



Card Swipe and iButton® Access Configuration

Model	CTM - 200
Revision:	Rev 1.4

3066 Beta Avenue | Burnaby, B.C. | V5G 4K4
© 2021 Cypress Solutions

Revision Control

Description	Revision	Date
Updating CTM200R1 images to CTM200R2	Rev 1.4	19-Sep-2016
Condensing and adding content	Rev 1.3	4-Jan-2016
Added card list example	Rev 1.2	12-Dec-2014
Removed serial connection and accelerometer sections. Updated hardware setup sections	Rev 1.1	13-May-2013
Customer Release	Rev 1.0	16-March-2012

Contents

Revision Control	1
Contents	2
1. Overview	3
2. Components	3
2.1 Access card reader solution components	3
2.2 Access card reader solution pinout	3
2.3 iButton solution components	4
2.4 iButton solution pinout	4
3. Configuring CTM for access card and iButton® reader	5
3.1 Access card reader configuration	5
3.2 iButton® reader configuration	5
4. Managing IDs	5
4.1 Card list format	5
4.2 Loading and saving authorized users	6
4.3 Deleting list of authorized users	7
4.4 Loading lock and unlock script	7
4.4.1 Lock Script	7
4.4.2 Unlock Script	8
5. Swipe card reporting	9
6. Specific access card application	10

I. Overview

CTM-200 supports integration with iButton® and access card readers for vehicle access control and logging. This feature allows the engine to be started based on a list of preapproved access cards. In addition to thwarting unauthorized vehicle use, card swipes that were approved can be reported to help provide a detailed history of the vehicle's use. Currently, access cards using the standard Wiegand 26-bit format, Corporate 1000's 35-bit and HID 37 bit H10302 format are supported.

2. Components

The iButton® and access card reader connects to the CTM-200 via GPIO.

2.1 Access card reader solution components



Card reader



Access card

2.2 Access card reader solution pinout



Wire Color	I/O Connector
Blue	2
Brown	4
Red	15
Black	16

Green	17
White	18

2.3 iButton solution components



iButton reader



iButton Fob

2.4 iButton solution pinout



Wire Color	I/O Connector
Red	15
Black	16
White	17

3. Configuring CTM for access card and iButton® reader

See link to the CTM200 command reference for more details on any commands:

http://www.cypress.bc.ca/documents/Command_Ref/CTM200/

Once the reader has been connected, CTM200 must be configured according to the authentication method used (access card or iButton®) and how they are connected to the CTM200. This section needs to be completed each time a reader is installed or the setup changed, or when a factory reset occurred. Otherwise, this section may be safely skipped in the future if only the approved IDs are to be changed.

3.1 Access card reader configuration

In command prompt, set the swipe card mode to I/O swipe card interface as follows:

```
cmd swipemode 10      # Configures GPIO for access card reader
cmd save              # saves configuration
```

You may now skip to section 6 to load the list of approved IDs for use with this CTM200 or vehicle.

3.2 iButton® reader configuration

In command prompt, set the swipe card mode to I/O iButton® reader interface as follows:

```
cmd swipemode 11      # Configures GPIO for iButton reader
cmd save              # saves configuration
```

You may now skip to section 6 to load the list of approved IDs for use with this CTM200 or vehicle.

4. Managing IDs

Users authorized to use a specific vehicle are managed by loading, saving and deleting a list of approved IDs. This list is only specific to each CTM200, making it vehicle specific. By default, if no list is added, then ANY user can access that vehicle. Custom engine lock and unlock scripts must also be added for engine enable and disable control.

4.1 Card list format

If a list of authorized users is used, the list must meet the following criteria:

- 1) UNIX text file format (i.e. each line in the file must end with linefeed/newline, and not contain carriage return.) Users may need to run dos2unix to convert the file before loading it to CTM200.
- 2) Each line in the file contains one authorized ID, in any format.
- 3) The list can be composed of a mix of IDs for different formats.
- 4) Wildcards can be used by replacing the corresponding digit to be ignored with 'X' or 'x'.
- 5) 5 or 6 digit long IDs are converted to XXX-#### or XXX-##### format respectively.
- 6) If the ID is for an HID 37 bit card, it must be 16 characters long and padded with leading zeroes or wildcard.

Example	Description
039-30395	Wiegand 26 bit format including facility code 39 and user ID 30395
04663	5 digit user ID (4663), ignoring facility code (will be converted to xxx-04663)
0466X	Allow any users with ID between 4660 and 4669 (will be converted to xxx-0466X)
XXX-18755	Facility code is wildcarded, so only user ID will be used for comparison
013-XXXXX	Allow any users with facility code 13
01X-123XX	Allow any users with facility code 10-19 and ID between 12300 and 12399
0000000010122233	HID 37 bit card with ID 10122233 (padded with leading 0's to be 16 chars long)
0000000000004663	HID 37 bit card with ID 4663 (padded with leading 0's to be 16 chars long)
XXXXXXXXXXXXXXXXX3	Allow any ID that ends with 3
XXXXXXXXXXXXXXXXX01	Allow all IDs (e.g. iButtons) that end with family code 01
D8000015AFDD4201	iButton ID (D8000015AFDD4201) with family code 01

Sample list:

```
039-30395
04663
XXX-18755
0000000010122233
D8000015AFDD4201
```

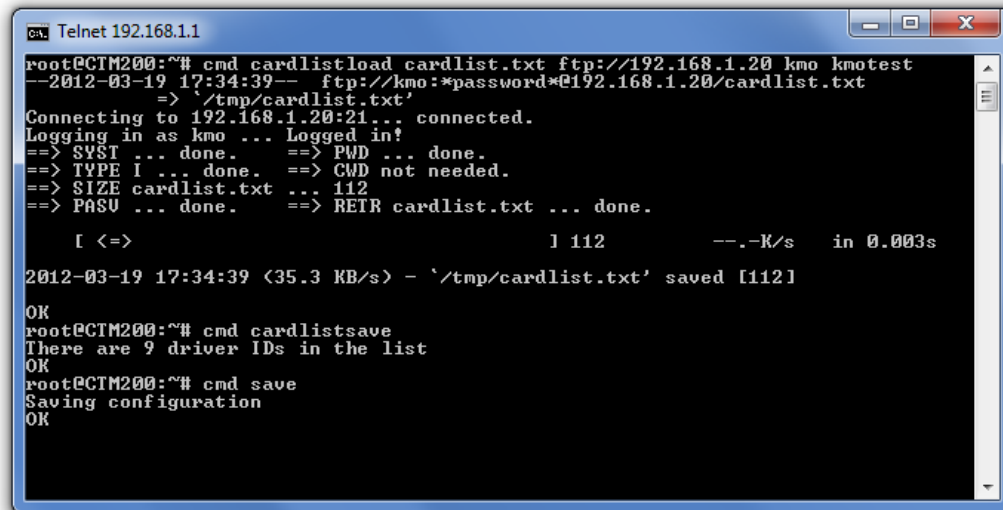
4.2 Loading and saving authorized users

Authorized users must first be added and saved by loading a list of approved IDs using the following command:

```
cmd cardlistload approvedlist.txt testrepository.com
```

Referring to the example above **cmd cardlistload** will download approvedlist.txt and place it in the **/var/data/ctm/acl/approved.txt** location (approvedlist.txt is renamed to approved.txt).

Note: The list must be in UNIX text file format (i.e. each line in the file must end with linefeed/newline, and not contain carriage return.) You may need to run dos2unix to convert the file before loading it to CTM200.



```

Telnet 192.168.1.1
root@CTM200:~# cmd cardlistload cardlist.txt ftp://192.168.1.20 kmo kmotest
--2012-03-19 17:34:39-- ftp://kmo:*password@192.168.1.20/cardlist.txt
=> '/tmp/cardlist.txt'
Connecting to 192.168.1.20:21... connected.
Logging in as kmo ... Logged in!
==> SYST ... done. ==> PWD ... done.
==> TYPE I ... done. ==> CWD not needed.
==> SIZE cardlist.txt ... 112
==> PASV ... done. ==> RETR cardlist.txt ... done.

[ <=> ] 112 --.-K/s in 0.003s
2012-03-19 17:34:39 <35.3 KB/s> - '/tmp/cardlist.txt' saved [112]
OK
root@CTM200:~# cmd cardlistsave
There are 9 driver IDs in the list
OK
root@CTM200:~# cmd save
Saving configuration
OK

```

4.3 Deleting list of authorized users

If authorized users are to be deleted, or to allow all users to access the vehicle, the following command must be used:

```
cmd cardlistclear
```

4.4 Loading lock and unlock script

Customs scripts can be run based on a **valid swipe from card/iButton** OR an **ignition off** event. See below for command information and functionality:

4.4.1 Lock Script

A script can be triggered when the CTM200 senses ignition off for at least 30 seconds. You will need to increase the CTM-200 ignition off timer and enable the vehiclepwrmgmt command for this feature to work. Below is an example of how this can be done:

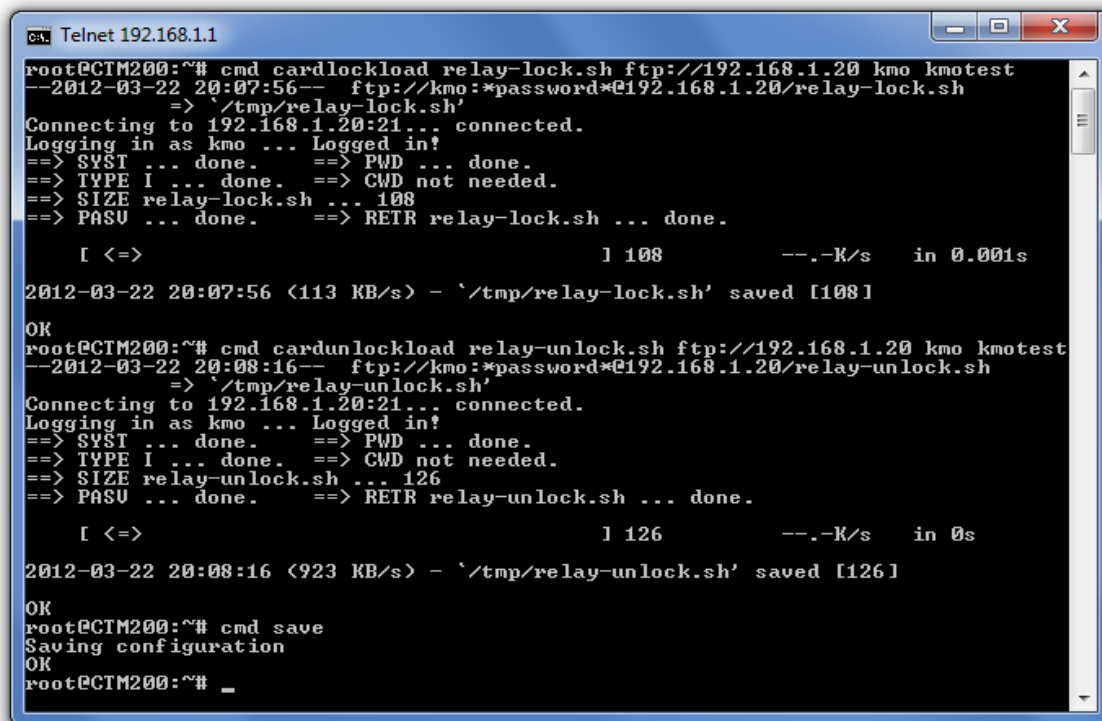

```

cmd cardlockload test1.sh testrepository.com # download test1.sh from testrepository.com
cmd pwr sdwn 1 60 # increase ignition off timer past 30 seconds
cmd vehiclepwrmgmt 1 # enable feature to trigger lock.sh script
cmd save # save configuration
cmd pwr mode 2 # power cycle gateway

```

Referring to the example above, **cmd cardlockload** will download test1.sh and place it in the **/var/data/ctm/acl/lock.sh** location (test1.sh is renamed to lock.sh).

Note: This configuration will also place your CTM-200 into low power mode after 20 minutes



```

Telnet 192.168.1.1
root@CTM200:~# cmd cardlockload relay-lock.sh ftp://192.168.1.20 kmo kmotest
--2012-03-22 20:07:56-- ftp://kmo:*password*@192.168.1.20/relay-lock.sh
=> '/tmp/relay-lock.sh'
Connecting to 192.168.1.20:21... connected.
Logging in as kmo ... Logged in!
==> SYST ... done. ==> PWD ... done.
==> TYPE I ... done. ==> CWD not needed.
==> SIZE relay-lock.sh ... 108
==> PASV ... done. ==> RETR relay-lock.sh ... done.

[ <=> ] 108 --.-K/s in 0.001s
2012-03-22 20:07:56 (113 KB/s) - '/tmp/relay-lock.sh' saved [108]
OK
root@CTM200:~# cmd cardunlockload relay-unlock.sh ftp://192.168.1.20 kmo kmotest
--2012-03-22 20:08:16-- ftp://kmo:*password*@192.168.1.20/relay-unlock.sh
=> '/tmp/relay-unlock.sh'
Connecting to 192.168.1.20:21... connected.
Logging in as kmo ... Logged in!
==> SYST ... done. ==> PWD ... done.
==> TYPE I ... done. ==> CWD not needed.
==> SIZE relay-unlock.sh ... 126
==> PASV ... done. ==> RETR relay-unlock.sh ... done.

[ <=> ] 126 --.-K/s in 0s
2012-03-22 20:08:16 (923 KB/s) - '/tmp/relay-unlock.sh' saved [126]
OK
root@CTM200:~# cmd save
Saving configuration
OK
root@CTM200:~# _

```

4.4.2 Unlock Script

A script can be triggered when a valid card/iButton is swiped. Below is an example of how to download a valid swipe script:

```
cmd cardunlockload test2.sh testrepository.com # download test2.sh from testrepository.com
```

Referring to the example above, **cmd cardunlockload** will download test2.sh and place it in the **/var/data/ctm/acl/swipe.sh** location (test2.sh is renamed to swipe.sh).

Note: this configuration assumes that you have a list of approved ID's (see section 6.2 for more detail).

5. Swipe card reporting

CTM200 has the ability to send reports when the swipe card reader has read an ID. These reports will also include a modified message #114. This message is the standard message #114 with the addition of the ID read as well as whether the ID is valid or not. To enable this feature and configure which general reports will have the modified message, use the following command:

```
cmd swipereport r1 [r2] [r3] [r4] [r5] [r6] [r7] [r8]
```

where **r1** to **r8** are general reports to include the ID in.

A sample message #114 as well as a modified message #114 is given below.

Original message #114:

```
$PGPS,171134.00,A,4915.3868,N,12259.8049,W,000.0,000.0,170609,+00004,5,09604890958*4A
```

Modified message #114 with ID and ID status:

```
$PGPS,171134.00,A,4915.3868,N,12259.8049,W,000.0,000.0,170609,+00004,5,09604890958,039-30391,V*33
```

'V' above means a valid ID was read when compared against the list of authorized users. If an invalid ID is read, then it would be shown as 'F'.

Finish by configuring general reports:

```
cmd retype n l r s
```

where **n** is the general report number, **l** is the local report destination, **r** is the remote report destination and **s** is whether to store and forward report.

Since there are numerous commands for reporting, please refer to the commands for **retype**, **repaddmes**, **repedmes**, **replocip**, **replocport**, **reprempip**, **reprempport**, **repemail**, and **repsms**. When finished, use **cmd save** to save.

For example, to use general report #2 with store and forward enabled for swipe card reader triggered reports:

```
cmd swipereport 2
cmd reptime 2 0 3 1
cmd repremip 2 192.168.1.204
cmd repremport 2 5006
cmd repaddmes 2 80
cmd save
```

Note that message 114 does not have to be added as `swipereport` will do that automatically.

6. Specific access card application

It is possible to configure the CTM200 to explicitly notify a driver if they have not authenticated with their access card. This CTM200 configuration does not prevent or inhibit the operation of the vehicle, but it is a useful way to ensure that authorized drivers are authenticating with their access cards.

For more details about the hardware/configuration for this setup, see the application note below:

[HID card swipe access](#)

Technical Support

**Cypress Solutions Service
Support Group**

1.844.462.9773 or 778.372.4603

9.00am to 5.00pm PST

support@cypress.bc.ca