# New York City's MTA *Exposed!*

**Joseph Battaglia**

sephail@sephail.net http://www.sephail.net

Originally appearing in *2600 Magazine*, Spring 2005

## Introduction

In this article, I will explain many of the inner workings of the New York City Transit Authority fare collection system and expose the content of MetroCards. I will start off with a description of the various devices of the fare collection system, proceeding into the details of how to decode the MetroCard's magnetic stripe. This article is the result of many hours of experimentation, plenty of cash spent on MetroCards (you're welcome, MTA), and lots of help from several people. I'd like to thank everyone at *2600*, *Off The Hook*, and all those who have mailed in cards and various other information.

Becoming familiar with how magnetic stripe technology works will help you understand much of what is discussed in the sections describing how to decode MetroCards. More information on this, including additional recommended reading, can be found in “Magnetic Stripe Reading” also in this issue.

## Terms

These terms will be used throughout the article:

FSK - Frequency Shift Keying
: A type of frequency modulation in which the signal's frequency is shifted between two discrete values.

MVM - MetroCard Vending Machine
: MVMs can be found in every subway station. They are the large vending machines which accept cash in addition to credit and debit.

MEM - MetroCard Express Machine
: MEMs are vending machines that accept only credit and debit. They are often located beside a batch of MVMs.

MTA - Metropolitan Transportation Authority
: A public benefit corporation of the State of New York responsible for implementing a unified mass transportation policy for NYC and counties within the "Transportation District".

NYCTA - New York City Transit Authority
: Under the control of the MTA, the NYCTA is a public benefit corporation responsible for operating buses and subway trains in NYC.

RFM - Reduced-Fare MetroCard
: RFMs are available to the elderly or people with qualifying disabilities. Typical RFM fare is half or less than half of the standard fare.

"Common MetroCard"
: This term will refer to any MetroCard available to the public without special requirements. Examples include standard pay-per-ride cards, standard unlimited cards, and single-ride cards.

"Special MetroCard"

This term will refer to any MetroCard not available to the general public. Examples include reduced-fare cards, student cards, and employee cards.

"Single-Track MetroCard"

This term will refer to any MetroCard that has a one-track magnetic stripe (although there is no visible difference between the stripes of these cards and the stripes of two-track cards). The following types of cards are single-track: Single-Ride and Bus Transfer MetroCards.

"Dual-Track MetroCard"

This term will refer to all MetroCards with the exception of the single-track MetroCards mentioned above. The following types of cards are some examples of dual-track cards: pay-per-ride, pre-valued, unlimited, and reduced-fare.

"Passback period"

This term will refer to the time period before an access device will allow you to use an unlimited card again after swiping it. During this period, the devices generally respond with the message "JUST USED".

"Standard cards" and "standard readers"

These terms will refer to cards containing a magnetic stripe (credit, banking, etc) or readers of these cards that conform to the standards set forth in any or all of the following ISO specifications: 7810, 7811, 7813, and 4909.

# Cubic Transportation Systems

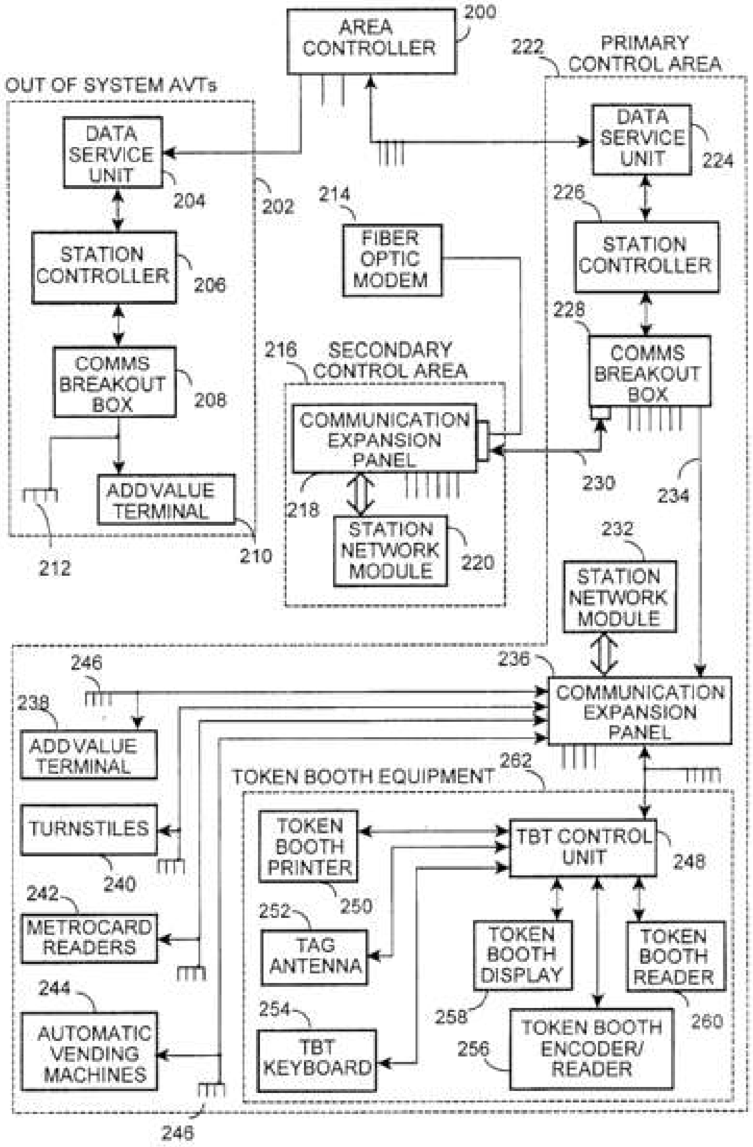
The fare collection system the MTA uses is developed by Cubic Transportation Systems, a subsidiary of Cubic Corporation. The patents I found to be related to the current New York City system filed by Cubic Corporation are as follows:

- 4,877,179 - FAREBOX SECURITY DEVICE
- 5,056,261 - TURNSTILE SYSTEM
- 5,072,543 - TURNSTILE MECHANISM
- 5,191,195 - FARE CARD READ-WRITER WHICH OVERWRITES OLDEST OR INVALID DATA
- 5,215,383 - TICKET STOCK AND TICKET DISPENSER
- 5,298,726 - FARE CARD READ-WRITER WHICH OVERWRITES OLDEST OR INVALID DATA
- 5,333,410 - CONTROLLABLE BARRIER SYSTEM FOR PREVENTING UNPAID ADMISSION TO A FEE-PAID AREA
- 5,574,441 - MASS TRANSIT INDUCTIVE DATA COMMUNICATION SYSTEM
- 5,612,684 - MASS TRANSIT INDUCTIVE DATA COMMUNICATION SYSTEM
- 6,595,416 - SYSTEM FOR RAPIDLY DISPENSING AND ADDING VALUE TO FARE CARDS
- 6,655,587 - CUSTOMER ADMINISTERED AUTOLOAD
- 6,789,736 - DISTRIBUTED ARCHITECTURE FOR MAGNETIC FARE CARD PROCESSING

Servicing, apart from routine collection of fares, on MTA equipment seems to be done by Cubic employees, not the MTA.

# The MetroCard System

At the core of the MTA fare collection system is the MetroCard. Preceded by a token-based system, the MetroCard is now used for every aspect of fare collection and allows for fare options that would never have been previously possible (e.g., Employee, Reduced-Fare, and Student MetroCards). MetroCards can currently be purchased at MVMs, MEMs, token booths, and various merchants throughout the New York City area. I will categorize the MetroCard access devices into two types: reading devices and fare collection devices. Both of these devices are networked in a complex system which allows the MTA, within minutes, to have up-to-date information on every card that has been issued. This also allows them to disable any card at will. The hierarchy of the network is shown below (as described in patent 6,789,736).



The physical characteristics of MetroCards follow those of standard cards (see Terms) almost exactly, but are one third the thickness. They have a diagonal notch cut out in the upper-right hand corner 3 1/8" from the left and 5/16" from the top of the card. Additionally, they have a 1/8" diameter hole, with its center 1/4" from the left and 5/16" from the top of the card, which is used to aid machines that suck your card in (bus fare boxes, MEMs/MVMs, handicapped entry/exit machines, etc.).

# Vending Machines

MEMs and MVMs are located throughout the subway system. They allow you to purchase or refill various common MetroCards with either cash or a credit card. RFMs can't be purchased at machines but can be refilled. On the front of the MEM or MVM is a tag with the machine's unique ID number.

The BIOS System Configuration screen from an MEM looks like this:

```
  AMIBIOS System Configuration (C) 1985-1997, American Megatrends Inc.,

Main Processor   : Celeron(tm)        Base Memory Size : 640KB
Math Processor   : Built-In           Ext. Memory Size : 14336KB
Floppy Drive A:  : None               Display Type     : VGA/EGA
Floppy Drive B:  : None               Serial Port(s)   : 3F8,2F8
AMIBIOS Date     : 07/15/95           Parallel Port(s) : 378
Processor Clock  : 300A MHz           External Cache   : 128KB,Enabled

ATA(PI) Device(s) Type         Size           LBA    32Bit  Block   PIO
                                              Mode   Mode   Mode    Mode
Primary Master   : Hard Disk   5729MB         LBA    On     16Sec   4

PCI Devices:                          PCI Onboard USB Controller, IRQ11
PCI Onboard Bridge Device             PCI Onboard Ethernet, IRQ15
PCI Onboard IDE
PCI Onboard VGA

FPGA ver. C, Base Address: 500h
BSP CPU.....Microcode OK
```

I have no reason to believe that the MVM hardware is any different.

# Receipts

Receipts can be obtained from MEM and MVM machines by answering "yes" when prompted. They possess a lot of information about the MEM/MVM, subway station, and card. You can match a receipt to a card by comparing the serial numbers. Let's take a look at some samples:

```
      MVM RECEIPT              MVM RECEIPT              MEM RECEIPT

MTA NYC TRANSIT          MTA NYC TRANSIT          MTA NYC TRANSIT
ASTOR PLACE              NASSAU AV & MANHATTAN AV 14TH STREET & 6TH AVENUE
NEW YORK CITY NY         NEW YORK CITY NY         NEW YORK CITY NY

MVM #: 0545(R219  0701)  MVM #: 1738(N408A  0500) MEM #: 5383(N513 0400)

Sun 14 Nov 04 21:28      Mon 04 Oct 04 14:22      Wed 17 Nov 04 12:14

Trans: Sale OK           Trans: Sale OK           Trans: Add Time OK
Payment Mode: Cash       Payment Mode: Credit     Amount:         $ 10.50
Amount:         $  7.00  Amount:         $ 21.00  Initial Type:030
Card Value:     $  0.00  Card Value:     $  0.00   7-DAY RFM UNLIMITED
Change Due:     $  3.00                           Time Added:  030
                         Credit Card #: XX5346     7-DAY RFM UNLIMITED
Serial #:1059909877      Auth#: 000008
Type: 023                Ref #: 060615762129      ATM Card #: XX0952
   1-DAY UNLIMITED                                Auth#: 760346
                         Serial #:1027066848      Ref #: 029089559668
       Questions?        Type: 024
Call (212) METROCARD        7-DAY UNLIMITED       Serial #:0987218036

                                Questions?               Questions?
                         Call (212) METROCARD     Call (212) METROCARD
```

Most of the information on the receipt is fairly obvious, but notice the line that begins with "MEM #" or "MVM #". The first four digits correspond to the actual MEM or MVM ID number as found on the machine. The next letter and following three digits inside the parenthesis correspond to the closest token booth. This ID can also be found on the booth itself. The meaning of the next four digits is currently unknown. However, they are unique to each machine that has the same booth ID, but are not unique among machines with different booth IDs. They seem to simply be a unique ID for each MEM/MVM in the station, possibly grouped by location. See "MEM/MVMs" for a table.

Now look to the bottom of the receipt. The line that begins with "Type:" (or "Initial Type:" if an RFM is being refilled) gives the numerical card subtype value followed by a description of the type on the following line.

Receipts purchased with a credit card contain additional fields that allow the MTA to verify the credit card holder in the case that he/she decides to lose the MetroCard.

# Turnstiles

The use of a turnstile is the most common way to enter the subway. Entry is granted by swiping a valid MetroCard through the reader/writer located on the outside of each turnstile. Once swiped, the LCD display on the turnstile will display a message. Some common messages:

GO

Message displayed for unlimited MetroCards

GO<br>
1 RIDE LEFT

Message displayed for Student MetroCards, where '1' is the number of rides left for the day.

JUST USED

The passback period for the unlimited MetroCard is not up

GO<br>
1 XFER OK

Message displayed when transferring from a bus.

Above the LCD there are a series of round indicators. Of these, one has an arrow pointing in the direction of the turnstile in which you would enter after paying your fare, and another reads "No" and a do-not-enter bar which, when lit, indicates that the turnstile is not active. After paying your fare, another indicator below the green arrow lights to indicate that you may proceed through the turnstile without smashing your groin into the arm.

Above those, there are three horizontal bar indicators contained within a rectangular cutout. When a Reduced-Fare MetroCard is swiped, the top indicator (red) will light. When a Student MetroCard is swiped, the middle indicator (yellow) will light. When an Employee MetroCard is swiped, the bottom indicator (the color of which I'm unsure of) will light. These indicators are present on both sides of the turnstiles and they allow transit cops, many of whom are undercover, to monitor the types of cards being used by riders. This helps detect, for example, when Student MetroCards are being used at times when school is not in session or when an obvious misuse of an Employee or Reduced-Fare MetroCard occurs.

# Reading MetroCards

MetroCards are relatively difficult to read. You will not be able to read them with off-the-shelf magnetic stripe readers, so please don't waste your money. The reason for this is not that the format is different; MetroCards use Aiken Biphase (also known as frequency shift keying (FSK)) just like standard cards. However, the hardware that ships with these readers is designed for a completely different (and well-documented) specification. They require many "clocking bits," which consist of a string of zero-bits at the beginning of the stripe to aid in setting a reference frequency for decoding. Additionally, most readers also look for a standard start and end sentinel that exists on standard cards to denote the start of a particular track. On top of that, characters on these cards are defined as either four or six bit blocks (depending on the track) and contain a longitudinal redundancy check (LRC) character after the end sentinel to verify data integrity. Needless to say, MetroCards don't have any of these properties and contain fields of arbitrary length; thus, another method of reading and decoding is required.

Fortunately, magnetic heads are everywhere (e.g., cassette tape players) and the output from magnetic heads when passed over a magnetic stripe consists of voltage spikes in the audible frequency range. Since sound cards are excellent A/D converters for this range of input and are readily available and very cheap, we can use the microphone input interfaced to a magnetic head for the purpose of creating our own reader (for a lot less than the MTA is paying, I'm sure!). See the article “Magnetic Stripe Reading” in this issue for more details.

For the same reason that reading was initially difficult, writing to MetroCards is extremely difficult, and is still a work-in-progress which will not be discussed in this article. A technique similar to that of the decoder (in reverse) can be used to write to cards, although it is much more difficult to implement and obviously requires more equipment than just a sound card and a magnetic head. For those of you who realize how this can be done and have the ability to build the equipment, kudos, but keep in mind the ramifications of being caught using a card you wrote to yourself. Modifying the data on cards does work. But the MetroCard system is very complex and allows for the surveillance of this sort of activity. The goal of this project is to learn how the system works, how it can be theoretically defeated, but certainly not to get stuck in prison.

Apart from these difficulties, MetroCard tracks are defined as follows: Dual-Track MetroCards have two tracks - one track being twice the width of the other - and will be referred to as track 1-2 and track 3; Paper MetroCards have one track which will be referred to as track 1-2. These track names (as I refer to them) correspond to the same track fields that have been established by ISO 7811.

# Decoding Dual-Track MetroCards - Track 3

Track 3 on dual-track MetroCards contains static data. It is written when the card is produced and the serial number is printed on the back, and is not written to thereafter by any machine. Some data found on this track can also be found by looking at the information printed on the back of the card. The track format is as follows:

|  | Track 3 Content | Offset | Length |
|---|---|---|---|
| 1: | Start Sentinel | 0 | 15 |
| 2: | Card Type | 15 | 4 |
| 3: | Unknown | 19 | 4 |
| 4: | Expiration Date | 23 | 12 |
| 5: | Unknown | 35 | 4 |
| 6: | Constant | 39 | 8 |
| 7: | Unknown | 47 | 8 |
| 8: | Serial Number | 55 | 80 |
| 9: | Unused | 135 | 16 |
| 10: | Unknown | 151 | 16 |
| 11: | End Sentinel | 167 | 93 |

Decoding track 3 is accomplished as follows:

1. Constant: 000000011000111
2. Convert binary to decimal
   - See "Card Types" for a lookup table.
3. Use is not yet known
4. To determine the expiration date for common MetroCards:
   - Convert binary to decimal
   - Divide the decimal value by 2, round up
   - Convert the decimal value to year / month format as follows:
     - Year: Integer value of the decimal value divided by 12
     - Month: Value of the modulus of the decimal value and 12
   - Add 1992 to the year
   - The expiration date is the last day of the previous month
   - Note: Non-common MetroCards seem to have different date offsets (FIXME: find offsets)
   - Note: This expiration date is the date the physical card can no longer be used and is considered invalid. See the track 1-2 expiration date field for more information.
5. Use is not yet known
6. Constant: 00001101
7. Use is not yet known
8. Convert binary to decimal
9. Unused field
10. Use is not yet known
11. Constant:
    00100101001100100110100101100101010011001010010
    1001100110101010011010010101001101001010110101

# Decoding Dual-Track MetroCards - Track 1-2

Track 1-2 on dual-track MetroCards contains variable data. It is written to by every machine used for fare collection, reading devices excluded. Interestingly enough, track 1-2 does not only contain information pertaining to the last use, but also to the use before that. These two records are separated by a strange set of field separating bits, which contains in it a bit that seems to be half of the one-bit frequency (which is a non-standard use of FSK). The most reliable way to find the second track is to search for a second start sentinel, both of which are identical for each record. The track format is as follows:

```
    Content              Offset  Length
    -------------------  ------  ------
 1: Start Sentinel         0       10
 2: Time                  10        2
 3: Card Sub-Type         12        6
 4: Time                  18        6
 5: Date                  24       10
 6: Times Used            34        6
 7: Expiration Date       40       10
 8: Transfer Bit          50        1
 9: Last Used ID          51       15
10: Card Value            66       16
11: Purchase ID           82       16
12: Unknown               98       20
```

Decoding track 1-2 is accomplished as follows:

1. Constant: 0011010111
2. See 4
3. Convert binary to decimal
    - The card sub-type corresponds to the sub-type as indicated on the receipt if one was obtained from an MEM/MVM.
    - See "Card Types" for a lookup table.
4. To deal with the limited storage space on the MetroCard stripe, each bit in this field and field (2) represents 6 minutes. To determine the last time used for common MetroCards:
    - Concatenate the binary from (2) with the binary from this field
    - Convert to decimal
    - Multiply decimal value by 6
    - Result is the number of minutes since 01:00 that the card was last used
5. Convert binary to decimal
    - This field contains the last usage date, which can be determined by calculating an offset based on a card of the same type with a last usage on a known date. However, since this field only has 10 bits, dates will most likely roll over after 1024 (2^10) days and a new offset will have to be determined. Offsets also seem to differ with different types of MetroCards.
6. Convert binary to decimal
    - The times used field is incremented every time you use the card to pay a fare except during a transfer. In that case, the transfer bit is set and the times used field remains the same.
7. Convert binary to decimal
    - Determine offset based on the description in 5a to determine the exact expiration date of a card. Alternatively, subtract the date field from this field to determine how many days after the last usage the card expires.
    - Do not confuse this field with the expiration date field on track 3; it is only used on cards which expire a set number of days after you first use them (eg. unlimited cards) and will not be set for cards such as pay-per-ride which do not have an expiration date.
8. Bit is 1 if the last use was for a transfer, 0 otherwise
9. Convert binary to decimal
    - This field seems to have a completely separate lookup table that is used internally by the fare collection system.
    - See "Last Used IDs" for a lookup table.
10. Convert binary to decimal
    - The result is the value remaining on the card in cents.

11. Convert binary to decimal
    - This field seems to have a completely separate lookup table that is used internally by the fare collection system to match the value of this field with a MVM ID number (such as those you can find on receipts).

# Card Types

| Type | Subtype | Description |
|---|---|---|
| 0 | 0 | FULL FARE |
| 0 | 10 | PRE-VALUED |
| 0 | 12 | PRE-VALUED ($10.00) |
| 0 | 13 | PRE-VALUED ($2.00) |
| 0 | 14 | LIRR |
| 0 | 19 | PRE-VALUED ($4.00) |
| 0 | 23 | 1-DAY UNLIMITED ($2.00 fare) |
| 0 | 24 | 7-DAY UNLIMITED ($2.00 fare) |
| 0 | 25 | 7-day Express Bus Unlimited ($4.00 fare) |
| 0 | 26 | 30-DAY UNLIMITED ($2.00 fare) |
| 0 | 29 | AIRTRAIN |
| 0 | 30 | 7-DAY RFM UNLIMITED ($2.00 fare) |
| 0 | 43 | TransitChek |
| 0 | 46 | TransitChek |
| 0 | 47 | TransitChek |
| 0 | 48 | TransitChek 30-DAY UNLIMITED |
| 0 | 56 | 1-DAY UNLIMITED ($1.50 fare) |
| 0 | 57 | 7-DAY UNLIMITED ($1.50 fare) |
| 0 | 59 | 30-DAY UNLIMITED ($1.50 fare) |
| 0 | 62 | SingleRide ($1.50 fare) |
| 0 | 87 | SingleRide ($2.00 fare) |
| 4 | 2 | Two-Trip Special Program Pass |
| 4 | 5 | Grades 7-12 |
| 4 | 13 | 1/2 Fare - Grades K-12 |

# Last Used Ids (sample)

| | |
|---|---|
| 1513 | 14th St/Union Sq |
| 1519 | 8th St/Broadway (A39) |
| 1880 | Lexington Ave (N601) |
| 1942 | ASTOR PLACE (R219) |
| 2157 | 34th St/6th Ave (N506) |
| 2204 | 42nd St/Grand Central |
| 2278 | 9th Street PATH |

## MEM/MVMs (sample)

```
Location                    Type  ID
14TH ST. - UNION SQUARE     MVM   0530(A033  0400)
14TH ST. - UNION SQUARE     MVM   0400(A033  0700)
14TH ST. - UNION SQUARE     MVM   0481(A033  0701)
14TH ST. - UNION SQUARE     MVM   1122(A034  0400)
14TH ST. - UNION SQUARE     MVM   0216(A034  0700)
14TH ST. - UNION SQUARE     MVM   0215(A034  0701)
14TH ST. - UNION SQUARE     MVM   1370(A035  0700)
14TH ST. - UNION SQUARE     MVM   0541(A037  0700)
14TH ST. - UNION SQUARE     MVM   0265(A037  0701)
8TH STREET & BROADWAY       MEM   5462(A039  0400)
8TH STREET & BROADWAY       MEM   5662(A038  0401)
95TH ST & FT. HAMILTON      MVM   0982(C028  0700)
14TH STREET & 8TH AVE       MEM   5314(H001  0702)
1ST AVE & 14TH STREET       MVM   1358(H007  0700)
1ST AVE & 14TH STREET       MVM   1145(H007  0701)
175 ST/FT. WASHINGTON AV    MVM   1632(N010  0400)
175 ST/FT. WASHINGTON AV    MVM   1611(N010  0700)
175 ST/FT. WASHINGTON AV    MEM   5274(N010  0701)
W 4TH ST - WASHINGTON SQ    MVM   0321(N080  0700)
W 4TH ST - WASHINGTON SQ    MVM   0109(N080  0701)
FORDHAM ROAD                MVM   0550(N218  0700)
LEXINGTON AVE - 3RD AVE     MVM   0740(N305  0401)
NASSAU AV & MANHATTAN AV    MVM   1738(N408A 0500)
34TH STREET/SIXTH AVENUE    MVM   1428(N506  0702)
34TH STREET/SIXTH AVENUE    MVM   0540(N507  0701)
14TH STREET & 6TH AVENUE    MEM   5383(N513  0400)
CHRISTOPHER STREET          MVM   0637(R125  0700)
CHRISTOPHER STREET          MVM   0063(R125  0701)
14TH STREET - 7TH AVENUE    MVM   0294(R127  0400)
14TH STREET - 7TH AVENUE    MVM   1643(R127  0401)
14TH STREET - 7TH AVENUE    MVM   0357(R127  0700)
14TH STREET - 7TH AVENUE    MVM   0376(R127  0701)
34TH STREET-PENN STATION    MVM   0553(R138  0701)
WALL STREET & BROADWAY      MVM   1123(R203  0400)
WALL STREET & BROADWAY      MVM   1038(R203  0700)
ASTOR PLACE                 MVM   0654(R219  0400)
ASTOR PLACE                 MVM   0586(R219  0700)
ASTOR PLACE                 MVM   0545(R219  0701)
ASTOR PLACE                 MVM   0744(R220  0700)
ASTOR PLACE                 MVM   0318(R220  0701)
14TH ST. - UNION SQUARE     MVM   0576(R221  0400)
14TH ST. - UNION SQUARE     MVM   0514(R221  0401)
14TH ST. - UNION SQUARE     MVM   0475(R221  0700)
14TH ST. - UNION SQUARE     MVM   0564(R221  0701)
23RD STREET - PARK AVE      MVM   0489(R227  0701)
28TH STREET - PARK AVE      MVM   1228(R229  0700)
```

## Conclusion

As you may have noticed, I haven't provided a way to decode the Single-Track MetroCards yet. Bus Transfer MetroCards are collected after use and the magnetic stripe of Single-Ride MetroCards is written with bogus data after use. We simply haven't received enough unused samples to be able to reverse-engineer all the information contained on these cards.

This project is far from over, and we still have tons of data that needs to be collected. You can help in many ways:

- Collect receipts every time you purchase a MetroCard. This will help us expand (and keep updated) our database of the booths and MEMs/MVMs contained within each station. Also, if possible, keep the MetroCard associated with the receipt.

- If you notice anything unusual, such as a frozen MTA kiosk (eg. MEM, MVM, reader, etc), open equipment (eg. while repairs are being done), or anything else, take some good pictures. As of now, photography bans are being proposed for New York City subway system, but are not yet in place. It is not yet illegal to take pictures, so know your rights.
- If you're paying for a bus ride with change, get a Bus Transfer MetroCard and send it to us if you don't intend to use it. Make sure you note the route, direction, time, date, and any other applicable information.

New things are being discovered and more data is being collected every day, so consider this article a "snapshot" of a work in progress. You can find and contribute to the data being collected on this system at http://www.2600.com/mta and by sending us additional information at 2600 Metrocard Project, PO Box 752, Middle Island, NY 11953 USA.

*A script to decode the cards described here is provided as an appendix and was not included in the original article.*

# Code Listing (mcdecode.pl)

```
#!/usr/bin/perl
#
# Display MetroCard Raw And Parsed Data
# Version 0.01
# Written in 2004 by Joseph Battaglia <sephail@sephail.net>
#
# Permission is hereby granted, free of charge, to any person obtaining a copy
# of this software and associated documentation files (the "Software"), to deal
# in the Software without restriction, including without limitation the rights
# to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the Software is
# furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice shall be included in
# all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
# AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
# OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
# SOFTWARE.
#
#
# Notes on input:
#  A line that begins with a '#' is not processed and not printed
#  A line that begins with a '%' is printed as a comment
#

$card_type = 0; #FIXME: this should be set by track 3, but since I only have
                # a single-track reader...

$typesfile = '/enc/sephail/development/wip/metrocard/apps/types.txt';
$lidfile = '/enc/sephail/development/wip/metrocard/apps/lids.txt';




# track 1-2 field names
@T12_FIELDS = (
  'Time',         'Sub-Type',    'Time',        'Date',
  'Times Used',  'Expiration',  'Transfer',    'Last ID',
  'Value',        'Purch ID',    'Unknown'
);

# track 3 field names
@T3_FIELDS = (
  'Type',         'Unknown',     'Expiration',  'Unkonwn',
  'Unknown',     'Unknown',     'Serial',      'Unused',
  'Unkonwn',     'Constant'
);

# generate regexps for track 1-2
@t12_lens = (2, 6, 6, 10, 6, 10, 1, 15, 16, 16); # lengths of track 1-2 fields
$t12_re1 = '11010111'; # start sentinel
$t12_re1 .= join('', map("([01]{$_})", @t12_lens)); # track 1-2 fields
$t12_re2 .= $t12_re1 . '(.*)' . $t12_re1; # regexp for dual records
# generate regexp for track 3
@t3_lens = (4, 4, 12, 4, 8, 8, 80, 16, 16); # lengths of track 3 fields
$t3_re = '000011000111'; # start sentinel
$t3_re .= join('', map("([01]{$_})", @t3_lens)); # track 3 fields
$t3_re .= '0010010100110010011010010110010101001100101001'; # end sentinel
$t3_re .= '0100110011010101001101001010100110100101011010';




# lookup card type/subtype in typesfile and return its name
sub lookup_type($$)
{
  ($in_type, $in_subtype) = @_; # read arguments
  open(FH, $typesfile) or die "Can't open $typesfile: $!"; # open typesfile
  while (<FH>) { # loop through each record
    ($type, $subtype, $name) = split(/:/); # split fields
    if (($type eq $in_type) and ($subtype eq $in_subtype)) { # look for match
      chomp($name); # remove newline
      return $name; # return name
    }
  }
  return 'UNKNOWN'; # could not find type/subtype; return 'UNKNOWN'
}


# lookup last id in lidfile and return its name
sub lookup_lid($)
{
  ($in_value) = @_; # read arguments
  open(FH, $lidfile) or die "Can't open $lidfile: $!"; # open lidfile
  while (<FH>) { # loop through each record
    ($value, $name) = split(/:/); # split fields
    if (($in_value eq $value)) { # look for match
      chomp($name); # remove newline
      return $name; # return name
    }
  }
  return 'UNKNOWN'; # could not find lid; return 'UNKNOWN'
}


# print header
sub print_header($)
{
  ($title) = @_; # read arguments
  print("$title\n"); # print title
  print("Field           Hex        Decimal    Parsed\n"); # print header
  print("--------------- ---------- ---------- ------\n");
}


# print field as hex and decimal and return decimal value
sub print_field($$$)
{
  ($track, $field, $value) = @_; # read arguments
  $tvar = 'T' . $track . '_FIELDS'; # create variable name
  $value = oct('0b' . $value); # convert to decimal
```

```
  printf('%2d: %-11s ', $field, $$tvar[$field - 1]);
  printf('%10X %10d ', $value, $value); # print base 10/16 values
  return $value; # return decimal value
}


# parse field
sub parse_field($$$)
{
  ($track, $field, $value) = @_; # read arguments
  $decvalue = oct('0b' . $value);

  if ($track == 12) { # track 1-2 fields

    if ($field == 1 or $field == 3) { # time
      if ($card_type == 0) {
        $time = $decvalue * 6;
        $hr = int($time / 60) - 1;
        $min = $time % 60;
        printf("%.2d:%.2d", $hr, $min);
      }
    }

    if ($field == 2) { # subtype
      print(lookup_type($card_type, $decvalue));
    }

    if ($field == 5) { # times used
      print($decvalue);
    }

    if ($field == 7) { # transfer
      print(($decvalue == 1) ? "YES" : "NO");
    }

    if ($field == 8) { # last id
      print(lookup_lid($decvalue));
    }

    if ($field == 9) { # value
      $dollars = int($decvalue / 100);
      $cents = $decvalue % 100;
      printf("\$%d.%.2d", $dollars, $cents);
    }
  }

  if ($track == 3) { # track 3 fields

    if ($field == 1) { # type
      print(lookup_type('M', $decvalue));
      $card_type = $decvalue;
    }

    if ($card_type == 0 and $field == 3) { # expiration
      $day = 30;
      if (!$decvalue % 2) {
        $decvalue++;
      }
      $decvalue /= 2;
      my $year = int ($decvalue / 12) + 1992;
      my $month = $decvalue % 12;
      if ($month > 1) {
        $month--;
      }
      else {
        $year--;
        $month = 12;
      }
      if ($month < 8) {
        if ($month % 2) {
          $day = "31";
        }
      }
      else {
        if (!($month % 2)) {
          $day = "31";
        }
      }
      if ($month == 2) {
        if (!($year % 4)) {
          $day = 29;
        }
        else {
          $day = 28;
        }
      }
      print("$month/$day/$year");
    }

    if ($field == 7) { # serial
      printf("%.10d", $decvalue);
    }
  }

  print("\n");
}


# main loop
while (<STDIN>) {

  if (/^%(.*)/) { # process printed comments
    print("### $1\n"); # print
    next; # process next input line
  }
  if (/^#/) { # process ignored comments
    next; # process next input line
  }

  # track 3
  if (/$t3_re/ or reverse =~ /$t3_re/) {
    print_header 'Track 3 Record';
    for ($i = 1; $i <= ($#t3_lens + 2); $i++) {
      $field = $i; # set field
      print_field(3, $field, $$i); # print each field
      parse_field(3, $field, $$i); # parse each field
    }
    print "\n";
  }

  # track 1-2 dual records
  if (/$t12_re2/ or reverse =~ /$t12_re2/) { # regexp
    print_header 'Track 1-2   Record 1'; # print header
    for ($i = 1; $i <= ($#t12_lens + 2); $i++) {
      $field = $i; # set field
      print_field(12, $field, $$i); # print/parse each field
      if ($field == 1 or $field == 3) { # time field is split in two
        parse_field(12, $field, $1 . $3); # combine time fields
      }
      else { # everything else is normal
        parse_field(12, $field, $$i); # parse each field
      }
    }
    print "\n";
    print_header 'Track 1-2   Record 2'; # print header
    for ($i = ($#t12_lens + 3); $i <= (($#t12_lens * 2) + 4); $i++) {
      $field = $i - ($#t12_lens + 2); # set field
      print_field(12, $field, $$i); # print/parse each field
      if ($field == 1 or $field == 3) { # time field is split in two
        $timevar1 = ($#t12_lens + 3);
        $timevar2 = ($#t12_lens + 5);
        parse_field(12, $field, $$timevar1 . $$timevar2); # combine time fields }
      }
      else { # everything else is normal
        parse_field(12, $field, $$i); # parse each field
      }
    }
    print "\n";
    next; # process next input line
  }

  # track 1-2 single record
  if (/$t12_re1/ or reverse =~ /$t12_re1/) { # regexp
    print_header 'Track 1-2   (Read Error)'; # print header
    for ($i = 1; $i <= ($#t12_lens + 2); $i++) {
      $field = $i; # set field
      print_field(12, $field, $$i); # print each field
      if ($field == 1 or $field == 3) { # time field is split in two
        parse_field(12, $field, $1 . $3); # combine time fields
      }
      else { # everything else is normal
        parse_field(12, $field, $$i); # parse each field
      }
    }
    print "\n";
    next; # process next input line
  }
```

# Code Listing (types.txt – sample)

```
M:0:Standard
M:4:Student
0:0:FULL FARE
0:10:PRE-VALUED
0:12:PRE-VALUED ($10.00)
0:13:PRE-VALUED ($2.00)
0:14:LIRR
0:19:PRE-VALUED ($4.00)
0:23:1-DAY UNLIMITED ($2.00 fare)
0:24:7-DAY UNLIMITED ($2.00 fare)
0:25:7-day Express Bus Unlimited ($4.00 fare)
0:26:30-DAY UNLIMITED ($2.00 fare)
0:29:AIRTRAIN
0:30:7-DAY RFM UNLIMITED ($2.00 fare)
0:43:TransitChek
0:46:TransitChek
0:47:TransitChek
0:48:TransitChek 30-DAY UNLIMITED
0:56:1-DAY UNLIMITED ($1.50 fare)
0:57:7-DAY UNLIMITED ($1.50 fare)
0:59:30-DAY UNLIMITED ($1.50 fare)
0:62:SingleRide ($1.50 fare)
0:87:SingleRide ($2.00 fare)
4:2:Two-Trip Special Program Pass
4:5:Grades 7-12
4:13:1/2 Fare - Grades K-12
```

# Code Listing (lids.txt – sample)

```
1513:14th St/Union Sq
1519:8th St/Broadway (A39)
1880:Lexington Ave (N601)
1942:ASTOR PLACE (R219)
2157:34th St/6th Ave (N506)
2204:42nd St/Grand Central
2278:9th Street PATH
```