

# CITIZEN MITSUBISHI

## DEVICE DRIVER

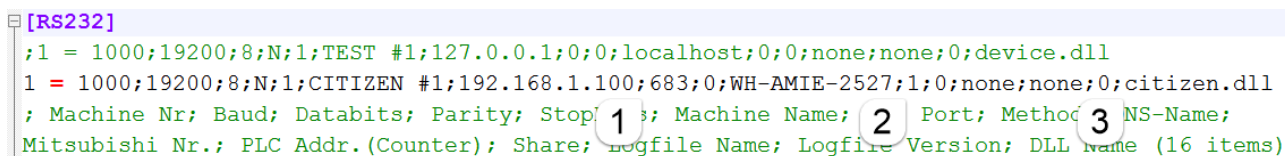
This highly configurable **device driver for CNCnetPDM** enables you to acquire and output all items that can be read from Mitsubishi CNC controllers on CITIZEN and MAZAK machines. [Here](#) you can find a list of sections with all items. By using writable commands you can also change parameters at the controller from a remote PC.

You can dynamically define your own set of items (up to 30) that can be enabled and disabled on a per machine basis by an INI file that is automatically created for each machine configured. This INI file also allows you to define your own names for every tag.

Tips: Make sure that you have a connection from your PC to the machine and get data, use the **MitsubishiGui.exe** utility to do so. Please also open port 683 in your Firewall.

## SETUP CNCNETPDM

- **Download** CNCnetPDM and install it as described in the **quick-start-guide**.
- Download the Citizen Mitsubishi driver, extract all content of Mitsubishi\_citizen\_dll.zip into the folder where you have CNCnetPDM installed.
- Edit CNCnetPDM.ini and add a new device as described in the **quick start guide** point 6. If you already have an entry in section [RS232] for a machine from Citizen or Mazak with a Mitsubishi controller you can simply copy and paste this line and change its content. You may comment out the previous entry by prepending a semicolon ';' if it's the same machine.
- Change the line as follows:



```
[RS232]
;1 = 1000;19200;8;N;1;TEST #1;127.0.0.1;0;0;localhost;0;0;none;none;0;device.dll
1 = 1000;19200;8;N;1;CITIZEN #1;192.168.1.100;683;0;WH-AMIE-2527;1;0;none;none;0;citizen.dll
; Machine Nr; Baud; Databits; Parity; Stop; 1; Machine Name; 2 Port; Method; 3 NS-Name;
Mitsubishi Nr.; PLC Addr. (Counter); Share; logfile Name; Logfile Version; DLL Name (16 items)
```

FIG 1: Configure device Citizen Mitsubishi in CNCnetPDM.ini

For Citizen machines with Mitsubishi controllers the following 3 parameters are important

1. Right to the IP Address of the controller enter number 683 (=port)
2. Right to the DNS Hostname of your machine enter the Mitsubishi number as configured, see section 'CITIZEN MITSUBISHI SETUP'.
3. Change the driver name to citizen.dll

---

## CITIZEN MITSUBISHI SETUP

- Extract all contents of Mitsubishi\_citizen\_dll.zip to the directory where CNCnetPDM is installed.
- As the driver identifies and accesses controllers by using machine numbers the first thing you have to do is to edit the configuration file melcfg.ini.
- Citizen Mitsubishi machine numbers are defined in sections, one for each device. The machine number is the numeric part after Machine, e.g. [Machine01] = 1. Note: The second part of this numeric value uses hexadecimal notation from 1 to F, e.g. 01 = 1 and 0F = 15, remember your math lessons.
- If you're working with one machine it is recommended to leave section [Machine01] as it is. Only in case it doesn't work you can change the value for Controller from M8NX to NCXJ, NCXK, M6B, NX or M7NX.
- It is necessary to adjust section [HOSTS] according to your network configuration e.g. TCP1=192.168.100.1,683 means that the IP Address of your controller is 192.168.100.1 683 after the comma is the port to be used to access the controller, do not change that.
- For additional machines you have to copy and paste the whole section [Machine01], then change its name e.g. [Machine02] and Device= to a different device in section [HOSTS] e.g. Device=TCP2. In section [HOSTS] you have to add a new device e.g. TCP2=192.168.100.2,683.
- Save melcfg.ini.

## USAGE

- Start CNCnetPDM, foreground program is sufficient (Start thread)
- CNCnetPDM automatically copies the original citizen.dll and appends the machine number as configured in the INI file, e.g. citizen\_1000.dll for machine 1000.
- In addition an INI file with the same name is created by the device driver, e.g. citizen\_1000.ini for your first machine.
- Double click CNCnetControl, if your device number is 1000 and the device name is CITIZEN #1 the output should be similar to the one below:

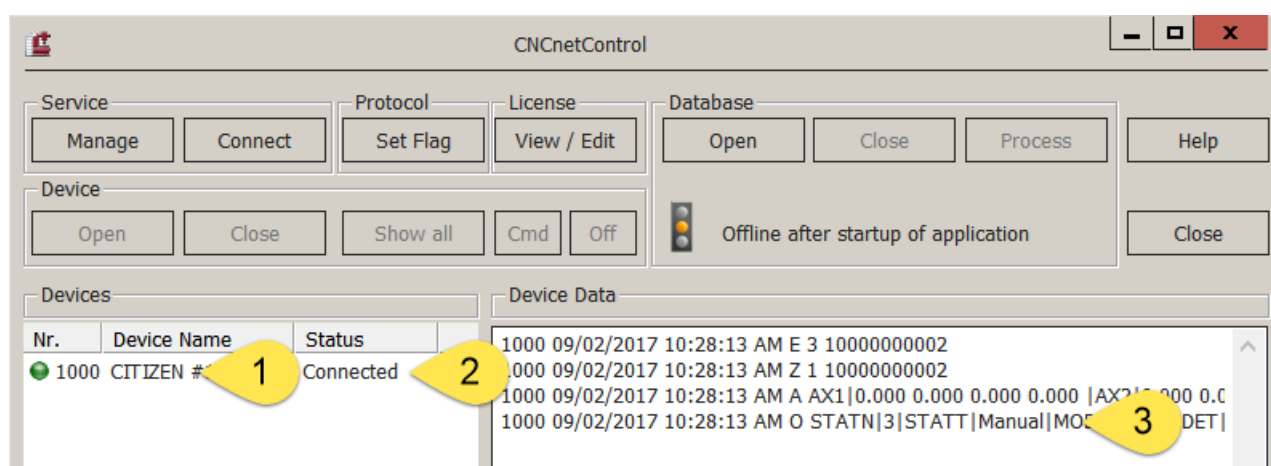


FIG 2: CNCnetControl (Citizen Mitsubishi)

The machine (1) shows up as connected (2) which is good. On the right side (3) you see the acquired data:

- The line starting with O contains most of items defined in the INI file of the device (e.g. Citizen\_1000.ini). If you didn't change the INI file you get something like the following:

```
1000 09/02/2020 10:28:13 AM O STATN| 3| STATT| Manual| MODEN| 8| MODET| INC feed|
PRGCU| 1| PRGMN| 4711| SEQCU| 1| FEEDR| 0| SPSPD| 0| ALMST| 0| TOOL1| 0| TOOL2| 0|
ALMTX| No alarm| 100000000002
```

Every item has a description followed by its value. All descriptions and values are delimited by pipe '|' symbols. This allows the most recent version of CNCnetPDM to create a database record and/or MTConnect Adapter output for every item.

With an unmodified INI file the initial items are:

ID	NAME	DESCRIPTION
1	STATN	Machine state number
2	MODEN	Controller mode number
3	PRGCU	Current NC program
4	PRGCM	Program comment
5	CFEED	Commanded feed rate
6	FEEDR	Current feed rate
7	PRGMN	Main NC program
8	SPDNR	Spindle number
9	CV501	Common variable #501
10	ERR0 - ERR9	Up to 10 current alarm messages sorted by priority

FIG 3: IDs, names and description of items

## WRITE DATA

This device driver allows you to create items or change data at the controller by using writable commands f.i. common variable  $\geq 500$  **SECT\_COMMON\_VAL1(x)**.

Note: Please use our **diagnostic tool to read & write data** first to check if a writing operation succeeds!

To write to a controller for security reasons a section with the respective command has to exist in your machine INI file. With an unmodified machine INI file you can use command 9 (Reads common variable 501) for testing. For commands with an (x) parameter the driver allows to write data for any valid parameter number e.g. with **SECT\_COMMON\_VAL1(x)** you can set all common variables  $\geq 500$ .

The most simple way to test writing operations is to use CNCnetControl:

To change the value of common variable 501 click on the machine (1) in the left area followed by button [Cmd] (2). In the dialog window (3) enter Number 9 (4). As command text use 0|501|4.055 (5). Items in the command text are Axis number, parameter number and new item value delimited by pipe '|' characters. Clicking on [OK] (6) executes the command. If successful the output of tag CV501 changes to the new value.

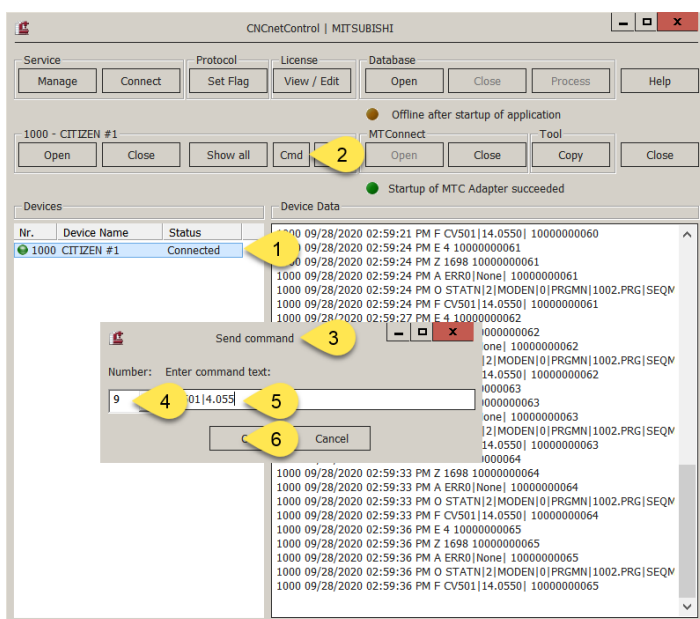


FIG 1: Set new value for common variable 501 with CNCnetControl

You can also write data by using the CNCnetPDM MTConnect API with your own applications (see [Free Open Source Client C#](#)) or the [MTConnect Agent Tester](#). In this case commands have to be input by using the following format (for the above example):

\* device|1000|command|9|0|501|4.055

\* device|1000 = Select device number 1000

command|9 = Command number 9

0|501|4.055 = Axis flag 0, parameter 501, new value 4.055

All items have to be delimited by using a pipe '|' character.

Results of all writing operations are written to the devices log file log\_devicenr\_date.txt in subfolder \log of the CNCnetPDM directory.

## ADJUST ITEMS

This device driver enables to dynamically add, group, enable or disable items and change their names. The INI file automatically created by the device driver for every machine contains sections that allow you to control its behavior.

Section [GENERAL] contains information about global parameters used by the driver on startup, usually there's no need to change the preset values for Axisflag and System. Commands defines the number of commands you'd like to execute, maximum value = 30.

Numeric section [0] defines how you'd like to detect if the controller is accessible. If this command does not succeed the driver assumes that the controller is not accessible. In this case all other commands are not executed.

Description of the numeric sections:

Entry	Description
[0]	Section identifier, numbers > 0 are queried items
Axisflag	If a command is related to an axis (e.g. Axis position) you can enter a number to get data related to a specific axis, e.g. 1
ParameterNumber	If column Command in the documentation ends with '(X)' you can enter a number the driver should use for this command. If you enter 500 for SECT_COMMON_VAL1(x) the driver queries common variable 500

Datatype	<p>Datatype to be used for output. As this is not well documented it's a bit of guesswork but shouldn't be too difficult. For commands that obviously output integer (numeric) values like program number you can use T_CHAR, T_SHORT, T_LONG, T_UCHAR, T_USHORT, T_ULONG. Recommended = T_LONG</p> <p>For floating point numbers like axis position you have to use T_DOUBLE or T_FLOATBIN. Recommended = T_FLOATBIN</p> <p>For characters like program comment you have to use T_STR</p>
Command	<p>Command to be executed e.g. SECT_RUN_STATUS for running status. To change a command first select a <b>section</b>. Then, simply copy the selected command (e.g. SECT_COMMON_VAL1(x)) and paste it into the INI file. Make sure that you also enter an appropriate data type (e.g. T_LONG) for the command in Datatype.</p>
Active	<p>If you set this to 0 the command is not executed</p>
Name	<p>A short name (max 5 characters) to describe the item e.g. STATN for status number</p>
Comment	<p>A comment that describes the command (optional)</p>
Output section	<p>You can output data to 3 sections that may contain up to 256 characters. 1 goes to section 'O', 2 to 'A' and 3 to 'F'. If data in one section exceeds the maximum length you can send items to a different section. Make sure you have entries CollectOrders, CollectFeeder and CollectQuality enabled (=1) in CNCnetPDM.ini.</p>

To change commands, switch items on or off or alter its name while CNCnetPDM is running open the INI file with a text editor such as notepad, make the desired changes and save the file. To apply the changes immediately you can click on the machine in CNCnetControl on the left side followed by clicking buttons 'Close' and 'Open' above the section 'Devices'.

## TIPS

An overview of all sections that lead to pages that contain commands belonging to the specific section can be found [here](#).

It is recommended to start with simple commands in sections like 'Data on current NC program and running status' or 'Data on variable (#\_\_)'.

If you enter a wrong data type for output of data in the INI file you may get data but the values are incorrect.

If you want to add additional items first adjust the number of commands in section [GENERAL]. Then, copy and paste the last numeric section and adjust the section number and its content.

## TROUBLESHOOTING

If the machine shows a red icon, state disconnected and you get just output E = 0 the machine is not reachable at all. This has nothing to do with the Citizen Mitsubishi API, it's a network issue, DNS Hostname or IP Address is wrong.

If the machine shows a green icon, state connected but you just get output E = 1 the machine is reachable but the API does not respond to commands, check your Citizen Mitsubishi setup and the parameters used in CNCnetPDM.ini (esp. Port number). You can also use the tool [MitsubishiGui.exe](#) to check if the machine replies to commands.

If you do not get any of the preconfigured items please make sure that CollectOrders is set to 1 in section [General] of CNCnetPDM.ini. If you forced items to sections 2 or 3 set CollectQuality and CollectFeeder to 1.

If specific items are not acquired check first with [MitsubishiGui.exe](#) that you get output. Also make sure that the numeric section of the item in the device drivers INI file is activated and has an entry for 'Name', inactive or items with empty names are not acquired.

## LICENSING

This device driver requires the most recent version of [CNCnetPDM](#) and also works with a free license. However, in this mode you only get output for the first item configured in your INI file. With a valid license you are able to output the result of up to 30 functions per reading cycle, see [licensing](#) for details.



## CONTROLLERS

The driver was tested and works with the following controllers on Citizen machines, others may work as well:

Controller	Mitsubishi Series
MELDAS FCA70LPC-2A	M70
MELDAS M635L	M635
M635L	M635
MELDAS FCA70LPC-2AV	M70
MELDAS FCA70LPC-2AVU	M70
M735LUC	M700
MELDAS FCA70LPC-2AVU	M70
MELDAS FCA70LPC-2AV	M70
M500	M500
MELDAS FCA730LPC-4VS	M730