Un nuovo insetto vola nell'aria del Satellitare.
== Le Mosche ==
By LeonePrimo

L'origine di queste note, nasce dalla constatazione della necessita' di dare dei punti fermi, sul piano informativo, al lavoro svolto dal Picc2rd Team.
Puo' sembrare strano, ma ci sono ancora molte persone che confondono la Piccard2 con la MoscAS.
Nel contempo, si cerchera', nel caso della MoscAS, di proporre, sia la traduzione del manualetto a corredo dello zip di distribuzione, che delle indicazioni sulla programmazione della MoscAS in particolare.

Gli amici spagnoli del Picc2rd team, stanno svolgendo un lavoro notevole. Le loro realizzazioni piu' conosciute ed usate, sono sicuramente:

- **Piccard2**, una wafer con Pic 16F876 ed eventualmente con una eeprom esterna (dalla 24C32 in su), con installato il loro firmware Culluh (giunto al momento della preparazione delle presenti note alla versione 2.0), corredata di 4 led's indicatori dello stato operativo della scheda;
- **MoscAS (o Mosche)**, utilizza lo stesso hardware dell Piccard2, con Pic 16F876 ed eventualmente con una eeprom esterna (dalla 24C32 in su), (giunta al momento della preparazione delle presenti note alla versione 1.1). Anche la MoscAS, e' corredata di 4 led's indicatori dello stato operativo della scheda;
- **MatrixStudio** (giunto al momento della preparazione delle presenti note alla versione 6.2) un ambiente di "sviluppo" vero e proprio, capace di dialogare, in ogni momento con gli elementi delle realizzazioni del Picc2rd Team. A seguito di una semplicissima configurazione, e' possibile collegarlo, attraverso le porte seriali di un PC, sia ad un programmatore di PIC ed EEPROM, del tipo LudiPipo o Multipipo, che ad un programmatore di Carte, tipo Phoenix, con quarzo a 3.57 MHz.

Essendo queste note, dedicate essenzialmente alla MoscAS, cominciamo a parlarne piu' dettagliatamente.
La MoscAS acronimo di Mosc Advanced Simulation e' un progetto che ha come suo obiettivo la realizzazione di una scheda che, con il suo hardware e con il firmware proposto, sia un clone, in pratica simuli perfettamente il comportamento di una carta originale SEGA.
Notevoli, sono le motivazioni che hanno spinto gli amici spagnoli a questa realizzazione: dare la possibilita' a tutti coloro che sono interessati, di studiare piu' approfonditamente, di fare pratica in proprio su una carta SEGA, senza far correre rischi alla Card originale. Questo significa aprire il mondo SEGA, ad una piu' approfondita e ragionata conoscenza.
D'altra parte chi non condividesse queste motivazioni, potrebbe sempre rivolgersi alla Piccard2, giunta ormai alla quasi perfezione, sul piano tecnico.
Qui di seguito viene riportata la traduzione in italiano, del manuale della MoscAS, con l'indicazione delle caratteristiche tecniche, nel dettaglio e con delle prime indicazioni sulla programmazione.

MoscAS 1.0
Manuale dell'Utente

Caratteristiche Principali

- Testato su ASTON cam e Goldbox (di varie marche)
- Verifica dei dati esistenti: MK's, PPUA, Termination Date, Operation Keys e RegionalByte per evitare riscritture nella eeprom
- Allocazione automatica dello spazio nell'EEPROM alla ricezione delle chiavi
- Gestione dei comandi per il Provider 00 SE*A
- Totale gestione e Controllo del PBM
- Autoupdate, le chiavi si aggiornano automaticamente
- Supporta la SuperEncriptyon con PK+SK
- Controllo e verifica della firma (signature) nei messaggi EMM e EMC
- Blocco dei canali con codice d'accesso modificabile
- Comunicazione all'utente mediante diodi led's di:
    - Elaborazione di EMC (Led 0).
    - Chiave Falsa / Segnalazione di INS errate (3c., 40.. etc) (Led 1)
    - Ricezione di comando non supportato (Led 2)
    - Aggiornamento dell Chiavi (Led 3)
- Modo depurazione. Aggiornamento di tutti i parametri del firmware tramite l'interfaccia SmartCard
- Configurazione del tipo di diodi led's utilizzati.
- Supporta i dati dei providers in eeprom esterna o nella propria Flash (Modo Single Chip)
- Autorilevazione della eeprom esterna
- Invio del traffico con la scheda per RC6/TX
- Backdoor nel SE*A con MK00p (00 00 00 00 00 00 00 00)
- All'inserimento della scheda, durante l'ATR si accendono tutti i led's (TEST)
- AutoPPUA nel caso che la selezionata per il CUSTWP-BITMAP non e' attiva.
- Il file MoscAS v1.0 e' configurato di default per lavorare con flash-eeprom interna e con i led's configurati con catodo comune (il primo byte dell'eeprom del pic settato a 02)
- Ottimizzazione generale del codice e raggruppamento delle funzioni di aggiornamento della eeprom.
- Aggiunto supporto per nuove istruzioni non documentate:
  02, 04, 48, 5C, 36, 38, 42, 4A, 4C, 50, 54, 56, 7C, 8A
- Migliorata la risposta delle ins. 3C, 3A, 16 e 12
- Aggiunta memorizzazione di eventi (PPV). In questa maniera, una volta selezionato un canale di questo tipo, non sara' necessaria una nuova autorizzazione. Questo permette di cambiare rapidamente da un canale all'altro
- Eliminata l'opzione della cattura dei logs e sostituita la opzione per la SIMULAZIONE DEL PPV.
- Supportati nuovi nano delle ins. 40 e 3C: 03, 17, 23, 24, 26, 01, 02, 19
- Modificate le risposte di alcuni nano, sia con aggiornamento o no di alcuni dati, ad esempio il 41 o l'80. Sono state implementate anche le risposte del tipo 97 xx (aggiornamento NANO xx non necessario).
- Corretto il nano D0
- Gestione dei records

- Riprodotto il bug 3c / 3a cosi' che il metodo Pippo funziona perfettamente con questa simulazione.

INS 40 Nano implementati nella MoscAS 1.0

SE*A NANO 23
Crea provider
[23] d1 d2

Crea il provider specificato con d1 d2

SE*A NANO 24
Usa Provider
[24] d1 d2

Se esite ed e' abilitato resta attaccato per i no_SE*A_nanos che lo seguono

SE*A NANO 26
Abilita la modifica dei registri
[26] d1 d2

Abilita/Disabilita la scrittura dei registri

NANO F0
Segue l'F0-bitmap
[F0] d1 d2 ... d32

Se l'ultimo byte della ppua (custwp_byte) e' adeguato continua l'elaborazione. In caso contrario restituisce 9009

NANO 17
Codice di Regione
[17] d1

NANO 03
Annulla il PIN
[03]

Pone i 4 bytes del pin zero nella eeprom

NANO 80
Crea PBM record 9x
[80] d1 d2 d3 d4 d5 d6 d7 d8

- Se disabilitato il nano_80 restituisce 9015
- In caso contrario crea/modifica il registro PBM 9x lasciandolo:

d1 d2 d3 d4 d5 d6 d7 d8 00 00 00 9x
in cui d1..d8 e' il PBM y x il provider

NANO 90
Crea chiave primaria 8x
[90] ik k1 k2 k3 k4 k5 k6 k7 k8

Segue l'indice della chiave e la chiave primaria cifrata

NANO 91
Crea la chiave secondaria 9x
[91] ik k1 k2 k3 k4 k5 k6 k7 k8

Segue l'indice della chiave e la chiave secondaria cifrata

NANO 10
Cancella la chiave 8x,Cx
[10] ik

Cancella la chiave dell'indice ik primaria e secondaria se esistono

NANO 41
Crea PPUA
[41] d1 d2 d3 d4

Si scrive la ppua

NANO 01
Disabilita il nano 24
[01]

Disabilita il nano_24 per il provider in uso

NANO 02
Abilita il nano 24
[02]

Abilita il nano_24 per il provider in uso

NANO D0
Prov_ID string
[D0] d1..d16

Scrive il ProviderID String

NANO B0
Crea SE*A Record Ex
[B0] d1..d11

Scrive un record SE*A con indice d1

NANO 40
Cancella Bx range
[40] d1 d2 d3 d4

Cancella i records Bx compresi tra [d1d2 d3d4]

Programmazione di MoscAs 1.0

Di default, MoscAS 1.0 e' fornito del Provider SE*A attivo ed una backdoor (MK00 primario SE*A 00 00 00 00 00 00 00 00)
Con questa backdoor e vari nano del sistema SE*A MEDIAGU*RD, è possibile creare, modificare e cancellare ogni tipo di record.

Il primo passo da compiere è inserire il vostro numero di serie (UA), modificabile per l'INS protetto 0C Es.:
C1 0C 00 00 08 XX XX 00 00 00 XX XX XX

Nota importante!!!
L'INS 0C è protetta, e funziona nello stesso modo delle ins 20, 22, 24, 26, 28 e pertanto può utilizzarsi solamente in modo depurazione.

Per utilizzare questa INS in forma automatica e' possibile utilizzare l'ultima versione di MatrixStudio 6.2 che la genera nella sezione "Mini MOSC" in relazione al numero di serie introdotto nella sezione "Providers".

Di seguito dettagliamo la procedura, nel caso in cui si volesse reallizzarlo manualmente tutto ma, per comodita', possono usarsi programmi per MOSC come CAMELMOSC, SEKABLASTER, MKFIND, SECAMOSC, MATRIXSTUDIO.....

LL è il numero (in hex) di ottetti compresi dopo LL fino a 82 compreso + gli 8 ottetti della firma.

SignMK0 è la firma della backdoor conosciuta.

- Creazione di un nuovo fornitore pp pp:
C1 40 00 00 LL 23 PP PP 82 SIGNMK0

- Aggiungere il nome del fornitore:
C1 40 00 00 LL 24 PP PP D0 NN NN NN NN NN NN NN NN NN NN NN NN NN NN NN NN 82 SIGNMK0

Es.: per il fornitore 00 04 (CANALSATELLITE), inserire invece di nn:
43 41 4E 41 4C 53 41 54 45 4C 4C 49 54 45 20 20
C A N A L S A T E L L I T E

Es.: per il fornitore 00 03 (Canale +), inserire:
43 41 4E 41 4C 2B 20 20 20 20 20 20 20 20 20 20
C A N A L +

- Autorizzazione (per abilitare scrittura PBM -> FF FF)
C1 40 00 00 LL 24 PP PP 26 FF FF 82 SIGNMK0

- Inserire PPUA:
C1 40 00 00 LL 24 PP PP 41 UU UU UU UU 82 SIGNMK0

Es.: canalsatellite: 00 1E D9 01

CANALE +: 00 10 F3 01
biglietteria: 00 07 29 01
TELEPIÙ: 00 12 B4 E4

- Inserire MK01:
m1 m2 m3 m4 m5 m6 m7 m8 criptata con MK0 si trasforma in n1 n2 n3 n4 n5 n6 n7 n8
C1 40 00 00 LL 24 PP PP 90 51 N1 N2 N3 N4 N5 N6 N7 N8 82 SIGNMK0

- Inserire codice regionale + bitmap fornitore:
C1 40 00 02 LL 24 PP PP 17 02 80 XX XX XX XX XX XX XX XX 82 SIGNMK0

Es.: canalsatellite: 00 10 10 80 08 00 00 5E
canale +: 00 00 00 00 00 00 00 02

- Inserzione della data (max FF FF):
C1 40 00 00 LL 24 PP PP 21 FF FF 82 SIGNMK0

- Attivazione della carta:
C1 40 00 00 LL B0 01 00 00 00 00 00 00 00 02 00 93 00 00 00 82 SIGNMK0

Logicamente se la MK1 e la PPUA del fornitore sono valide, disponiamo di autoupdate.

Se volete introdurre manualmente le chiavi:

- per la chiave 0C:
C1 40 00 00 LL 24 PP PP 90 5C A1 A2 A3 A4 A5 A6 A7 A8 82 SIGNMK0

- per la chiave 0D:
C1 40 00 00 LL 24 PP PP 90 5D A1 A2 A3 A4 A5 A6 A7 A8 82 SIGNMK0

- per la chiave 0E:
C1 40 00 00 LL 24 PP PP 90 5E A1 A2 A3 A4 A5 A6 A7 A8 82 SIGNMK0

a1 a2 a3 a4 a5 a6 a7 a8 è sono le key 0C, 0D o 0E criptate con MK0.

Ovviamente questa è una mini guida per divertirsi ed imparare, sul sistema SE*A MEDIGU*ARD, poiché è completamente proibita la visione di canali a pagamento senza abbonamento, in tutta la CE.

MoscAS Verso MatrixStudio e Mkfind

Veniamo ora a discutere su come realizzare una MoscAS, utilizzando MatrixStudio 6.2, per la programmazione e Mkfind 4.1, per verificare la riuscita delle operazioni con MatrixStudio.
In queste note, si realizzera' un MoscAS, senza Eeprom esterna. Si aggiungera' al provider SEGA esistente, un provider operativo (es. Tele-), con tutti i suoi parametri, in grado di aggiornare le chiavi e la data. Alla fine di questa lettura, ognuno sara' in grado di decidere su come configurare definitivamente la propria realizzazione.
Vediamo qualche schermata di MatrixStudio 6.2, per poter parlare, visualizzando qualcosa.
Ecco, in Fig. 1, allora la sezione "Config" di MatrixStudio 6.2 (ognuno, dovra' apportare le modifiche proprie): "Port interface", "Port Logger/JDM" ecc.

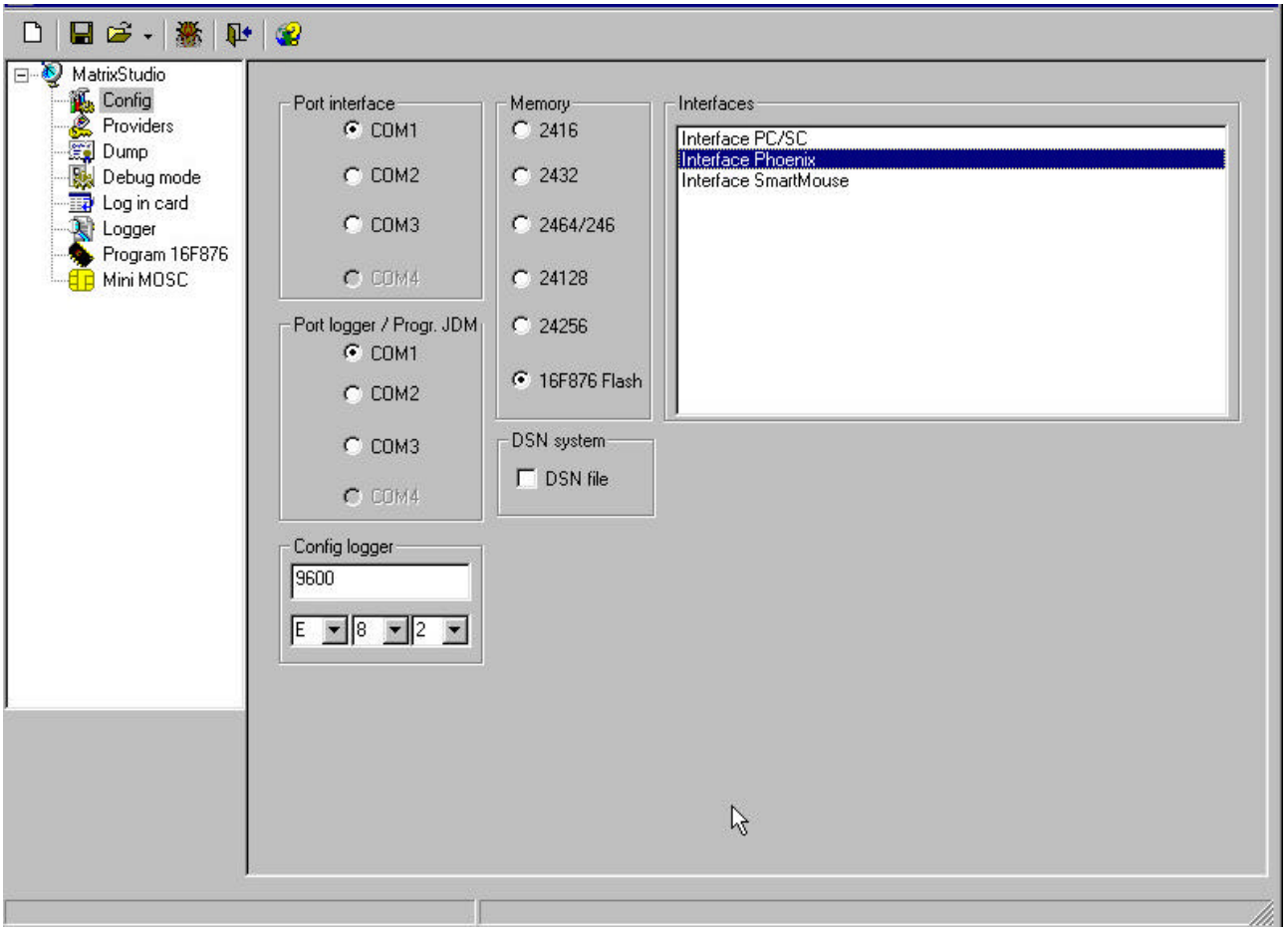

Fig. 1

Andiamo alla Sezione "provider". Il primo provider SEGA, in Fig.2, dovra' avere questa configurazione:

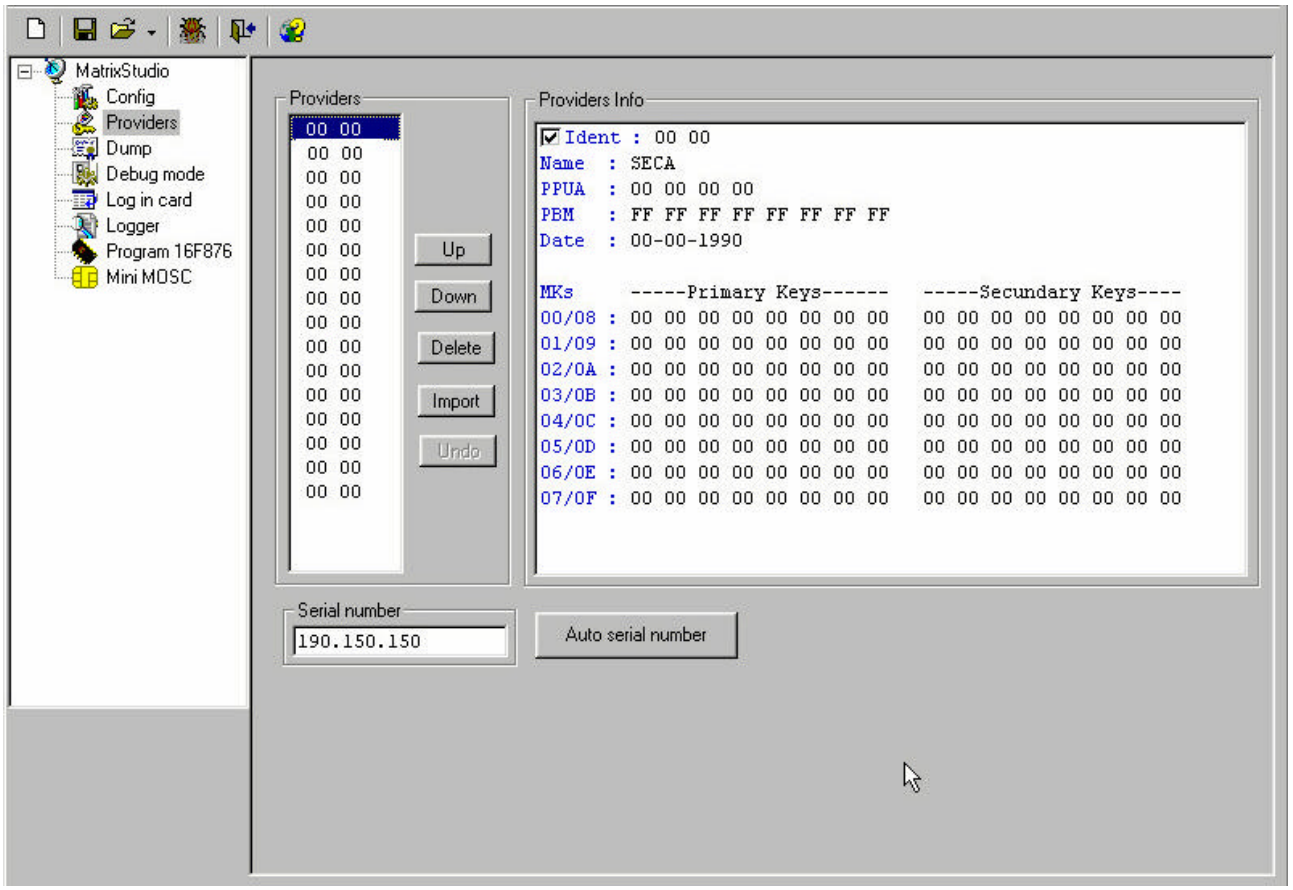

Fig.2

Si dovranno editare i campi che dovranno essere modificati per avere una pagina simile a quella mostrata.
Si passa, quindi, al nostro primo provider operativo, andando con il mouse sul secondo provider dell'elenco, che nel caso di prima editazione sara' ancora 00 00
Ed ecco, in Fig. 3, cosa si dovra' visualizzare ed editare:

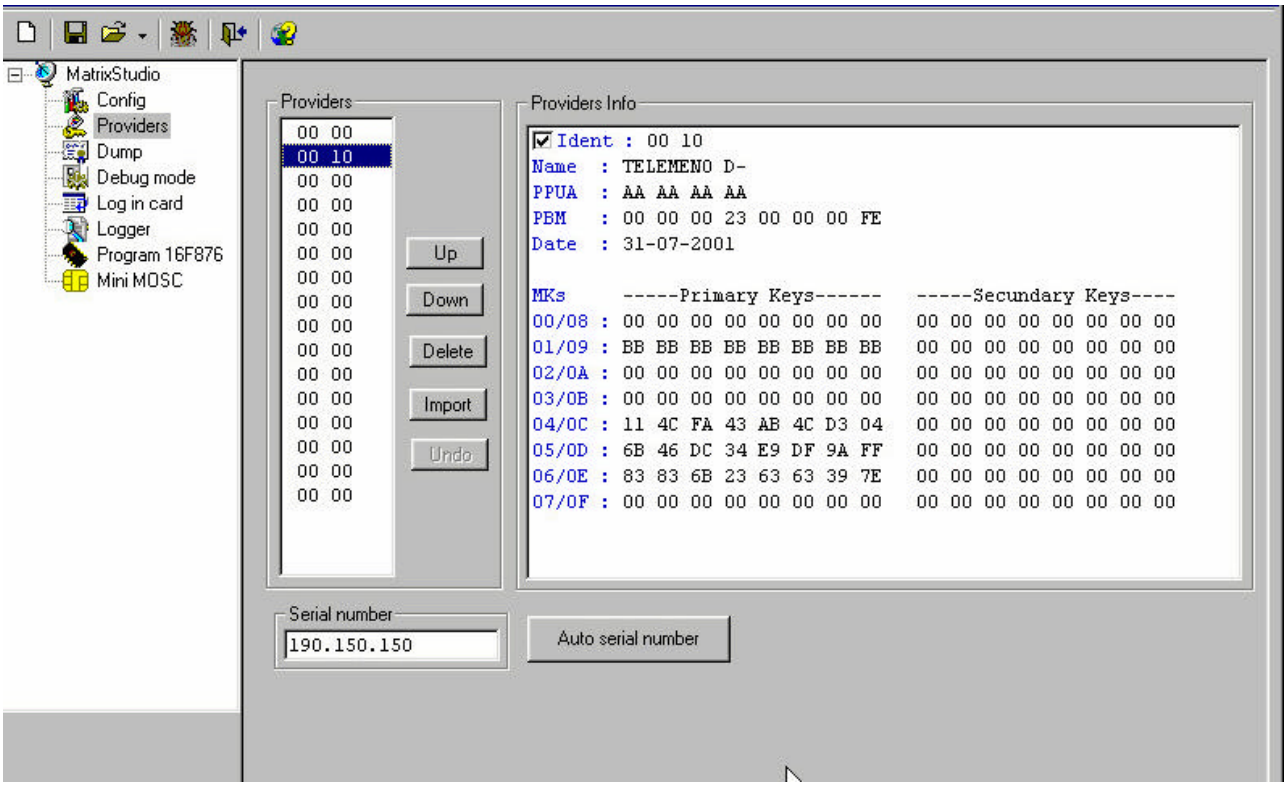

Fig. 3

Chiaramente, per quanto riguarda la PPUA, al posto delle AA, andra' inserita la PPUA del proprio abbonamento e, al posto delle BB, alla MKs 01/09, si dovra' inserire la propria. L'ident dovra' essere 00 10.
Per quanto riguarda le chiavi 0C, 0D e 0E, quelle riportate erano quelle valide al momento della stesura del presente documento. Lo stesso vale per il Date. Si ricorda ancora una volta che, inserendo PPUA e MK1 valide, le chiavi e la data si aggiorneranno automaticamente.
Il serial number mostrato e' sicuramente valido, se lo si vuole cambiare, bastera' clickare sul bottone "Auto serial number".
Una volta completata l'editazione di tutti i campi, si salvino questi provider in un file con il nome che si vuole, ad esempio "MoscTest.hex", sara' utile in un secondo momento.

Ora si passi a programmare il PIC 16F876, con il file della MoscAS. Si vada alla sezione "Program 16F876" e, si colleghi il MultiPipo alla seriale, come da configurazione. Si carichi, con il "load File" della sezione, il file "Moscas10, della versione piu' aggiornata (al momento della stesura del presente documento era la 1.1), si smarchi l'opzione "All" nel settore "Read/Write/Verify".

La videata che si avra', sara' quella di Fig. 4:

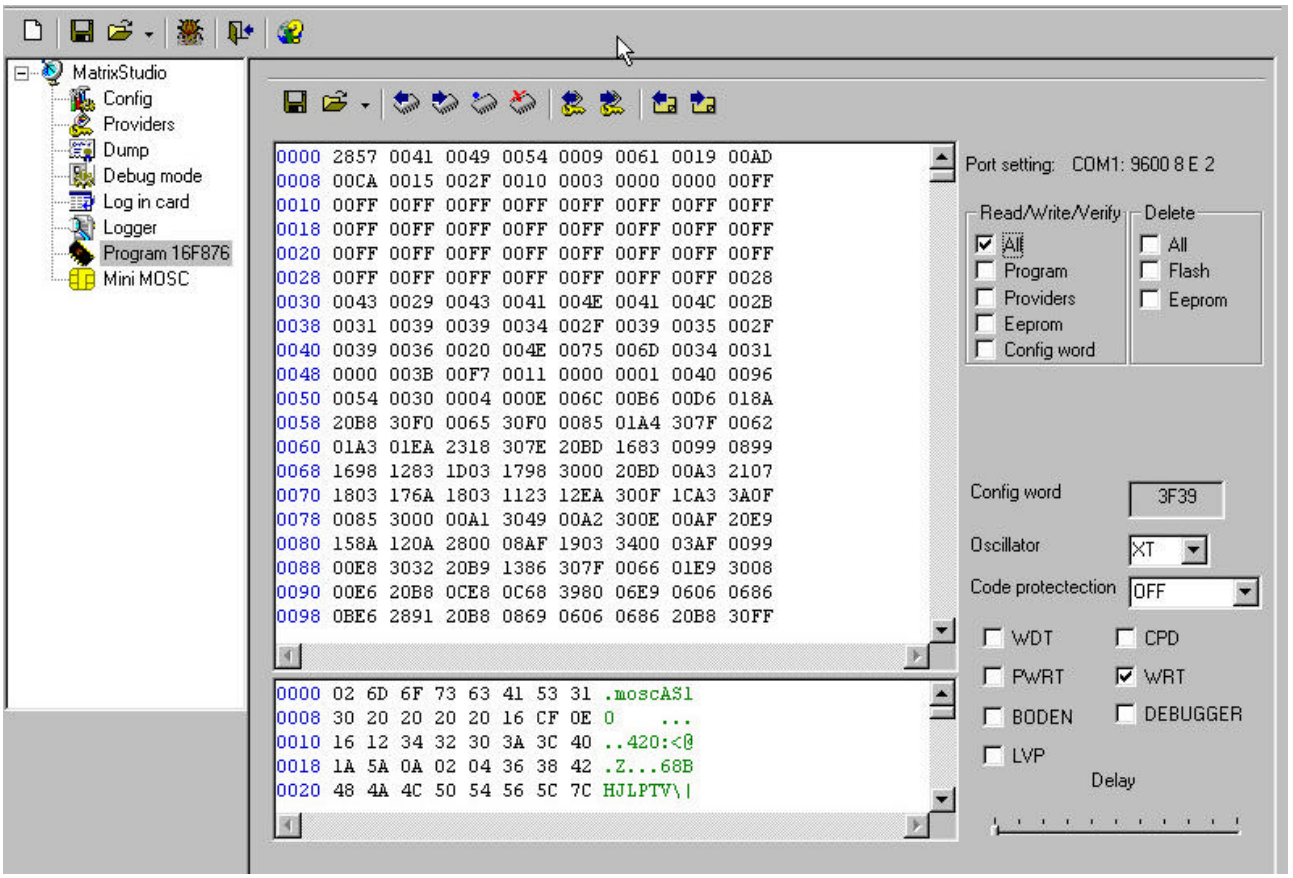

Fig. 4

A questo punto, clickando sul bottone “Write”, si programmera’ il 16F876.

Se durante la programmazione, o durante la verifica, vengono fuori messaggi d’errore, ignorarli. MatrixStudio programma perfettamente in barba ai messaggi d’errore (verifica fatta con IC Prog 1.04).

Ora si ha una wafer con il 16F876, programmato con il file .Hex della MoscAs, e’ giunto il momento di metterci il nostro provider.
Si torni alla Sezione “Provider” e si carichi il file “MoscTest.hex”, salvato in precedenza. Quindi, si vada alla sezione “Mini MOSC”. Si colleghi alla seriale, come da configurazione, il Phoenix con quarzo da 3.57 MHz, e si infili la wafer nel Phoenix.

Ricordiamo quanto detto nel manualetto della MoscAS:
Il primo passo da compiere è inserire il vostro numero di serie (UA). Il comando relativo sara’ del tipo: C1 0C 00 00 08 XX XX 00 00 00 XX XX XX

Ebbene, avendo smarcato, nel settore providers, l’opzione “Update”, si clicki sul bottone “Change SN”. Nel campo “Ins”, comparira’ una riga simile a quella necessaria, ovvero c1 0c 00 00 08 002500000B557606 (che riporta, oltre al comando di scrittura, il numero seriale “190.150.150” (che si era salvato in precedenza nel file in “MoscTest.hex”) in esadecimale “B557606”.
Si clicki sul bottone “Send INS” e dopo qualche istante, sulla riga di stato, comparira’ un 90 00, che e’ il messaggio di scrittura andata a buon fine.

A questo punto e' possibile fare un primo controllo di quello che si e' fatto, utilizzando Mkfind 4.1. (e' possibile utilizzarlo, senza uscire da MatrixStudio, purche' ogni volta che si termina di utilizzarlo, si dia il comando di disconnessione della carta).
Si lanci, quindi, Mkfind 4.1, configurato sulla giusta seriale, si clicki sul bottone "Connect/Disconnect" , quindi su "Get Card Info" e comparira' una schermata simile a quella mostrata in Fig. 5

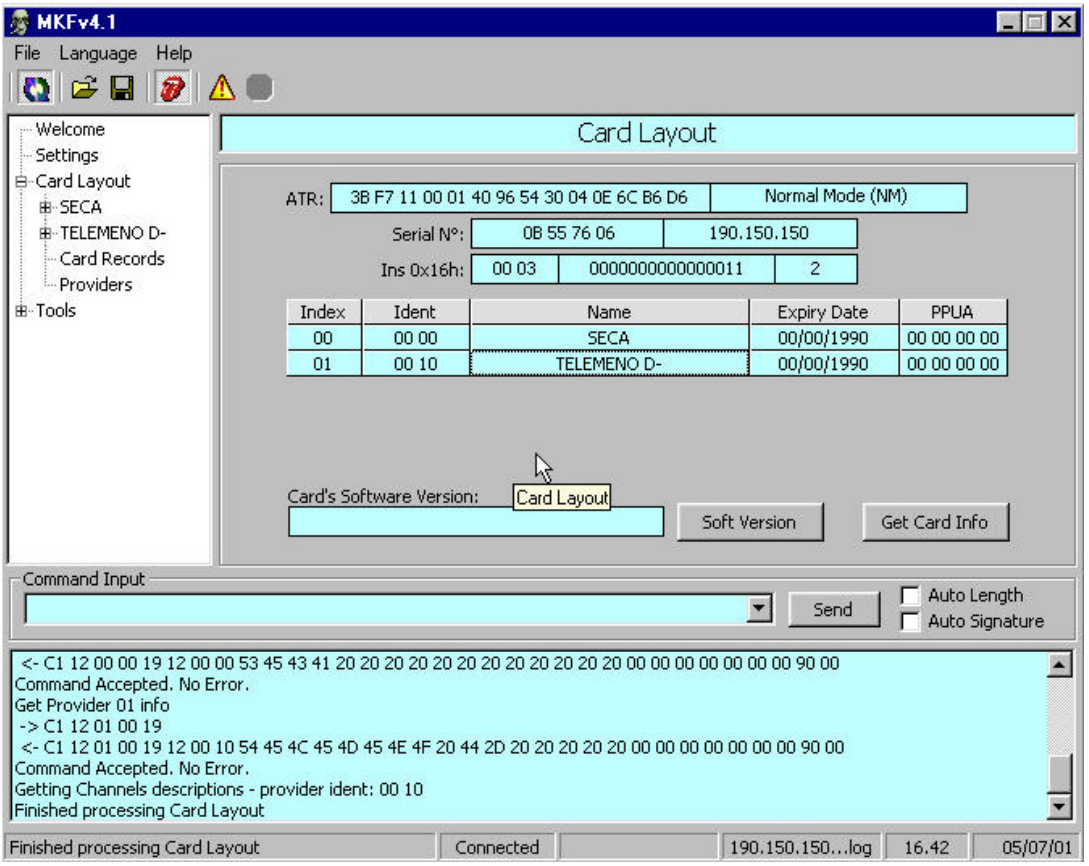

Fig. 5

Si disconnetta la carta.

A questo punto si dovranno creare i records del nostro 1° provider.
Si torni a MatrixStudio, sezione "Mini MOSC", Fig. 6. Si smarchi "Add", nel settore "Provider", selezionando 1 0010 TELEMENO D-, in quello "signature, selezionare il provider 0 "SEGA" com Mk 0. Clickando sul bottone "Make ins", verra' visualizzata l'istruzione: c1 40 00 00 20 23 0010 26 ff ff d0 54 45 4C 45 4D 45 4E 4F 20 44 2D 20 20 20 20 20 82 F4 9B 33 C1 7B CA 5B 38, nel campo "Ins". Clickando, ora su "Send INS", s'inviera' il comando alla MoscAS, se tutto andra' bene, come prevedibile, si avra' la risposta 90 00, nella linea di status.

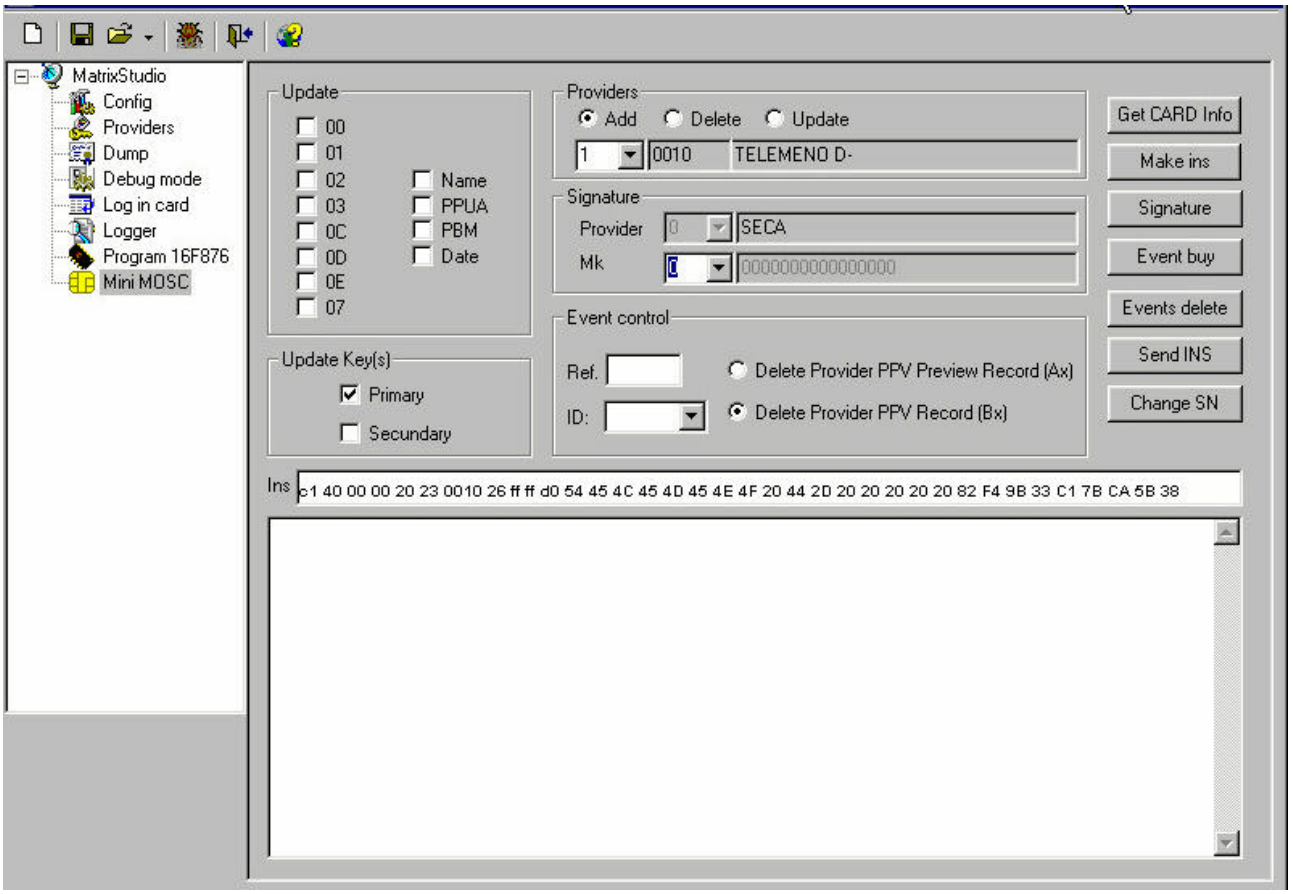

Fig. 6

In pratica non si e' generato il primo record ma, si e' inserito il nome e l'ident del nuovo provider, che si potra' vedere, lanciano Mkfind 4.1, connettendo la scheda, e clickando, nella sezione "Card Layout" di Mkfind, sul bottone "Get Card Info". Il nuovo provider, invece, non comparira' se, andando nella sezione "Card Records", si clickera' su "Get Record". Questo perche', non si e' ancora dato il comando per l'abilitazione per la scrittura del PBM (che deve essere il primo record, per ogni nuovo provider), e cioe' un comando del tipo: C1 40 00 00 LL 24 PP PP 26 FF FF 82 SIGNMK0.
Disconnessione da Mkfind, ritorno a MatrixStudio, stessa sezione. Smarcare "Update", in "Provider", lasciando selezionati il n. 1 (Telemeno D-) e "PBM", nel settore "Update", lasciando sempre "SEGA" e Mk 0, nel settore "Signature". Poi si clicka, come al solito su "Make ins" e, l'istruzione che comparira' nel campo "Ins" dovra' essere, come detto, del tipo: C1 40 00 00 LL 24 PP PP 26 FF FF 82 SIGNMK0.
Come sempre, con un "Send INS", si inviera' il comando alla scheda, con il ritorno di sempre: 90 00, se tutto e' andato bene.
Il passaggio successivo sara' quello d'inserire la PPUA.
Cosi', come di fatto si e' inserito (Update) il PBM, si inserira' la PPUA, smarcando le opzioni adeguate: "Update", in "Provider", sempre con 1 (Telemeno D-) e "PPUA", nel settore "Update", idem come sopra nel settore "Signature". Poi si clicka, come al solito su "Make ins" e, l'istruzione che comparira' nel campo "Ins" dovra' essere simile a: C1 40 00 00 LL 24 PP PP 41 UU UU UU UU 82 SIGNMK0, dove il gruppo UU UU UU UU e' la PPUA del proprio abbo a Telemeno.
Solita risposta 90 00.

Stessa solfa per l'inserimento delle chiavi e della data. Si sottolinea che e' fondamentale inserire (cioe' aggiornare) la Mk1 del provider, perche' con PPUA e Mk1 valide, le chiavi 0C, 0D e 0E e la data, si aggiorneranno automaticamente.
Quindi per inserire la chiave 01, smarcare "Update", in "Provider", con scelto 1 (Telemeno D-) e "01", nel settore "Update", lasciando sempre "SEGA" e Mk 0, nel settore "Signature". Poi si clicka, come al solito su "Make ins" e, l'istruzione che comparira' nel campo "Ins" dovra' essere del tipo: C1 40 00 00 LL 24 PP PP 90 51 N1 N2 N3 N4 N5 N6 N7 N8 82 SIGNMK0, ricordando che m1 m2 m3 m4 m5 m6 m7 m8 criptata con MK0 si trasforma in n1 n2 n3 n4 n5 n6 n7 n8. Inviando con "Send INS", il comando alla scheda, si ricevera' 90 00, come ritorno.

In questa fase, e' anche necessario inserire le tre chiavi operative, non tanto per averle aggiornate, ma per creare i records necessari a contenerle.

Per inserire le chiavi 0C, 0D e 0E, si seguira' lo stesso procedimento usato per la chiave 01, quindi, per ognuna, smarcando "Update", sempre con 1 (Telemeno D-), in "Provider" e "0C" o "0D" o "0E", nel settore "Update" e lasciando sempre "SEGA" e Mk 0, nel settore "Signature", ripetendo, con la stessa tecnica, i comandi interessati. Stesso procedimento per la data.
Da ora, se la PPUA e la Mk1, sono dati di una carta reale, si avra' l'autoupdate delle chiavi operative e della data.

Terminata questa fase, si potra' vedere con Mkfind cosa si e' ottenuto.
In figura 5, si e' mostrato cosa compare, una volta connessi e clickato su "Get Card Info". Nella figura seguente, Fig. 7, e' possibile vedere cosa apparira', andando nella sezione "Card Records", clcickando sul bottone "Get Records".
Si potranno cosi vedere sia i records relativi al provider SEGA che, quelli del nostro provider operativo.

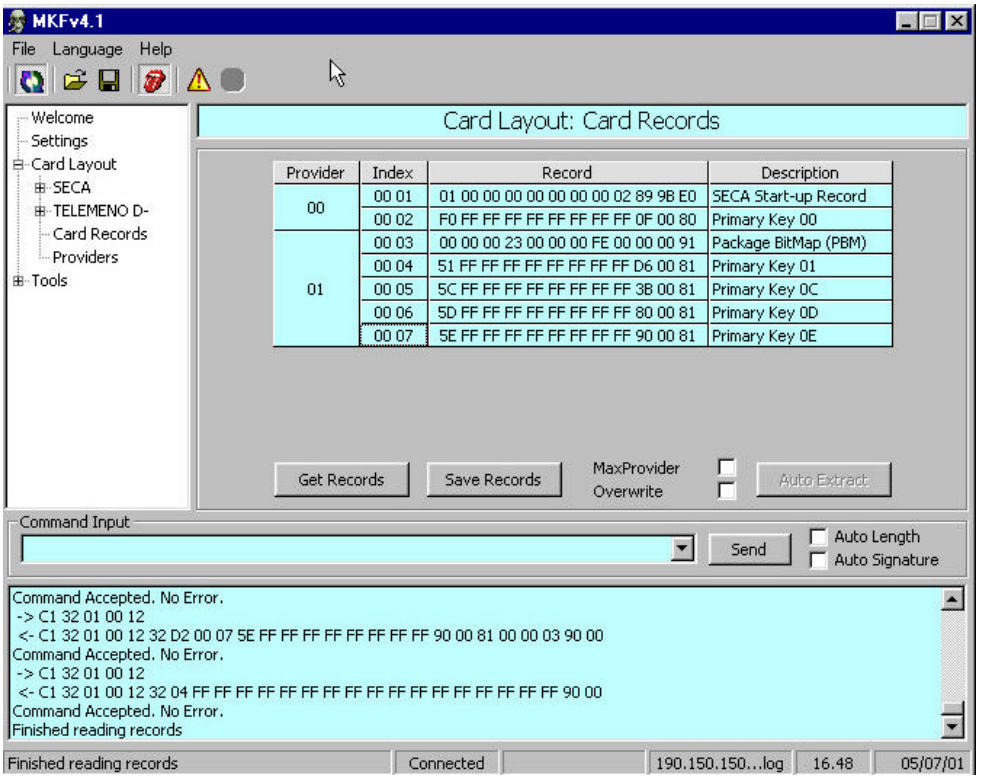

Fig. 7

In pratica la fatica e' terminata. Ora si e' in possesso di una scheda funzionante con un provider attivo. Questa puo' essere la base di partenza per seguitare ad esplorare il mondo SEGA. Come sempre letture consigliate sono la SEGAFaq e tutte quelle simili (anche piu' pallottolose), ma necessarie, se si vuole capire cosa si sta' facendo.

Sicuramente, per tutti la procedura e' stata chiarissima ma, se ci fosse qualcuno ancora con qualche dubbio, ricapitoliamo:

- con MatrixStudio 6.2, abbiamo, prima editato il nostro provider operativo;
- poi, abbiamo caricato, con il programmatore di Eeprom, l'hex di MoscAS, sul PIC 16F876;
- successivamente, abbiamo inserito nella nostra MoscAS, il numero seriale (UA);
- quindi si e' creato il nostro provider, con il suo nome;
- abbiamo abilitato la scrittura con l'inserimento del PBM;
- si e' inserita la PPUA;
- si e' inserita la Mk1;
- sono state inserite le chiavi 0C, 0D e 0E;
- e' stata inserita la data.

Di fatto, si e' utilizzato Mkfind 4.1, soltanto per verificare che i comandi dati con il MatrixStudio, fossero andati a segno (cosa, peraltro inutile), in quanto se la risposta ad un comando dato e' stata sempre 90 00, significa che non ci sono stati errori.

Come sulla Piccard2, con i comandi Pers, 6, 1 su Telecomando del G*ldB*x, si potra' visualizzare lo stato della propria carta.

Questa e', quindi la procedura da adottare, anche per inserire altri provider.

Prove ed esperienze, potranno fare avanzare le conoscenze.

Ognuno potra' variare, sempre con MatrixStudio nella sezione "debug mode', la configurazione dei diodi led's, scegliere o no la eeprom esterna. Decidendo anche se si vuole la gestione della PPV.

Insomma i Picc2rd Boys, hanno dato le tartine, lo Champagne, i bicchieri, il locale ecc., qualcuno vorra' portare le dame?!.

DEDICA

Queste note, sono dedicate e, al tempo stesso vogliono essere un simpatico omaggio ed un ringraziamento, a tutti i membri del Picc2rd Team, in quanto con il loro lavoro, svolto sempre alla luce del sole ed offerto all'uso ed al dibattito di tutti, senza far pesare su nessuno, inferiorita' od ignoranza, hanno consentito e stanno consentendo una grande operazione culturale di alfabetizzazione in un campo di per se' assai difficile.

Grazie