transferring valuable data securely into a portable module;

FIG. **5** is an exemplary technique for transferring valuable data securely out of a portable module;

FIG. **6** is an exemplary organization of the software and firmware within a secure microprocessor based device; and

FIG. **7** is an exemplary configuration of software and firmware within a secure microprocessor based device.

DETAILED DESCRIPTION OF A PRESENTLY PREFERRED EXEMPLARY EMBODIMENT

FIG. **1** depicts a block diagram of an exemplary system **100** for transferring valuable information to and from a portable module. A portable module **102**, which will be described in more detail later, communicates to a microprocessor based device **104**. The portable module **102** may contain information that represents units of exchange or a currency equivalent. The microprocessor based device **104** can be any of an unlimited number of devices. For example, the microprocessor based device **104** could be a personal computer, an add-a-fare machine at a train or bus station (similar to those in today's District of Columbia metro stations), a turn style, a toll booth, a bank's terminal, a ride at a carnival, a washing machine at a Laundromat, a locking device, a mail metering device or any device that controls access, or meters a monetary equivalent, etc.

The means for communication **106** between the portable module **102** and the microprocessor based device **104** is preferably via a single wire or contact connection. The single wire connection **106** preferably incorporates a communication protocol that allows the portable module **102** and the microprocessor based device **104** to communicate in a bidirectional manner. Preferably the communication protocol is a one-wire protocol developed by Dallas Semiconductor. It is understood that the means for communicating **106** is not limited to a single wire connection. The communication means **106** could be multiple wires, a wireless communication system, infrared light, any electromagnetic means, a magnetic technique, or any other similar technique.

The microprocessor based device **104** is electrically connected to another microprocessor based device, which is preferably a secure device **108**. The term secure device means that the device is designed to contain a secret code and the secret code is extremely difficult to learn. An example of a secure device **108** is explained later in this document.

The microprocessor based device **104** can be connected to a variety of other devices. Such devices include, but are not limited to a cash acceptor **110**, an automatic teller machine (ATM) **112**, a credit card reader **114**, and a phone line **116**.

The cash acceptor **110** is adapted to receive cash in the form of currency, such as dollar bills or coins. The cash acceptor **110**, preferably, determines the value of the accepted currency. The cash acceptor **110** communicates to the microprocessor based device **104** and informs the device **104** of how much currency has been deposited in the cash acceptor **110**.

The cash acceptor **110** can also be a device which provides currency. That is, the cash accepter **110** in response to a communication from the microprocessor based device **104**, may provide a metered amount of currency to a person.

The credit card reader **114**, and ATM **112** can also be attached to the microprocessor based device **104**. The credit card reader **114** could be used to read a user's credit card and then, when authorized, either communicate to the microprocessor based device **104** that units of exchange need to be added to the portable module or that units of exchange need to be extracted from the portable module to pay for a good, service or credit card bill.

The ATM **112** may also be connected to the microprocessor based device. Via communications from the ATM **112**, the microprocessor based device **104** can be informed that units of exchange need to be added or subtracted from the portable module **102**.

Furthermore, it is also possible that the microprocessor based device **104** is connected to a phone line **116**. The phone line may be used for a variety of things. Most importantly, the phone line may be used to allow the microprocessor based device **104** to communicate with a network of devices. Such telephonic communication may be for validating transactions or for aiding the accounting of transactions that are performed via the microprocessor based device's **104** aid. It is further understood that the phone line may be any of a vast variety of communication lines including wireless lines. Video, analog, or digital information may be communicated over the phone line **116**.

FIG. 2 depicts a preferred exemplary portable module **102**. The portable module **102** is preferably a rugged read/write data carrier that can act as a localized data base and be easily accessed with minimal hardware. The module can be incorporated in a vast variety of portable items which includes, but is not limited to a durable micro-can package that is highly resistant to environmental hazards such as dirt, moisture, and shock. The module can be incorporated into any object that can be articulated by a human or thing, such as a ring, bracelet, wallet, name tag, necklace, baggage, machine, robotic device, etc. Furthermore, the module **102** could be attached to a stationary item and the microprocessor based device **104** may be articulated to the portable module **102**. For example, the module **102** may be attached to a piece of cargo and a module reader may be touched to or brought near the module **102**. The module reader may be part of the microprocessor based device **104**.

The portable module **102** comprises a memory **202** that is preferably, at least in part, nonvolatile memory for storing and retrieving vital information pertaining to the system to which the module **102** may become attached to. The memory **202** may contain a scratchpad memory which may act as a buffer when writing into memory. Data is first written to the scratchpad where it can be read back. After data has been verified, the data is transferred into the memory.

The module **102** also comprises a counter **206** for keeping track of the number of transactions the module has performed (the number of times certain data in the memory of the module has been changed). A timer **102** may be provided in the module to provide the ability to time stamp transactions performed by the module. A memory controller **204** controls the reading and writing of data into and out of the memory **202**.

The module also may comprise an identification number **210**. The identification number preferably uniquely identifies the portable module from any other portable module.

An input/output control circuit **212** controls the data flow into and out of the portable module **102**. The input/output control ("I/O") **212** preferably has an input buffer and an output buffer and interface circuitry **214**. As stated above, the interface circuitry **214** is preferably a one-wire interface. Again, it is understood that a variety of technologies can be used to interface the portable module **102** to another electronic device. A single wire or single connection is preferred because the mechanics of making a complete connection is simplified. It is envisioned that a proximity/wireless communication technique is also a technique for communicating between the module **102** and another device. Thus, the interface circuit **214** can be a single wire, multiple wire, wireless, electromagnetic, magnetic, light, or proximity, interface circuit.

FIG. 3 depicts a block diagram of an exemplary secure microprocessor based device ("secure device") **108**. The secure device circuitry can be a single integrated circuit. It is understood that the secure device **108** could also be a monolithic or multiple circuits combined together. The secure device **108** preferably comprises a microprocessor **12**, a real time clock **14**, control circuitry **16**, a math coprocessor **18**, memory circuitry **20**, input/output circuitry **26**, and an energy circuit **34**.

The secure device **108** could be made small enough to be incorporated into a variety of objects including, but not limited to a token, a card, a ring, a computer, a wallet, a key fob, a badge, jewelry, a stamp, or practically any object that can be grasped and/or articulated by a user of the object. In the present system **100**, the secure device **108** is preferably adapted to be a trusted certifying authority. That is the secure device **108** is a trusted computer. The secure device **108** comprises a numeric coprocessor **18** optimized for math intensive encryption. The BIOS is immune to alteration and is specifically designed for secure transactions. This secure device **108** is preferably encased in a durable, dirt, moisture and shock resistant stainless steel enclosure, but could be encased in wide variety of structures so long as specific contents of the secure device **108** are extremely difficult to decipher. The secure device **108**. The secure device **108** may have the ability to store or create a private/public key set, whereby the private key never leaves the secure device **108** and is not revealed under almost any circumstance. Furthermore, the secure module **108** is preferably designed to prevent discovery of the private key by an active self-destruction of the key upon wrongful entry.

The microprocessor **12** is preferably an 8-bit microprocessor, but could be 16, 32, 64 or any operable number of bits. The clock **14** provides timing for the module circuitry. There can also be separate clock circuitry **14** that provides a continuously running real time clock.

The math coprocessor circuitry **18** is designed and used to handle very large numbers. In particular, the coprocessor will handle the complex mathematics of RSA encryption and decryption or other types of math intensive encryption or decryption techniques.

The memory circuitry **20** may contain both read-only-memory and non-volatile random-access-memory.

Furthermore, one of ordinary skill in the art would understand that volatile memory, EPROM, SRAM and a variety of other types of memory circuitry might be used to create an equivalent device.

Control circuitry **16** provides timing, latching and various necessary control functions for the entire circuit.

An input/output circuit **26** enables bidirectional communication with the secure module **108**. The input/output circuitry **26** preferably comprises at least an output buffer and an input buffer. For communication via a one-wire bus, one-wire interface circuitry can be included with the input/output circuitry **26**. It is understood that the input/output circuitry **26** of the secure device **108** can be designed to operate on a single wire, a plurality of wires or any means for communicating is information between the secure module **108** and the microprocessor based device **104**.

An energy circuit **34** may be necessary to maintain stored information in the memory circuitry **20** and/or aid in powering the other circuitry in the module **108**. The energy circuit **34** could consist of a battery, capacitor, R/C circuit, photo-voltaic cell, or any other equivalent energy producing circuit or means.

The firmware architecture of the secure module **108** and how it operates within the exemplary system for transferring valuable information, such as units of exchange or currency, between the secure module **108** and a portable module **102** will now be discussed. The secure module **108** provides encryption and decryption services for confidential data transfer through the microprocessor based device **104**. The following examples are intended to illustrate a preferred feature set of the secure module **108** and to explain the services that the exemplary system **100** can offer. These applications and examples by no means limit the capabilities of the invention, but instead bring to light a sampling of its capabilities.

I. Overview of the Preferred Secure Module **108** and its Firmware Design

Referring to FIG. **3** again, the secure module **108** preferably contains a general-purpose, 8051-compatible micro controller **12** or a reasonably similar product, a continuously running real-time clock **14**, a high-speed modular exponentiation accelerator for large integers (math coprocessor) **18**, input and output buffers **28, 30** with a one-wire interface **32** for sending and receiving data, 32 Kbytes of ROM memory **22** with preprogrammed firmware, 8 Kbytes of NVRAM (non-volatile RAM) **24** for storage of critical data, and control circuitry **16** that enables the micro controller **12** to be powered up to interpret and act on the data placed in an input data object. The module **108** draws its operating power from a single wire, one-wire communication line. The micro controller **12**, clock **14**, memory **20**, buffers **28, 30**, one-wire front-end **32**, modular exponentiation accelerator **18**, and control circuitry **16** are preferably integrated on a single silicon chip and packaged in a stainless steel micro can using packaging techniques which make it virtually impossible to probe the data in the NVRAM **24** without destroying the data. Initially, most of the NVRAM **24** is available for use to support applications such as those described below. One of ordinary skill will understand that there are many comparable variations of the module design. For example, volatile memory might be used, or an interface other than a one-wire interface could be used.

The secure module **108** is preferably intended to be used first by a Service Provider who loads the secure module **108** with data to enable it to perform useful functions, and second by an End User who issues commands to the secure module **108** to perform operations on behalf of the Service Provider for the benefit of the End User. For this reason, the secure module **108** offers functions to support the Service Provider in setting up the module for an intended application. It also offers functions to allow the End User to invoke the services offered by the Service Provider.

Each Service Provider can reserve a block of NVRAM memory to support its services by creating a transaction group **40** (refer to FIGS. **6** and **7**). A transaction group **40** is simply a set of software objects **42** that are defined by the Service Provider. These objects **42** include both data objects (encryption keys, transaction counts, money amounts, date/time stamps, etc.) and transaction scripts **44** which specify how to combine the data objects in useful ways. Each Service Provider creates his own transaction group **40**, which is independent of every other transaction group **40**. Hence, multiple Service Providers can offer different services in the same module **108**. The number of independent Service Providers that can be supported depends on the number and complexity of the objects **42** defined in each transaction group **40**. Examples of some of the objects **42** that can be defined within a transaction group **40** are the following:

RSA Modulus
RSA Exponent
Transaction Script
Transaction Counter
Money Register
Destructor
Clock Offset
Random SALT
Configuration Data
Input Data
Output Data

Within each transaction group **40** the secure module **108** will initially accept certain commands which have an irreversible effect. Once any of these irreversible commands are executed in a transaction group **40**, they remain in effect until the end of the module's useful life or until the transaction group **40**, to which it applies, is deleted from the secure module **108**. In addition, there are certain commands which have an irreversible effect until the end of the module's life or until a master erase command is issued to erase the entire contents of the secure module **108**. These commands will be discussed further below. These commands are essential to give the Service Provider the necessary control over the operations that can be performed by the End User. Examples of some of the irreversible commands are:

Privatize Object
Lock Transaction Group
Lock Object
Lock Micro-In-A-Can ™

Since much of the module's utility centers on its ability to keep a secret, the Privatize command is a very important irreversible command.

Once the secure module **108**, as a whole, is locked, the remaining NVRAM memory **24** is allocated for a circular buffer for holding an audit trail of previous transactions. Each of the transactions are identified by the number of the transaction group, the number of objects **42** within the specified group, and the date/time stamp.

The fundamental concept implemented by the firmware is that the Service Provider can store transaction scripts **44** in a transaction group **40** to perform only those operations among objects that he wishes the End User to be able to perform. The Service Provider can also store and privatize RSA key or keys (encryption keys) that allow the secure module **108** to "sign" transactions on behalf of the Service Provider, thereby guaranteeing their authenticity. By privatizing and/or locking one or more objects **42** in the transaction group **40**, the Service Provider maintains control over what the secure module **108** is allowed to do on his behalf. The End User cannot add new transaction scripts **44** and is therefore limited to the operations on objects **42** that can be performed with the transaction scripts **44** programmed by the Service Provider.

II. Usage Models of the Secure Module **108** and Portable Module **102**

This section presents practical applications of the system **100**. Each of these applications is described in enough detail to make it clear why the secure module **108** and portable module **102** are important to the system application.

A. Transferring Units of Exchange Out of a Portable Module **102**

This section describes an example of how a portable module **102** and a secure module **108** operate in conjunction with the microprocessor based device **104** so that units of exchange can be securely transferred out of the portable module **102** and deposited into the secure module **108** and/or potentially communicated to at least one of the cash acceptor **110**, ATM **112**, credit card reader **114**, or the phone line **116**.

Referring to FIG. **4**, initially the portable module **102** contains its ID number, a count within its transaction counter and an encrypted data packet stored in memory. Encrypted within the data packet is the portable modules ID number, the portable modules transaction count number, and the amount of value (the monetary value) of the portable module at the present time X1.

The user of the portable module touches, or somehow puts the portable module **102** into communication with the microprocessor based device **104**. For explanation purposes, suppose the portable module **102** is being used as a token used to pay for a train fare. Thus, the microprocessor based device **104** could be, in this case, a turn style that allows the user to enter a train platform. The cost of entering the train platform is known by the microprocessor based device **104**.

The microprocessor based device **104** reads the portable module's serial number, transaction count, and the encrypted data packet X2. This data could be referred to as a first data.

The microprocessor device **104** then provides the first data along with a first value, being the amount of value to be debited from the portable token (the train fare), to the secure module **108** X3. The secure module **108** decrypts the encrypted data found in the first data using a public key X4.

Next, the secure module **108** makes a few comparisons to make sure that the data received is good data and not counterfeit. The secure module **108** compares the serial number received in the first data with the decrypted serial number X5. If the two serial numbers match then the secure module **108** compares the transaction count received in the first data with the decrypted transaction count X6. If the two transaction counts match then the secure module is comfortable that the data received is not counterfeit data. It is understood that the comparisons can be done in any order.

Furthermore, there may have been a time stamp sent from the portable module **102**. The time stamp may indicate a variety of things. One thing could be an indication of whether the portable module is still valid or the time stamp may further enable the secure module to decide if the data is or is not counterfeit.

Assuming all the data passed to the secure module **108** is determined to be valid data, the secure module **108** subtracts the first value, the train fare, from the monetary value of the portable module **102** X7. The decrypted transaction count is then incremented.

A register within the secure module **108** is increased by the amount of the first value, the train fare, so that the secure module can keep an accounting of the amount of "money" it has collected X8. The secure module **108** creates a data packet, a second data, which comprises at least the portable module's serial number, the incremented transaction count, and the reduced monetary value of the portable module **102**. The second data packet is then encrypted by the secure module **108** using a private key X9.

The microprocessor based device **104** receives the encrypted second data packet, passes the encrypted second data packet to the portable module **102** X10, and opens the turn style to let the module's user onto the train platform. The portable module **102** receives the encrypted second data packet and stores it in memory X11. The portable module also increments its transaction count indicating that another transaction has occurred X12.

Thus, the above description indicates how valuable information can be transferred between a portable insecure module **102** and a secure module **108** wherein there is a conservation of value. That is, no value is gained or lost. Value that was in the portable module **102** was decreased by the same amount value was added to the secure module **108**. In the example provided, the decrease and increase in value was equal to a train fare. Such an increment or decrement can also be equal to an amount provided by an ATM, credit card transaction, cash acceptor, etc.

It is also understood that the insecure portable is module **102** could be another secure module similar to the secure module in the system, but programed to act like a portable module **102**.

B. Transferring Units of Exchange Into the Portable Module **102**

In this example, for simplicity, suppose the portable module does not have any monetary value and the user of the portable module wishes to "fill it up" with value. Suppose the user wishes to take cash out of an ATM machine and instead of pocketing the cash, the user wishes to put the cash value into the portable module **102**.

Referring to FIG. **5**, the portable module **102** contains its ID number, a transaction count and an encrypted data packet containing the portable module's ID number, transaction count and the monetary value of the portable module **102** Y1. The microprocessor based device **104**, which in this example could be part of the ATM machine **112**, receives the information contained in the portable module **102** when a communication is initiated between the portable module **102** and the microprocessor based device **104** Y2.

The microprocessor based device **104** passes the module's serial number, transaction count, and encrypted data packet as a first data packet to the secure module **108**. The microprocessor based device also passes the amount of amount of monetary value to add to the portable module **102**, as indicated by the ATM **112**, to the secure module **108** Y3.

The secure module **108** decrypts the encrypted data passed to it using a public key Y4. The secure module **108** then makes a few comparisons to make sure that the data it has just received is valid and not counterfeit. The secure module **108** compares the serial number (ID number) received in the first data packet with the serial number (ID number) found in the decrypted data Y5. The secure module **108** also compares the transaction count passed the first data packet with the transaction count found in the decrypted data Y6. If the serial numbers and transaction counters match, then the secure module decides that the data received is valid and the secure module adds the monetary value, indicated by the ATM to the monetary value of the decrypted data Y7. The decrypted transaction count is incremented Y8. A register within the secure module may be decremented by the same amount that the monetary value of the decrypted data was increased Y8.

The secure module 108 creates a second data packet, that contains the portable module's ID number, the incremented transaction counter and the increased monetary value. The second data packet is then encrypted using a private key Y10.

The microprocessor based device 104 reads the encrypted second data packet and sends it to the portable module 102 Y11. The portable module receives the encrypted second data packet and stores it in memory Y12. The portable module also advances its transaction counter Y13. The result being that the portable module now has the value of the cash withdrawn from the ATM 112. Furthermore, a record of the transaction may have been recorded and kept in the secure module, as well as by the bank that operates the ATM 112.

Exemplary Firmware Definitions for Use With the Secure Module

Object The most primitive data structure accepted by and operated on by the secure modules firmware. A list of valid objects and their definitions is provided in the next section.

Group A self-contained collection of objects. An object's scope is restricted to the group of which it is a member.

Group ID A number preferably between 0 and 255 representing a specific group.

Object ID A number preferably between 0 and 255 representing a specific object within a specific group.

Object Type Preferably a 1-byte type specifier that describes a specific object.

PIN An alphanumeric Personal Identification number that is preferably eight bytes in length.

Common PIN The PIN that controls access to shared resources such as the audit trail. It is also used to control the host's ability to create and delete groups.

Group PIN The PIN that controls access to operations specific to objects within a group.

Audit Trail A record of transactions occurring after the secure module has been locked.

Locked Object An object which has been locked by executing the lock object command. Once an object is locked it is not directly readable.

Private Object An object which has been privatized by executing the privatize object command. Once an object is private, it is not directly readable or writable.

Locked Group A group which has been locked using the locked group command. After a group has been locked it will not allow object creation.

Composite Object A combination of several objects. The individual objects inherit the attributes of the composite object.

Exemplary Object Definitions

RSA Modulus A large integer preferably of at most 1024 bits in length. It is the product of 2 large prime numbers that are each about half the number of bits in length of the desired modulus size. The RSA modulus is used in the following equations for encrypting and decrypting a message M:

$$\text{Encryption: } C=M^e \pmod N \quad (1)$$

$$\text{Decryption: } M=C^d \pmod N \quad (2)$$

where C is the cyphertext, d and e are the RSA exponents (see below), and N is the RSA modulus.

RSA Exponent Both e and d (shown in equations 1 and 2 above) are RSA exponents. They are typically large numbers but are smaller than the modulus (N). RSA exponents can be either private or public. When RSA exponents are created in the secure module, they may be declared as either. Once created an exponent may be changed from a public exponent to a private exponent. After an exponent has been made private, however, it will remain private until the transaction group 40 to which it belongs is destroyed.

Transaction Script A transaction script is a series of instructions to be carried out by the secure module. When invoked the secure module firmware interprets the instructions in the script and places the results in the output data object (see below). The actual script is simply a list of objects. The order in which the objects are listed specifies the operations to be performed on the objects. transaction scripts 44 preferably may be as long as 128 bytes.

Transaction Counter The transaction counter object is preferably 4 bytes in length and is usually initialized to zero when it is created. Every time a transaction script, which references this object, is invoked, the transaction counter increments by 1. Once a transaction counter has been locked it is read only and provides an irreversible counter.

Money Register The money register object is preferably 4 bytes in length and may be used to represent money or some other form of credit. Once this object has been created, it must be locked to prevent a user from tampering with its value. Once locked the value of this object can be altered only by invoking a transaction script. A typical transaction group 40 which performs monetary transactions might have one script for withdrawals from the money register and one for deposits to the money register.

Clock Offset This object is preferably a 4 byte number which contains the difference between the reading of the secure module's real-time clock and some convenient time (e.g., 12:00 a.m., Jan. 1, 1970). The true time can then be obtained from the secure module by adding the value of the clock offset to the real-time clock.

SALT A SALT object is preferably 20 bytes in length and should be initialized with random data when it is created. When a host transmits a generate random SALT command, the secure module combines the previous SALT with the secure module's random number (produced preferably by randomly occurring power-ups) to generate a new random SALT. If the SALT object has not been privatized it may subsequently be read by issuing a read object command.

Configuration Data This is a user defined structure with preferably a maximum length of 128 bytes. This object is typically used to store configuration information specific to its transaction group 40. For example, the configuration data object may be used to specify the format of the money register object (i.e., the type of currency it represents). Since this object has no pre-defined structure, it may never be used by a transaction object.

Input Data An input data object is simply an input buffer with preferably a maximum length of 128 bytes. A transaction group may have multiple input objects. The host uses input data objects to store data to be processed by transaction scripts 44.

Output Data The output data object is used by transaction scripts as an output buffer. This object is automatically created when the transaction group is created. It is preferably 512 bytes in length and inherits password protection from its group.

Random Fill When the script interpreter encounters this type of object it automatically pads the current message so that its length is 1 bit smaller than the length of the preceding modulus. A handle to this object is automatically created when the transaction group is created. It is a private object and may not be read using the read object command.

Working Register This object is used by the script interpreter as working space and may be used in a transaction script. A handle to this object is automatically created when the transaction group is created. It is a private object and may not be read using the read object command.

ROM Data This object is automatically created when the transaction group is created. It is a locked object and may not be altered using the write object command. This object is 8 bytes and length and its contents are identical to the 8 by ROM data of the Micro-In-A-Can™.

Preferred Secure Module Firmware Command Set

Set Common PIN(01H)
Transmit (to secure module)
01H, old PIN, new PIN, PIN option byte
Receive data
CSB (command status byte) = 0 if successful, appropriate error code otherwise
Output length = 0
Output Data = 0

Notes:

The PIN option byte may be the bitwise-or of any of the following values:

PIN_TO_ERASE	00000001b (require PIN for Master Erase)
PIN_TO_CREATE	00000010b (require PIN for group creation).

Initially the secure module has a PIN (Personal Identification Number) of 0 (Null) and an option byte of 0. Once a PIN has been established it can only be changed by providing the old PIN or by a Master Erase. However, if the PIN_TO_ERASE bit is set in the option byte, the PIN can only be changed through the set common PIN command.

Possible error codes for the set common PIN command:

ERR_BAD_COMMON_PIN	(Common PIN match failed)
ERR_BAD_PIN_LENGTH	(New PIN length > 8 bytes)
ERR_BAD_OPTION_BYTE	(Unrecognizable option byte)

For all commands described in this section, data received by the host will be in the form of a return packet. A return packet has the following structure:

Command status byte	(0 if command successful, error code otherwise, 1 byte)
Output data length	(Command output length, 2 bytes)
Output data	(Command output, length specified above).

Master Erase (02H)
Transmit data
02H, Common PIN
Receive data
CSB = 0 if command was successful, ERR_BAD_COMMON_PIN otherwise
Output length = 0
Output data = 0

Notes:

If the LSB (least significant bit) of the PIN option is clear (i.e. PIN not required for Master Erase) then a 0 is transmitted for the Common PIN value. In general this text will always assume a PIN is required. If no PIN has been established a 0 should be transmitted as the PIN. This is true of the common PIN and group PINS (see below). If the PIN was correct the firmware deletes all groups (see below) and all objects within the groups. The common PIN and common PIN option byte are both reset to zero.

After everything has been erased the secure module transmits the return packet. The CSB is as described above. The output data length and output data fields are both set to 0.

Create Group (03H)
Transmit data
03H, Common PIN, Group name, Group PIN
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = 1 if successful, 0 otherwise
Output data = Group ID if successful, 0 otherwise

Notes:

The maximum group name length is 16 bytes and the maximum PIN length is eight bytes. If the PIN_TO_CREATE bit is set in the common PIN option byte and the PIN transmitted does not match the common PIN the secure module will set the OSC to ERR_BAD_COMMON_PIN.

Possible error return codes for the create group command:

ERR_BAD_COMMON_PIN	(Incorrect common PIN)
ERR_BAD_NAME_LENGTH	(If group name length > 16 bytes)
ERR BAD_PIN_LENGTH	(If group PIN length > 8 bytes)
ERR_MIAC_LOCKED	(The secure modulc has been locked)
ERR_INSUFFICIENT_RAM	(Not enough memory for new group)

Set Group PIN (04H)
Transmit data
04H, Group ID, old GPIN, new GPIN
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = 0
Output data = 0

Notes:

The Group PIN only restricts access to objects within the group specified by the group ID transmitted in the command packet.

Possible error codes for the set group PIN command:

ERR_BAD_GROUP_PIN	(Group PIN match failed)
ERR_BAD_PIN_LENGTH	(New group PIN length > 8 bytes)

Create Object (05H)
Transmit data
05H, Group ID, Group PIN, Object type, Object attributes, Object data
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = 1 if successful, 0 otherwise
Output data = object ID if successful, 0 otherwise

Notes:

If the Create Object command is successful the secure module firmware returns the object's ID within the group specified by the Group ID. If the PIN supplied by the host was incorrect or the group has been locked by the Lock Group command (described below) the secure module returns an error code in the CSB. An object creation will also fail if the object is invalid for any reason. For example, if the object being created is an RSA modulus (type 0) and it is greater than 1024 bits in length. transaction script creation will succeed if it obeys all transaction scripts rules.

Possible error return codes for the create object command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)	
ERR_GROUP_LOCKED	(The group has been locked)	
ERR_MIAC_LOCKED	(The secure module has been locked)	
ERR_INVALID_TYPE	(The object type specified is invalid)	
ERR_BAD_SIZE	(The objects length was invalid)	
ERR_INSUFFICIENT_RAM	(Not enough memory for new object)	
Object types:	RSA modulus	0
	RSA exponent	1
	Money register	2
	Transaction counter	3
	Transaction script	4
	Clock offset	5
	Random SALT	6
	Configuration object	7
	Input data object	8
	Output data object	9
Object Attributes:	Locked	00000001b
	Privatized	00000010b

Objects may also be locked and privatized after creation by using the Lock Object and Privatize Object commands described below.

Lock Object (06H)
Transmit data
06H, Group ID, Group PIN, Object ID
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = 0
Output data = 0

Notes:

If the Group ID, Group PIN and Object ID are all correct, the secure module will lock the specified object. Locking an object is an irreversible operation.

Possible error return codes for the lock object command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_GROUP_LOCKED	(The group has already been locked)
ERR_MIAC_LOCKED	(The secure module has been locked)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Specified object does not exist)

Privatize Object (07H)
Transmit data
07H, Group ID, Group PIN, Object ID
Receive data
CSB = 0 if successful, appropriate error code otherwise

Notes:

If the Group ID, Group PIN and Object ID were valid the object will be privatized. Privatized objects share all the properties of locked objects but are not readable. Privatized objects are only modifiable through transaction scripts. Note that locking a privatized object is legal, but has no meaning since object privatization is a stronger operation than object locking. Privatizing an object is an irreversible operation.

Possible error return codes for the privatize object command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_GROUP_LOCKED	(The group has already been locked)
ERR_MIAC_LOCKED	(The secure module has been locked)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Specified object does not exist)

Make Object Destructable (08H)
Transmit data
08H, Group ID, Group PIN, Object ID
Receive data
CSB = 0 if successful, appropriate error code otherwise

Notes:

If the Group ID, Group PIN and Object ID were valid the object will be made destructable. If an object is destructable it becomes unusable by a transaction script after the groups destructor becomes active. If no destructor object exists within the transaction group the destructible object attribute bit has no affect. Making an object destructable is an irreversible operation.

Possible error return codes for the make object destructable command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_GROUP_LOCKED	(The group has already been locked)
ERR_MIAC_LOCKED	(The secure module has been locked)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Specified object does not exist)

Lock Secure module (09H)
Transmit data
09H, Common PIN
Receive data
CSB = 0 if successful, appropriate error code otherwise
Output length = 2 if successful, 0 otherwise
Output data = audit trail size if successful, 0 otherwise

Notes:

If the host supplied Common PIN is correct and the secure module has not previously been locked, the command will succeed. When the secure module is locked it will not accept any new groups or objects. This implies that all groups are automatically locked. The RAM not used by the system or by groups will be used for an audit trail. There is no audit trail until the secure module has successfully been locked!

An audit trail record is six bytes long and has the following structure:

Group ID|Object ID|Date/Time stamp.

Once an audit trail has been established, a record of the form shown above will be stored in the first available size byte location every time a transaction script is executed. Note that since the secure module must be locked before the audit trail begins, neither the group ID nor any object ID is subject to change. This will always allow an application processing the audit trail to uniquely identify the transaction script that was executed. Once the audit trail has consumed all of its available memory, it will store new transaction records over the oldest transaction records.

Possible error codes for the lock secure module command:

ERR_BAD_COMMON_PIN	(Supplied common PIN was incorrect)
ERR_MIAC_LOCKED	(Secure module was already locked)

Lock Group (0AH)
Transmit data
0AH, Group ID, Group PIN
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = 0
Output data = 0

Notes:

If the group PIN provided is correct the secure module BIOS will not allow further object creation within the specified group. Since groups are completely self-contained entities they may be deleted by executing the Delete Group command (described below).

Possible error return codes for the lock group command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_GROUP_LOCKED	(The group has already been locked)
ERR_MIAC_LOCKED	(The secure module has been locked)
ERR_BAD_GROUP_ID	(Specified group does not exist)

Invoke Transaction Script (0BH)
Transmit data
0BH, Group ID, Group PIN, Object ID
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = 1 if successful, 0 otherwise
Output data = estimated completion time

Notes:

The time estimate returned by the secure module is in sixteenths of a second. If an error code was returned in the CSB, the time estimate will be 0.

Possible error return codes for the execution transaction script command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Script object did not exist in group)

Read Object (0CH)
Transmit data
0CH, Group ID, Group PIN, Object ID
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = object length if successful, 0 otherwise
Output data = object data if successful, 0 otherwise

Notes:

If the Group ID, Group PIN and Object ID were correct, the secure module checks the attribute byte of the specified object. If the object has not been privatized the secure module will transmit the object data to the host. If the Group PIN was invalid or the object has been privatized the secure module will return a 0 in the output length, and data fields of the return packet.

Possible error codes for the read object command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Object did not exist in group)
ERR_OBJECT_PRIVATIZED	(Object has been privatized)

Write Object (0DH)

Transmit data
0DH, Group ID, Group PIN, Object ID, Object size, Object Data
Receive data
CSB = 0 if successful, appropriate error code otherwise
Output length = 0
Output data = 0

Notes:

If the Group ID, Group PIN and Object ID were correct, the secure module checks the attribute byte of the specified object. If the object has not been locked or privatized the secure module will clear the objects previous size and data and replace it with the new object data. Note that the object type and attribute byte are not affected.

Possible error codes for the write object command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT ID	(Object did not exist in group)
ERR_BAD_OBJECT_SIZE	(Illegal object size specified)
ERR_OBJECT_LOCKED	(Object has been locked)
ERR_OBJECT_PRIVATIZED	(Object has been privatized)

Read Group Name (0EH)

Transmit data
0EH, Group ID
Receive data
CSB = 0
Output Length = length of group name
Output data = group name

Notes:

The group name length is a maximum of 16 bytes. All byte values are legal in a group name.

Delete Group (0FH)

Transmit data
0FH, Group ID, Group PIN
Receive data
CSB = 0 if successful, appropriate error code otherwise
Output length = 0
Output data = 0

Notes:

If the group PIN and group ID are correct the secure module will delete the specified group. Deleting a group causes the automatic destruction of all objects within the group. If the secure module has been locked the Delete Group command will fail.

Possible error codes for the delete group command:

ERR_BAD_CROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_MIAC_LOCKED	(Secure module has been locked)

Get Command Status Info (10H)

Transmit data
10H
Receive data
CSB = 0
Output length = 6
Output data = secure module status structure (see below)

Notes:

This operation requires no PIN and never fails. The status structure is defined as follows:

Last command executed	(1 byte)
Last command status	(1 byte)
Time command received	(4 bytes)

Get Secure module Configuration Info (11H)

Transmit data
11H
Receive data
CSB = 0
Output length = 4
Output data = secure module configuration structure

Notes:

This operation requires no PIN and never fails. The configuration structure is defined as follows:

Number of groups	(1 byte)
Flag byte (see below)	(1 byte)
Audit trail size/Free RAM	(2 bytes)

The flag byte is the bitwise-or of any of the following values:

00000001b	(Secure module is locked)
00000010b	(Common PIN required for access)

Read Audit Trail Info (12H)

Transmit data
12H, Common PIN
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = audit trail structure size (5) if successful, 0 otherwise

-continued

Read Audit Trail Info (12H)
Output data = audit trail info structure if successful, 0 otherwise

Notes:

If the transmitted Common PIN is valid and the secure module has been locked, it returns audit trail configuration information as follows:

Number of used transaction records	(2 bytes)
Number of free transaction records	(2 bytes)
A boolean specifying whether or not the audit trail rolled since previous read command	(1 byte)

Possible error codes for the read audit trail info command:

ERR_BAD_COMMON_PIN	(Common PIN was incorrect)
ERR_MIAC_NOT_LOCKED	(Secure module is not locked)

Read Audit Trail (13H)
Transmit data
13H, Common PIN
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = # of new records * 6 if successful, 0 otherwise
Output data = new audit trail records

Notes:

If the transmitted common PIN is valid and the secure module has been locked, it will transfer all new transaction records to the host.

Possible error codes for the read audit trail command:

ERR_BAD_COMMON_PIN	(Common PIN was incorrect).
ERR_MIAC_NOT_LOCKED	secure module is not locked

Read Group Audit Trail (14H)
Transmit data
14H, Group ID, Group PIN
Receive data
CSB = 0 if command successful, appropriate error code otherwise
Output length = # or records for group * 6 if successful, 0 otherwise
Output data = audit trail records for group

Notes:

This command is identical to the read audit trail command, except that only records involving the group ID specified in the transmit data are returned to the host. This allows transaction groups to record track their own activities without seeing other groups records.

Possible error codes for the read group audit trail command:

ERR_BAD_GROUP_ID	(Group ID does not exist)
ERR_PAD_GROUP_PIN	(Common PIN was incorrect)
ERR MIAC_NOT_LOCKED	(The secure module is not locked)

Read Real Time Clock (15H)
Transmit data
15H, Common PIN
Receive data
CSB = 0 if the common PIN matches and ERR_BAD_COMMON_PIN otherwise
Output length = 4
Output data = 4 most significant bytes of the real time clock

Notes:

This value is not adjusted with a clock offset. This command is normally used by a service provider to compute a clock offset during transaction group creation.

Read Real Time Clock Adjusted (16H)
Transmit data
16H, Group ID, Group PIN, ID of offset object
Receive data
CSB = 0 if successful, appropriate error code otherwise
Output length = 4 if successful, 0 otherwise
Output data = Real time clock + clock offset ID

Notes:

This command succeeds if the group ID and group PIN are valid, and the object ID is the ID of a clock offset. The secure module adds the clock offset to the current value of the 4 most significant bytes of the RTC and returns that value in the output data field. Note that a transaction script may be written to perform the same task and put the result in the output data object.

Possible error codes for the real time clock adjusted command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD OBJECT_TYPE	(Object ID is not a clock offset)

Get Random Data (17H)
Transmit data
17H, Length (L)
Receive data
CSB = 0 if successful, appropriate error code otherwise
Output length = L if successful, 0 otherwise
Output data = L bytes of random data if successful

Notes:

This command provides a good source of cryptographically useful random numbers.

Possible error codes for the get random data command are:

ERR_BAD_SIZE	(Requested number of bytes > 128)

Get Firmware Version ID (18H)
Transmit data
18H
Receive data
CSB = 0
Output length = Length of firmware version ID string
Output data = Firmware version ID string

Notes:

This command returns the firmware version ID as a Pascal type string (length+data).

Get Free RAM (19H)
Transmit data
19H
Receive data
CSB = 0
Output length = 2
Output data = 2 byte value containing the amount of free RAM

Notes:

If the secure module has been locked the output data bytes will both be 0 indicating that all memory not used by transaction groups has been reserved for the audit trail.

Change Group Name (1AH)
Transmit data
1AH, Group ID, Group PIN, New Group name
Receive data
CSB = 0 if successful or an appropriate error code otherwise
Output length = 0
Output data = 0

Notes:

If the group ID specified exists in the secure module and the PIN supplied is correct, the transaction group name is replaced by the new group name supplied by the host. If a group ID of 0 is supplied the PIN transmitted must be the common PIN. If it is correct, the secure module name is replaced by the new name supplied by the host.

Possible error codes for the change group name command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GRQUP_ID	(Specified group does not exist)
ERR_BAD NAME_LENGTH	(New group name > 16 bytes)

ERROR CODE DEFINITIONS

ERR_BAD_COMMAND (80H)

This error code occurs when the secure module firmware does not recognize the command just transmitted by the host.

ERR_BAD_COMMON_PIN (81H)

This error code will be returned when a command requires a common PIN and the PIN supplied does not match the secure module's common PIN. Initially the common PIN is set to 0.

ERR_BAD_GROUP_PIN (82H)

Transaction groups may have their own PIN, FIG. 6. If this PIN has been set (by a set group PIN command) it must be supplied to access any of the objects within the group. If the Group PIN supplied does not match the actual group PIN, the secure module will return the ERR_BAD_GROUP_PIN error code.

ERR_BAD_PIN_LENGTH (83H)

There are 2 commands which can change PIN values. The set group PIN and the set common PIN commands. Both of these require the new PIN as well as the old PIN. The ERR_BAD_PIN_LENGTH error code will be returned if the old PIN supplied was correct, but the new PIN was greater than 8 characters in length.

ERR_BAD_OPTION_BYTE (84H)

The option byte only applies to the common PIN. When the set common PIN command is executed the last byte the host supplies is the option byte (described in command section). If this byte is unrecognizable to the secure module, it will return the ERR_BAD_OPTION_BYTE error code.

ERR_BAD_NAME_LENGTH (85H)

When the create transaction group command is executed, one of the data structures supplied by the host is the group's name. The group name may not exceed 16 characters in length. If the name supplied is longer than 16 characters, the ERR_BAD_NAME_LENGTH error code is returned.

ERR_INSUFFICIENT_RAM (86H)

The create transaction group and create object commands return this error code when there is not enough heap available in the secure module.

ERR_MIAC_LOCKED (87H)

When the secure module has been locked, no groups or objects can be created or destroyed. Any attempts to create or delete objects will generate an ERR_MIAC_LOCKED error code.

ERR_MIAC_NOT_LOCKED (88H)

If the secure module has not been locked there is no audit trail. If one of the audit trail commands is executed this error code will be returned.

ERR_GROUP_LOCKED (89H)

Once a transaction group has been locked object creation within that group is not possible. Also the objects attributes and types are frozen. Any attempt to create objects or modify their attribute or type bytes will generate an ERR_GROUP_LOCKED error code.

ERR_BAD_OBJECT_TYPE (8AH)

When the host sends a create object command to the secure module, one of the parameters it supplies is an object type (see command section). If the object type is not recognized by the firmware it will return an ERR_BAD_OBJECT_TYPE error code.

ERR_BAD_OBJECT_ATTR (8BH)

When the host sends a create object command to the secure module, one of the parameters it supplies is an object attribute byte (see command section). If the object attribute byte is not recognized by the firmware it will return an ERR_BAD_OBJECT_ATTR error code.

ERR_BAD_SIZE (8CH)

An ERR_BAD_SIZE error code is normally generated when creating or writing an object. It will only occur when the object data supplied by the host has an invalid length.

ERR_BAD_GROUP_ID (8DH)

All commands that operate at the transaction group level require the group ID to be supplied in the command packet. If the group ID specified does not exist in the secure module it will generate an ERR_BAD_GROUP_ID error code.

ERR_BAD_OBJECT_ID (8EH)

All commands that operate at the object level require the object ID to be supplied in the command packet. If the object ID specified does not exist within the specific transaction group (also specified in the command packet) the secure module will generate an ERR_BAD_OBJECT_ID error code.

ERR_INSUFFICIENT_FUNDS (8FH)

If a script object that executes financial transactions is invoked and the value of the money register is less than the withdrawal amount requested an ERR_INSUFFICIENT_FUNDS error code will be returned.

ERR_OBJECT_LOCKED (90H)

Locked objects are read only. If a write object command is attempted and it specifies the object ID of a locked object the secure module will return an ERR_OBJECT_LOCKED error code.

ERR_OBJECT_PRIVATE (91H)

Private objects are not directly readable or writable. If a read object command or a write object command is attempted, and it specifies the object ID of a private object, the secure module will return an ERR_OBJECT_PRIVATE error code.

ERR_OBJECT_DESTRUCTED (92H)

If an object is destructible and the transaction group's destructor is active the object may not be used by a script. If a script is invoked which uses an object which has been destructed, an ERR_OBJECT_DESTRUCTED error code will be returned by the secure module.

The exemplary embodiment of the present invention is preferably placed within a durable stainless steel, token-like can. It is understood that an exemplary secure module can be placed in virtually any articulatable item. Examples of articulatable items include credit cards, rings, watches, wallets, purses, necklaces, jewelry, ID badges, pens, clipboards, etc.

The secure module **108** preferably is a single chip "trusted computer". By the word "trusted" it is meant that the computer is extremely secure from tampering by unwarranted means. The secure module incorporates a numeric coprocessor optimized for math intensive encryption. The BIOS is preferably immune to alteration and specifically designed for very secure transactions.

Each secure module can have a random "seed" generator with the ability to create a private/public key set. The private key never leaves the secure module and is only known by the secure module. Furthermore, discovery of the private key is prevented by active self-destruction upon wrongful entry into the secure module. The secure module can be bound to the user by a personal identification number (PIN).

When transactions are performed by the secure module **108** certificates of authentication are created by either or both the secure module and a system the secure module communicates with. The certificate can contain a variety of information. In particular, the certificate may contain:

1) who is the secure module user via a unique registration number and a certified public key.

2) when the transaction took place via a true-time stamping of the transaction.

3) where the transaction took place via a registered secure module interface site identification.

4) security information via uniquely serialized transactions and digital sign on message digests.

5) secure module status indicated as valid, lost, or expired.

Although a preferred embodiment of the method and apparatus of the present invention has been illustrated in the accompanying Drawings and described in the foregoing Detailed Description, it will be understood that the invention is not limited to the embodiment disclosed, but is capable of numerous rearrangements, modifications and substitutions without departing from the spirit of the invention as set forth and defined by the following claims.

What is claimed is:

1. A method for electronically transferring units of exchange between a first module and a second module, comprising the steps of:

a. initiating communication between said first module and an electronic device;

b. passing a first value datum from said first module to said electronic device;

c. passing said first value datum from said electronic device to said second module;

d. performing a mathematical calculation on said first value datum thereby creating a second value datum;

e. passing said second value datum from said second module to said electronic device;

f. passing said second value datum from said electronic device to said first module;

g. storing said second value datum in said first module; and

h. discontinuing communication between said first module and said electronic device.

2. The method of claim 1, wherein said first value datum represents a monetary equivalent.

3. The method of claim 1, wherein said first value datum is encrypted.

4. The method of claim 1, wherein said second value datum is encrypted.

5. The method of claim 3, wherein the step of performing a mathematical calculation comprises the steps of:

m. decrypting said first value datum with a public key thereby creating a decrypted value;

n. performing at least one of an addition function and a subtraction function on said decrypted value thereby creating a value result; and

o. encrypting said value result with a private key thereby creating said second value datum.

6. The method of claim 1, wherein the step (b) of passing is performed over at least a single conductive contact.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,949,880 Page 1 of 2
DATED : Sep. 7, 1999
INVENTOR(S) : Curry et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 2, line 57	Replace ‟electromagnetic″ With --electro-magnetic--
Column 5, line 15	Before ‟information″ Remove --is--
Column 8, line 26	Before ‟module″ Remove --is--
Column 12, line 47	Replace ‟ERR BAD_PIN LENGTH″ With --ERR_BAD_PIN_LENGTH--
Column 17, line 34	Replace ‟ERR_BAD_OBJECT ID″ With --ERR_BAD_OBJECT_ID--

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,949,880 Page 2 of 2

DATED : Sept 7, 1999

INVENTOR(S) : Curry et al

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 20, line 6	Replace "ERR MIAC_NOT_LOCKED" With --ERR_MIAC_NOT_LOCKED--
Column 20, line 48	Replace "ERR_BAD OBJECT_TYPE" With --ERR_BAD_OBJECT_TYPE--
Column 21, line 58	Replace "ERR_BAD NAME_LENGTH" With --ERR_BAD_NAME_LENGTH--

Signed and Sealed this

Twenty-fifth Day of April, 2000

Attest:

Q. Todd

Q. TODD DICKINSON

Attesting Officer

Director of Patents and Trademarks

PATENT APPLICATION SERIAL NO. ____________________

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE
FEE RECORD SHEET

```
03/06/1998 NTRAN1  00000005 DA#:040031   08978798
01 FC:101          790.00 CH
```

PTO-1556
(5/87)

#41

5-18-98

PATENT APPLICATION
DOCKET NO.:20661-429D1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
REQUEST FOR FILING RULE 60 APPLICATION

In re Application of:

Stephen M. Curry, et al.

This Application is a: **DO NOT USE FOR CIPs**

X Divisional) application filed under 37 CFR 1.60

of pending parent application:

Serial No.:	08/594,975
Filed:	January 31, 1996
Examiner:	White, C.
Group:	2202

Title: TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

Box Application
To The Assistant Commissioner
for Patents
Washington, D.C. 20231

CERTIFICATE OF MAILING BY EXPRESS MAIL

"EXPRESS MAIL" Mailing Label No. EM492669214US
Date of Deposit 11-26-97
I hereby certify that this paper or fee is being deposited with the U.S. Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231
Type or Print Name CAROL MARSTALLER
Carol Marstaller
Signature

Dear Sir:

1. **Attached** is a copy of the parent application as **originally** filed (The Applicant has enclosed the best copy which is presently available. Please disregard the stray marks. If necessary, a substitute specification will be filed at a later date.), including:

X Abstract

X Specification, claims and attachments (unamended clean copy) as originally filed (97 pages, including Abstract) **(must be attached)**

_X_ Drawings (**<u>must</u> be attached if originally filed):** <u>8</u> sheet(s) of 6 figures _x_ informal; __ Formal of size ____ 8½ x 11" ____ A4 ____ 13" ____ 14"

1A. <u>Always</u> X one box, only:

_X_ 1. <u>Signed</u> declaration or oath as originally filed in prior application <u>attached</u>

____ 2. <u>NO</u> Declaration or fee is enclosed; this is a filing under Rule 60(d).

NOTE: No amendments (if any) referred to in the Oath/Declaration filed to complete the prior application introduced new matter.

____ 2. This Rule 60 application is hereby filed by <u>less than all of the inventors</u> named in the prior application. Petition is hereby made requesting deletion as inventor(s) of the following who is/are **<u>not</u>** inventor(s) of the invention being claimed in this Rule 60 application:

1. ____ 2. ____
3. ____ 4. ____

____ 3. Transfer the drawings from the prior application to this application and **<u>abandon</u>** said prior application as of the filing date accorded this application. A <u>third</u> copy of this letter is <u>attached</u> for filing in the prior application file.

____ 4. Priority is claimed under 35 U.S.C. 119/365 based on filing in ____ (country) of

Application No.	Filing Date	Application No.	Filing Date
(1) ____	____	(4) ____	____
(2) ____	____	(5) ____	____
(3) ____	____	(6) ____	____

____ a. ___ (No.) Certified copy/copies attached.

____ b. Certified copy/copies previously filed on ____ in U.S. Application No. ____, filed on ____.

____ c. Certified copy/copies filed during International stage of PCT/____.

____ d. Priority is also claimed from PCT/____ filed ____.

X 5. Prior application is assigned to **Dallas Semiconductor Corporation** by means of an Assignment recorded on May 6, 1996, Reel 8029, Frame 0098.

____ 6. Attached is an Assignment and Cover Sheet. Please return the recorded Assignment to the undersigned.
(NOTE: add assignment filing fee below.)

X 7. The power of attorney in the prior application is to at least:
The address of whom is in item 8.

X	JEFFERY E. BACON Reg. No. 35,055
X	THOMAS L. CANTRELL Reg. No. 20,849
____	GEORGE E. CLARK Reg. No. 25,133
X	THOMAS L. CRISMAN Reg. No. 24,846
X	STUART D. DWORK Reg. No. 31,103
X	H. MATHEWS GARLAND Reg. No. 19,129
X	J. KEVIN GRAY Reg. No. 37,141
X	STEVEN R. GREENFIELD Reg. No. 38,166
X	CRAIG A. HOERSTEN Reg. No. 38,917
____	ROBERT H. KELLY Reg. No. 33,922
____	JOHN R. KIRK JR. Reg. No. 24,477
X	ROGER L. MAXWELL Reg. No. 31,855
____	ROBERT McFALL Reg. No. 28,968
____	MICHELE MOBLEY Reg. No. 35,616
X	STANLEY R. MOORE Reg. No. 26,958
X	P. WESTON MUSSELMAN JR. Reg. No. 31,644
X	ANDRE M. SZUWALSKI Reg. No. 35,701
X	GERALD T. WELCH Reg. No. 30,332

____ 7a. Recognize Steven R. Greenfield, Reg. No. 38,166 as having associate power of attorney.
(Name and Reg. No.; Address as in item 8 unless otherwise indicated)

____ 7b. Steven R. Greenfield, Reg. No. 38,166, was recognized as associate power of attorney in the parent application.

____ 7c. Since a power does not appear in the original papers, a copy of the power in the prior application is attached.

8. Address all future communications to:

Steven R. Greenfield
Jenkens & Gilchrist, P.C.
1445 Ross Avenue, Suite 3200
Dallas, Texas 75202

X 9. Amend the specification by inserting before the first line of the application the sentence: -- This application is a Divisional of Application No. 08/594,975 filed on January 31, 1996.

A1

____ 10. ____ (No.) Verified Statement(s) establishing "small entity" status under Rules 9 & 27
____ have been filed in above prior application (and hence are applicable hereto)
____ are attached hereto.

X 11. **PETITION to extend the life** of the above prior application to at least the date hereof. (One box must be X'd)
___ is being concurrently filed in that prior application.
____ was previously filed in that prior application (Check length of prior extension).
x is not necessary for copendency **(Double check** before X'ing this box).

X 12. **INFORMATION DISCLOSURE STATEMENT:** Attached is Form PTO-1449 listing documents cited by Applicant or the PTO in the parent application(s) relied upon under 35 USC 120 and referenced in item 9 above Please fully consider those documents and advise that they have been considered in this new application as by returning a copy of the enclosed Form PTO-1449 with the Examiner's initials in the left column per MPEP 609.

____ 13. Attached is a Rule 103(a) Petition to Suspend Action.

X 14. **PRELIMINARY AMENDMENT to be entered before fee calculation:** (Do not make amendments here except for correction of improper multiple dependencies or cancellation of whole claims or multiple dependencies for purpose of reducing the filing fee per MPEP §§ 506 and 607; do not cancel all claims).

96

Prior to a first Office Action, Kindly amend the Application as follows:

_x_ Please cancel Claims 1-15.

____ Please amend the claims as follows:

15. The following Filing Fee calculation is based on the claims filed less any claims canceled by the Preliminary Amendment of Item 14.

NOTE: If box 1A2 is X'd, do not pay any fees at this time

					SMALL ENTITY RATE		LARGE ENTITY RATE		
BASIC FEE			Design Appln.		$150	OR	$300	=	$___
BASIC FEE			Utility Appln.		$395	OR	$790	=	$ 790.00
	NUMBER FILED			NUMBER EXTRA					
TOTAL CLAIMS	6	-20	=	0 (at least 0)	x 11	OR	x 22	=	+$ 0
INDEP. CLAIMS	1	- 3	=	0 (at least 0)	x 41	OR	x 82	=	+$ 0
If any proper multiple dependent claim (ignore improper) is present					+$130	OR	+$260	=	+$___
If assignment is x'ed (Item 6)............... add recording fee ($40.00)									+$___
If "petition" Item 13 above is X'd ....add petition fee ($130.00)									+$___
TOTAL FILING FEE	=								**$790.00**

____ 16. ATTACHED: Drawing Change Request and Replacement Drawings

____ 17. Please enter the Preliminary Amendment attached hereto after assigning an Appln. No. The Fee for entering the attached Preliminary Amendment is calculated below:

	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST # PREVIOUSLY PAID FOR		PRESENT EXTRA	SMALL ENTITY RATE			LARGE ENTITY RATE		
TOTAL CLAIMS	____	-	20 (at least 20)	=	____ (at least 0)	x11	=	OR	x22	=	$ 0
INDEP. CLAIMS	____	-	3 (at least 3)	=	____ (at least 0)	x41	=	OR	x82	=	+$ 0
If amendment enters proper multiple dependent claim(s) into this application for first time, add (per application)						+120	=	OR	+240	=	+$___
	plus **TOTAL FILING FEE** from Item 15										+$ 0
	TOTAL FEE	=									**$ 790.00**

___ 18. A check in the amount of $_____ to cover the TOTAL FEE is attached. Please charge any deficiency or credit any overpayment to Deposit Account No. 10-0447.

X 19. Please charge **Dallas Semiconductor Corporation Deposit Account No. 04-0031** in the amount of $790.00 to cover the TOTAL FEE. This sheet is attached in duplicate.

CHARGE STATEMENT: The Commissioner is hereby authorized to charge any fee specifically authorized hereafter, or any missing or insufficient fee(s) filed, or asserted to be filed, or which should have been filed herewith or concerning any paper filed hereafter, and may be required under 37 CFR 1.16-1.18 (missing or insufficiencies only) now or hereafter relative to this application and for the resulting Official Document under 37 CFR 1.20, OR credit any overpayment to **Dallas Semiconductor Corporation Deposit Account No. 04-0031**, for which purpose a duplicate copy of this sheet is attached. In the event that Dallas Semiconductor Corporation Deposit Account No. 04-0031 cannot be charged hereby to cover the TOTAL FEE, please charge the TOTAL FEE to my Deposit Account No. 10-0447.

This CHARGE STATEMENT does not authorize charge of the issue fee until/unless an issue fee transmittal form is filed.

Respectfully submitted,

JENKENS & GILCHRIST, P.C.

By: ______________________
Steven R. Greenfield
Registration No. 38,166

Dated: November 26, 1997

JENKENS & GILCHRIST, P.C.
1445 Ross Avenue, Suite 3200
Dallas, Texas 75202
Tel: 214/855-4789
Fax: 214/855-4300

NOTE: File this Request in duplicate with a return postcard and attachments or in triplicate if item 3 is marked.

08/978798

Patent Application
Docket #20661/429

CERTIFICATE OF MAILING BY EXPRESS MAIL

"EXPRESS MAIL" Mailing Label No. IB885275721US
Date of Deposit January 31, 1996
I hereby certify that this paper or fee is being deposited with the U.S. Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, D.C. 20231

Type or Print Name JEANNE A. HOWARD

Jeanne A. Howard
Signature

TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

CROSS REFERENCE TO OTHER APPLICATIONS

The following applications of common assignee contains related subject matter and is hereby incorporated by reference:

Serial No. UNKNOWN, filed January 31, 1996, entitled METHOD, APPARATUS, SYSTEM AND FIRMWARE FOR SECURE TRANSACTIONS; and

Serial No. UNKNOWN, filed January 31, 1996, entitled METHOD, APPARATUS AND SYSTEM FOR TRANSFERRING UNITS OF VALUE.

08/978798

Patent Application
Docket #20661/429

Ins A1

BACKGROUND OF THE INVENTION

Technical Field of the Invention

The present invention relates to a method and system for transferring valuable information securely between a secure module and another module. More particularly, the present invention relates to transferring units of value between a microprocessor based secure module and another module used for carrying a monetary equivalent.

Description of Related Art

In the past the preferred means for paying for an item was cash. As our society has become more advanced, credit cards have become an accepted way to pay for merchandise or services. The payment is not a payment to the merchant, but instead is a credit given by a bank to the user that the merchant accepts as payment. The merchant collects money from the bank based on the credit. As time goes on, cash is used less and less, and money transfers between parties are becoming purely electronic.

3

Present credit cards have magnetic strips to identify the owner of the card and the credit provider. Some credit cards have electronic circuitry installed that identifies the credit card owner and the credit or service provider (the bank).

The magnetic strips installed in present credit cards do not enable the card to be used as cash. That is the modern credit card does not allow the consumer to buy something with the credit card and the merchant to receive cash at the time of the transaction. Instead, when the consumer buys something on credit, the merchant must later request that the bank pay for the item that the consumer bought. The bank then bills the consumer for the item that was bought.

Thus, there is a need for an electronic system that allows a consumer to fill an electronic module with a cash equivalent in the same way a consumer fills his wallet with cash. When the consumer buys a product or service from a merchant, the consumer's module can be debited and the merchant's cash drawer can be credited without any further transactions with a bank or service provider.

SUMMARY OF THE INVENTION

The present invention is an apparatus, system and method for communicating a cash equivalent electronically to and from a portable module. The portable module can be used as a cash equivalent when buying products and services in the market place.

The present invention comprises a portable module that can communicate to a secure module via a microprocessor based device. The portable module can be carried by a consumer, filled with electronic money at an add-money station, and be debited by a merchant when a product or service is purchased by the consumer. As a result of a purchase, the merchant's cash drawer will indicate an increase in cash value.

BRIEF DESCRIPTION OF THE DRAWINGS

A more complete understanding of the method and apparatus of the present invention may be had by reference to the following Detailed Description when taken in conjunction with the accompanying Drawings wherein:

FIGURE 1 depicts an exemplary system for transferring valuable information between a module and a secure device;

FIGURE 2 is a block diagram of an embodiment of a portable module;

FIGURE 3 is a block diagram of an embodiment of a microprocessor based module;

FIGURE 4 is an exemplary technique for transferring valuable data securely into a portable module;

FIGURE 5 is an exemplary technique for transferring valuable data securely out of a portable module;

FIGURE 6 is an exemplary organization of the software and firmware within a secure microprocessor based device; and

FIGURE 7 is an exemplary configuration of software and firmware within a secure microprocessor based device.

DETAILED DESCRIPTION OF A PRESENTLY PREFERRED EXEMPLARY EMBODIMENT

FIGURE 1 depicts a block diagram of an exemplary system 100 for transferring valuable information to and from a portable module. A portable module 102, which will be described in more detail later, communicates to a microprocessor based device 104. The portable module 102 may contain information that represents units of exchange or a currency equivalent. The microprocessor based device 104 can be any of an unlimited number of devices. For example, the microprocessor based device 104 could be a personal computer, an add-a-fare machine at a train or bus station (similar to those in today's District of Columbia metro stations), a turn style, a toll booth, a bank's terminal, a ride at a carnival, a washing machine at a Laundromat, a locking device, a mail metering device or any device that controls access, or meters a monetary equivalent, etc.

The means for communication 106 between the portable module 102 and the microprocessor based device 104 is preferably via a single wire or contact connection. The single wire connection 106 preferably incorporates a communication protocol that allows the portable module 102 and the microprocessor based device 104 to communicate in a bidirectional manner. Preferably the communication protocol is a one-wire protocol developed by Dallas Semiconductor. It is understood that the means for communicating 106 is not limited to a single wire connection. The communication means 106 could be multiple wires, a wireless communication system, infrared

8

light, any electro-magnetic means, a magnetic technique, or any other similar technique.

The microprocessor based device 104 is electrically connected to another microprocessor based device, which is preferably a secure device 108. The term secure device means that the device is designed to contain a secret code and the secret code is extremely difficult to learn. An example of a secure device 108 is explained later in this document.

The microprocessor based device 104 can be connected to a variety of other devices. Such devices include, but are not limited to a cash acceptor 110, an automatic teller machine (ATM) 112, a credit card reader 114, and a phone line 116.

The cash acceptor 110 is adapted to receive cash in the form of currency, such as dollar bills or coins. The cash acceptor 110, preferably, determines the value of the accepted currency. The cash acceptor 110 communicates to the microprocessor based device 104 and informs the device 104 of how much currency has been deposited in the cash acceptor 110.

The cash acceptor 110 can also be a device which provides currency. That is, the cash accepter 110 in response to a communication from the microprocessor based device 104, may provide a metered amount of currency to a person.

The credit card reader 114, and ATM 112 can also be attached to the microprocessor based device 104. The credit card reader 114 could be used to read a user's credit card and then, when authorized, either communicate to the microprocessor based device 104 that units of exchange need to be added to the portable module or that units of exchange need to be extracted from the portable module to pay for a good, service or credit card bill.

The ATM 112 may also be connected to the microprocessor based device. Via communications from the ATM 112, the microprocessor based device 104 can be informed that units of exchange need to be added or subtracted from the portable module 102.

Furthermore, it is also possible that the microprocessor based device 104 is connected to a phone line 116. The phone line may be used for a variety of things. Most importantly, the phone line may be used to allow the microprocessor based device 104 to communicate with a network of devices. Such telephonic communication may be for validating transactions or for aiding the accounting of transactions that are performed via the microprocessor based device's 104 aid. It is further understood that the phone line may be any of a vast variety of communication lines including wireless lines. Video, analog, or digital information may be communicated over the phone line 116.

FIGURE 2 depicts a preferred exemplary portable module 102. The portable module 102 is preferably a rugged read/write data carrier that can act as a localized data base and be easily accessed with minimal hardware. The module can be incorporated in a vast variety of portable items which includes, but is not limited to a durable micro-can package that is highly resistant to environmental hazards such as dirt, moisture, and shock. The module can be incorporated into any object that can be articulated by a human or thing, such as a ring, bracelet, wallet, name tag, necklace, baggage, machine, robotic device, etc. Furthermore, the module 102 could be attached to a stationary item and the microprocessor based device 104 may be articulated to the portable module 102. For example, the module 102 may be attached to a piece of cargo and a module reader may be touched to or brought near the module 102. The module reader may be part of the microprocessor based device 104.

The portable module 102 comprises a memory 202 that is preferably, at least in part, nonvolatile memory for storing and retrieving vital information pertaining to the system to which the module 102 may become attached to. The memory 202 may contain a scratchpad memory which may act as a buffer when writing into memory. Data is first written to the scratchpad where it can be read back. After data has been verified, the data is transferred into the memory.

The module 102 also comprises a counter 206 for keeping track of the number of transactions the module has performed (the number of times certain data in the memory of the module has been changed). A timer 102 may be provided in the module to provide the ability to time stamp transactions performed by the module. A memory controller 204 controls the reading and writing of data into and out of the memory 202.

The module also may comprise an identification number 210. The identification number preferably uniquely identifies the portable module from any other portable module.

An input/output control circuit 212 controls the data flow into and out of the portable module 102. The input/output control ("I/O") 212 preferably has an input buffer and an output buffer and interface circuitry 214. As stated above, the interface circuitry 214 is preferably a one-wire interface. Again, it is understood that a variety of technologies can be used to interface the portable module 102 to another electronic device. A single wire or single connection is preferred because the mechanics of making a complete connection is simplified. It is envisioned that a proximity/wireless communication technique is also a technique for communicating between the module 102 and another device. Thus, the interface circuit 214 can be a single wire, multiple wire, wireless, electromagnetic, magnetic, light, or proximity, interface circuit.

FIGURE 3 depicts a block diagram of an exemplary secure microprocessor based device ("secure device") 108. The secure device circuitry can be a single integrated circuit. It is understood that the secure device 108 could also be a monolithic or multiple circuits combined together. The secure device 108 preferably comprises a microprocessor 12, a real time clock 14, control circuitry 16, a math coprocessor 18, memory circuitry 20, input/output circuitry 26, and an energy circuit 34.

The secure device 108 could be made small enough to be incorporated into a variety of objects including, but not limited to a token, a card, a ring, a computer, a wallet, a key fob, a badge, jewelry, a stamp, or practically any object that can be grasped and/or articulated by a user of the object. In the present system 100, the secure device 108 is preferably adapted to be a trusted certifying authority. That is the secure device 108 is a trusted computer. The secure device 108 comprises a numeric coprocessor 18 optimized for math intensive encryption. The BIOS is immune to alteration and is specifically designed for secure transactions. This secure device 108 is preferably encased in a durable, dirt, moisture and shock resistant stainless steel enclosure, but could be encased in wide variety of structures so long as specific contents of the secure device 108 are extremely difficult to decipher. The secure device 108. The secure device 108 may have the ability to store or create a private/public key set, whereby the private key never leaves the secure device 108 and is not revealed under almost any circumstance.

Furthermore, the secure module 108 is preferably designed to prevent discovery of the private key by an active self-destruction of the key upon wrongful entry.

The microprocessor 12 is preferably an 8-bit microprocessor, but could be 16, 32, 64 or any operable number of bits. The clock 14 provides timing for the module circuitry. There can also be separate clock circuitry 14 that provides a continuously running real time clock.

The math coprocessor circuitry 18 is designed and used to handle very large numbers. In particular, the coprocessor will handle the complex mathematics of RSA encryption and decryption or other types of math intensive encryption or decryption techniques.

The memory circuitry 20 may contain both read-only-memory and non-volatile random-access-memory. Furthermore, one of ordinary skill in the art would understand that volatile memory, EPROM, SRAM and a variety of other types of memory circuitry might be used to create an equivalent device.

Control circuitry 16 provides timing, latching and various necessary control functions for the entire circuit.

An input/output circuit 26 enables bidirectional communication with the secure module 108. The input/output circuitry 26 preferably comprises at least an output buffer and an input buffer. For communication via a one-wire bus, one-wire interface circuitry can be included with the input/output circuitry 26. It is understood that the input/output circuitry 26 of the secure device 108 can be designed to operate on a single wire, a plurality of wires or any means for communicating information between the secure module 108 and the microprocessor based device 104.

An energy circuit 34 may be necessary to maintain stored information in the memory circuitry 20 and/or aid in powering the other circuitry in the module 108. The energy circuit 34 could consist of a battery, capacitor, R/C circuit, photo-voltaic cell, or any other equivalent energy producing circuit or means.

The firmware architecture of the secure module 108 and how it operates within the exemplary system for transferring valuable information, such as units of exchange or currency, between the secure module 108 and a portable module 102 will now be discussed. The secure module 108 provides encryption and decryption services for confidential data transfer through the microprocessor based device 104. The following examples are intended to illustrate a preferred feature set of the secure module 108 and to explain the services that the exemplary system 100 can offer. These applications and examples by no means limit the capabilities of the invention, but instead bring to light a sampling of its capabilities.

I. OVERVIEW OF THE PREFERRED SECURE MODULE 108 AND ITS FIRMWARE DESIGN

Referring to FIGURE 3 again, the secure module 108 preferably contains a general-purpose, 8051-compatible micro controller 12 or a reasonably similar product, a continuously running real-time clock 14, a high-speed modular exponentiation accelerator for large integers (math coprocessor) 18, input and output buffers 28, 30 with a one-wire interface 32 for sending and receiving data, 32 Kbytes of ROM memory 22 with preprogrammed firmware, 8 Kbytes of NVRAM (non-volatile RAM) 24 for storage of critical data, and control circuitry 16 that enables the micro controller 12 to be powered up to interpret and act on the data placed in an input data object. The module 108 draws its operating power from a single wire, one-wire communication line. The micro controller 12, clock 14, memory 20, buffers 28, 30, one-wire front-end 32, modular exponentiation accelerator 18, and control circuitry 16 are preferably integrated on a single silicon chip and packaged in a stainless steel micro can using packaging techniques which make it virtually impossible to probe the data in the NVRAM 24 without destroying the data. Initially, most of the NVRAM 24 is available for use to support applications such as those described below. One of ordinary skill will understand that there are many comparable variations of the module design. For example, volatile memory might be used, or an interface other than a one-wire interface could be used.

The secure module 108 is preferably intended to be used first by a Service Provider who loads the secure module 108 with data to enable it to perform useful functions, and second by an End User who issues commands to the secure module 108 to perform operations on behalf of the Service Provider for the benefit of the End User. For this reason, the secure module 108 offers functions to support the Service Provider in setting up the module for an intended application. It also offers functions to allow the End User to invoke the services offered by the Service Provider.

Each Service Provider can reserve a block of NVRAM memory to support its services by creating a transaction group 40(refer to FIGURES 6 and 7). A transaction group 40 is simply a set of software objects 42 that are defined by the Service Provider. These objects 42 include both data objects (encryption keys, transaction counts, money amounts, date/time stamps, etc.) and transaction scripts 44 which specify how to combine the data objects in useful ways. Each Service Provider creates his own transaction group 40, which is independent of every other transaction group 40. Hence, multiple Service Providers can offer different services in the same module 108. The number of independent Service Providers that can be supported depends on the number and complexity of the objects 42 defined in each transaction group 40. Examples of some of the objects 42 that can be defined within a transaction group 40 are the following:

RSA Modulus
RSA Exponent
Transaction Script
Clock Offset
Random SALT
Configuration Data

Transaction Counter
Money Register
Destructor
Input Data
Output Data

Within each transaction group 40 the secure module 108 will initially accept certain commands which have an irreversible effect. Once any of these irreversible commands are executed in a transaction group 40, they remain in effect until the end of the module's useful life or until the transaction group 40, to which it applies, is deleted from the secure module 108. In addition, there are certain commands which have an irreversible effect until the end of the module's life or until a master erase command is issued to erase the entire contents of the secure module 108. These commands will be discussed further below. These commands are essential to give the Service Provider the necessary control over the operations that can be performed by the End User. Examples of some of the irreversible commands are:

Privatize Object
Lock Transaction Group
Lock Object
Lock Micro-In-A-Can™

Since much of the module's utility centers on its ability to keep a secret, the Privatize command is a very important irreversible command.

Once the secure module 108, as a whole, is locked, the remaining NVRAM memory 24 is allocated for a circular buffer for holding an audit trail of previous transactions. Each of the transactions are identified by the number of the transaction group, the number of objects 42 within the specified group, and the date/time stamp.

The fundamental concept implemented by the firmware is that the Service Provider can store transaction scripts 44 in a transaction group 40 to perform only those operations among objects that he wishes the End User to be able to perform. The Service Provider can also store and privatize RSA key or keys (encryption keys) that allow the secure module 108 to "sign" transactions on behalf of the Service Provider, thereby guaranteeing their authenticity. By privatizing and/or locking one or more objects 42 in the transaction group 40, the Service Provider maintains control over what the secure module 108 is allowed to do on his behalf. The End User cannot add new transaction scripts 44 and is therefore limited to the operations on objects 42 that can be performed with the transaction scripts 44 programmed by the Service Provider.

II. USAGE MODELS OF THE SECURE MODULE 108 AND PORTABLE MODULE 102

This section presents practical applications of the system 100. Each of these applications is described in enough detail to make it clear why the secure module 108 and portable module 102 are important to the system application.

A. TRANSFERRING UNITS OF EXCHANGE OUT OF A PORTABLE MODULE 102

This section describes an example of how a portable module 102 and a secure module 108 operate in conjunction with the microprocessor based device 104 so that units of exchange can be securely transferred out of the portable module 102 and deposited into the secure module 108 and/or potentially communicated to at least one of the cash acceptor 110, ATM 112, credit card reader 114, or the phone line 116.

Referring to FIGURE 4, initially the portable module 102 contains its ID number, a count within its transaction counter and an encrypted data packet stored in memory. Encrypted within the data packet is the portable modules ID number, the portable modules transaction count number, and the amount of value (the monetary value) of the portable module at the present time X1.

The user of the portable module touches, or somehow puts the portable module 102 into communication with the microprocessor based device 104. For explanation purposes, suppose the portable module 102 is being used as a token used to pay for a train fare. Thus, the microprocessor based device 104 could be, in this case, a turn style that allows the user to enter a train platform. The cost of entering the train platform is known by the microprocessor based device 104.

The microprocessor based device 104 reads the portable module's serial number, transaction count, and the encrypted data packet X2. This data could be referred to as a first data.

The microprocessor device 104 then provides the first data along with a first value, being the amount of value to be debited from the portable token (the train fare), to the secure module 108 X3. The secure module 108 decrypts the encrypted data found in the first data using a public key X4.

Next, the secure module 108 makes a few comparisons to make sure that the data received is good data and not counterfeit. The secure module 108 compares the serial number received in the first data with the decrypted serial number X5. If the two serial numbers match then the secure module 108 compares the transaction count received in the first data with the decrypted transaction count X6. If the two transaction counts match then the secure module is comfortable that the data received is not counterfeit data. It is understood that the comparisons can be done in any order.

Furthermore, there may have been a time stamp sent from the portable module 102. The time stamp may indicate a variety of things. One thing could be an indication of whether the portable module is still valid or the time stamp may further enable the secure module to decide if the data is or is not counterfeit.

Assuming all the data passed to the secure module 108 is determined to be valid data, the secure module 108 subtracts the first value, the train fare, from the monetary value of the portable module 102 X7. The decrypted transaction count is then incremented.

A register within the secure module 108 is increased by the amount of the first value, the train fare, so that the secure module can keep an accounting of the amount of "money" it has collected X8. The secure module 108 creates a data packet, a second data, which comprises at least the portable module's serial number, the incremented transaction count, and the reduced monetary value of the portable module 102. The second data packet is then encrypted by the secure module 108 using a private key X9.

The microprocessor based device 104 receives the encrypted second data packet, passes the encrypted second data packet to the portable module 102 X10,and opens the turn style, to let the module's user onto the train platform. The portable module 102 receives the encrypted second data packet and stores it in memory X11. The portable module also increments its transaction count indicating that another transaction has occurred X12.

Thus, the above description indicates how valuable information can be transferred between a portable insecure module 102 and a secure module 108 wherein there is a conservation of value. That is, no value is gained or lost. Value that was in the portable module 102 was decreased by the same amount value was added to the secure module 108. In the example provided, the decrease and increase in value was equal to a train fare. Such an increment or decrement can also be equal to an amount provided by an ATM, credit card transaction, cash acceptor, etc.

It is also understood that the insecure portable module 102 could be another secure module similar to the secure module in the system, but programed to act like a portable module 102.

B. TRANSFERRING UNITS OF EXCHANGE INTO THE PORTABLE MODULE 102

In this example, for simplicity, suppose the portable module does not have any monetary value and the user of the portable module wishes to "fill it up" with value. Suppose the user wishes to take cash out of an ATM machine and instead of pocketing the cash, the user wishes to put the cash value into the portable module 102.

Referring to FIGURE 5, the portable module 102 contains its ID number, a transaction count and an encrypted data packet containing the portable module's ID number, transaction count and the monetary value of the portable module 102 Y1. The microprocessor based device 104, which in this example could be part of the ATM machine 112, receives the information contained in the portable module 102 when a communication is initiated between the portable module 102 and the microprocessor based device 104 Y2.

The microprocessor based device 104 passes the module's serial number, transaction count, and encrypted data packet as a first data packet to the secure module 108. The microprocessor based device also passes the amount of amount of monetary value to add to the portable module 102, as indicated by the ATM 112, to the secure module 108 Y3.

The secure module 108 decrypts the encrypted data passed to it using a public key Y4. The secure module 108 then makes a few comparisons to make sure that the data it has just received is valid and not counterfeit. The secure module 108 compares the serial number (ID number) received in the first data packet with the serial number (ID number) found in the decrypted data Y5. The secure module 108 also compares the transaction count passed the first data packet with the transaction count found in the decrypted data Y6. If the serial numbers and transaction counters match, then the secure module decides that the data received is valid and the secure module adds the monetary value, indicated by the ATM to the monetary value of the decrypted data Y7. The decrypted transaction count is incremented Y8. A register within the secure module may be decremented by the same amount that the monetary value of the decrypted data was increased Y8.

The secure module 108 creates a second data packet, that contains the portable module's ID number, the incremented transaction counter and the increased monetary value. The second data packet is then encrypted using a private key Y10.

The microprocessor based device 104 reads the encrypted second data packet and sends it to the portable module 102 Y11. The portable module receives the encrypted second data packet and stores it in memory Y12. The portable module also advances its transaction counter Y13. The result being that the portable module now has the value of the cash withdrawn from the ATM 112. Furthermore, a record of the transaction may have been recorded and kept in the secure module, as well as by the bank that operates the ATM 112.

Exemplary Firmware Definitions for Use With the Secure Module

Object The most primitive data structure accepted by and operated on by the secure modules firmware. A list of valid objects and their definitions is provided in the next section.

Group A self-contained collection of objects. An object's scope is restricted to the group of which it is a member.

Group ID A number preferably between 0 and 255 representing a specific group.

Object ID	A number preferably between 0 and 255 representing a specific object within a specific group.
Object Type	Preferably a 1-byte type specifier that describes a specific object.
PIN	An alphanumeric Personal Identification number that is preferably eight bytes in length.
Common PIN	The PIN that controls access to shared resources such as the audit trail. It is also used to control the host's ability to create and delete groups.
Group PIN	The PIN that controls access to operations specific to objects within a group.

Audit Trail A record of transactions occurring after the secure module has been locked.

Locked Object An object which has been locked by executing the lock object command. Once an object is locked it is not directly readable.

Private Object An object which has been privatized by executing the privatize object command. Once an object is private, it is not directly readable or writable.

Locked Group A group which has been locked using the locked group command. After a group has been locked it will not allow object creation.

35

Composite Object A combination of several objects. The individual objects inherit the attributes of the composite object.

Exemplary Object Definitions

RSA Modulus A large integer preferably of at most 1024 bits in length. It is the product of 2 large prime numbers that are each about half the number of bits in length of the desired modulus size. The RSA modulus is used in the following equations for encrypting and decrypting a message M:

$$\text{Encryption:} \quad C = M^{e} \pmod N \tag{1}$$

$$\text{Decryption:} \quad M = C^{d} \pmod N \tag{2}$$

where C is the cyphertext, d and e are the RSA exponents (see below), and N is the RSA modulus.

RSA Exponent Both e and d (shown in equations 1 and 2 above) are RSA exponents. They are typically large numbers but are smaller than the modulus (N). RSA exponents can be either private or public. When RSA exponents are created in the secure module, they may be declared as either. Once created an exponent may be changed from a public exponent to a private exponent. After an exponent has been made private, however, it will remain private until the transaction group 40 to which it belongs is destroyed.

Transaction Script A transaction script is a series of instructions to be carried out by the secure module. When invoked the secure module firmware interprets the instructions in the script and

38

places the results in the output data object (see below). The actual script is simply a list of objects. The order in which the objects are listed specifies the operations to be performed on the objects. transaction scripts 44 preferably may be as long as 128 bytes.

Transaction Counter The transaction counter object is preferably 4 bytes in length and is usually initialized to zero when it is created. Every time a transaction script, which references this object, is invoked, the transaction counter increments by 1. Once a transaction counter has been locked it is read only and provides an irreversible counter.

Money Register The money register object is preferably 4 bytes in length and may be used to represent money or some other form of credit. Once this object has been created, it must be locked to prevent a user from tampering with its value. Once locked the value of this object can be altered only by invoking a transaction script. A typical transaction group 40 which performs monetary transactions might have one script for withdrawals from the money register and one for deposits to the money register.

Clock Offset This object is preferably a 4 byte number which contains the difference between the reading of the secure module's real-time clock and some convenient time (e.g., 12:00 a.m., January 1, 1970). The true time can then be obtained from the secure module by adding the value of the clock offset to the real-time clock.

SALT A SALT object is preferably 20 bytes in length and should be initialized with random data when it is created. When a host transmits a generate random SALT command, the secure module combines the previous SALT with the secure module's random number (produced preferably by randomly occurring power-ups) to generate a new random SALT. If the SALT object has not been privatized it may subsequently be read by issuing a read object command.

Configuration Data This is a user defined structure with preferably a maximum length of

128 bytes. This object is typically used to store configuration information specific to its transaction group 40. For example, the configuration data object may be used to specify the format of the money register object (i.e., the type of currency it represents). Since this object has no pre-defined structure, it may never be used by a transaction object.

Input Data

An input data object is simply an input buffer with preferably a maximum length of 128 bytes. A transaction group may have multiple input objects. The host uses input data objects to store data to be processed by transaction scripts 44.

Output Data

The output data object is used by transaction scripts as an output buffer. This object is automatically created when the transaction group is created. It is preferably 512 bytes in length and inherits password protection from its group.

Random Fill

When the script interpreter encounters this type of object it automatically pads the current message so that its length is 1 bit smaller than the length of the preceding modulus. A handle to this object is automatically created when the transaction group is created. It is a private object and may not be read using the read object command.

Working Register This object is used by the script interpreter as working space and may be used in a transaction script. A handle to this object is automatically created when the transaction group is created. It is a private object and may not be read using the read object command.

ROM Data This object is automatically created when the transaction group is created. It is a locked object and may not be altered using the write object command. This object is 8 bytes and length and its contents are identical to the 8 by ROM data of the Micro-In-A-Can™.

Preferred Secure module Firmware Command Set

Set Common PIN(01H)

Transmit (to secure module)

01H, old PIN, new PIN, PIN option byte

Receive data

CSB (command status byte) = 0 if successful, appropriate error code otherwise

Output length = 0

Output Data = 0

Notes:

The PIN option byte may be the bitwise-or of any of the following values:

PIN_TO_ERASE 00000001b (require PIN for Master Erase)

PIN_TO_CREATE 00000010b (require PIN for group creation).

Initially the secure module has a PIN (Personal Identification Number) of 0 (Null) and an option byte of 0. Once a PIN has been established it can only be changed by providing the old PIN or by a Master Erase. However, if the PIN_TO_ERASE bit is set in the option byte, the PIN can only be changed through the set common PIN command.

Possible error codes for the set common PIN command:

ERR_BAD_COMMON_PIN (Common PIN match failed)

ERR_BAD_PIN_LENGTH (New PIN length > 8 bytes)

ERR_BAD_OPTION_BYTE (Unrecognizable option byte)

For all commands described in this section, data received by the host will be in the form of a return packet. A return packet has the following structure:

Command status byte (0 if command successful, error code otherwise, 1 byte)

Output data length (Command output length, 2 bytes)

T4.70X

Output data (Command output, length specified above).

Master Erase (02H)

Transmit data

02H, Common PIN

T4.71X

Receive data

CSB = 0 if command was successful, ERR_BAD_COMMON_PIN otherwise

Output length = 0

Output data = 0

Notes:

If the LSB (least significant bit) of the PIN option is clear (i.e. PIN not required for Master Erase) then a

47

0 is transmitted for the Common PIN value. In general this text will always assume a PIN is required. If no PIN has been established a 0 should be transmitted as the PIN. This is true of the common PIN and group PINS (see below). If the PIN was correct the firmware deletes all groups (see below) and all objects within the groups. The common PIN and common PIN option byte are both reset to zero.

After everything has been erased the secure module transmits the return packet. The CSB is as described above. The output data length and output data fields are both set to 0.

Create Group (03H)

Transmit data

03H, Common PIN, Group name, Group PIN

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = 1 if successful, 0 otherwise

Output data = Group ID if successful, 0 otherwise

Notes:

The maximum group name length is 16 bytes and the maximum PIN length is eight bytes. If the PIN_TO_CREATE bit is set in the common PIN option byte and the PIN transmitted does not match the common PIN the secure module will set the OSC to ERR_BAD_COMMON_PIN.

Possible error return codes for the create group command:

ERR_BAD_COMMON_PIN (Incorrect common PIN)

✓ ERR_BAD_NAME_LENGTH (If group name length > 16 bytes)

T490x

49

ERR_BAD_PIN_LENGTH (If group PIN length > 8 bytes)

ERR_MIAC_LOCKED (The secure module has been locked)

ERR_INSUFFICIENT_RAM (Not enough memory for new group)

<u>Set Group PIN (04H)</u>

Transmit data

04H, Group ID, old GPIN, new GPIN

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = 0

Output data = 0

Notes:

The Group PIN only restricts access to objects within the group specified by the group ID transmitted in the command packet.

Possible error codes for the set group PIN command:

ERR_BAD_GROUP_PIN (Group PIN match failed)

ERR_BAD_PIN_LENGTH (New group PIN length > 8 bytes)

Create Object (05H)

Transmit data

05H, Group ID, Group PIN, Object type, Object attributes, Object data

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = 1 if successful, 0 otherwise

Output data = object ID if successful, 0 otherwise

Notes:

If the Create Object command is successful the secure module firmware returns the object's ID within the group specified by the Group ID. If the PIN supplied by the host was incorrect or the group has been locked by the Lock Group command (described below) the secure module returns an error code in the CSB. An object creation will also fail if the object is invalid for any reason. For example, if the object being created is an RSA modulus (type 0) and it is greater than 1024 bits in length. transaction script creation will succeed if it obeys all transaction scripts rules.

Possible error return codes for the create object command:

ERR_BAD_GROUP_PIN (Incorrect group PIN)

ERR_GROUP_LOCKED (The group has been locked)

ERR_MIAC_LOCKED (The secure module has been locked)

ERR_INVALID_TYPE (The object type specified is invalid)

ERR_BAD_SIZE (The objects length was invalid)

ERR_INSUFFICIENT_RAM (Not enough memory for new object)

Object types:	RSA modulus	0
	RSA exponent	1
	Money register	2
	Transaction counter	3
	Transaction script	4
	Clock offset	5
	Random SALT	6
	Configuration object	7
	Input data object	8
	Output data object	9

Object Attributes:	Locked	00000001b
	Privatized	00000010b

Objects may also be locked and privatized after creation by using the Lock Object and Privatize Object commands described below.

Lock Object (06H)

Transmit data

06H, Group ID, Group PIN, Object ID

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = 0

Output data = 0

Notes:

If the Group ID, Group PIN and Object ID are all correct, the secure module will lock the specified object. Locking an object is an irreversible operation.

Possible error return codes for the lock object command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_GROUP_LOCKED	(The group has already been locked)
ERR_MIAC_LOCKED	(The secure module has been locked)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Specified object does not exist)

Privatize Object (07H)

Transmit data

07H, Group ID, Group PIN, Object ID

Receive data

CSB = 0 if successful, appropriate error code otherwise

Notes:

If the Group ID, Group PIN and Object ID were valid the object will be privatized. Privatized objects share all the properties of locked objects but are not readable. Privatized objects are only modifiable through transaction scripts. Note that locking a privatized object is legal, but has no meaning since object privatization is a stronger operation than object locking. <u>Privatizing an object is an irreversible operation</u>.

Possible error return codes for the privatize object command:

ERR_BAD_GROUP_PIN (Incorrect group PIN)

T56x

56

ERR_GROUP_LOCKED	(The group has already been locked)
ERR_MIAC_LOCKED	(The secure module has been locked)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Specified object does not exist)

Make Object Destructable (08H)

Transmit data

08H, Group ID, Group PIN, Object ID

Receive data

CSB = 0 if successful, appropriate error code otherwise

Notes:

If the Group ID, Group PIN and Object ID were valid the object will be made destructable. If an object is destructable it becomes unusable by a transaction script after the groups destructor becomes active. If no destructor object exists within the transaction group the destructible object attribute bit has no affect. Making an object destructable is an irreversible operation.

Possible error return codes for the make object destructable command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_GROUP_LOCKED	(The group has already been locked)
ERR_MIAC_LOCKED	(The secure module has been locked)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Specified object does not exist)

T580X

<u>Lock Secure module (09H)</u>

T581X

Transmit data

09H, Common PIN

58

Receive data

CSB = 0 if successful, appropriate error code otherwise

Output length = 2 if successful, 0 otherwise

Output data = audit trail size if successful, 0 otherwise

Notes:

If the host supplied Common PIN is correct and the secure module has not previously been locked, the command will succeed. When the secure module is locked it will not accept any new groups or objects. This implies that all groups are automatically locked. The RAM not used by the system or by groups will be used for an audit trail. There is no audit trail until the secure module has successfully been locked!

An audit trail record is six bytes long and has the following structure:

Group ID | Object ID | Date/Time stamp.

Once an audit trail has been established, a record of the form shown above will be stored in the first available size byte location every time a transaction script is executed. Note that since the secure module must be locked before the audit trail begins, neither the group ID nor any object ID is subject to change. This will always allow an application processing the audit trail to uniquely identify the transaction script that was executed. Once the audit trail has consumed all of its available memory, it will store new transaction records over the oldest transaction records.

Possible error codes for the lock secure module command:

ERR_BAD_COMMON_PIN (Supplied common PIN was incorrect)

ERR_MIAC_LOCKED (Secure module was already locked)

Lock Group (0AH)

Transmit data

0AH, Group ID, Group PIN

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = 0

Output data = 0

Notes:

If the group PIN provided is correct the secure module BIOS will not allow further object creation within the specified group. Since groups are completely self-contained entities they may be deleted by executing the Delete Group command (described below).

Possible error return codes for the lock group command:

61

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_GROUP_LOCKED	(The group has already been locked)
ERR_MIAC_LOCKED	(The secure module has been locked)
ERR_BAD_GROUP_ID	(Specified group does not exist)

Invoke Transaction Script (0BH)

Transmit data

0BH, Group ID, Group PIN, Object ID

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = 1 if successful, 0 otherwise

Output data = estimated completion time

62

Notes:

The time estimate returned by the secure module is in sixteenths of a second. If an error code was returned in the CSB, the time estimate will be 0.

Possible error return codes for the execution transaction script command:

T630X

ERR_BAD_GROUP_PIN (Incorrect group PIN)

ERR_BAD_GROUP_ID (Specified group does not exist)

ERR_BAD_OBJECT_ID (Script object did not exist in group)

<u>Read Object (0CH)</u>

Transmit data

0CH, Group ID, Group PIN, Object ID

T631X

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = object length if successful, 0 otherwise

Output data = object data if successful, 0 otherwise

Notes:

If the Group ID, Group PIN and Object ID were correct, the secure module checks the attribute byte of the specified object. If the object has not been privatized the secure module will transmit the object data to the host. If the Group PIN was invalid or the object has been privatized the secure module will return a 0 in the output length, and data fields of the return packet.

Possible error codes for the read object command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)

ERR_BAD_OBJECT_ID (Object did not exist in group)

ERR_OBJECT_PRIVATIZED (Object has been privatized)

Write Object (0DH)

Transmit data

0DH, Group ID, Group PIN, Object ID, Object size, Object Data

Receive data

CSB = 0 if successful, appropriate error code otherwise

Output length = 0

Output data = 0

Notes:

If the Group ID, Group PIN and Object ID were correct, the secure module checks the attribute byte of the specified object. If the object has not been locked or privatized the secure module will clear the objects previous size and data and replace it with the new object data. Note that the object type and attribute byte are not affected.

Possible error codes for the write object command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_ID	(Object did not exist in group)
ERR_BAD_OBJECT_SIZE	(Illegal object size specified)
ERR_OBJECT_LOCKED	(Object has been locked)
ERR_OBJECT_PRIVATIZED	(Object has been privatized)

Read Group Name (0EH)

Transmit data

0EH, Group ID

Receive data

CSB = 0

Output Length = length of group name

Output data = group name

Notes:

The group name length is a maximum of 16 bytes. All byte values are legal in a group name.

Delete Group (0FH)

Transmit data

0FH, Group ID, Group PIN

Receive data

CSB = 0 if successful, appropriate error code otherwise

Output length = 0

Output data = 0

Notes:

If the group PIN and group ID are correct the secure module will delete the specified group. Deleting a group causes the automatic destruction of all objects within the group. If the secure module has been locked the Delete Group command will fail.

Possible error codes for the delete group command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_MIAC_LOCKED	(Secure module has been locked)

T6808

68

Get Command Status Info (10H)

Transmit data

10H

T690X

Receive data

CSB = 0

Output length = 6

Output data = secure module status structure (see below)

Notes:

This operation requires no PIN and never fails. The status structure is defined as follows:

Last command executed	(1 byte)
Last command status	(1 byte)
Time command received	(4 bytes)

T691X

Get Secure module Configuration Info (11H)

Transmit data

11H

Receive data

CSB = 0

Output length = 4

Output data = secure module configuration structure

Notes:

This operation requires no PIN and never fails. The configuration structure is defined as follows:

Number of groups	(1 byte)
Flag byte (see below)	(1 byte)
Audit trail size/Free RAM	(2 bytes)

The flag byte is the bitwise-or of any of the following values:

00000001b (Secure module is locked)

00000010b (Common PIN required for access)

Read Audit Trail Info (12H)

Transmit data

12H, Common PIN

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = audit trail structure size (5) if successful, 0 otherwise

Output data = audit trail info structure if successful, 0 otherwise

Notes:

If the transmitted Common PIN is valid and the secure module has been locked, it returns audit trail configuration information as follows:

Number of used transaction records	(2 bytes)
Number of free transaction records	(2 bytes)
A boolean specifying whether or not the audit trail rolled since previous read command	(1 byte)

Possible error codes for the read audit trail info command:

ERR_BAD_COMMON_PIN (Common PIN was incorrect)

ERR_MIAC_NOT_LOCKED (Secure module is not locked)

Read Audit Trail (13H)

Transmit data

13H, Common PIN

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = # of new records * 6 if successful, 0 otherwise

Output data = new audit trail records

Notes:

If the transmitted common PIN is valid and the secure module has been locked, it will transfer all new transaction records to the host.

Possible error codes for the read audit trail command:

ERR_BAD_COMMON_PIN (Common PIN was incorrect)

ERR_MIAC_NOT_LOCKED secure module is not locked

Read Group Audit Trail (14H)

Transmit data

14H, Group ID, Group PIN

Receive data

CSB = 0 if command successful, appropriate error code otherwise

Output length = # or records for group * 6 if successful, 0 otherwise

Output data = audit trail records for group

Notes:

This command is identical to the read audit trail command, except that only records involving the group ID specified in the transmit data are returned to the host. This allows transaction groups to record track their own activities without seeing other groups records.

Possible error codes for the read group audit trail command:

ERR_BAD_GROUP_ID (Group ID does not exist)

ERR_BAD_GROUP_PIN (Common PIN was incorrect)

T7504

ERR_MIAC_NOT_LOCKED (The secure module is not locked)

Read Real Time Clock (15H)

Transmit data

15H, Common PIN

Receive data

T7514

CSB = 0 if the common PIN matches and ERR_BAD_COMMON_PIN otherwise

Output length = 4

Output data = 4 most significant bytes of the real time clock

Notes:

This value is not adjusted with a clock offset. This command is normally used by a service provider to compute a clock offset during transaction group creation.

Read Real Time Clock Adjusted (16H)

Transmit data

16H, Group ID, Group PIN, ID of offset object

Receive data

CSB = 0 if successful, appropriate error code otherwise

Output length = 4 if successful, 0 otherwise

Output data = Real time clock + clock offset ID

Notes:

This command succeeds if the group ID and group PIN are valid, and the object ID is the ID of a clock offset. The secure module adds the clock offset to the current value of the 4 most significant bytes of the RTC and returns that value in the output data field. Note that a transaction script may be written to perform the same task and put the result in the output data object.

Possible error codes for the real time clock adjusted command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_OBJECT_TYPE	(Object ID is not a clock offset)

T770Y

Get Random Data (17H)

T771Y

Transmit data

17H, Length (L)

Receive data

CSB = 0 if successful, appropriate error code otherwise

Output length = L if successful, 0 otherwise

Output data = L bytes of random data if successful

Notes:

This command provides a good source of cryptographically useful random numbers.

Possible error codes for the get random data command are:

T780X

ERR_BAD_SIZE (Requested number of bytes > 128)

Get Firmware Version ID (18H)

T781Y

Transmit data

18H

Receive data

CSB = 0

78

Output length = Length of firmware version ID string

Output data = Firmware version ID string

Notes:

This command returns the firmware version ID as a Pascal type string (length + data).

Get Free RAM (19H)

Transmit data

19H

Receive data

CSB = 0

Output length = 2

Output data = 2 byte value containing the amount of free RAM

Notes:

If the secure module has been locked the output data bytes will both be 0 indicating that all memory not used by transaction groups has been reserved for the audit trail.

Change Group Name (1AH)

Transmit data

1AH, Group ID, Group PIN, New Group name

Receive data

CSB = 0 if successful or an appropriate error code otherwise

Output length = 0

Output data = 0

Notes:

If the group ID specified exists in the secure module and the PIN supplied is correct, the transaction group name is replaced by the new group name supplied by

80

the host. If a group ID of 0 is supplied the PIN transmitted must be the common PIN. If it is correct, the secure module name is replaced by the new name supplied by the host.

Possible error codes for the change group name command:

ERR_BAD_GROUP_PIN	(Incorrect group PIN)
ERR_BAD_GROUP_ID	(Specified group does not exist)
ERR_BAD_NAME_LENGTH	(New group name > 16 bytes)

ERROR CODE DEFINITIONS

ERR_BAD_COMMAND (80H)

This error code occurs when the secure module firmware does not recognize the command just transmitted by the host.

ERR_BAD_COMMON_PIN (81H)

This error code will be returned when a command requires a common PIN and the PIN supplied does not match the secure module's common PIN. Initially the common PIN is set to 0.

ERR_BAD_GROUP_PIN (82H)

Transaction groups may have their own PIN, FIGURE 6. If this PIN has been set (by a set group PIN command) it must be supplied to access any of the objects within the group. If the Group PIN supplied does not match the actual group PIN, the secure module will return the ERR_BAD_GROUP_PIN error code.

ERR_BAD_PIN_LENGTH (83H)

There are 2 commands which can change PIN values. The set group PIN and the set common PIN commands. Both of these require the new PIN as well as the old PIN. The ERR_BAD_PIN_LENGTH error code will be returned if the old PIN supplied was correct, but the new PIN was greater than 8 characters in length.

ERR_BAD_OPTION_BYTE (84H)

The option byte only applies to the common PIN. When the set common PIN command is executed the last byte the host supplies is the option byte (described in command section). If this byte is unrecognizable to the secure module, it will return the ERR_BAD_OPTION_BYTE error code.

ERR_BAD_NAME_LENGTH (85H)

When the create transaction group command is executed, one of the data structures supplied by the host is the group's name. The group name may not exceed 16 characters in length. If the name supplied is longer than 16 characters, the ERR_BAD_NAME_LENGTH error code is returned.

ERR_INSUFFICIENT_RAM (86H)

The create transaction group and create object commands return this error code when there is not enough heap available in the secure module.

ERR_MIAC_LOCKED (87H)

When the secure module has been locked, no groups or objects can be created or destroyed. Any attempts to create or delete objects will generate an ERR_MIAC_LOCKED error code.

84

ERR_MIAC_NOT_LOCKED (88H)

If the secure module has not been locked there is no audit trail. If one of the audit trail commands is executed this error code will be returned.

ERR_GROUP_LOCKED (89H)

Once a transaction group has been locked object creation within that group is not possible. Also the objects attributes and types are frozen. Any attempt to create objects or modify their attribute or type bytes will generate an ERR_GROUP_LOCKED error code.

ERR_BAD_OBJECT_TYPE (8AH)

When the host sends a create object command to the secure module, one of the parameters it supplies is an object type (see command section). If the object type is not recognized by the firmware it will return an ERR_BAD_OBJECT_TYPE error code.

ERR_BAD_OBJECT_ATTR (8BH)

When the host sends a create object command to the secure module, one of the parameters it supplies is an object attribute byte (see command section). If the object attribute byte is not recognized by the firmware it will return an ERR_BAD_OBJECT_ATTR error code.

ERR_BAD_SIZE (8CH)

An ERR_BAD_SIZE error code is normally generated when creating or writing an object. It will only occur when the object data supplied by the host has an invalid length.

ERR_BAD_GROUP_ID (8DH)

All commands that operate at the transaction group level require the group ID to be supplied in the command packet. If the group ID specified does not exist in the

86

secure module it will generate an ERR_BAD_GROUP_ID error code.

ERR_BAD_OBJECT_ID (8EH)

All commands that operate at the object level require the object ID to be supplied in the command packet. If the object ID specified does not exist within the specific transaction group (also specified in the command packet) the secure module will generate an ERR_BAD_OBJECT_ID error code.

ERR_INSUFFICIENT_FUNDS (8FH)

If a script object that executes financial transactions is invoked and the value of the money register is less than the withdrawal amount requested an ERR_INSUFFICIENT_FUNDS error code will be returned.

ERR_OBJECT_LOCKED (90H)

Locked objects are read only. If a write object command is attempted and it specifies the object ID of a locked object the secure module will return an ERR_OBJECT_LOCKED error code.

ERR_OBJECT_PRIVATE (91H)

Private objects are not directly readable or writable. If a read object command or a write object command is attempted, and it specifies the object ID of a private object, the secure module will return an ERR_OBJECT_PRIVATE error code.

ERR_OBJECT_DESTRUCTED (92H)

If an object is destructible and the transaction group's destructor is active the object may not be used by a script. If a script is invoked which uses an object which has been destructed, an ERR_OBJECT_DESTRUCTED error code will be returned by the secure module.

The exemplary embodiment of the present invention is preferably placed within a durable stainless steel, token-like can. It is understood that an exemplary secure module can be placed in virtually any articulatable item. Examples of articulatable items include credit cards, rings, watches, wallets, purses, necklaces, jewelry, ID badges, pens, clipboards, etc.

The secure module 108 preferably is a single chip "trusted computer". By the word "trusted" it is meant that the computer is extremely secure from tampering by unwarranted means. The secure module incorporates a numeric coprocessor optimized for math intensive encryption. The BIOS is preferably immune to alteration and specifically designed for very secure transactions.

Each secure module can have a random "seed" generator with the ability to create a private/public key set. The private key never leaves the secure module and is only known by the secure module. Furthermore, discovery of the private key is prevented by active self-destruction upon wrongful entry into the secure module. The secure module can be bound to the user by a personal identification number (PIN).

When transactions are performed by the secure module 108 certificates of authentication are created by either or both the secure module and a system the secure module communicates with. The certificate can contain a variety of information. In particular, the certificate may contain:

1) who is the secure module user via a unique registration number and a certified public key.

2) when the transaction took place via a true-time stamping of the transaction.

3) where the transaction took place via a registered secure module interface site identification.

4) security information via uniquely serialized transactions and digital sign on message digests.

5) secure module status indicated as valid, lost, or expired.

Although a preferred embodiment of the method and apparatus of the present invention has been illustrated in the accompanying Drawings and described in the foregoing Detailed Description, it will be understood that the invention is not limited to the embodiment disclosed, but is capable of numerous rearrangements, modifications and substitutions without departing from the spirit of the invention as set forth and defined by the following claims.

WHAT IS CLAIMED IS:

~~1. A system for communicating data securely, comprising:~~

~~a first module for containing a first data;~~

~~an electronic system comprising a secure module, said electronic system adapted to be able to communicate with said first module.~~

~~2. The system of claim 1, wherein said first module is a portable module.~~

~~3. The system of claim 1, wherein said first module comprises a memory circuit for storing said first data.~~

~~4. The system of claim 3, wherein said memory circuit contains an encrypted data.~~

~~5. The system of claim 1, wherein said first module comprises an identification means for identifying said first module to said electronic system.~~

~~6. The system of claim 1, wherein said first module comprises a counter for counting a number of transactions said first module performed with said electronic system.~~

~~7. The system of claim 6, wherein said number of transactions represent the number of times a memory data is changed in said module.~~

~~8. The system of claim 1, wherein said electronic system is adapted to communicate with said first module via a single conductive contact.~~

~~9. The system of claim 1, wherein said electronic system is adapted to communicate with said first module via a one-wire bus.~~

~~10. The system of claim 1, wherein said first module is another secure module.~~

~~11. A system of claim 1, wherein said secure module is adapted to receive said first data.~~

~~12. The system of claim 1, wherein said secure module is adapted to receive said first data and create a second data that contains at least one information that was in said first data.~~

~~13. The system of claim 12, wherein said second data is encrypted.~~

~~14. The system of claim 1, wherein said secure module contains a substantially inaccessible private key in memory portion of said secure module.~~

~~15. The system of claim 1, wherein said electronic system is connected to at least one of a credit card reader, a cash accepter, a cash provider, an automatic teller machine and a communication line.~~

1 ~~16.~~ A method for electronically transferring units of exchange between a first module and a second module, comprising the steps of:

a. initiating communication between said first module and an electronic device;

b. passing a first value datum from said first module to said electronic device;

c. passing said first value datum from said electronic device to said second module;

d. performing a mathematical calculation on said first value datum thereby creating a second value datum;

e. passing said second value datum from said second module to said electronic device;

f. passing said second value datum from said electronic device to said first module;

g. storing said second value datum in said first module; and

h. discontinuing communication between said first module and said electronic device.

~~17~~ 2. The method of claim ~~16~~ 1, wherein said first value datum represents a monetary equivalent.

~~18~~ 3. The method of claim ~~16~~ 1, wherein said first value datum is encrypted.

~~19~~ 4. The method of claim ~~16~~ 1, wherein said second value datum is encrypted.

~~20~~ 5. The method of claim ~~18~~ 3, wherein the step of performing a mathematical calculation comprises the steps of:

m. decrypting said first value datum with a public key thereby creating a decrypted value;

n: performing at least one of an addition function and a subtraction function on said decrypted value thereby creating a value result; and

o. encrypting said value result with a private key thereby creating said second value datum.

B

6 ~~21~~. The method of claim ~~16~~ 1, wherein the step (b) of passing is performed over at least a single conductive contact.

08/978798

Patent Application
Docket #20661/429

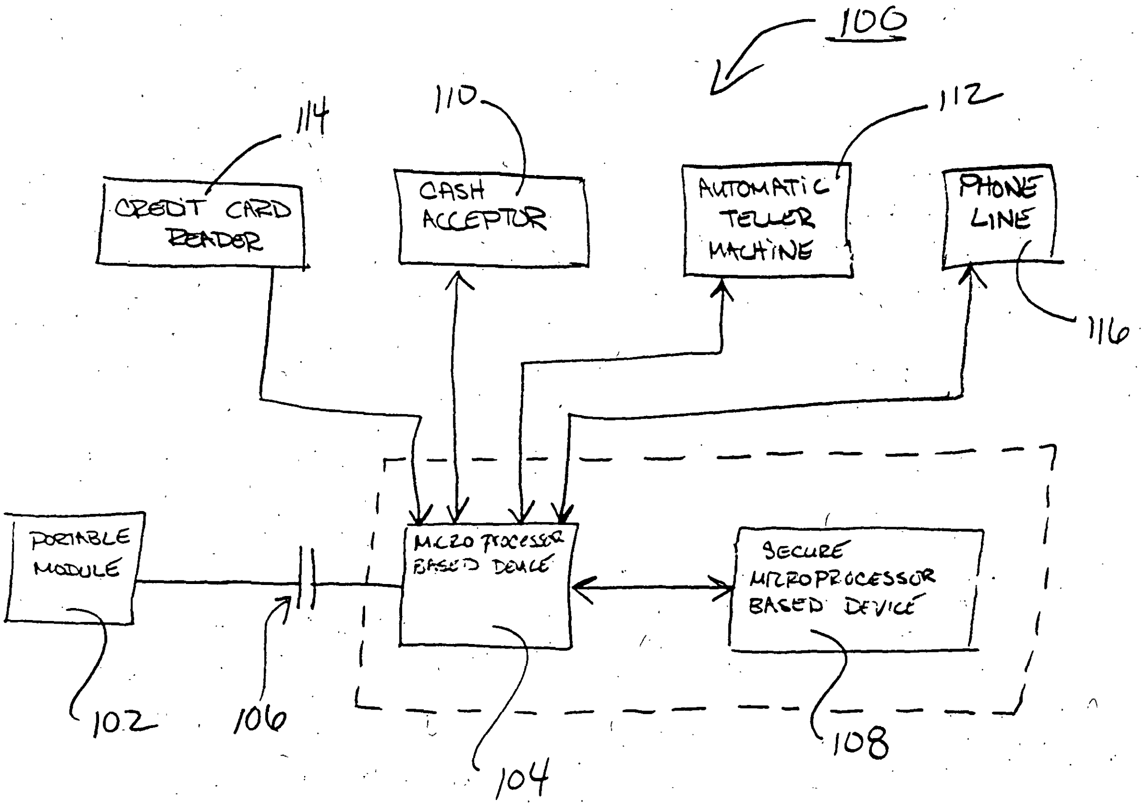
ABSTRACT OF THE DISCLOSURE

The present invention relates to system, apparatus and method for communicating valuable data from a portable module to another module via an electronic device. More specifically, the disclosed system, apparatus and method are useful for enabling a user to fill a portable module with a cash equivalent and to spend the cash equivalent at a variety of locations. The disclosed system incorporates an encryption/decryption method.

08/978798

20661-429

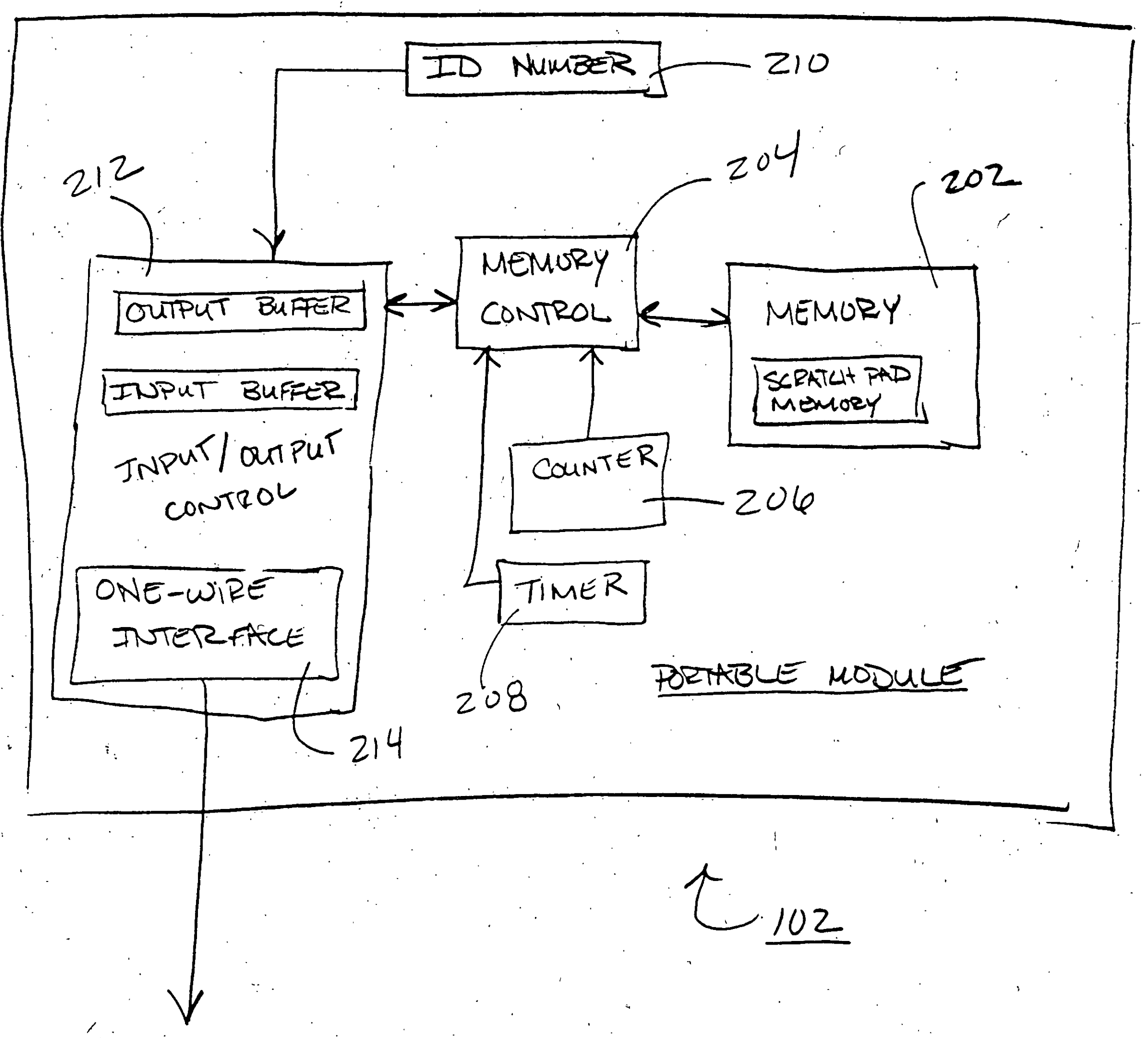
FIGURE 1



09/978798

20661-429

FIGURE 2



09/979798

20661-429

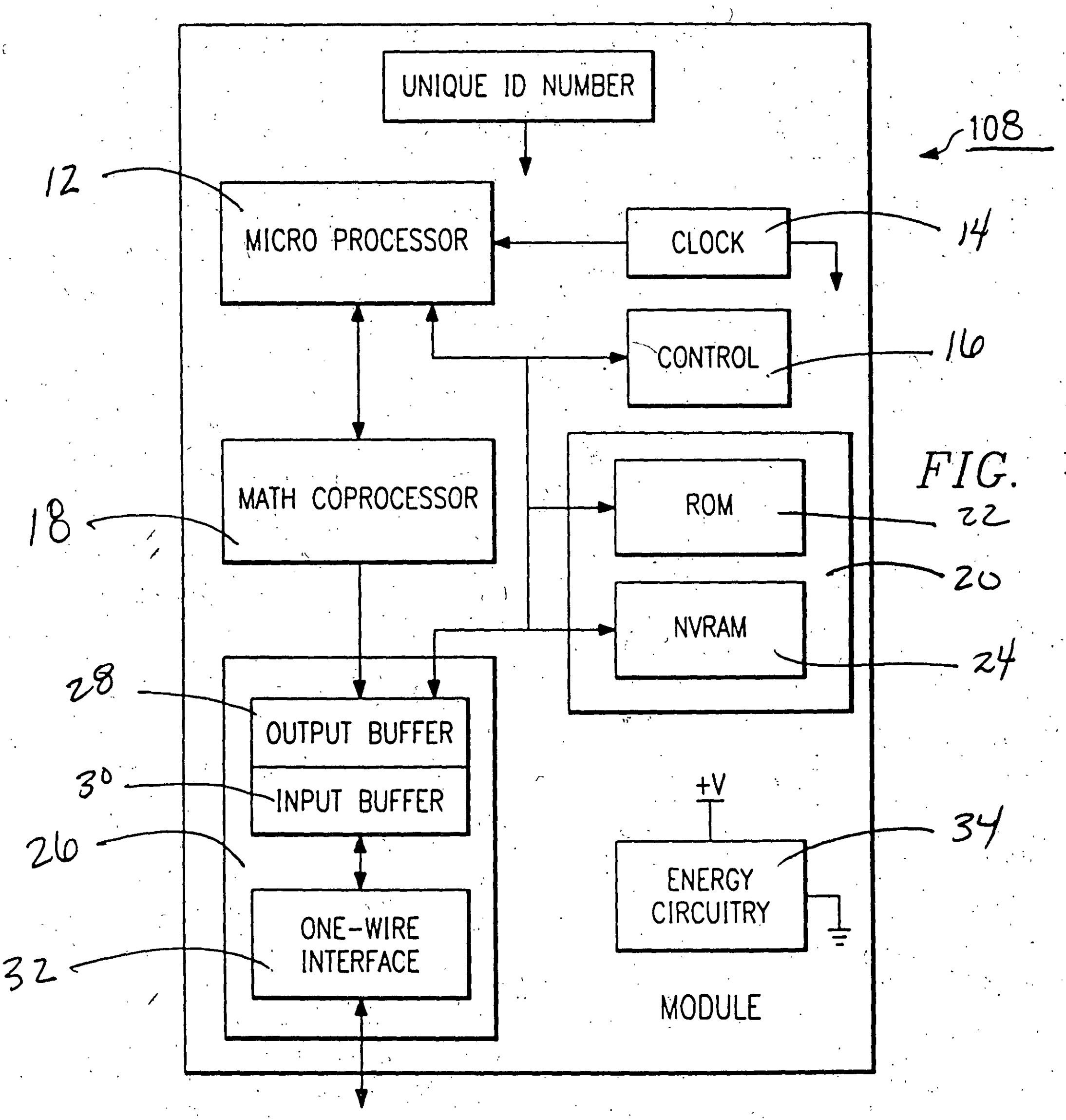
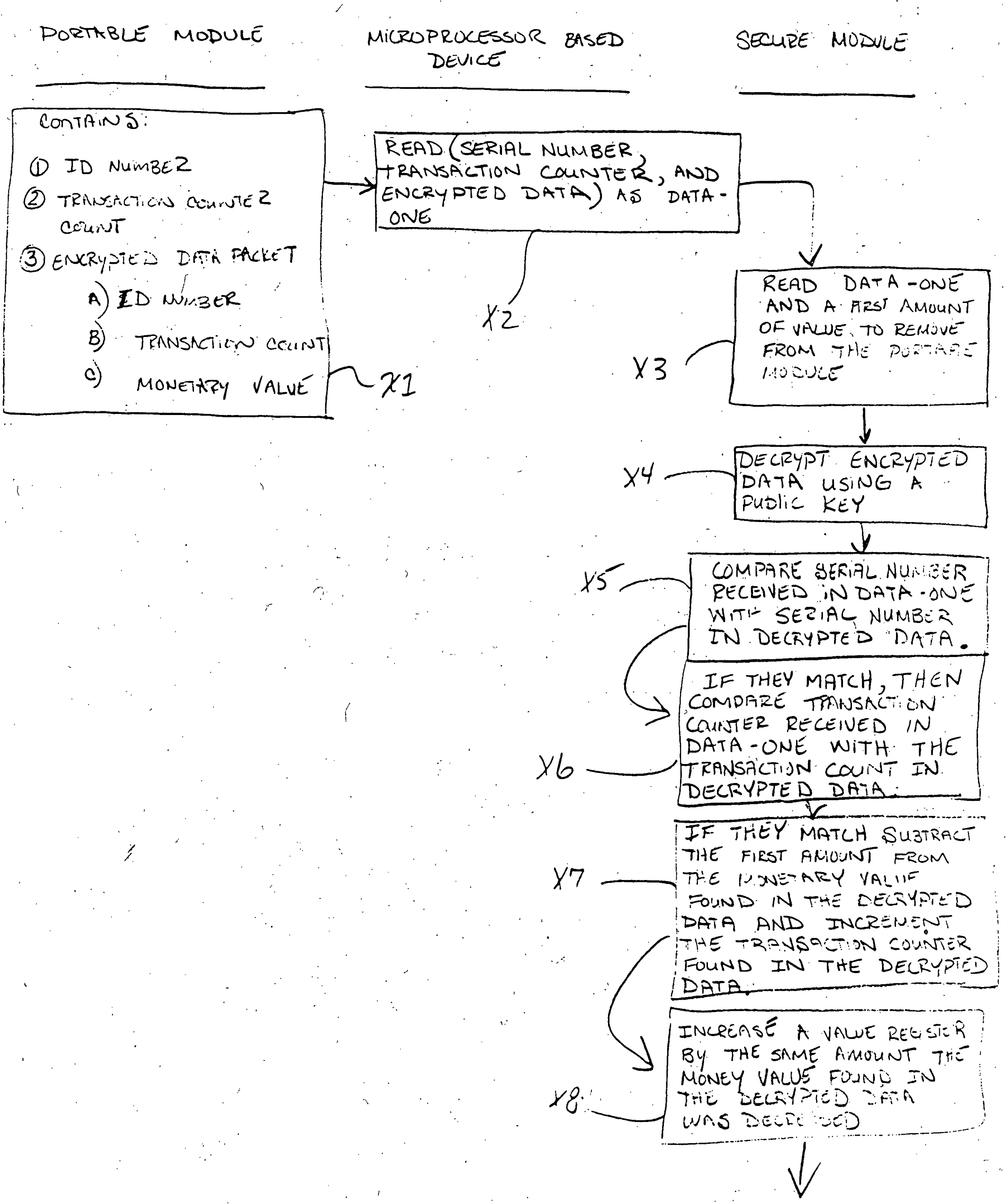


FIG. 3

09/978798

FIGURE 4

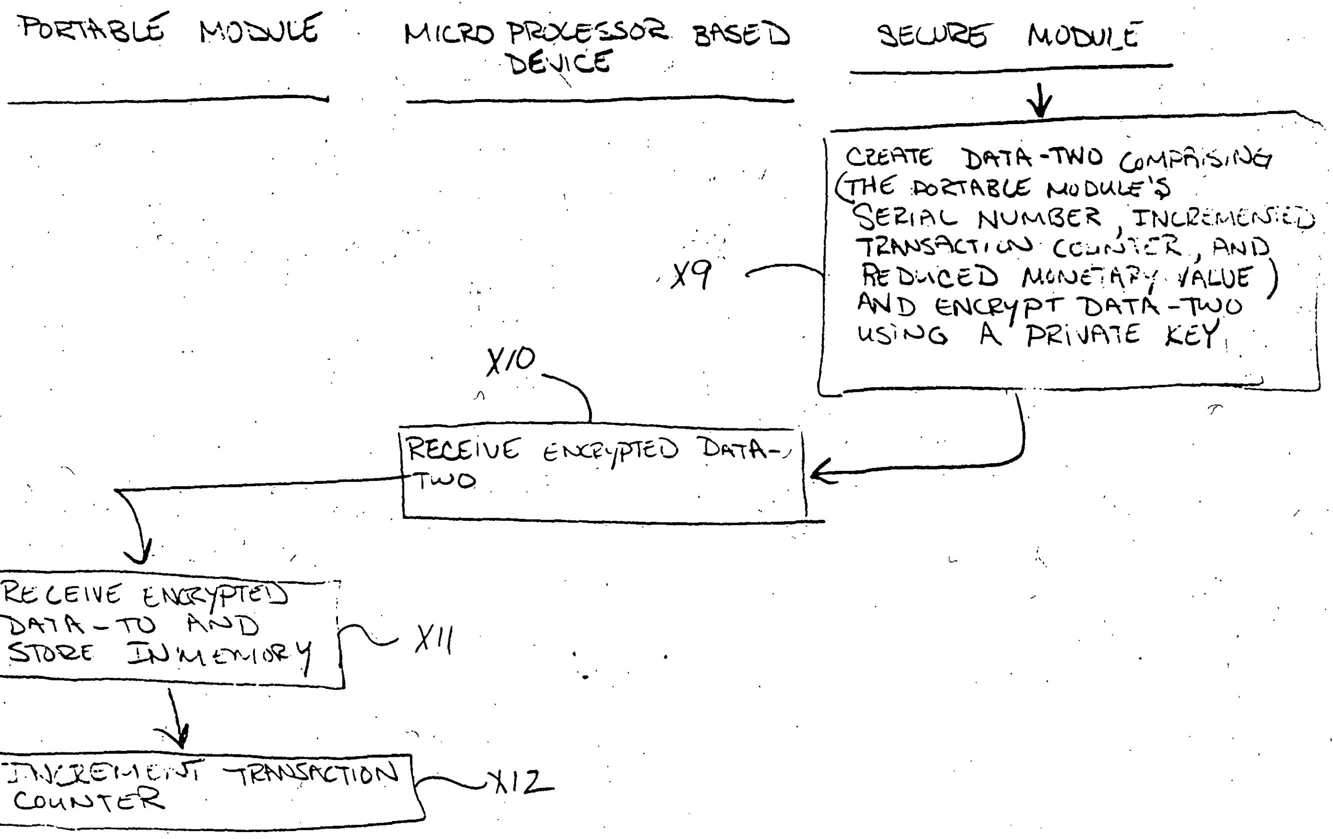
20661-429


PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CONTAINS:
① ID NUMBER
② TRANSACTION COUNTER COUNT
③ ENCRYPTED DATA PACKET
A) ID NUMBER
B) TRANSACTION COUNT
c) MONETARY VALUE
X1
READ (SERIAL NUMBER, TRANSACTION COUNTER, AND ENCRYPTED DATA) AS DATA-ONE
X2
READ DATA-ONE AND A FIRST AMOUNT OF VALUE TO REMOVE FROM THE PORTABLE MODULE
X3
DECRYPT ENCRYPTED DATA USING A PUBLIC KEY
X4
COMPARE SERIAL NUMBER RECEIVED IN DATA-ONE WITH SERIAL NUMBER IN DECRYPTED DATA.
X5
IF THEY MATCH, THEN COMPARE TRANSACTION COUNTER RECEIVED IN DATA-ONE WITH THE TRANSACTION COUNT IN DECRYPTED DATA.
X6
IF THEY MATCH SUBTRACT THE FIRST AMOUNT FROM THE MONETARY VALUE FOUND IN THE DECRYPTED DATA AND INCREMENT THE TRANSACTION COUNTER FOUND IN THE DECRYPTED DATA.
X7
INCREASE A VALUE REGISTER BY THE SAME AMOUNT THE MONEY VALUE FOUND IN THE DECRYPTED DATA WAS DECREASED
X8

08/978798

FIGURE 4 CONTINUED

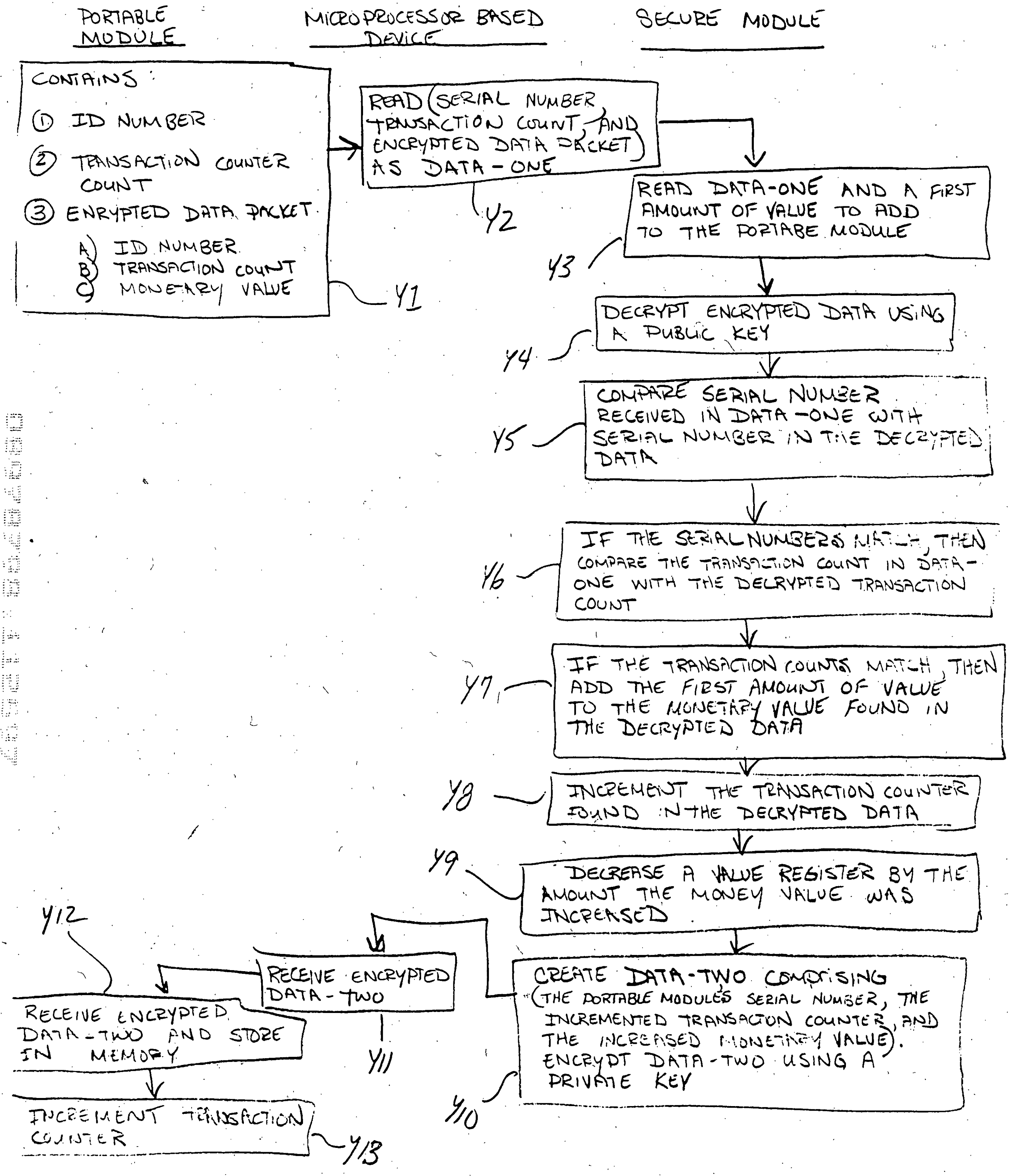
20661-429


PORTABLE MODULE
MICRO PROCESSOR BASED DEVICE
SECURE MODULE
CREATE DATA-TWO COMPRISING (THE PORTABLE MODULE'S SERIAL NUMBER, INCREMENTED TRANSACTION COUNTER, AND REDUCED MONETARY VALUE) AND ENCRYPT DATA-TWO USING A PRIVATE KEY
X9
X10
RECEIVE ENCRYPTED DATA-TWO
RECEIVE ENCRYPTED DATA-TO AND STORE IN MEMORY
X11
INCREMENT TRANSACTION COUNTER
X12

09/978798

20661-429

FIGURE 5



CONTAINS:
① ID NUMBER
② TRANSACTION COUNTER COUNT
③ ENRYPTED DATA PACKET
A) ID NUMBER
B) TRANSACTION COUNT
C) MONETARY VALUE

Y2 READ (SERIAL NUMBER, TRANSACTION COUNT, AND ENCRYPTED DATA PACKET) AS DATA-ONE

Y3 READ DATA-ONE AND A FIRST AMOUNT OF VALUE TO ADD TO THE PORTABE MODULE

Y4 DECRYPT ENCRYPTED DATA USING A PUBLIC KEY

Y5 COMPARE SERIAL NUMBER RECEIVED IN DATA-ONE WITH SERIAL NUMBER IN THE DECRYPTED DATA

Y6 IF THE SERIAL NUMBERS MATCH, THEN COMPARE THE TRANSACTION COUNT IN DATA-ONE WITH THE DECRYPTED TRANSACTION COUNT

Y7 IF THE TRANSACTION COUNTS MATCH, THEN ADD THE FIRST AMOUNT OF VALUE TO THE MONETARY VALUE FOUND IN THE DECRYPTED DATA

Y8 INCREMENT THE TRANSACTION COUNTER FOUND IN THE DECRYPTED DATA

Y9 DECREASE A VALUE REGISTER BY THE AMOUNT THE MONEY VALUE WAS INCREASED

Y10 CREATE DATA-TWO COMPRISING (THE PORTABLE MODULES SERIAL NUMBER, THE INCREMENTED TRANSACTON COUNTER, AND THE INCREASED MONETARY VALUE). ENCRYPT DATA-TWO USING A PRIVATE KEY

Y11 RECEIVE ENCRYPTED DATA-TWO

Y12 RECEIVE ENCRYPTED DATA-TWO AND STORE IN MEMORY

Y13 INCREMENT TRANSACTION COUNTER

09/978798

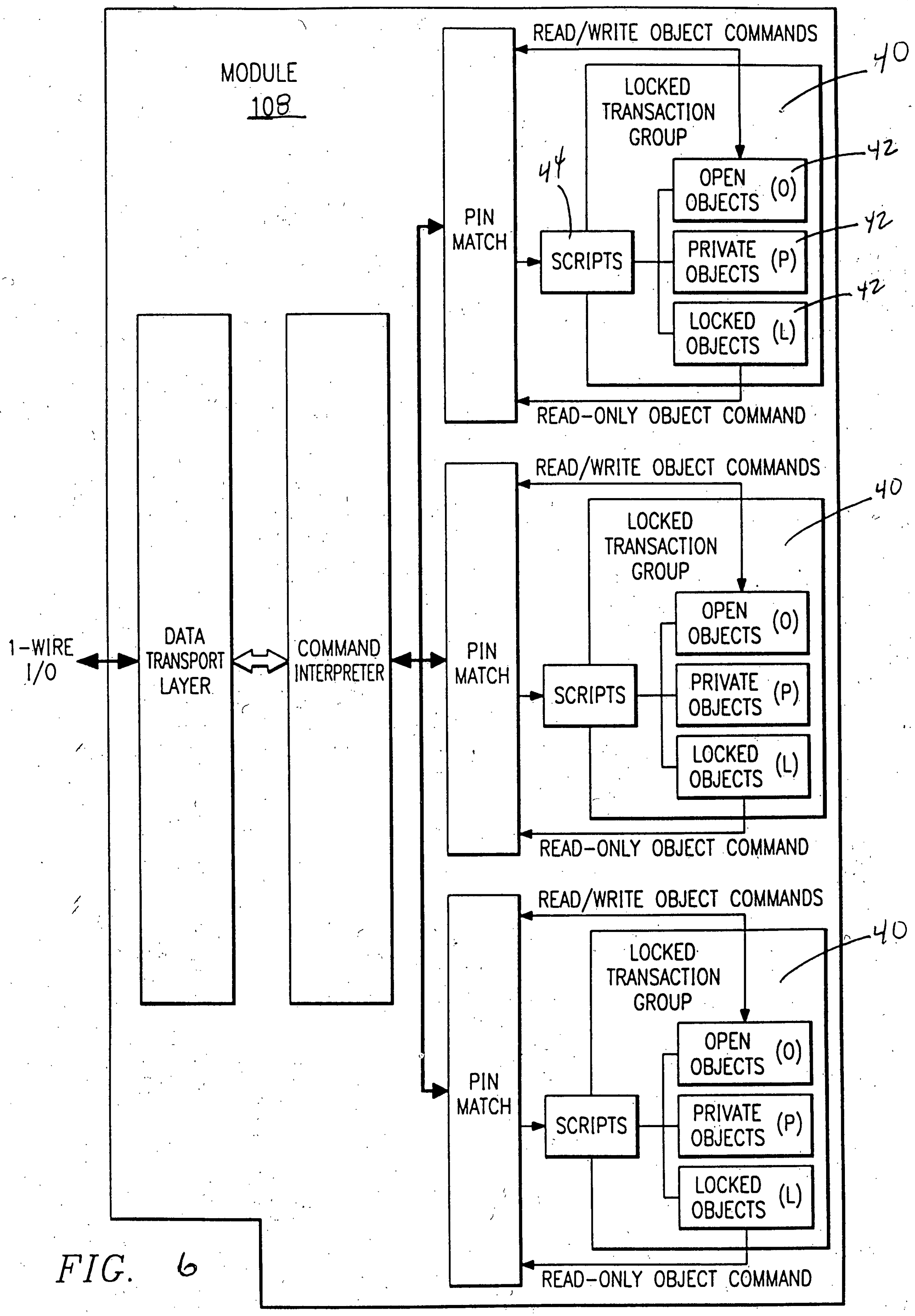
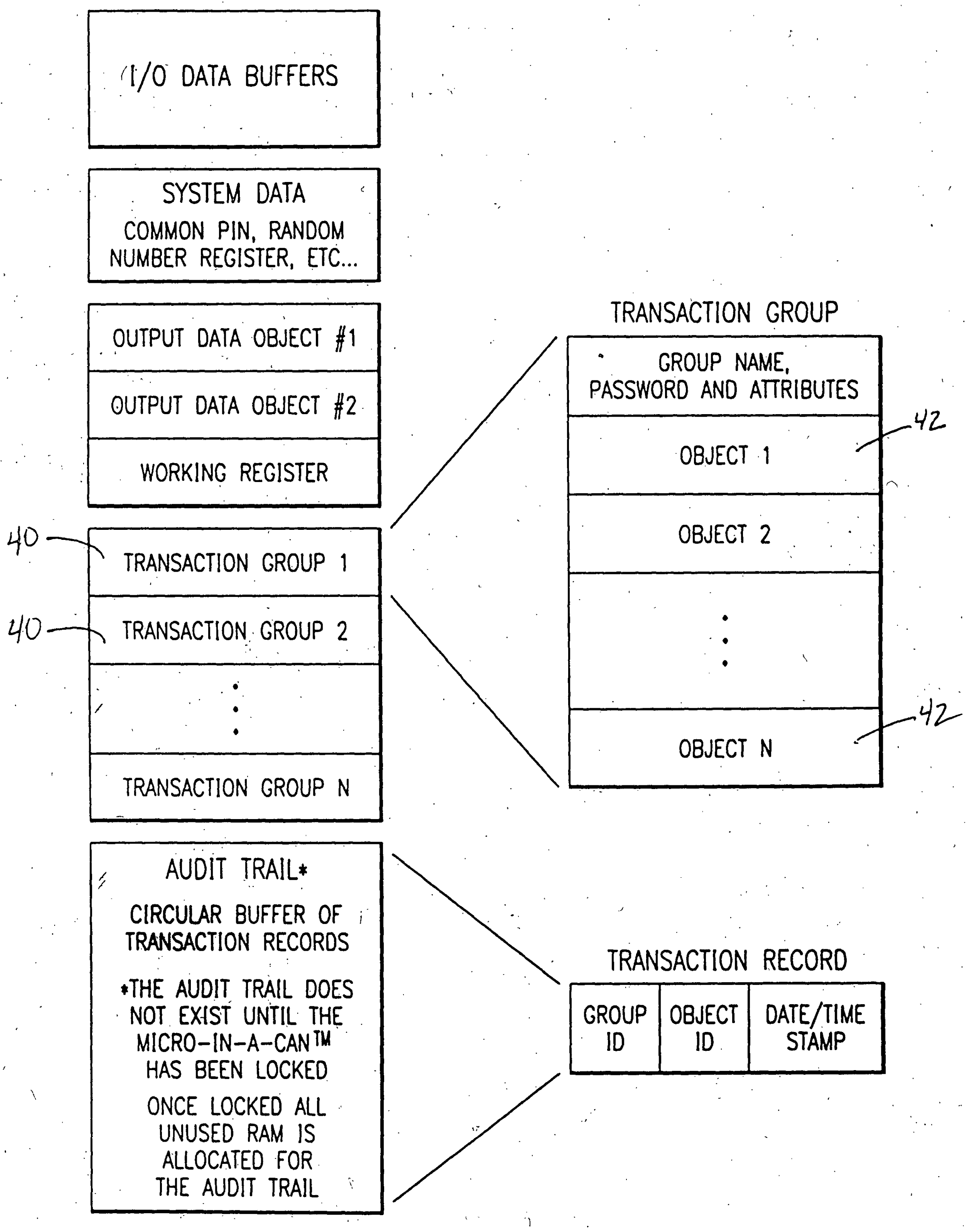


FIG. 6

20661 29

08/978798

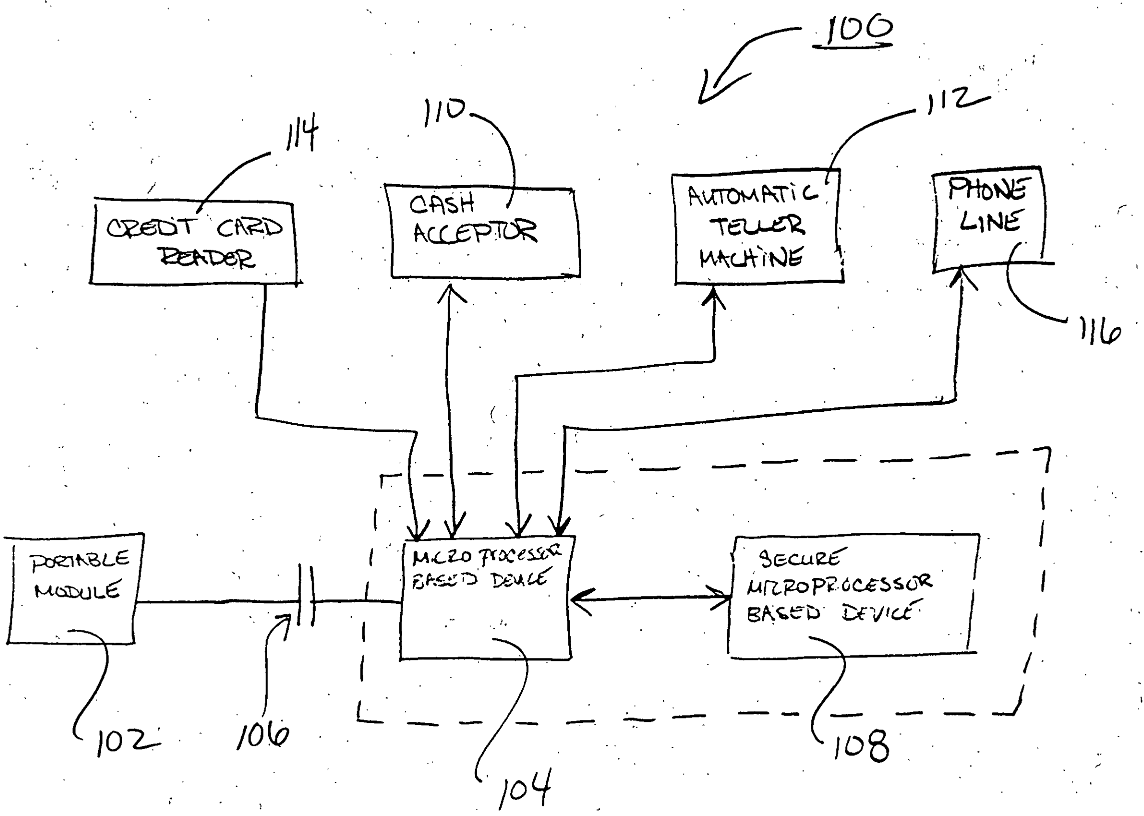
FIG. 7



08/979798

20661-429

FIGURE 1



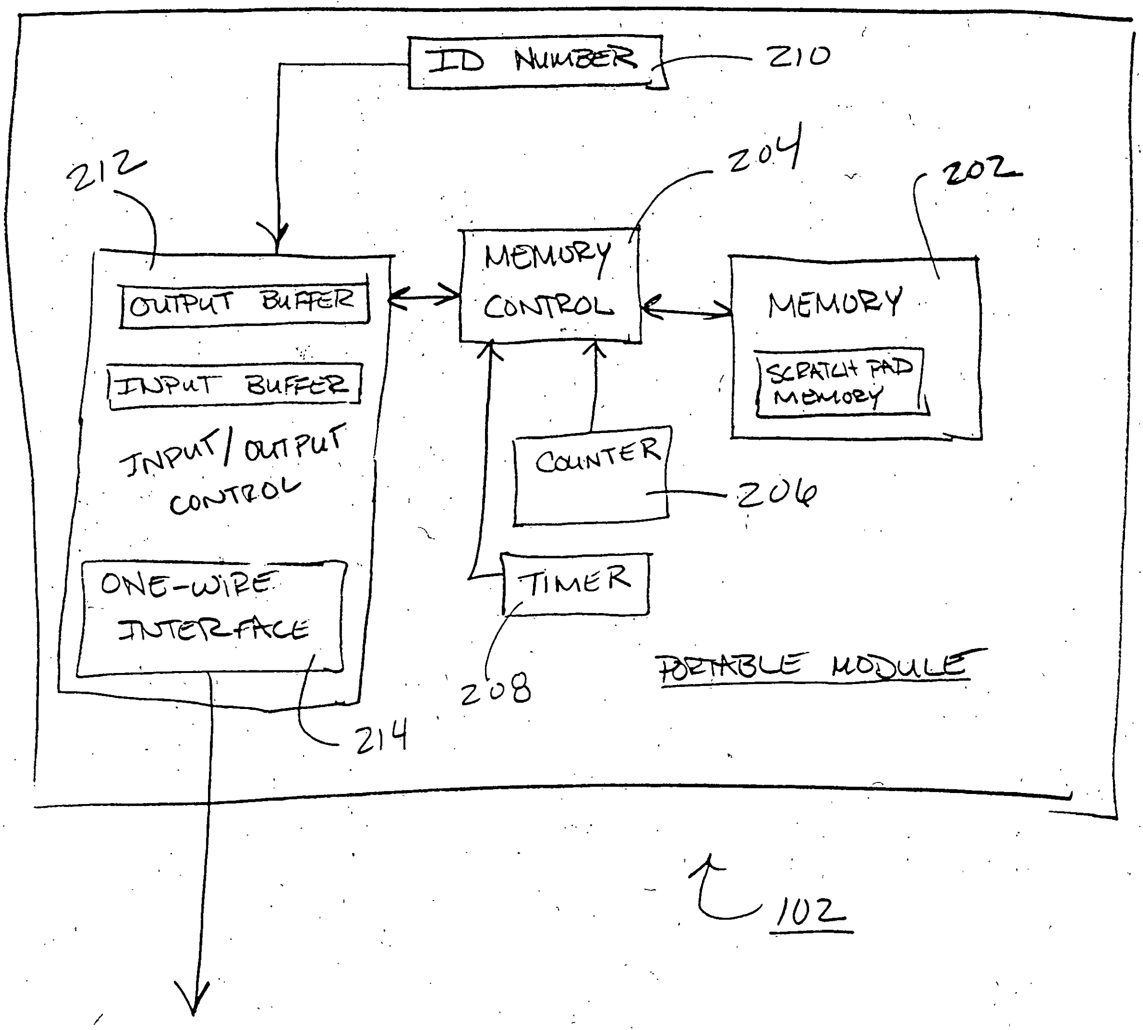
7

08/978798

20661-429

[redacted]

FIGURE 2



08/978798

20661-429

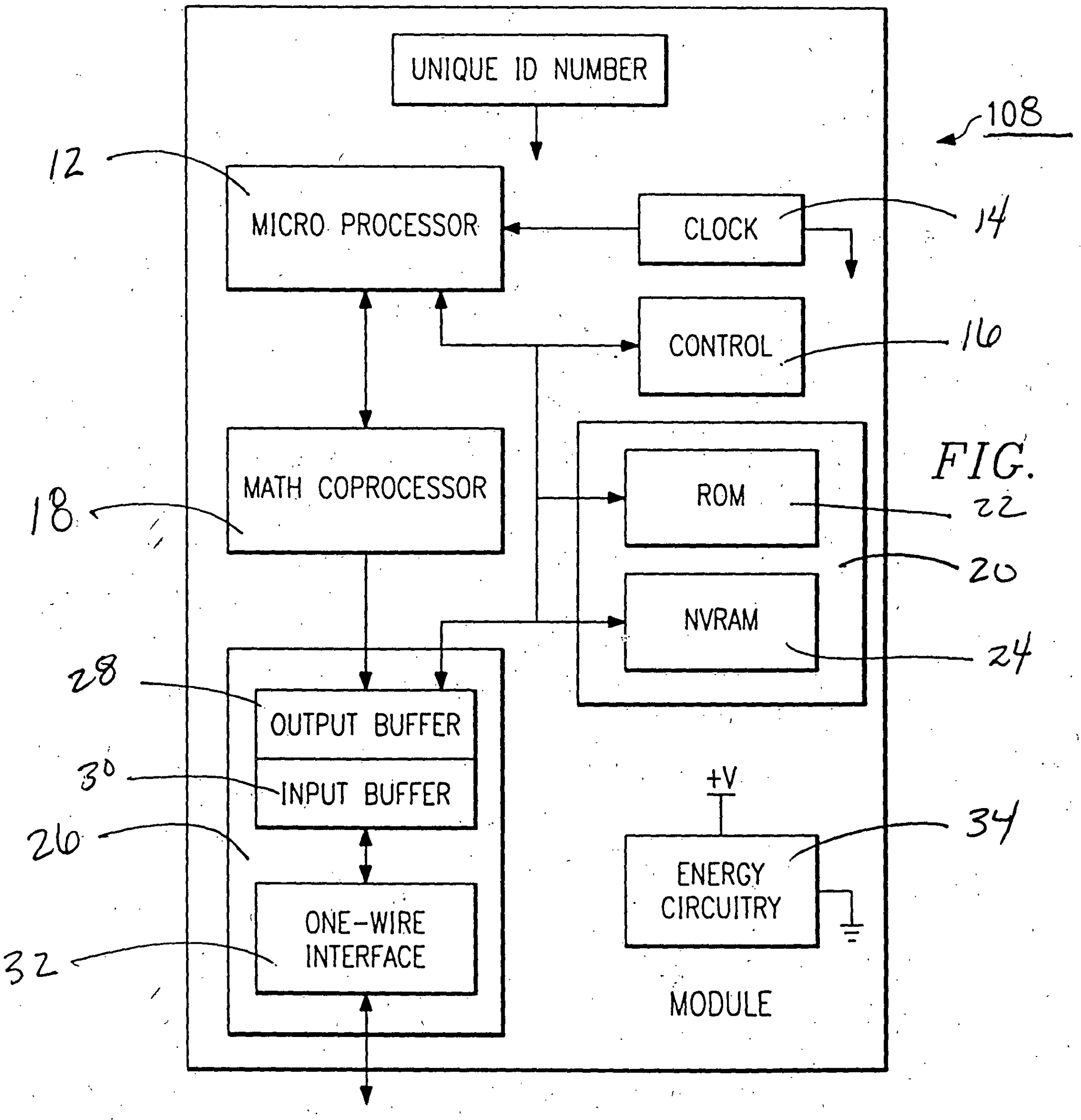
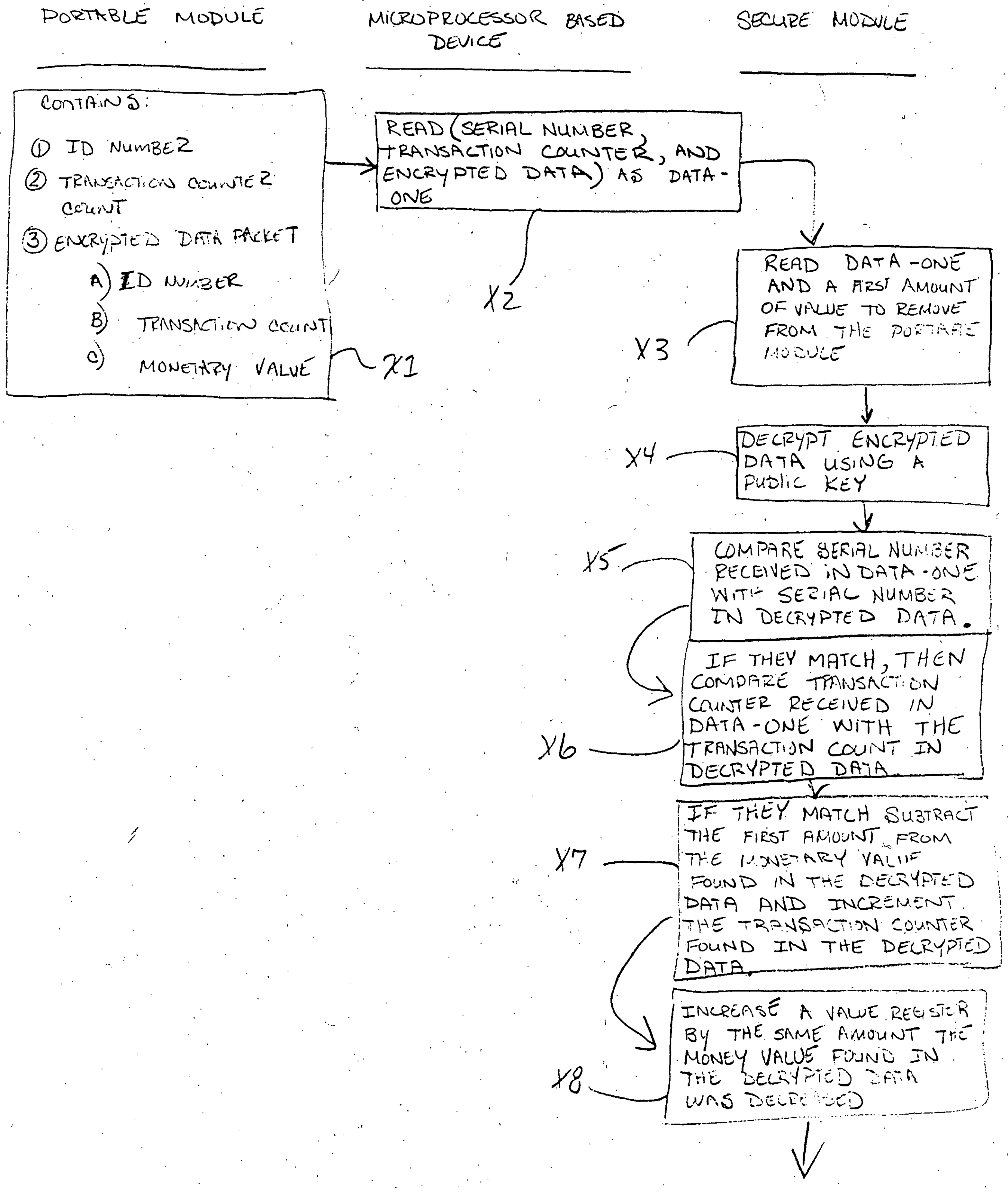


FIG. 3

08/978798

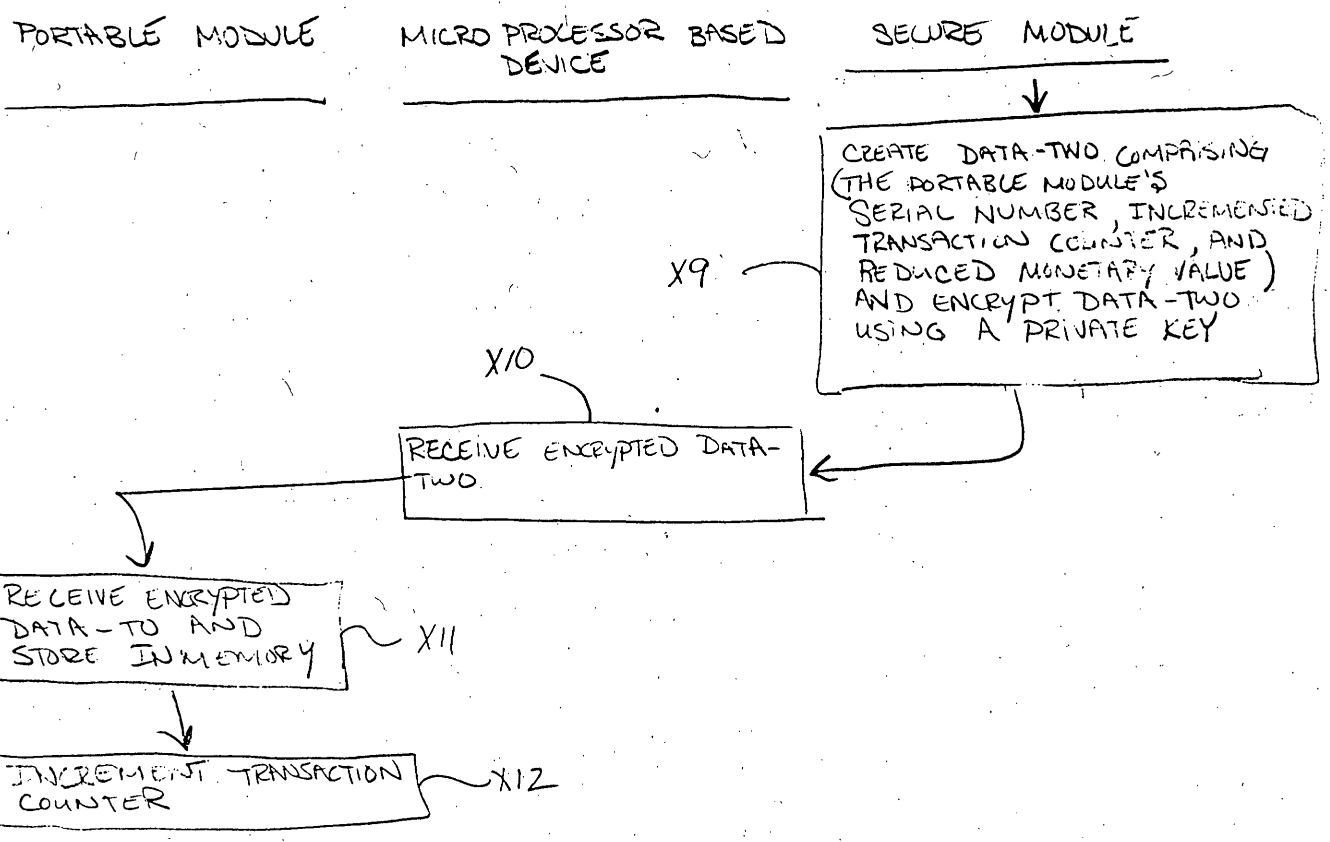
FIGURE 4

20661-429


PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CONTAINS:
① ID NUMBER
② TRANSACTION COUNTER COUNT
③ ENCRYPTED DATA PACKET
A) ID NUMBER
B) TRANSACTION COUNT
C) MONETARY VALUE
X1
READ (SERIAL NUMBER, TRANSACTION COUNTER, AND ENCRYPTED DATA) AS DATA-ONE
X2
READ DATA-ONE AND A FIRST AMOUNT OF VALUE TO REMOVE FROM THE PORTABLE MODULE
X3
DECRYPT ENCRYPTED DATA USING A PUBLIC KEY
X4
COMPARE SERIAL NUMBER RECEIVED IN DATA-ONE WITH SERIAL NUMBER IN DECRYPTED DATA.
X5
IF THEY MATCH, THEN COMPARE TRANSACTION COUNTER RECEIVED IN DATA-ONE WITH THE TRANSACTION COUNT IN DECRYPTED DATA.
X6
IF THEY MATCH SUBTRACT THE FIRST AMOUNT FROM THE MONETARY VALUE FOUND IN THE DECRYPTED DATA AND INCREMENT THE TRANSACTION COUNTER FOUND IN THE DECRYPTED DATA.
X7
INCREASE A VALUE REGISTER BY THE SAME AMOUNT THE MONEY VALUE FOUND IN THE DECRYPTED DATA WAS DECREASED
X8

08/978798

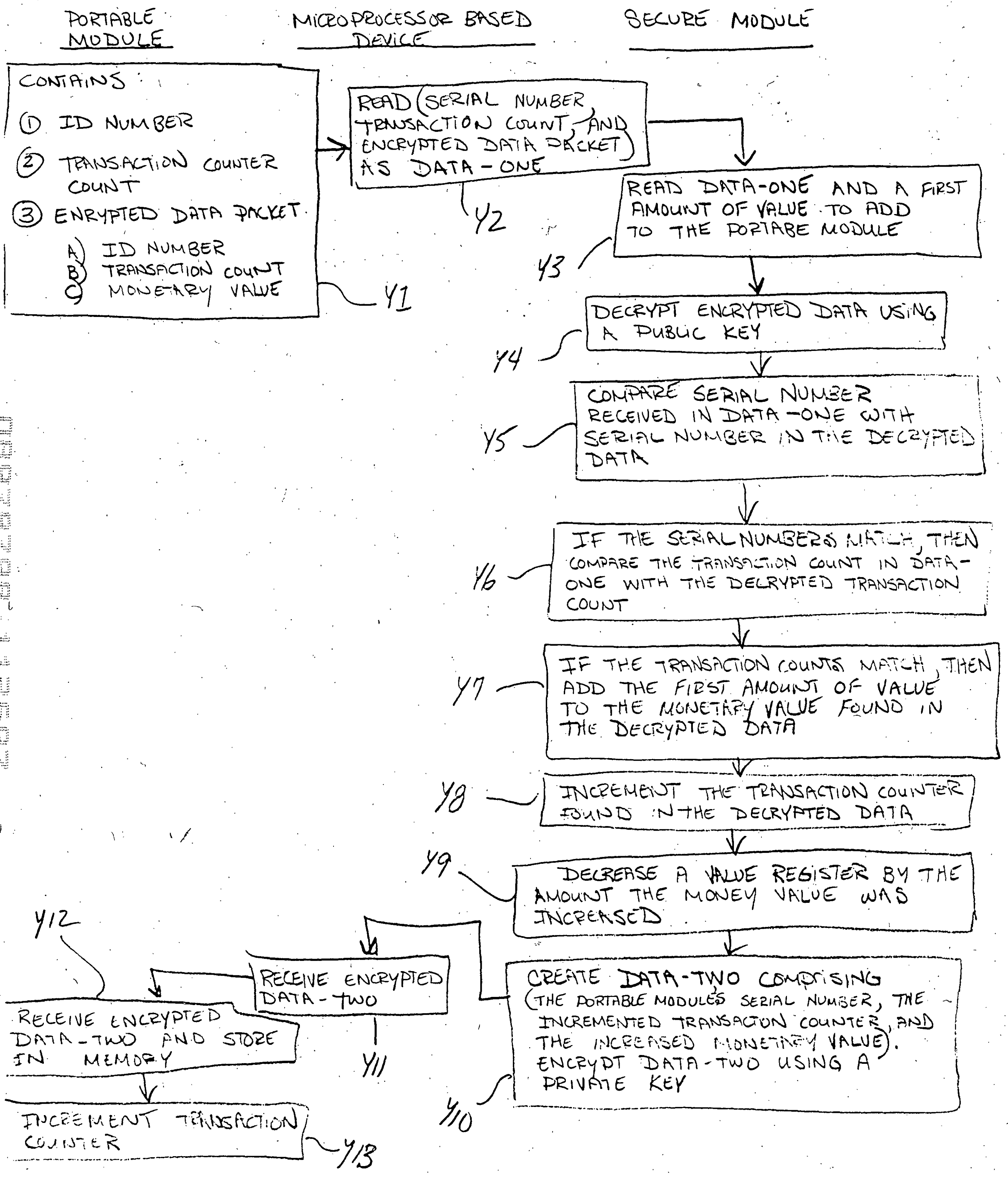
FIGURE 4 CONTINUED 20661-429


PORTABLE MODULE
MICRO PROCESSOR BASED DEVICE
SECURE MODULE
CREATE DATA-TWO COMPRISING (THE PORTABLE MODULE'S SERIAL NUMBER, INCREMENTED TRANSACTION COUNTER, AND REDUCED MONETARY VALUE) AND ENCRYPT DATA-TWO USING A PRIVATE KEY
X9
X10
RECEIVE ENCRYPTED DATA-TWO
RECEIVE ENCRYPTED DATA-TO AND STORE IN MEMORY
X11
INCREMENT TRANSACTION COUNTER
X12

08/978998

20661-429

FIGURE 5


PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CONTAINS:
① ID NUMBER
② TRANSACTION COUNTER COUNT
③ ENRYPTED DATA PACKET
A) ID NUMBER
B) TRANSACTION COUNT
C) MONETARY VALUE
41
READ (SERIAL NUMBER, TRANSACTION COUNT, AND ENCRYPTED DATA PACKET) AS DATA-ONE
42
READ DATA-ONE AND A FIRST AMOUNT OF VALUE TO ADD TO THE PORTABE MODULE
43
DECRYPT ENCRYPTED DATA USING A PUBLIC KEY
44
COMPARE SERIAL NUMBER RECEIVED IN DATA-ONE WITH SERIAL NUMBER IN THE DECRYPTED DATA
45
IF THE SERIAL NUMBERS MATCH, THEN COMPARE THE TRANSACTION COUNT IN DATA-ONE WITH THE DECRYPTED TRANSACTION COUNT
46
IF THE TRANSACTION COUNTS MATCH, THEN ADD THE FIRST AMOUNT OF VALUE TO THE MONETARY VALUE FOUND IN THE DECRYPTED DATA
47
INCREMENT THE TRANSACTION COUNTER FOUND IN THE DECRYPTED DATA
48
DECREASE A VALUE REGISTER BY THE AMOUNT THE MONEY VALUE WAS INCREASED
49
CREATE DATA-TWO COMPRISING (THE PORTABLE MODULE'S SERIAL NUMBER, THE INCREMENTED TRANSACTON COUNTER, AND THE INCREASED MONETARY VALUE). ENCRYPT DATA-TWO USING A PRIVATE KEY
410
RECEIVE ENCRYPTED DATA-TWO
411
RECEIVE ENCRYPTED DATA-TWO AND STORE IN MEMORY
412
INCREMENT TRANSACTION COUNTER
413

09/979798

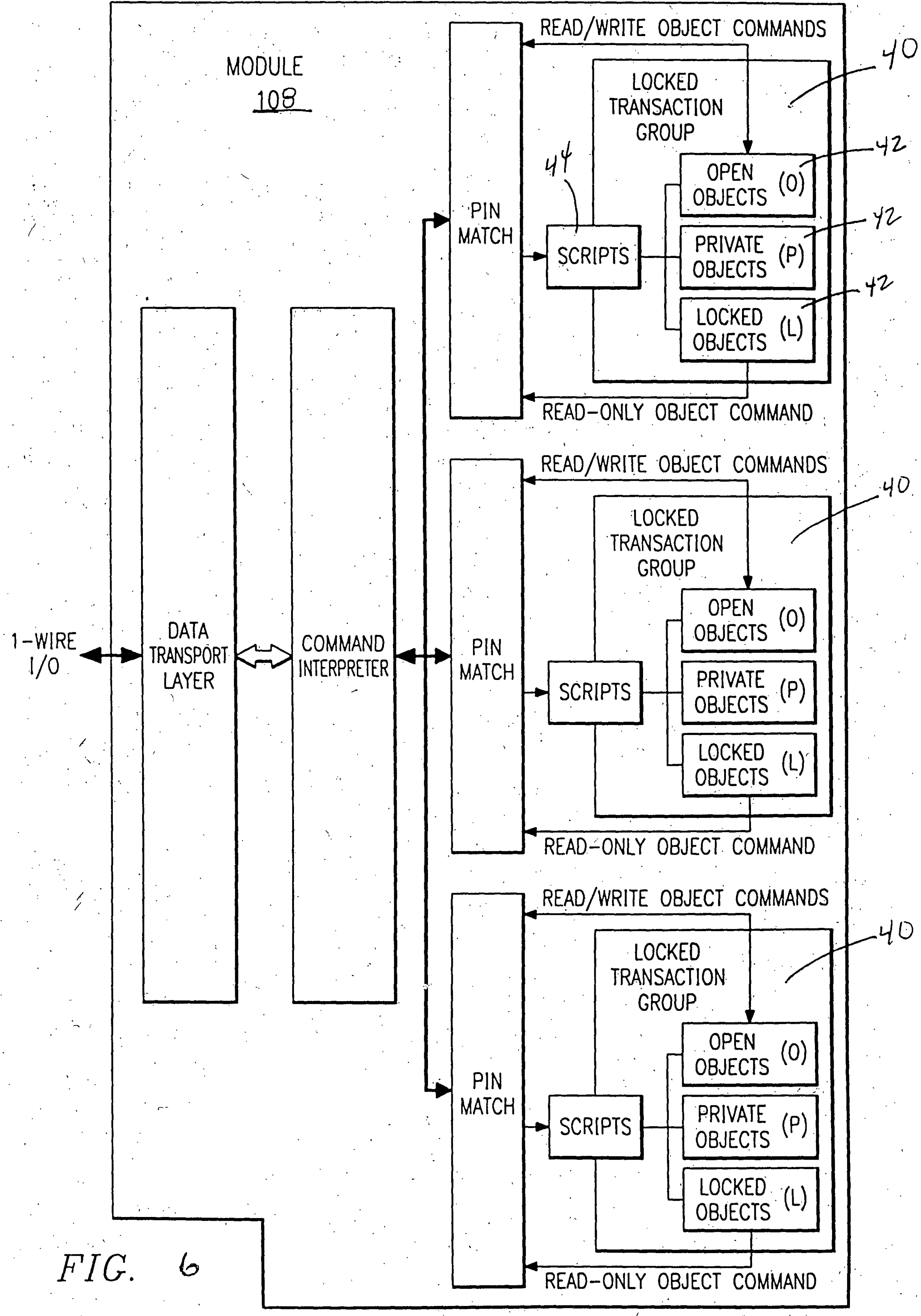
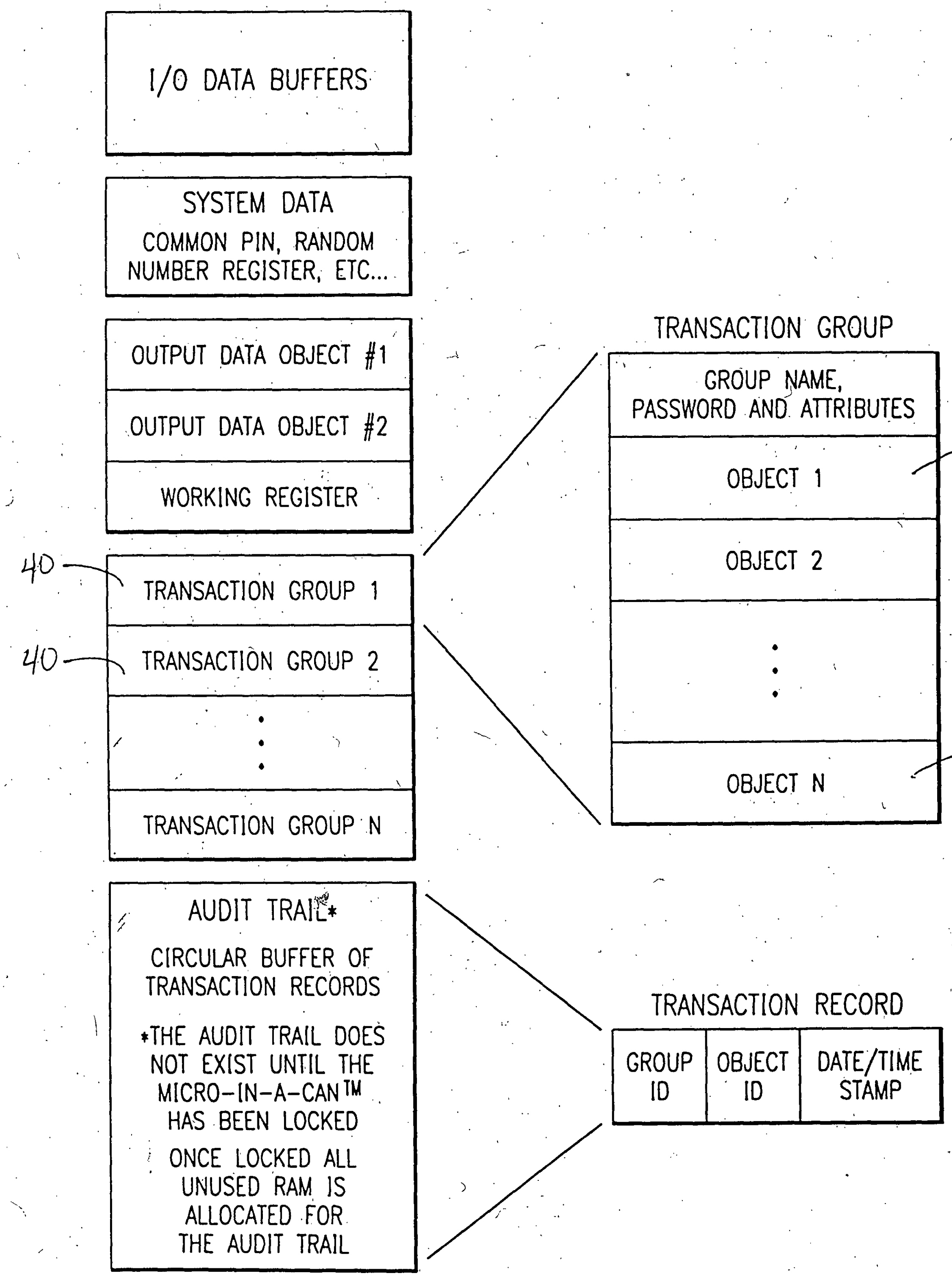


FIG. 6

2006 129 08/978798

FIG. 7



PATENT APPLICATION
DOCKET NO.: 20661/00429

RULES 63 AND 67 (37 C.F.R. 1.63 and 1.67)
DECLARATION AND POWER OF ATTORNEY

FOR UTILITY/DESIGN/CIP/PCT NATIONAL APPLICATIONS

As a named inventor, **STEPHEN M. CURRY, DONALD W. LOOMIS, and MICHAEL L. BOLAN**, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name; and

I believe that I am the original, first and sole inventor (if only one name is listed above) or an original, first and joint inventor (if plural names are listed above) of the subject matter which is claimed and for which a patent is sought on the invention entitled: **TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE**, the specification of which: (mark only one)

_____ (a) is attached hereto.
__X__ (b) was filed on January 31, 1996 as Application Serial No. 08/594,975.
_____ (c) was filed as PCT International Application No. PCT/_____ on ____ and was amended on _____ (if applicable).
_____ (d) was filed on _____________as Application Serial No. ____________and issued as Patent No. ___________on _____________.

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims as amended by any amendment referred to above or as allowed as indicated above.

I acknowledge the duty to disclose all information known to me to be material to the patentability of this application as defined in 37 CFR § 1.56. If this is a continuation-in-part (CIP) application, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of 35 U.S.C. § 112, I acknowledge the duty to disclose to the Office all information known to me to be material to patentability of the application as defined in 37 CFR § 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this CIP application.

I hereby claim foreign priority benefits under 35 U.S.C. § 119/365 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate filed by me or my assignee disclosing the subject matter claimed in this application and having a filing date (1) before that of the application on which my priority is claimed or, (2) if no priority is claimed, before the filing date of this application:

PRIOR FOREIGN PATENTS

Number	Country	Month/Day/Year Filed	Date first laid-open or Published	Date patented or Granted	Priority Claimed Yes	No
_____	_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____	_____

I hereby claim the benefit under 35 U.S.C. § 120/365 of any United States application(s) listed below and PCT international applications listed above or below:

PRIOR U.S. OR PCT APPLICATIONS

Application No. (series code/serial no.)	Month/Day/Year Filed	Status(pending, abandoned, patented)
_____	_____	_____
_____	_____	_____

I hereby appoint:

H. MATHEWS GARLAND, Reg. No. 19,129
THOMAS L. CANTRELL, Reg. No. 20,849
THOMAS L. CRISMAN, Reg. No. 24,846
STANLEY R. MOORE, Reg. No. 26,958
GERALD T. WELCH, Reg. No. 30,332
P. WESTON MUSSELMAN, JR., Reg No. 31,644
ROGER L. MAXWELL, Reg. No. 31,855
JEFFERY E. BACON, Reg. No. 35,055
ANDRE M. SZUWALSKI, Reg. No. 35,701
J. KEVIN GRAY, Reg. No. 37,141
STEVEN R. GREENFIELD, Reg. No. 38,166
CRAIG A. HOERSTEN, Reg. No. 38,917
STUART D. DWORK, Reg. No. 31,103

all of the firm of **JENKENS & GILCHRIST, P.C.**, 3200 Fountain Place, 1445 Ross Avenue, Dallas, Texas 75202-2799, as my attorneys and/or agents, with full power of substitution and revocation, to prosecute this application and to transact all business in the United States Patent and Trademark Office connected therewith, and to file and prosecute any international patent application filed thereon before any international authorities under the Patent Cooperation Treaty, and I hereby authorize them to act and rely on instructions from and communicate directly with the person/assignee/attorney/firm/organization who/which first sent this case to them and by whom/which I hereby declare that I have consented after full disclosure to be represented unless/until I instruct them in writing to the contrary.

Please address all correspondence and direct all telephone calls to:

PATENT APPLICATION
DOCKET NO.: 20661/00429

Steven R. Greenfield
Jenkens & Gilchrist, P.C.
3200 Fountain Place
1445 Ross Avenue
Dallas, Texas 75202-2799
214/855-4789
214/855-4300 (fax)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further 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, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

NAMED INVENTOR(S)

1	STEPHEN M. CURRY **Full Name**	Stephen M. Curry **Inventor's Signature**	April 16, 1996 **Date**
	6646 Clearhaven Circle Dallas, TX 75248 **Residence** (city, state, country)		USA **Citizenship**
	6646 Clearhaven Circle Dallas, TX 75248 **Post Office Address** (include zip code)		

2	DONALD W. LOOMIS **Full Name**	Donald W. Loomis **Inventor's Signature**	April 16, 1996 **Date**
	316 Dakota Lane Coppell, TX 75019 **Residence** (city, state, country)		USA **Citizenship**
	316 Dakota Lane Coppell, TX 75019 **Post Office Address** (include zip code)		

3	MICHAEL L. BOLAN **Full Name**	Michael L Bolan **Inventor's Signature**	4-18-98 **Date**
	6214 Misty Trail Dallas, TX 75248 **Residence** (city, state, country)		USA **Citizenship**
	6214 Misty Trail Dallas, TX 75248 **Post Office Address** (include zip code)		

(FOR ADDITIONAL INVENTORS, check here ____ and add additional sheet for inventor information regarding signature, name, date, citizenship, residence and address)

#3
IDS w/att
JB 3/18/98

Patent
Docket No. 20661-429C1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Curry et al.) Group Art Unit: 2766 ~~2202~~
)
Serial No.: Unknown) Examiner: White, C.
)
Filed: November 25, 1997)

For: Transfer of Valuable Information Between a Secure Module and Another Module

Assistant Commissioner for Patents
Washington DC 20231

CERTIFICATE OF MAILING BY EXPRESS MAIL

"EXPRESS MAIL" Mailing Label No. EM492669214US
Date of Deposit: November 26, 1997
I hereby certify that this paper or fee is being deposited with the U.S. Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, D.C. 20231

Type or Print Name: CAROL MARSTALLER

Carol Marstaller
Signature

Dear Sir:

INFORMATION DISCLOSURE STATEMENT

In accordance with Applicant's duty under 37 C.F.R. § 1.56 and 1.97, Applicant hereby submits the attached form PTO-1449 (modified) which lists art cited. The art listed therein, while of some relevance, is not necessarily considered to teach or suggest any aspect of the invention described and claimed in the above-identified patent application. This statement is also not to be construed as a representation that a search has, or has not, been conducted or that no better art exists. Rather, this statement discloses only the best art of which the Applicant is aware.

In considering the art set forth below, it may be noted by the Examiner that certain of the references may contain markings, underlinings or other notations. These markings or notations are not to be construed as drawing the Examiner's attention either to selected parts or away from other parts of the references. Any such markings were either present on the copies of the references obtained by Applicant, or were made thereon during the study of the references by the Applicant and/or his attorneys.

The Examiner is respectfully requested to consider each of the cited references, indicate such consideration by initialling each reference on the enclosed Form PTO-1449 (modified) and return a copy of the same with the next communication to the Applicant. For the convenience of the Examiner in considering the references, copies of the cited references are enclosed with this communication.

Respectfully submitted,

Steven R. Greenfield
Reg. No. 38,166

Date: November 26, 1997

Jenkens & Gilchrist, P.C.
3200 Fountain Place
1445 Ross Avenue
Dallas, Texas 75202-2799
214/855-4708

Sheet 1 of 1

Form PTO-1449 Modified List of Patents and Publications Cited by Applicant (Use several sheets if necessary) U.S. Patent Department of Commerce Patent and Trademark Office	**Docket No.:** 20661-429C1	**~~Prior~~ Serial No.:** ~~08/594,975~~ 08/978798
	Applicants: Curry et al.	
	Prior Filing Date: January 31, 1996	**Prior Group:** 2766 ~~2202~~ ~~3642~~

U.S. PATENT DOCUMENTS

Examiner Initial		Document No.	Date	Name	Class	Subclass
CW	AA	5,003,594	03/26/91	Shinagawa	380	24
CW	AB	5,546,463	08/13/96	Caputo et al.	380	25
CW	AC	5,621,796	04/15/97	Davis et al .	380	24
CW	AD	5,539,825	07/23/96	Akiyama et al.	380	24
CW	AE	5,577,121	11/19/96	Davis et al.	380	24
	AF					
	AG					
	AH					
	AI					
	AJ					

FOREIGN PATENT DOCUMENTS

Examiner Initial		Document No.	Date	Country	Translation	
					Yes	No
	AK					
	AL					
	AM					
	AN					

EXAMINER: C. White	DATE CONSIDERED: 5/26/98

CHG LOC 0380
LOC 3665

Receipt

Patent Application
Docket No. 20661-00429D1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In the Application of:

CURRY ET AL	§		
	§	Examiner:	UNKNOWN
Serial No.: 08/978,798	§		
	§	Group Art Unit:	~~3642~~ 2764
Filed: November 26, 1997	§		

For: TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

Assistant Commissioner For Patents
Washington, D.C. 20231

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited postage paid with the United States Postal Service postage paid as first class mail in an envelope addressed to: Assistant Commissioner For Patents, Washington, D.C. 20231

on 4/21/98

Signature

Dear Sir:

Attached is a copy of the official filing receipt received from the Patent and Trademark Office regarding this application. Please amend the official filing receipt as follows:

Please correct the title on the attached filing receipt as follows: --TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE--.

A corrected, marked copy of the original filing receipt is enclosed.

Applicants respectfully request that a new official filing receipt be provided having the corrected title thereon.

Applicants understand that there should be no fee.

Respectfully submitted,

JENKENS & GILCHRIST, P.C.

Date: April 21, 1998

Steven R. Greenfield
Reg. No. 38,166

Jenkens & Gilchrist,
A Professional Corporation
1445 Ross Avenue, Suite 3200
Dallas, Texas 75202-2799
214/855-4789
214/855-4300 (fax)

PTO-103X
(Rev. 8-95)

FILING RECEIPT

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office
ASSISTANT SECRETARY AND COMMISSIONER
OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

APPLICATION NUMBER	FILING DATE	GRP ART UNIT	FIL FEE REC'D	ATTORNEY DOCKET NO.	DRWGS	TOT CL	IND CL
08/978,798	11/26/97	3642	$0.00	20661-429D1	8	6	1

STEVEN R GREENFIELD
JENKENS & GILCHRIST
1445 ROSS AVENUE
SUITE 3200
DALLAS TX 75202

Receipt is acknowledged of this nonprovisional Patent Application. It will be considered in its order and you will be notified as to the results of the examination. Be sure to provide the U.S. APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION when inquiring about this application. Fees transmitted by check or draft are subject to collection. Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please write to the Application Processing Division's Customer Correction Branch within 10 days of receipt. Please provide a copy of the Filing Receipt with the changes noted thereon.

Applicant(s)
STEPHEN M. CURRY, DALLAS, TX; DONALD W. LOOMIS, COPPELL, TX; MICHAEL L. BOLAN, DALLAS, TX.

CONTINUING DATA AS CLAIMED BY APPLICANT-
THIS APPLN IS A DIV OF 08/594,975 01/31/96

FOREIGN FILING LICENSE GRANTED 03/04/98
TITLE
~~TRANPER~~ OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE TRANSFER

PRELIMINARY CLASS: 380

Complete Filing receipt 3/13/98

(see reverse)

Transaction History Date 1998-10-16
Date information retrieved from USPTO Patent
Application Information Retrieval (PAIR)
system records at www.uspto.gov

BEST COPY

UNITED STATL ᴗEPARTMENT OF COMMERCE
Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

APPLICATION NUMBER	FILING DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NO.
08/978,798	11/26/97	CURRY	S 20661-429D1

EXAMINER
HAYES, G

LM11/1016

STEVEN R GREENFIELD
JENKENS & GILCHRIST
1445 ROSS AVENUE
SUITE 3200
DALLAS TX 75202

ART UNIT	PAPER NUMBER
2766	5/B

DATE MAILED: 10/16/98

This is a communication from the examiner in charge of your application.
COMMISSIONER OF PATENTS AND TRADEMARKS

NOTICE OF ALLOWABILITY

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance and Issue Fee Due or other appropriate communication will be mailed in due course.

☒ This communication is responsive to papers filed on November 26, 1997.

☒ The allowed claim(s) is/are 16-21.

☐ The drawings filed on ______ are acceptable.

☐ Acknowledgement is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d).

☐ All ☐ Some* ☐ None of the CERTIFIED copies of the priority documents have been

☐ received.

☐ received in Application No. (Series Code/Serial Number) ______.

☐ received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

*Certified copies not received: ______.

☐ Acknowledgement is made of a claim for domestic priority under 35 U.S.C. § 119(e).

A SHORTENED STATUTORY PERIOD FOR RESPONSE to comply with the requirements noted below is set to EXPIRE **THREE MONTHS** FROM THE "DATE MAILED" of this Office action. Failure to timely comply will result in ABANDONMENT of this application. Extensions of time may be obtained under the provisions of 37 CFR 1.136(a).

☐ Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL APPLICATION, PTO-152, which discloses that the oath or declaration is deficient. A SUBSTITUTE OATH OR DECLARATION IS REQUIRED.

☒ Applicant MUST submit NEW FORMAL DRAWINGS

☐ because the originally filed drawings were declared by applicant to be informal.

☒ including changes required by the Notice of Draftperson's Patent Drawing Review, PTO-948, attached hereto or to Paper No. ______.

☐ including changes required by the proposed drawing correction filed on ______, which has been approved by the examiner.

☐ including changes required by the attached Examiner's Amendment/Comment.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the reverse side of the drawings. The drawings should be filed as a separate paper with a transmittal letter addressed to the Official Draftperson.

☐ Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Any response to this letter should include, in the upper right hand corner, the APPLICATION NUMBER (SERIES CODE/SERIAL NUMBER). If applicant has received a Notice of Allowance and Issue Fee Due, the ISSUE BATCH NUMBER and DATE of the NOTICE OF ALLOWANCE should also be included.

Attachment(s)

☒ Notice of References Cited, PTO-892

☒ Information Disclosure Statement(s), PTO-1449, Paper No(s). ______

☒ Notice of Draftsperson's Patent Drawing Review, PTO-948

☐ Notice of Informal Patent Application, PTO-152

☐ Interview Summary, PTO-413

☒ Examiner's Amendment/Comment

☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material

☒ Examiner's Statement of Reasons for Allowance

DETAILED ACTION

1. The following is an examiner's statement of reasons for allowance:

Neither Rosen ('419) or Rosen ('280) discloses passing said second value datum from said second module to said electronic device; passing said second value datum from said electronic device to said first module; and discontinuing communication between said first module and said electronic device.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

2. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Steven Greenfield on May 26, 1998.

3. The application has been amended as follows:

Claim 21 has been changed from "wherein the step (b) of passing is performed over at least a single conductive contact." to --The method of claim 16, wherein the step (b) of passing is performed over at least a single conductive contact.--

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Carmen White whose telephone number is (703) 305-4458.

Thomas H. Tarcza

THOMAS H. TARCZA
SUPERVISORY PATENT EXAMINER
GROUP ~~2200~~ 3640

File History Content Report

The following content is missing from the original file history record obtained from the United States Patent and Trademark Office. No additional information is available.

Document Date - 1998-10-16

Document Title - List of references cited by examiner

This page is not part of the official USPTO record. It has been determined that content identified on this document is missing from the original file history record.

FORM PTO 948 (REV. 01-97) U.S. DEPARTMENT OF COMMERCE-Patent and Trademark Office Application No. 978798

NOTICE OF DRAFTPERSON'S PATENT DRAWING REVIEW

The drawing filied (insert date) 11/26/97 are

A. ______ not objected to by the Draftperson under 37 CFR 1.84 or 1.152.

B. X ______ objected to by the Draftperson under 37 CFR 1.84 or 1.152 as indicated below. The Examiner will require submission of new, corrected drawings whe necessary. Corrected drawings must be submitted according to the instructions on the back of this notice.

1. DRAWINGS. 37 CFR 1.84(a): Acceptable categories of drawings: Black ink. Color.
   ______ Color drawing are not acceptable until petition is granted. Fig.(s) ______
   ______ Pencil and non black ink is not permitted. Fig(s) ______
2. PHOTOGRAPHS. 37 CFR 1.84(b)
   ______ Photographs are not acceptable until petition is granted,
   ______ 3 full-tone sets are required. Fig(s) ______
   ______ Photographs not properly mounted (must brystol board or photographic double-weight paper). Fig(s) ______
   ______ Poor quality (half-tone). Fig(s) ______
3. TYPE OF PAPER. 37 CFR 1.84(e)
   ______ Paper not flexible, strong, white and durable. Fig.(s) ______
   ______ Erasures, alterations, overwritings, interlineations, folds, copy machine marks not acceptable. (too thin)
   ______ Mylar, vellum paper is not acceptable (too thin). Fig(s) ______
4. SIZE OF PAPER. 37 CFR 1.84(F): Acceptable sizes:
   ______ 21.0 cm by 29.7 cm (DIN size A4)
   ______ 21.6 cm by 27.9 cm (8 1/2 x 11 inches)
   ______ All drawings sheets not the same size. Sheet(s) ______
5. MARGINS. 37 CFR 18.4(g): Acceptable margins:
   Top 2.5 cm Left 2.5 cm Right 1.5 cm Bottom 1.0 cm
   SIZE: A4 Size
   Top 2.5 cm Left 2.5 cm Right 1.5 cm Bottom 1.0 cm
   SIZE: 8 1/2 x 11
   ______ Margins not acceptable. Fig(s) 4-6,
   ______ Top (T) X ______ Left (L)
   ______ Right (R) ______ Bottom (B)
6. VIEWS. CFR 1.84(h)
   REMINDER: Specification may require revision to correspond to drawing changes.
   ______ Views connected by projection lines or lead lines. Fig.(s) ______
   Partial views. 37 CFR 1.84(h)(2)
   ______ Brackets needed to show figure as one entity. Fig.(s) ______
   X ______ Views not labeled separately or properly. Fig.(s) 4
   ______ Enlarged view not labeled separately or properly. Fig.(s) ______
7. SECTIONAL VIEWS. 37 CFR 1.84(h)(3)
   ______ Hatching not indicated for sectional portions of an object. Fig.(s) ______
   ______ Sectional designation should be noted with Arabic or Roman numbers. Fig.(s) ______
8. ARRANGEMENT OF VIEWS. 37 CFR 1.84(i)
   ______ Words do not appear on a horizontal, left-to-right fashion when page is either upright or turned, so that the top becomes the right side, except for graphs. Fig.(s) ______
   ______ Views not on the same plane on drawing sheet. Fig.(s) ______
9. SCALE. 37 CFR 1.84(k)
   ______ Scale not large enough to show mechansim with crowding when drawing is reduced in size to two-thirds in reproduction. Fig.(s) ______
10. CHARACTER OF LINES, NUMBERS, & LETTERS. 37 CFR 1.84(l)
    X ______ Lines, numbers & letters not uniformly thick and well defined, clean, durable and black (poor line quality). Fig.(s) 1-7
11. SHADING. 37 CFR 1.84(m)
    ______ Solid black areas pale. Fig.(s) ______
    ______ Solid black shading not permitted. Fig.(s) ______
    ______ Shade lines, pale, rough and blurred. Fig.(s)
12. NUMBERS, LETTERS, & REFERENCE CHARACTERS. 37 CFR 1.48(p)
    ______ Numbers and reference characters not plain and legible. Fig.(s) ______
    ______ Figure legends are poor. Fig.(s) ______
    ______ Numbers and reference characters not oriented in the same direction as the view. 37 CFR 1.84(p)(3) Fig.(s) ______
    ______ Engligh alphabet not used. 37 CFR 1.84(p)(3) Fig.(s) ______
    ______ Numbers, letters and reference characters must be at least .32 cm (1/8 inch) in height. 37 CFR 1.84(p)(3) Fig.(s) ______
13. LEAD LINES. 37 CFR 1.84(q)
    ______ Lead lines cross each other. Fig.(s) ______
    ______ Lead lines missing. Fig.(s) ______
14. NUMBERING OF SHEETS OF DRAWINGS. 37 CFR 1.48(t)
    ______ Sheets not numbered consecutively, and in Ababic numerals beginning with number 1. Fig.(s) ______
15. NUMBERING OF VIEWS. 37 CFR 1.84(u)
    ______ Views not numbered consecutively, and in Abrabic numerals, beginning with number 1. Fig.(s) ______
16. CORRECTIONS. 37 CFR 1.84(w)
    ______ Corrections not made from PTO-948 dated ______
17. DESIGN DRAWINGS. 37 CFR 1.152
    ______ Surface shading shown not appropriate. Fig.(s) ______
    ______ Solid black shading not used for color contrast. Fig.(s) ______

COMMENTS

REVIEWER A. D DATE 3/10/98 TELEPHONE NO. 7033058404

ATTACHMENT TO PAPER NO. ______

BEST COPY

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office

NOTICE OF ALLOWANCE AND ISSUE FEE DUE

LM11/1016

STEVEN R GREENFIELD
JENKENS & GILCHRIST
1445 ROSS AVENUE
SUITE 3200
DALLAS TX 75202

APPLICATION NO.	FILING DATE	TOTAL CLAIMS	EXAMINER AND GROUP ART UNIT	DATE MAILED
08/978,798	11/26/97	006	HAYES, G 2766	10/16/98

First Named Applicant CURRY, 35 USC 154(b) term ext. = 0 Days.

TITLE OF INVENTION RANFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

ATTY'S DOCKET NO.	CLASS-SUBCLASS	BATCH NO.	APPLN. TYPE	SMALL ENTITY	FEE DUE	DATE DUE
20661-429D1	380-024.000	K51	UTILITY	NO	$1320.00	01/19/99

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED.

THE ISSUE FEE MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED.

HOW TO RESPOND TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.
If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is changed, pay twice the amount of the FEE DUE shown above and notify the Patent and Trademark Office of the change in status, or
B. If the status is the same, pay the FEE DUE shown above.

If the SMALL ENTITY is shown as NO:

A. Pay FEE DUE shown above, or

B. File verified statement of Small Entity Status before, or with, payment of 1/2 the FEE DUE shown above.

II. Part B-Issue Fee Transmittal should be completed and returned to the Patent and Trademark Office (PTO) with your ISSUE FEE. Even if the ISSUE FEE has already been paid by charge to deposit account, Part B Issue Fee Transmittal should be completed and returned. If you are charging the ISSUE FEE to your deposit account, section "4b" of Part B-Issue Fee Transmittal should be completed and an extra copy of the form should be submitted.

III. All communications regarding this application must give application number and batch number.
Please direct all communications prior to issuance to Box ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PATENT AND TRADEMARK OFFICE COPY

PTOL-85 (REV. 10-96) Approved for use through 06/30/99. (0651-0033)

Jenkens & Gilchrist
A PROFESSIONAL CORPORATION

FOUNTAIN PLACE
1445 ROSS AVENUE, SUITE 3200
DALLAS, TX 75202

(214) 855-4500
TELECOPIER (214) 855-4300

AUSTIN, TEXAS
(512) 499-3800
HOUSTON, TEXAS
(713) 951-3300
SAN ANTONIO, TEXAS
(210) 308-3100
WASHINGTON, D.C.
(202) 326-1500

WRITER'S DIRECT DIAL NUMBER
Raymond Van Dyke
(214) 855-4708

Box ISSUE FEE
Assistant Commissioner
for Patents
Washington, D.C. 20231

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231

on 1/15/99

Carol Marstaller
Signature

Re:	Applicant(s):	Stephen Curry et al.
	Serial No.:	08/978,798
	Filed:	November 26, 1997
	Batch No.	K51
	NOA Mailed:	October 16, 1998
	For:	Transfer of Valuable Information Between a Secure Module and Another Module
	Docket No.:	20661-00429D1

Dear Sir:

Transmitted for filing with the Patent and Trademark Office are the following documents for the above-referenced patent application:

1. Part B Issue Fee Transmittal
2. Letter to Official Draftsperson
3. 8 Sheets of Formal Drawings
4. Check in the amount of $1,240.00 for issue fee and soft copies

Please address all communications related to this to:

Steven R. Greenfield
Jenkens & Gilchrist, P.C.
3200 Fountain Place
1445 Ross Avenue
Dallas, Texas 75202-2799

In the event there is an under or over payment, please debit or credit our Deposit Account #10-0447.

Respectfully submitted,

Steven R. Greenfield

Steven R. Greenfield
Registration No. 38,166

Jenkens & Gilchrist
A PROFESSIONAL CORPORATION

FOUNTAIN PLACE
1445 ROSS AVENUE, SUITE 3200
DALLAS, TX 75202

(214) 855-4500
TELECOPIER (214) 855-4300

AUSTIN, TEXAS
(512) 499-3800
HOUSTON, TEXAS
(713) 951-3300
SAN ANTONIO, TEXAS
(210) 308-3100
WASHINGTON, D.C.
(202) 326-1500

WRITER'S DIRECT DIAL NUMBER
Raymond Van Dyke
(214) 855-4708

Box ISSUE FEE
Assistant Commissioner
for Patents
Washington, D.C. 20231

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231

on 1/15/99

Carol Marstaller
Signature

Re:	Applicant(s):	Stephen Curry et al.
	Serial No.:	08/978,798
	Filed:	November 26, 1997
	Batch No.	K51
	NOA Mailed:	October 16, 1998
	For:	Transfer of Valuable Information Between a Secure Module and Another Module
	Docket No.:	20661-00429D1

Dear Sir:

Transmitted for filing with the Patent and Trademark Office are the following documents for the above-referenced patent application:

1. Part B Issue Fee Transmittal
2. Letter to Official Draftsperson
3. 8 Sheets of Formal Drawings
4. Check in the amount of $1,240.00 for issue fee and soft copies

Please address all communications related to this to:

Steven R. Greenfield
Jenkens & Gilchrist, P.C.
3200 Fountain Place
1445 Ross Avenue
Dallas, Texas 75202-2799

In the event there is an under or over payment, please debit or credit our Deposit Account #10-0447.

Respectfully submitted,

Steven R. Greenfield
Registration No. 38,166

PART B—I[illegible] FEE TRANSMITTAL

Complete and mail this form, together with [illegible]licable fees, to: Box ISSUE FEE
Assistant Commissioner for Patents
Washington, D.C. 20231

B$

MAILING INSTRUCTIONS: This form should be use[illegible] transmitting the ISSUE FEE. Blocks 1 through 4 should be completed where appropriate. All further correspondence including the Issue Fee Receipt, the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Legibly mark-up with any corrections or use Block 1)

LM11/1016

STEVEN R GREENFIELD
JENKENS & GILCHRIST
1445 ROSS AVENUE
SUITE 3200
DALLAS TX 75202

Note: The certificate of mailing below can only be used for domestic mailings of the Issue Fee Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing.

Certificate of Mailing

I hereby certify that this Issue Fee Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Box Issue Fee address above on the date indicated below.

CAROL MARSTALLER (Depositor's name)

Carol Marstaller (Signature)

1/15/99 (Date)

APPLICATION NO.	FILING DATE	TOTAL CLAIMS	EXAMINER AND GROUP ART UNIT		DATE MAILED
08/978,798	11/26/97	006	HAYES, G	2766	10/16/98

First Named Applicant CURRY. 35 USC 154(b) term ext. = 0 Days.

TITLE OF INVENTION RANFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

ATTY'S DOCKET NO.	CLASS-SUBCLASS	BATCH NO.	APPLN. TYPE	SMALL ENTITY	FEE DUE	DATE DUE
2 20661-429D1	380-024.000	K51	UTILITY	NO	$1320.00	01/19/99

1. Change of correspondence address or indication of " Fee Address" (37 CFR 1.363). Use of PTO form(s) and Customer Number are recommended, but not required.

   □ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

   □ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47) attached.

2. For printing on the patent front page, list (1) the names of up to 3 registered patent attorneys or agents OR, alternatively, (2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

   1 JENKENS + GILCHRIST
   2 ________
   3 ________

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)
   PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. Inclusion of assignee data is only appropiate when an assignment has been previously submitted to the PTO or is being submitted under separate cover. Completion of this form is NOT a subsititue for filing an assignment.

   (A) NAME OF ASSIGNEE DALLAS SEMICONDUCTOR CORPORATION

   (B) RESIDENCE: (CITY & STATE OR COUNTRY) DALLAS TX

   Please check the appropriate assignee category indicated below (will not be printed on the patent)

   □ individual ☒ corporation or other private group entity □ government

4a. The following fees are enclosed (make check payable to Commissioner of Patents and Trademarks):

☒ Issue Fee
☒ Advance Order - # of Copies 10

4b. The following fees or deficiency in these fees should be charged to:

DEPOSIT ACCOUNT NUMBER 10-0447
(ENCLOSE AN EXTRA COPY OF THIS FORM)

□ Issue Fee
□ Advance Order - # of Copies ________

The COMMISSIONER OF PATENTS AND TRADEMARKS IS requested to apply the Issue Fee to the application identified above.

(Authorized Signature) Steven R. Greenfield (Date) 1/15/99

NOTE; The Issue Fee will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the Patent and Trademark Office.

```
01/27/1999 RMAGAT1  00000026 08978798

01 FC:142                    1210.00 OP
02 FC:561                      30.00 OP
```

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending on the needs of the individual case. Any comments on the amount of time required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, D.C. 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND FEES AND THIS FORM TO: Box Issue Fee, Assistant Commissioner for Patents, Washington D.C. 20231

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMIT THIS FORM WITH FEE

km (1) 410 OIPE

DOCKET NO.: 20661-0u429D1

PATENT APPLICATION

Issue Batch No.: K51
Date of Notice
of Allowance : 10/16/98
Serial No. : 08/978,798

#6
CMS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent application of: Curry et al.

Serial No.: 08/978,798 Group No.: 2766

Filed: November 26, 1997 Examiner: Hayes, G.

For: **Transfer of Valuable Information Between a Secure Module and Another Module**

BOX ISSUE FEE
Assistant Commissioner for Patents
Washington, D.C. 20231

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231

on 1/15/99

Carol Marstaller
Signature

ATTN: Official Draftsperson

Sir:

TRANSMITTAL LETTER TO OFFICIAL DRAFTSPERSON

Enclosed please find 8 sheet(s) of formal drawings relating to the above-identified patent application.

The enclosed drawings each bear the Issue Batch No. K51, the date of the Notice of Allowance and Serial No. of the application on their reverse side.

In view of the above, the present application is believed to be in a condition ready for issuance.

Jenkens & Gilchrist, a Professional Corporation
1445 Ross Avenue, Ste. 3200
Dallas, Texas 75202-2799
214/855-4789
214/855-4300 FAX

Steven R. Greenfield
Registration No. 38,166

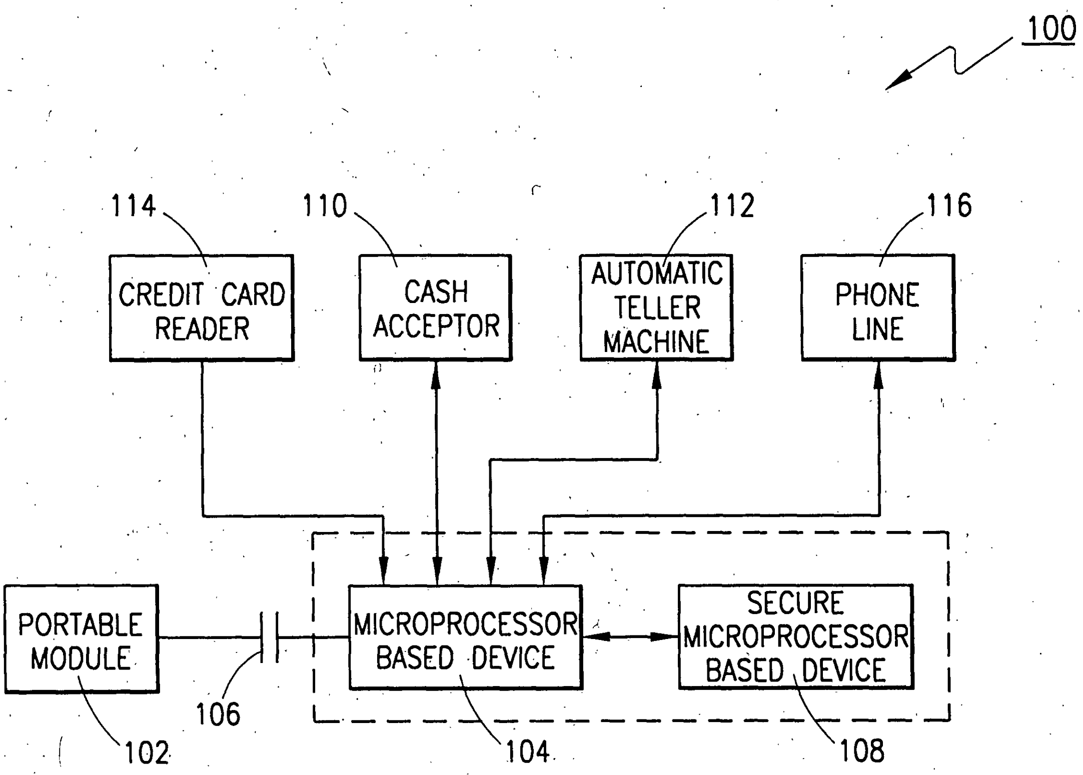

100
114
CREDIT CARD READER
110
CASH ACCEPTOR
112
AUTOMATIC TELLER MACHINE
116
PHONE LINE
PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MICROPROCESSOR BASED DEVICE
106
102
104
108

FIG. 1

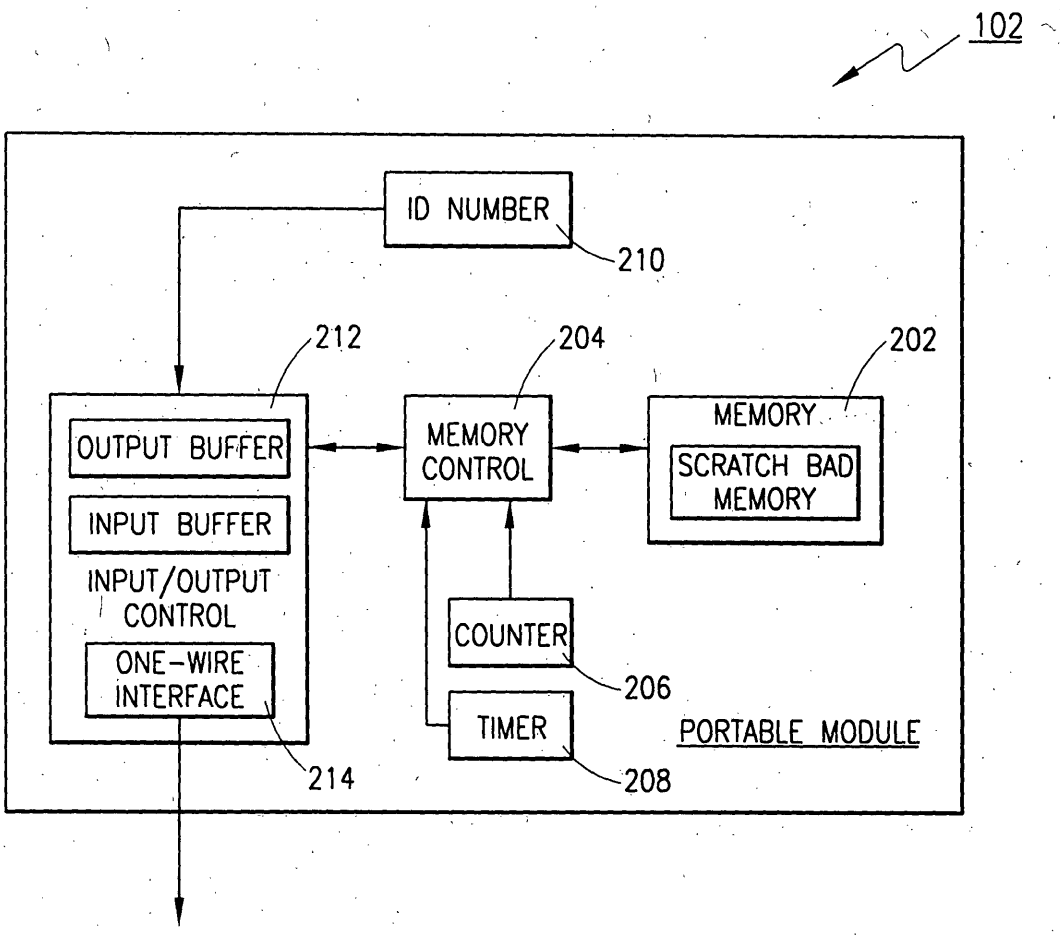

102
ID NUMBER
210
212
204
202
OUTPUT BUFFER
INPUT BUFFER
INPUT/OUTPUT CONTROL
ONE-WIRE INTERFACE
MEMORY CONTROL
MEMORY
SCRATCH BAD MEMORY
COUNTER
206
TIMER
208
214
PORTABLE MODULE

FIG. 2

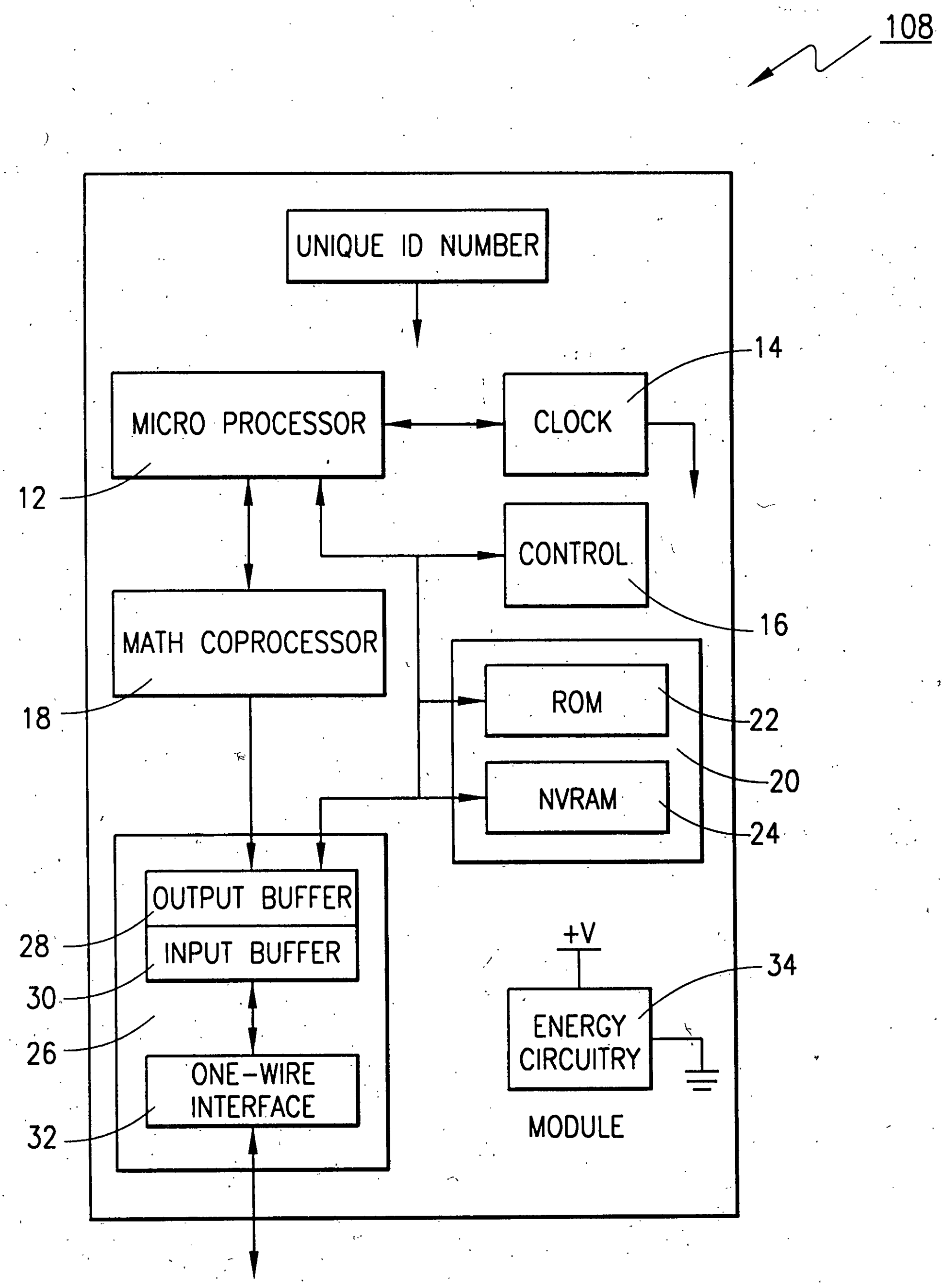

108
UNIQUE ID NUMBER
MICRO PROCESSOR
CLOCK
14
12
CONTROL
16
MATH COPROCESSOR
18
ROM
22
20
NVRAM
24
OUTPUT BUFFER
INPUT BUFFER
28
30
26
ONE-WIRE INTERFACE
32
+V
ENERGY CIRCUITRY
34
MODULE

FIG. 3

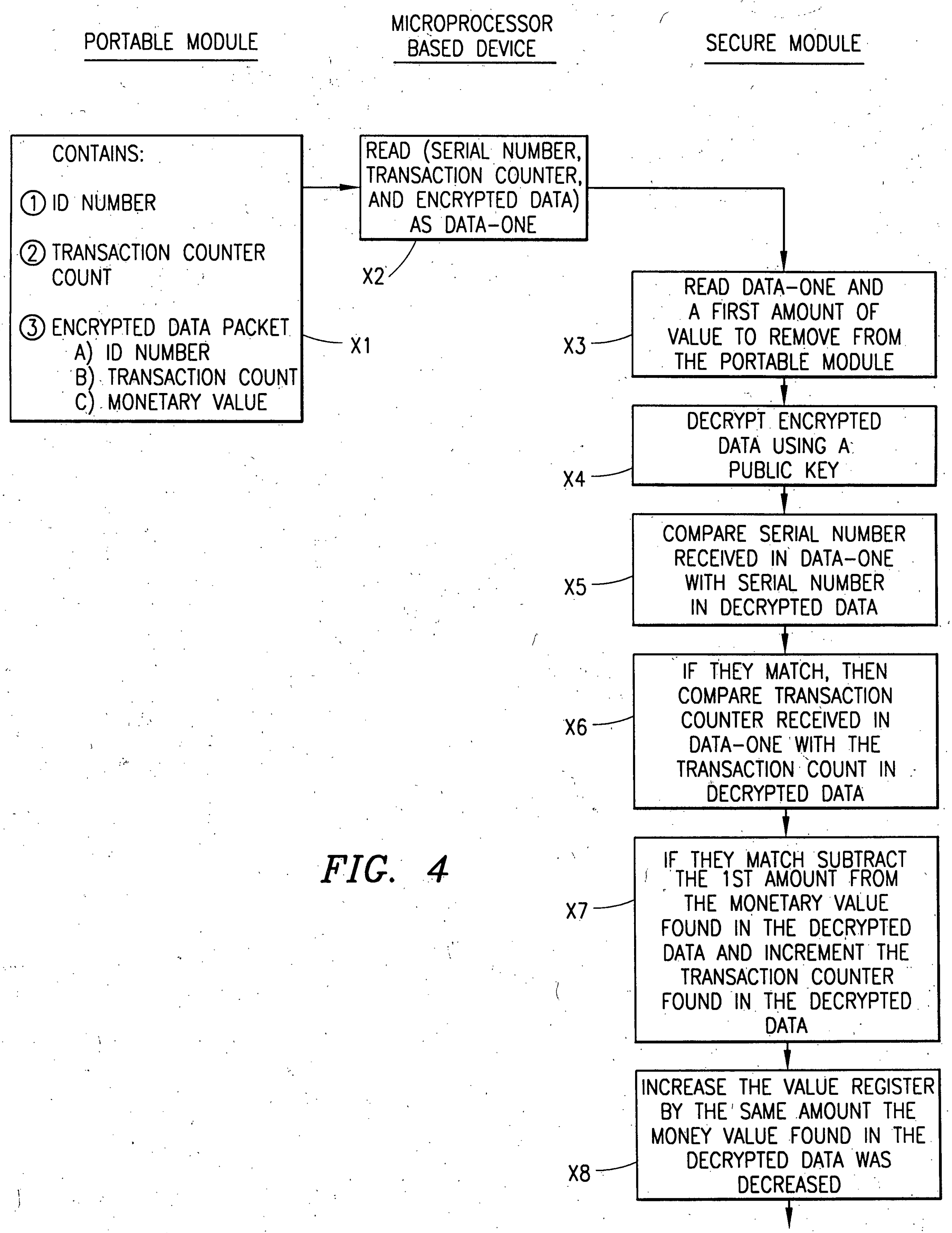

PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CONTAINS:
① ID NUMBER
② TRANSACTION COUNTER COUNT
③ ENCRYPTED DATA PACKET
A) ID NUMBER
B) TRANSACTION COUNT
C) MONETARY VALUE
X1
READ (SERIAL NUMBER, TRANSACTION COUNTER, AND ENCRYPTED DATA) AS DATA-ONE
X2
X3
READ DATA-ONE AND A FIRST AMOUNT OF VALUE TO REMOVE FROM THE PORTABLE MODULE
X4
DECRYPT ENCRYPTED DATA USING A PUBLIC KEY
X5
COMPARE SERIAL NUMBER RECEIVED IN DATA-ONE WITH SERIAL NUMBER IN DECRYPTED DATA
X6
IF THEY MATCH, THEN COMPARE TRANSACTION COUNTER RECEIVED IN DATA-ONE WITH THE TRANSACTION COUNT IN DECRYPTED DATA
X7
IF THEY MATCH SUBTRACT THE 1ST AMOUNT FROM THE MONETARY VALUE FOUND IN THE DECRYPTED DATA AND INCREMENT THE TRANSACTION COUNTER FOUND IN THE DECRYPTED DATA
X8
INCREASE THE VALUE REGISTER BY THE SAME AMOUNT THE MONEY VALUE FOUND IN THE DECRYPTED DATA WAS DECREASED

FIG. 4

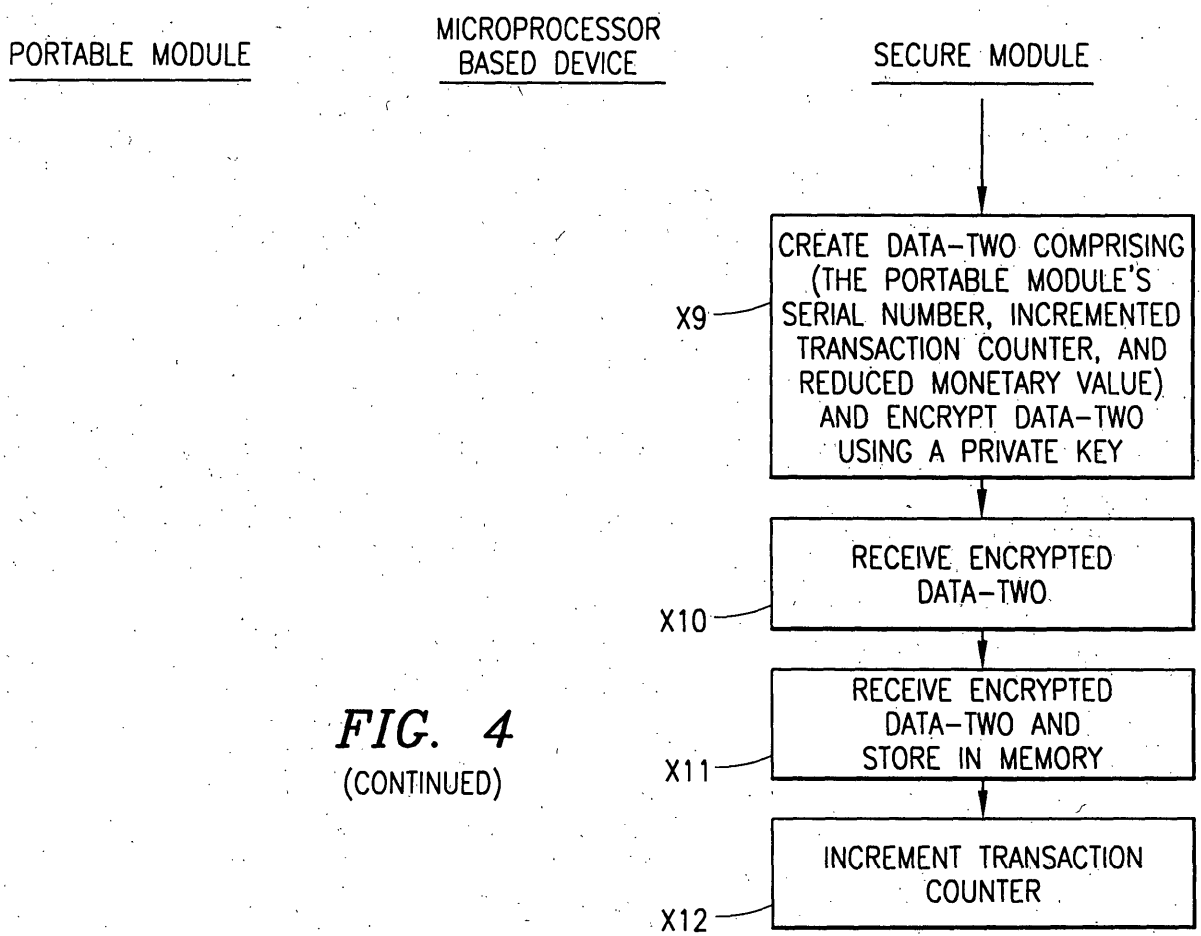

PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CREATE DATA–TWO COMPRISING (THE PORTABLE MODULE'S SERIAL NUMBER, INCREMENTED TRANSACTION COUNTER, AND REDUCED MONETARY VALUE) AND ENCRYPT DATA–TWO USING A PRIVATE KEY
X9
RECEIVE ENCRYPTED DATA–TWO
X10
RECEIVE ENCRYPTED DATA–TWO AND STORE IN MEMORY
X11
INCREMENT TRANSACTION COUNTER
X12

FIG. 4
(CONTINUED)

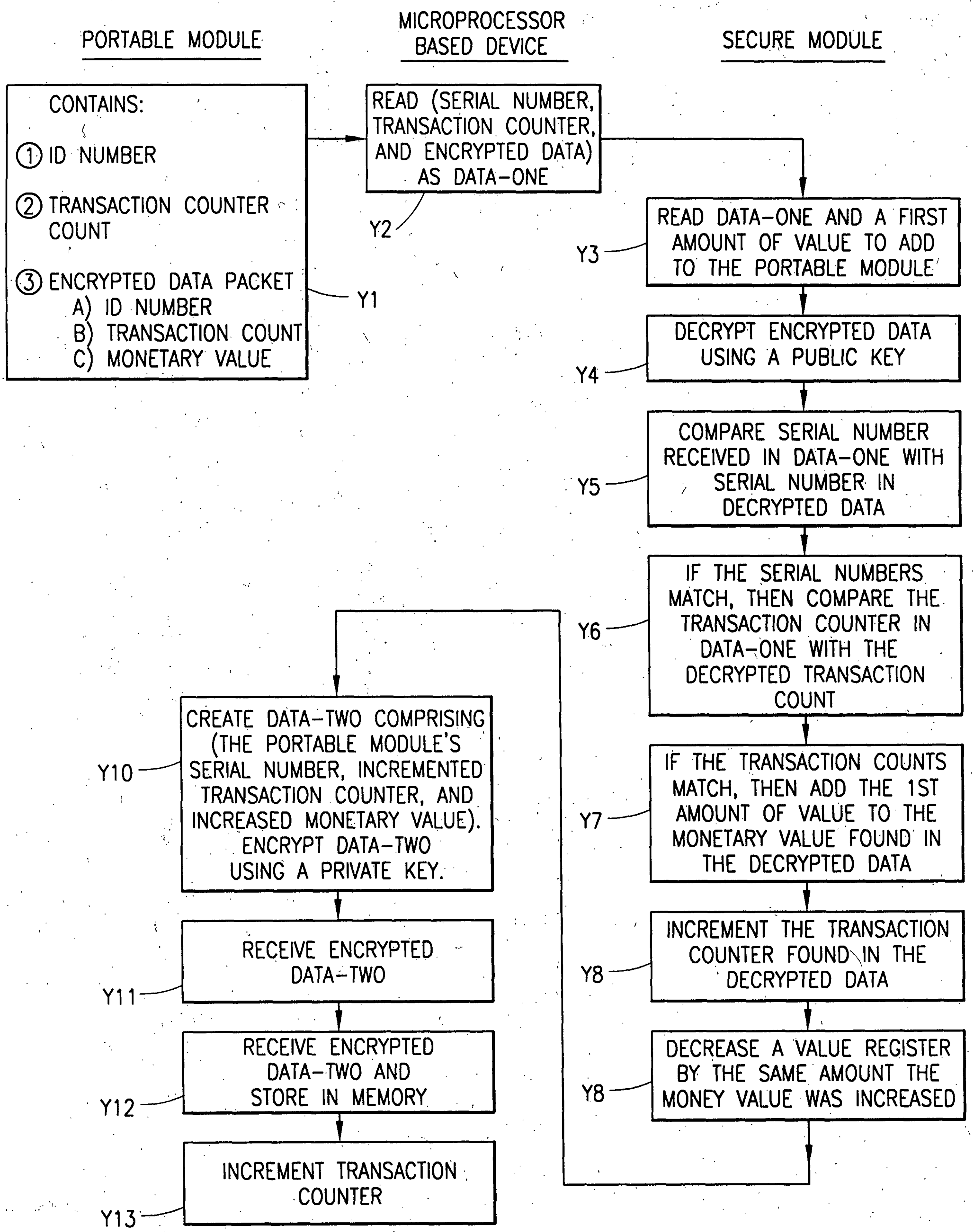

PORTABLE MODULE
MICROPROCESSOR BASED DEVICE
SECURE MODULE
CONTAINS:
① ID NUMBER
② TRANSACTION COUNTER COUNT
③ ENCRYPTED DATA PACKET
A) ID NUMBER
B) TRANSACTION COUNT
C) MONETARY VALUE
Y1
READ (SERIAL NUMBER, TRANSACTION COUNTER, AND ENCRYPTED DATA) AS DATA-ONE
Y2
READ DATA-ONE AND A FIRST AMOUNT OF VALUE TO ADD TO THE PORTABLE MODULE
Y3
DECRYPT ENCRYPTED DATA USING A PUBLIC KEY
Y4
COMPARE SERIAL NUMBER RECEIVED IN DATA-ONE WITH SERIAL NUMBER IN DECRYPTED DATA
Y5
IF THE SERIAL NUMBERS MATCH, THEN COMPARE THE TRANSACTION COUNTER IN DATA-ONE WITH THE DECRYPTED TRANSACTION COUNT
Y6
IF THE TRANSACTION COUNTS MATCH, THEN ADD THE 1ST AMOUNT OF VALUE TO THE MONETARY VALUE FOUND IN THE DECRYPTED DATA
Y7
INCREMENT THE TRANSACTION COUNTER FOUND IN THE DECRYPTED DATA
Y8
DECREASE A VALUE REGISTER BY THE SAME AMOUNT THE MONEY VALUE WAS INCREASED
Y8
CREATE DATA-TWO COMPRISING (THE PORTABLE MODULE'S SERIAL NUMBER, INCREMENTED TRANSACTION COUNTER, AND INCREASED MONETARY VALUE). ENCRYPT DATA-TWO USING A PRIVATE KEY.
Y10
RECEIVE ENCRYPTED DATA-TWO
Y11
RECEIVE ENCRYPTED DATA-TWO AND STORE IN MEMORY
Y12
INCREMENT TRANSACTION COUNTER
Y13

FIG. 5

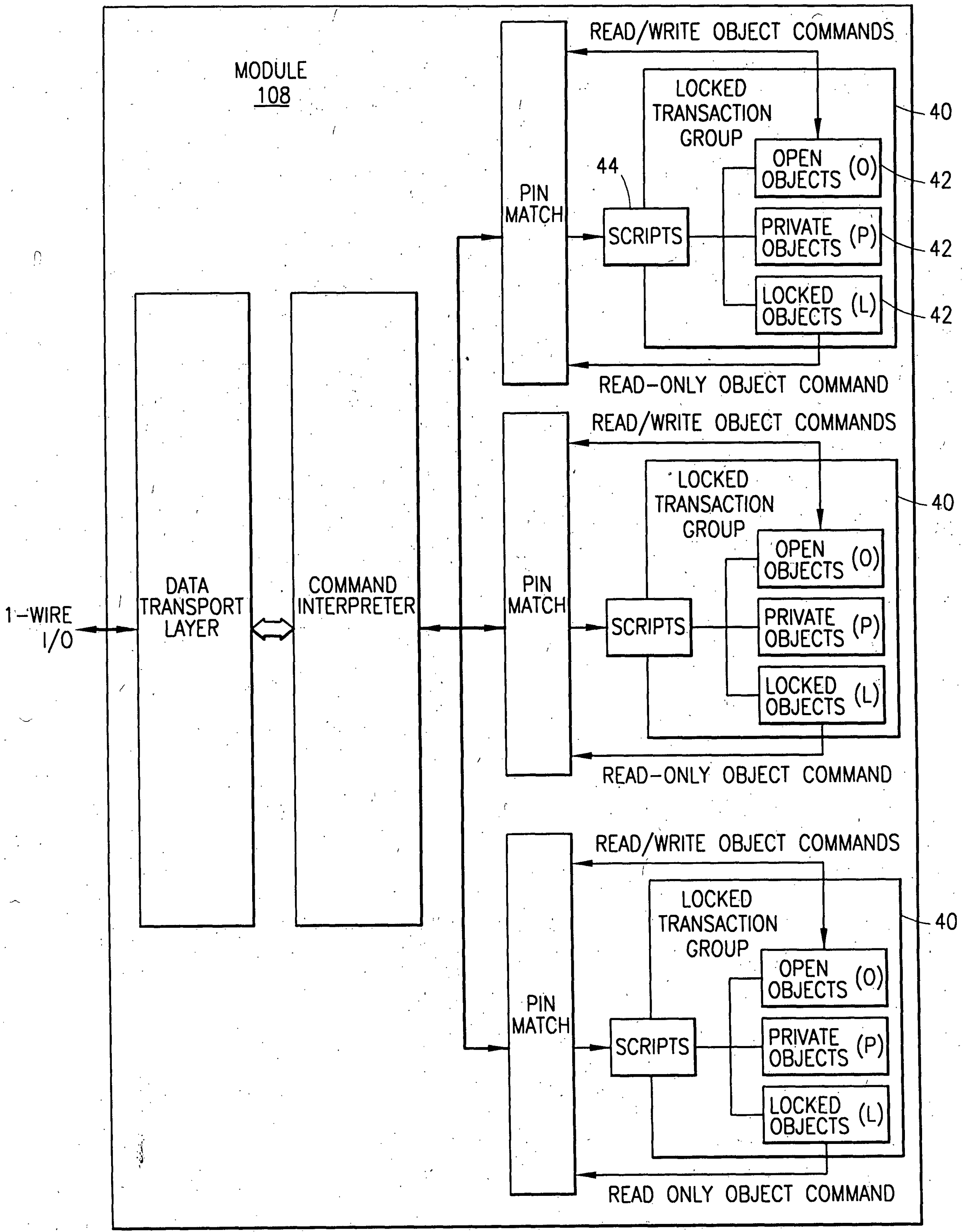

MODULE
108
1-WIRE I/O
DATA TRANSPORT LAYER
COMMAND INTERPRETER
PIN MATCH
SCRIPTS
44
LOCKED TRANSACTION GROUP
OPEN OBJECTS (O)
PRIVATE OBJECTS (P)
LOCKED OBJECTS (L)
40
42
42
42
READ/WRITE OBJECT COMMANDS
READ-ONLY OBJECT COMMAND
PIN MATCH
SCRIPTS
LOCKED TRANSACTION GROUP
OPEN OBJECTS (O)
PRIVATE OBJECTS (P)
LOCKED OBJECTS (L)
40
READ/WRITE OBJECT COMMANDS
READ-ONLY OBJECT COMMAND
PIN MATCH
SCRIPTS
LOCKED TRANSACTION GROUP
OPEN OBJECTS (O)
PRIVATE OBJECTS (P)
LOCKED OBJECTS (L)
40
READ/WRITE OBJECT COMMANDS
READ ONLY OBJECT COMMAND

FIG. 6

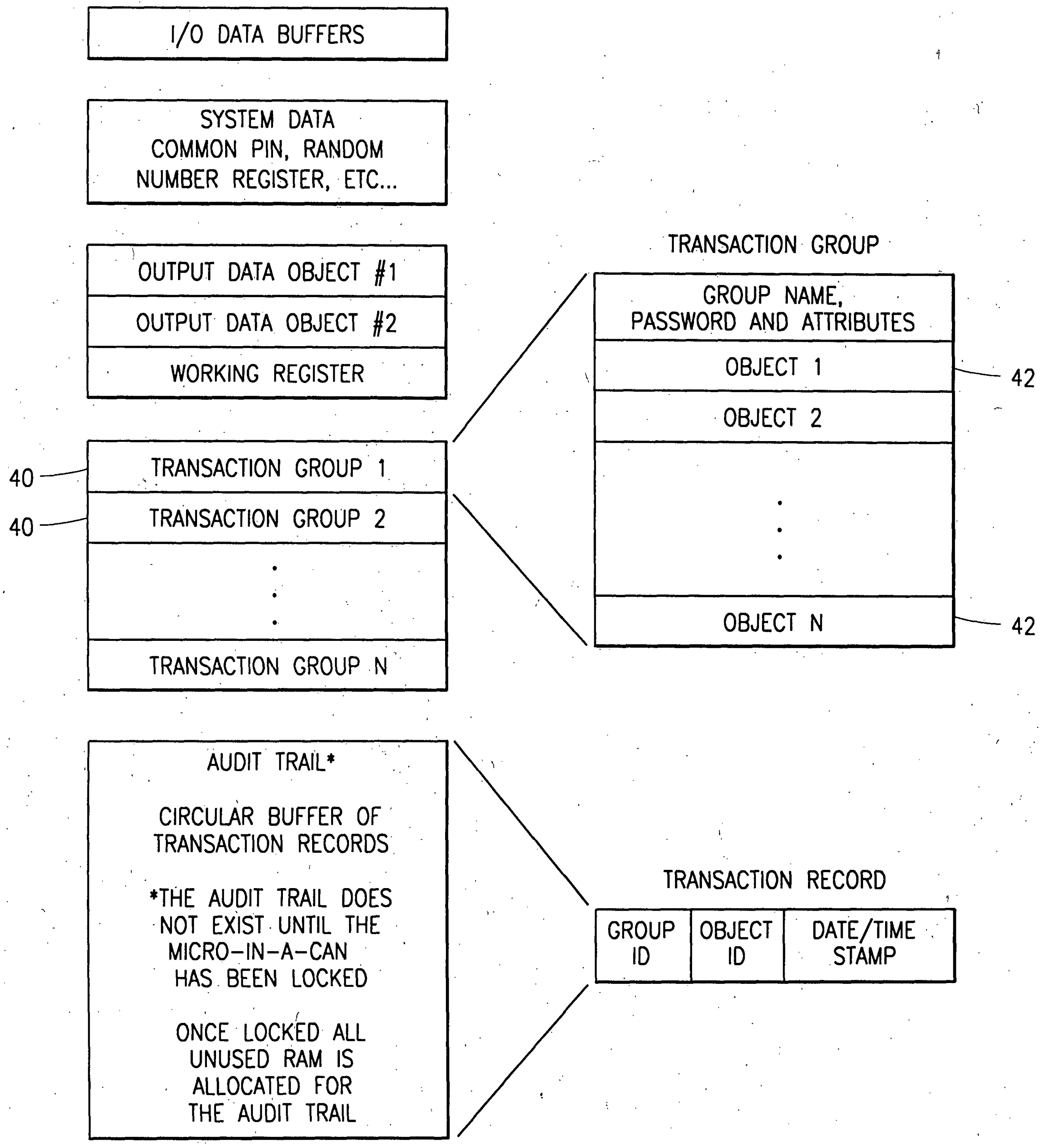

I/O DATA BUFFERS
SYSTEM DATA
COMMON PIN, RANDOM
NUMBER REGISTER, ETC...
OUTPUT DATA OBJECT #1
OUTPUT DATA OBJECT #2
WORKING REGISTER
40
TRANSACTION GROUP 1
40
TRANSACTION GROUP 2
TRANSACTION GROUP N
TRANSACTION GROUP
GROUP NAME,
PASSWORD AND ATTRIBUTES
OBJECT 1
42
OBJECT 2
OBJECT N
42
AUDIT TRAIL*
CIRCULAR BUFFER OF
TRANSACTION RECORDS
*THE AUDIT TRAIL DOES
NOT EXIST UNTIL THE
MICRO-IN-A-CAN
HAS BEEN LOCKED
ONCE LOCKED ALL
UNUSED RAM IS
ALLOCATED FOR
THE AUDIT TRAIL
TRANSACTION RECORD
GROUP ID
OBJECT ID
DATE/TIME STAMP

FIG. 7

#7

Jenkens & Gilchrist
A PROFESSIONAL CORPORATION

FOUNTAIN PLACE
1445 ROSS AVENUE, SUITE 3200
DALLAS, TX 75202

(214) 855-4500
TELECOPIER (214) 855-4300

AUSTIN, TEXAS
(512) 499-3800

HOUSTON, TEXAS
(713) 951-3300

SAN ANTONIO, TEXAS
(210) 308-3100

WASHINGTON, D.C.
(202) 326-1500

WRITER'S DIRECT DIAL NUMBER
Roger L. Maxwell
(214) 855-4787

Box Certificate of Correction
Assistant Commissioner
of Patents
Washington, D.C. 20231

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Box Certificate of Correction
Assistant Commissioner of Patents
Washington, D.C. 20231

on .... 20 October 1999 ....
Signature .... P. Guardiola ....
Printed Name P. Guardiola

Re:	Patent No.:	5,949,880
	Issued:	Sep. 7, 1999
	Title:	TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE
	Inventor:	Curry et al.

Dear Sir or Madam:

Transmitted for filing with the Patent and Trademark Office are the following documents for the above-referenced patent:

1. Request for Certificate of Correction of Patent to correct typographical errors in the patent, which does not introduce any new matter;
2. Form PTO-1050 (in duplicate); and
3. An acknowledgement postcard.

Please address all related communications to:

Roger L. Maxwell
Jenkens & Gilchrist, P.C.
1445 Ross Avenue, Suite 3200
Dallas, Texas 75202-2799

In the event there is an under- or over-payment, please debit or credit our Deposit Account #10-0447. This letter is being filed in duplicate to facilitate processing.

Very truly yours,

Roger L. Maxwell
Reg. No. 31,855

Jenkens & Gilchrist

A PROFESSIONAL CORPORATION

FOUNTAIN PLACE
1445 ROSS AVENUE, SUITE 3200
DALLAS, TX 75202

(214) 855-4500
TELECOPIER (214) 855-4300

AUSTIN, TEXAS
(512) 499-3800

HOUSTON, TEXAS
(713) 951-3300

SAN ANTONIO, TEXAS
(210) 308-3100

WASHINGTON, D.C.
(202) 326-1500

WRITER'S DIRECT DIAL NUMBER
Roger L. Maxwell
(214) 855-4787

Box Certificate of Correction
Assistant Commissioner
of Patents
Washington, D.C. 20231

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Box Certificate of Correction
Assistant Commissioner of Patents
Washington, D.C. 20231

on ......20 October 1999......
Signature ....P. Guardiola........
Printed Name P. Guardiola

Re:	Patent No.:	5,949,880
	Issued:	Sep. 7, 1999
	Title:	TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE
	Inventor:	Curry et al.

Dear Sir or Madam:

Transmitted for filing with the Patent and Trademark Office are the following documents for the above-referenced patent:

1. Request for Certificate of Correction of Patent to correct typographical errors in the patent, which does not introduce any new matter;
2. Form PTO-1050 (in duplicate); and
3. An acknowledgement postcard.

Please address all related communications to:

Roger L. Maxwell
Jenkens & Gilchrist, P.C.
1445 Ross Avenue, Suite 3200
Dallas, Texas 75202-2799

In the event there is an under- or over-payment, please debit or credit our Deposit Account #10-0447. This letter is being filed in duplicate to facilitate processing.

Very truly yours,

Roger L. Maxwell
Reg. No. 31,855

PATENT
Docket No. 20661-429D1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent Number:	5,949,880
Issued:	Sep. 7, 1999
Name of Patentee:	Curry et al.
Title of Invention:	TRANSFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MODULE

Box Certificate of Correction
Assistant Commissioner
of Patents
Washington, D.C. 20231

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Box Certificate of Correction
Assistant Commissioner of Patents
Washington, D.C. 20231

on 20 October 1999
Signature P. Guardiola
Printed Name P. Guardiola

Attention: Decision and Certificate of Correction Branch of the Patent Issue Division

REQUEST FOR CERTIFICATE OF CORRECTION OF PATENT
(37 CFR 1.322 (a))

Attached in duplicate is Form PTO-1050 with at least one copy being suitable for printing.

The exact location where the errors occur in the patent and where the matter appears correctly in the application file are:

Patent	Application File
Column 2, line 57	Page 8, line 1
Column 5, line 15	Page 16, line 15
Column 8, line 26	Page 28, line 15
Column 12, line 47	Page 49, line 1
Column 17, line 34	Page 65, line 8
Column 20, line 6	Page 74, line 5

Column 20, line 48	Page 76, line 9
Column 21, line 58	Page 80, line 10

The errors are printing errors by the Patent and Trademark Office and, accordingly, should be corrected without fee from applicant.

Please send the Certificate of Correction to:

Roger L. Maxwell
Jenkens & Gilchrist, P.C.
1445 Ross Avenue, Suite 3200
Dallas, Texas 75202-2799

Assignee: Dallas Semiconductor Corporation

Roger L. Maxwell
Assignee's Attorney
Reg. No. 31,855

/ X / Assignment recorded on
Reel/Frame 8029/0098 *et seq.*

/__/ Recordal of assignment attached

Staple Here Only !

PRINTER'S TRIM LINE

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,949,880
DATED : Sep. 7, 1999
INVENTOR(S) : Curry et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 2, line 57	Replace “electromagnetic” With --electro-magnetic--
Column 5, line 15	Before “information” Remove --is--
Column 8, line 26	Before “module” Remove --is--
Column 12, line 47	Replace “ERR BAD_PIN_LENGTH” With --ERR_BAD_PIN_LENGTH--
Column 17, line 34	Replace “ERR_BAD_OBJECT ID” With --ERR_BAD_OBJECT_ID--
Column 20, line 6	Replace “ERR MIAC_NOT_LOCKED” With --ERR_MIAC_NOT_LOCKED--
Column 20, line 48	Replace “ERR_BAD OBJECT_TYPE” With --ERR_BAD_OBJECT_TYPE--
Column 21, line 58	Replace “ERR_BAD NAME_LENGTH” With --ERR_BAD_NAME_LENGTH--

All checked
Transmittal needed

MAILING ADDRESS OF SENDER:
Roger L. Maxwell
1445 Ross Avenue
Suite 3200
Dallas, Texas 75202-2799

PATENT NO. 5,949,880

No. of add'l copies
@ 50¢ per page
1 of 1

20661-429D1

PATENT APPLICATION FEE DETERMINATION RECORD

Effective October 1, 1997

Application or Docket Number 08/978798

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)	SMALL ENTITY TYPE □		OR	OTHER THAN SMALL ENTITY	
FOR	NUMBER FILED	NUMBER EXTRA	RATE	FEE		RATE	FEE
BASIC FEE	▨	▨	▨	395.00	OR	▨	790.00
TOTAL CLAIMS	21 minus 20 =	* 1	x$11=		OR	x$22=	22.00
INDEPENDENT CLAIMS	2 minus 3 =	* ø	x41=		OR	x82=	
MULTIPLE DEPENDENT CLAIM PRESENT			+135=		OR	+270=	
			TOTAL		OR	TOTAL	22.00

* If the difference in column 1 is less than zero, enter "0" in column 2

CLAIMS AS AMENDED - PART II

		(Column 1)		(Column 2)	(Column 3)	SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
AMENDMENT A	▨	CLAIMS REMAINING AFTER AMENDMENT	▨	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
	Total	* 6	Minus	** 20	= —	x$11=		OR	x$22=	
	Independent	* 1	Minus	*** 3	= —	x41=		OR	x82=	
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM					+135=		OR	+270=	
						TOTAL ADDIT. FEE	—	OR	TOTAL ADDIT. FEE	—

		(Column 1)		(Column 2)	(Column 3)					
AMENDMENT B	▨	CLAIMS REMAINING AFTER AMENDMENT	▨	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
	Total	*	Minus	**	=	x$11=		OR	x$22=	
	Independent	*	Minus	***	=	x41=		OR	x82=	
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM					+135=		OR	+270=	
						TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

		(Column 1)		(Column 2)	(Column 3)					
AMENDMENT C	▨	CLAIMS REMAINING AFTER AMENDMENT	▨	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
	Total	*	Minus	**	=	x$11=		OR	x$22=	
	Independent	*	Minus	***	=	x41=		OR	x82=	
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM					+135=		OR	+270=	
						TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The "Highest Number Previously Paid F (Total or Independent) is the highest number found in t ppropriate box in column 1.

*U.S. Government Printing Office: 1997 - 430-571/69194

Patent & Trademark Office, U.S. DEPARTMENT OF COMMERCE

Form PTO 1130
(REV 2/94)

PACE DATA ENTRY CODING SHEET

U.S. DEPARTMENT OF COMMERCE
Patent and Trademark Office

1ST EXAMINER	DATE
2ND EXAMINER	DATE

APPLICATION NUMBER

TYPE APPL □

FILING DATE MONTH DAY YEAR

SPECIAL HANDLING □

GROUP ART UNIT

CLASS

SHEETS OF DRAWING

TOTAL CLAIMS

INDEPENDENT CLAIMS

SMALL ENTITY? □

FILING FEE

FOREIGN LICENSE □

ATTORNEY DOCKET NUMBER

CONTINUITY DATA

CONT CODE	STATUS CODE	PARENT APPLICATION SERIAL NUMBER	PCT APPLICATION SERIAL NUMBER	PARENT PATENT NUMBER	PARENT FILING DATE MONTH DAY YEAR
			P C T / /		
			P C T / /		
			P C T / /		
			P C T / /		
			P C T / /		

PCT/FOREIGN APPLICATION DATA

FOREIGN PRIORITY CLAIMED	COUNTRY CODE	PCT/FOREIGN APPLICATION SERIAL NUMBER	FOREIGN FILING DATE MONTH DAY YEAR

MPI Family Report (Family Bibliographic and Legal Status)

In the MPI Family report, all publication stages are collapsed into a single record, based on identical application data. The bibliographic information displayed in the collapsed record is taken from the latest publication.

Report Created Date: 2012-01-12

Name of Report:

Number of Families: 1

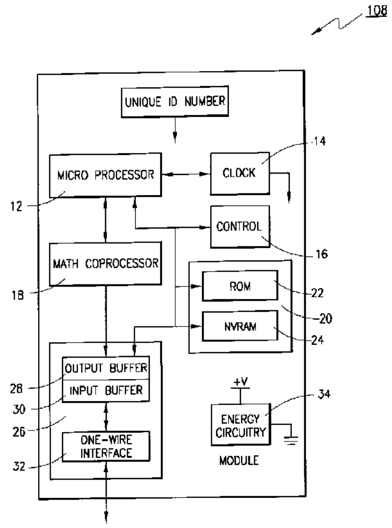
Comments:

Table of Contents

Family1

2 records in the family.

US5940510A 19990817



(ENG) Transfer of valuable information between a secure module and another module

Assignee: DALLAS SEMICONDUCTOR US

Inventor(s): CURRY STEPHEN M US ; LOOMIS DONALD W US ; BOLAN MICHAEL L US

Application No: US 59497596 A

Filing Date: 19960131

Issue/Publication Date: 19990817

Abstract: (ENG) The present invention rotates to system, apparatus and method for communicating valuable data from a portable module to another module via an electronic device. More specifically, the disclosed system, apparatus and method are useful for enabling a user to fill a portable module with a cash equivalent and to spend the cash equivalent at a variety of locations. The disclosed system incorporates an encryption/decryption method.

Priority Data: US 59497596 19960131 A Y;

IPC (International Class): G07F00710; G07F00708

ECLA (European Class): G07F00708C2B; G07F00710D4E

US Class: 705065; 705076; 713173

Publication Language: ENG

Filing Language: ENG

Agent(s): Jenkens & Gilchrist

Examiner Primary: Cangialosi, Salvatore

US Post Issuance:
--US Certificate of Correction: 20000222

Assignments Reported to USPTO:

Reel/Frame: 08029/0098 **Date Signed:** 19960416 **Date Recorded:** 19960506
Assignee: DALLAS SEMICONDUCTOR CORPORATION 4401 S. BELTWOOD PARKWAY DALLAS TEXAS 75244

Assignor: CURRY, STEPHEN M.; LOOMIS, DONALD W.; BOLAN, MICHAEL L.

Corres. Addr: JENKENS & GILCHRIST, P.C. STEVEN R. GREENFIELD, P.C 1445 ROSS AVENUE SUITE 3200 DALLAS, TX 75202-2799
Brief: ASSIGNMENT OF ASSIGNORS INTEREST (SEE DOCUMENT FOR DETAILS).

Reel/Frame: 21253/0637 **Date Signed:** 20080610 **Date Recorded:** 20080717
Assignee: MAXIM INTEGRATED PRODUCTS, INC. 120 SAN GABRIEL DRIVE SUNNYVALE CALIFORNIA 94086

Assignor: DALLAS SEMICONDUCTOR CORPORATION

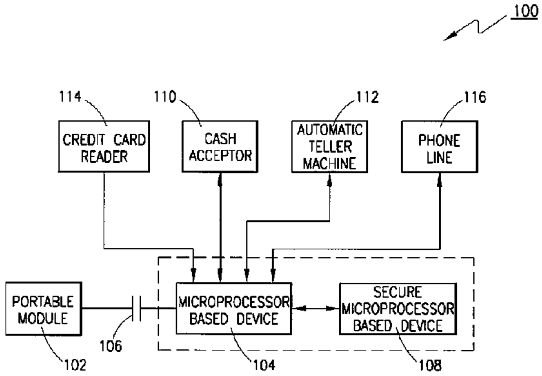
Corres. Addr: NORTH WEBER & BAUGH LLP ATTN: MICHAEL V. NORTH 2479 E. BAYSHORE
RD, SUITE 707 PALO ALTO, CA 94303

Brief: MERGER

Legal Status:

Date	+/-	Code	Description
19960506	()	AS	New owner name: DALLAS SEMICONDUCTOR CORPORATION, TEXAS; : ASSIGNMENT OF ASSIGNORS INTEREST;ASSIGNORS:CURRY, STEPHEN M.;LOOMIS, DONALD W.;BOLAN, MICHAEL L.;REEL/FRAME:008029/0098;SIGNING DATES FROM 19960416 TO 19960418;
20000222	( )	CC	CERTIFICATE OF CORRECTION
20021220	()	FPAY	Year of fee payment: 4;
20070302	()	FPAY	Year of fee payment: 8;
20070302	()	SULP	Year of fee payment: 7;
20080307 20070310	()	AS REMI	New owner name: MAXIM INTEGRATED PRODUCTS, INC., CALIFORNIA; : MERGER;ASSIGNOR:DALLAS SEMICONDUCTOR CORPORATION;REEL/FRAME:021253/0637; Effective date: 20080610;
20110321	()	REMI	

US5949880A 19990907



(ENG) Transfer of valuable information between a secure module and another module

Assignee: DALLAS SEMICONDUCTOR US

Inventor(s): CURRY STEPHEN M US ; LOOMIS DONALD W US ; BOLAN MICHAEL L US

Application No: US 97879897 A

Filing Date: 19971126

Issue/Publication Date: 19990907

Abstract: (ENG) The present invention relates to system, apparatus and method for communicating valuable data from a portable module to another module via an electronic device. More specifically, the disclosed system, apparatus and method are useful for enabling a user to fill a portable module with a cash equivalent and to spend the cash equivalent at a variety of locations. The disclosed system incorporates an encryption/decryption method.

Priority Data: US 97879897 19971126 A N; US 59497596 19960131 A 3 Y;

Related Application(s): 08/594975 19960131 US PENDING

IPC (International Class): G07F00710; G07F00708

ECLA (European Class): G07F00708C2B; G07F00710D4E

US Class: 705066; 705039; 705042; 705065

Publication Language: ENG

Filing Language: ENG

Agent(s): Jenkens & Gilchrist

Examiner Primary: Tarcza, Thomas H.

Examiner Assistant: White, Carmen D.

US Post Issuance:
--US Certificate of Correction: 20000425 20000425 a Certificate of Correction was issued for this patent

Assignments Reported to USPTO:

Reel/Frame: 06462/0935 **Date Signed:** 19930315 **Date Recorded:** 19930316
Assignee: MIDAS REX PNEUMATIC TOOLS, INC. 3001 RACE STREET FORT WORTH TEXAS 76111

Assignor: BARBER, FOREST C., JR., EXECUTOR OF ESTATE OF FOREST C. BARBER, M.D.; BARRETT, CARON HELEN
BARRETT, CARON HELEN I., EXECUTORS OF ESTATE OF FOREST C. BARBER, M.D.

Corres. Addr: JAMES E. BRADLEY FELSMAN, BARDLEY, GUNTER & DILLON, LLP 2600 CONTINENTAL PLAZA 777 MAIN STREET FORT WORTH, TX 76102
Brief: ASSIGNMENT OF ASSIGNORS INTEREST.

Reel/Frame: 08847/0336 **Date Signed:** 19971110 **Date Recorded:** 19971124
Assignee: MURATA MANUFACTURING CO., LTD. NAGAOKAKYO-SHI 26-10, 2-CHOME, TENJIN KYOTO 617 JAPAN

Assignor: SHIMOE, KAZUNOBU

Corres. Addr: GRAHAM & JAMES LLP ALBERT L. JACOBS, JR. INTELLECTUAL PROPERTY GROUP 885 THIRD AVENUE NEW YORK, NY 10022
Brief: ASSIGNMENT OF ASSIGNORS INTEREST (SEE DOCUMENT FOR DETAILS).

Reel/Frame: 21253/0637 **Date Signed:** 20080610 **Date Recorded:** 20080717
Assignee: MAXIM INTEGRATED PRODUCTS, INC. 120 SAN GABRIEL DRIVE SUNNYVALE CALIFORNIA 94086

Assignor: DALLAS SEMICONDUCTOR CORPORATION

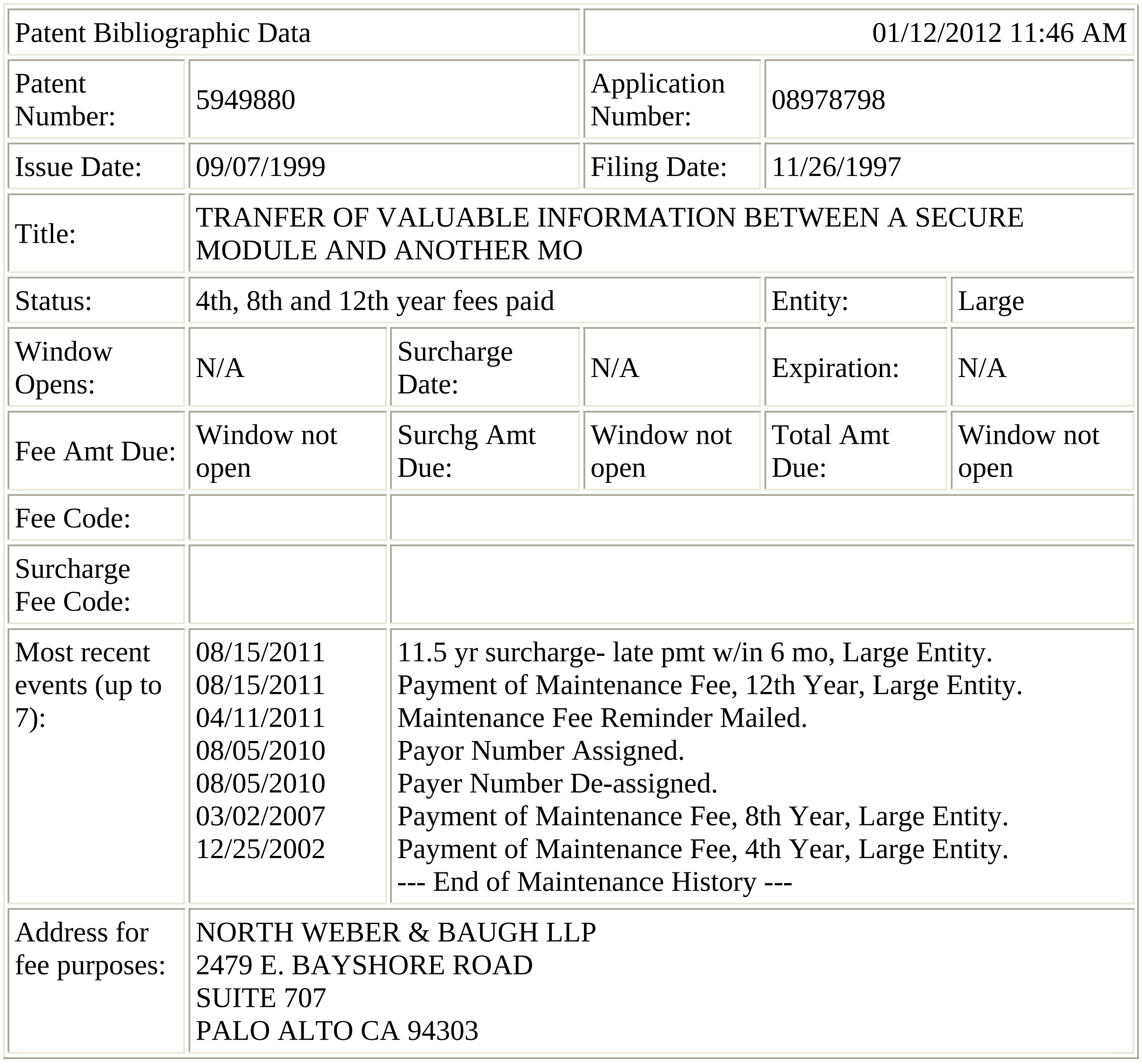
Corres. Addr: NORTH WEBER & BAUGH LLP ATTN: MICHAEL V. NORTH 2479 E. BAYSHORE RD, SUITE 707 PALO ALTO, CA 94303
Brief: MERGER

Legal Status:

Date	**+/-**	**Code**	**Description**
19930316	()	AS	New owner name: MIDAS REX PNEUMATIC TOOLS, INC., TEXAS; : ASSIGNMENT OF ASSIGNORS INTEREST.;ASSIGNORS:BARBER, FOREST C., JR., EXECUTOR OF ESTATE OF FOREST C.BARBER, M.D.;BARRETT, CARON HELEN I., EXECUTORS OF ESTATE OF FOREST C. BARBER, M.D.;REEL/FRAME:006462/0935; Effective date: 19930315;
19971124	()	AS	New owner name: MURATA MANUFACTURING CO., LTD., JAPAN; : ASSIGNMENT OF ASSIGNORS INTEREST;ASSIGNOR:SHIMOE, KAZUNOBU;REEL/FRAME:008847/0336; Effective date: 19971110;

20000425	( )	CC	CERTIFICATE OF CORRECTION
20021225	()	FPAY	Year of fee payment: 4;
20070302	()	FPAY	Year of fee payment: 8;
20080717	()	AS	New owner name: MAXIM INTEGRATED PRODUCTS, INC., CALIFORNIA; : MERGER;ASSIGNOR:DALLAS SEMICONDUCTOR CORPORATION;REEL/FRAME:021253/0637; Effective date: 20080610;
20110411	()	REMI	

USPTO Maintenance Report



Patent Bibliographic Data			01/12/2012 11:46 AM		
Patent Number:	5949880		Application Number:	08978798	
Issue Date:	09/07/1999		Filing Date:	11/26/1997	
Title:	TRANFER OF VALUABLE INFORMATION BETWEEN A SECURE MODULE AND ANOTHER MO				
Status:	4th, 8th and 12th year fees paid			Entity:	Large
Window Opens:	N/A	Surcharge Date:	N/A	Expiration:	N/A
Fee Amt Due:	Window not open	Surchg Amt Due:	Window not open	Total Amt Due:	Window not open
Fee Code:					
Surcharge Fee Code:					
Most recent events (up to 7):	08/15/2011 08/15/2011 04/11/2011 08/05/2010 08/05/2010 03/02/2007 12/25/2002	11.5 yr surcharge- late pmt w/in 6 mo, Large Entity. Payment of Maintenance Fee, 12th Year, Large Entity. Maintenance Fee Reminder Mailed. Payor Number Assigned. Payer Number De-assigned. Payment of Maintenance Fee, 8th Year, Large Entity. Payment of Maintenance Fee, 4th Year, Large Entity. --- End of Maintenance History ---			
Address for fee purposes:	NORTH WEBER & BAUGH LLP 2479 E. BAYSHORE ROAD SUITE 707 PALO ALTO CA 94303				