Voyager Series User Manual

SwissMicros GmbH

Table of Contents

Chapter 1. Family Members

The Voyager Series of calculators were introduced by Hewlett-Packard in 1981. Each model provided different capabilities and was aimed at different user markets.

The SwissMicros' Series of the Voyager calculators come in two sizes. See the Illustrations section for a visual size comparison. The credit card sized models (e.g. DM15) and the original sized models (e.g. DM15L) contain the same electronics and use the same firmware.

In this document, all models are referenced without trailing L.

- DM10 — basic scientific calculator
- DM11 — mid-range scientific calculator
- DM12 — business/financial calculator
- DM15 — advanced scientific calculator
- DM16 — computer programmer’s calculator
- DM41 — a landscape orientated clone of the HP-41CX

Chapter 2. SwissMicros improvements on the Voyager Series

- They do not have all the bugs found in the HP-15C LE, see the bug list!
- They run up to 30 times faster (at 48MHz) than the calculators introduced in the 80s.
- The USB connection features a command line interface for saving and restoring complete calculator states.
- Save, edit and restore complete calculator states with one of SwissMicros' online utility.
- Firmware update via USB is possible using the SwissMicros Firmware Update Tool (on Windows) or standard `lpc21isp` utility on Linux and MacOS.
- The reset button is located on the back in the lower left-hand side closest to the corner.
- They are still being produced

Chapter 3. References and Resources

- SwissMicros forum for DM1x/DM41
- Encode and decode utility (html with jscript) for Save and Restore State function
- Torsten Manz' excellent HP-15C Simulator supports data exchange with the DM15 under Windows, Linux and Mac OS X

Chapter 4. Battery Replacement

SwissMicros Voyager calculators are powered by a standard (non-rechargeable) CR2032 battery.

The battery should last many years with normal use.

Powering via USB is not supported.

- Turn calculator off; calculator data is preserved for about 30 seconds without battery
- Unscrew all four back screws
- Remove calculator’s back
- Replace the battery
- Put it back together, don’t tighten the screws too much

Chapter 5. Firmware Update

It is not possible to break or brick the calculator with any update procedure.

All calculator data (i.e. programs, registers, etc.) will be lost. Make a backup if needed.

5.1. Selecting Firmware

5.1.1. The Latest Firmware

The latest firmware versions are available at https://technical.swissmicros.com/voyager/firmware/.

Download files using the 'Target save as...' function.

5.1.2. Name Conventions

The **basic firmware name** consist of two parts

```
<model>_<version>.hex

Example: DM12_V31.hex
```

where

<model>
: Calculator model. One of `DM10`, `DM11`, `D12`, `DM15`, `DM16`, `DM41`

<version>
: Firmware version, e.g. `V31`.

Extended firmware name adds name of the extension

```
<model>_<ext>_<version>.hex

Example: DM15_M80_V31.hex
```

where `<model>` and `<version>` are the same as above and

<ext>
: Name of the extension e.g. `M80`, `M1B`

5.1.3. Special DM15 Firmware Versions

There are two special firmware versions for the DM15, version `M80` and version `M1B`. Both contain

the original ROM with modifications to support more memory.

According to the HP-15C Owner's Handbook, Appendix C: Memory Allocation (pg. 214), the total allocatable memory is 64 registers and initial Memory Status is `19-46 0-0`.

The following table summarizes how this limit is modified in extended DM15 ROMs.

ROM	Alocatable registers	RAM base addr.	Initial Memory Status
DM15 (Original ROM)	64	0xc0	`19-46 0-0`
DM15_M80	128	0x80	`19-110 0-0`
DM15_M1B	229	0x1b	`19-211 0-0`

The default distribution version on DM15 is `DM15_M1B` with 229 Registers to supply users with as much memory as possible. Please, keep in mind the original calculator ROM isn't designed to handle so much memory and there are known at least two marginal cases when the calculator can give confusing info or wrong result:

Known problems

- The program step counter is only three digits wide. Programs with more than 999 steps should not be created.
- Solving lin.eq and matrix inversions is limited to matrices with sizes up to 8x8. Unpredictable things may happen when operating with matrices larger than 8x8, it may even completely block the calculator. More info about this is available in Appendix chapter Matrices in Extended DM15.

To avoid potential problems, the firmware version `DM15_M80` can be used, where such effects should be less prominent, or even firmware version `DM15` with the original ROM code.

5.2. Update on Windows

5.2.1. Required software

- Install Serial Port driver
- Find Serial Port Number
- Download and unpack the SwissMicros Voyage Firmware Tool for updating the calculator.

5.2.2. Update

- Connect the calculator with a miniUSB cable to the computer.
- Start the Voyage Firmware Tool
- Press "Open File ..." and choose the firmware file
- Choose "Serial Port", see find Serial Port Number
- Press the "Program" button in the application.

- Press the reset button with a paperclip to initiate the flash process.
  If the calculator beeps and displays `SERIAL CONSOLE`, reset again until the flash process begins.
- The flash progress is displayed in the application and takes about 10-20 seconds depending on the firmware version.
- After the flash procedure is done, press the reset button again and disconnect from the computer.

5.3. Update on MacOS

5.3.1. Required Software

- Ensure the Serial Port driver is installed.
- Find Serial Port Name
- Download and install MacPort (www.macports.org)
- Open the Terminal application (Applications→Utilities→Terminal) and run the following commands:

```
sudo port selfupdate
sudo port upgrade outdated
sudo port install lpc21isp
```

- Close the Terminal application

5.3.2. Firmware Update

- Open the Terminal application (Applications→Utilities→Terminal)
- Connect the calculator with a miniUSB cable to the computer.
- Find USB Serial Port Name
- Run command:

```
sudo lpc21isp -control -controlinv -hex <firmware_file> <serial_port> 115200 12000
```

replace `<firmware_file>` with the firmware file name
and `<serial_port>` with Serial Port Name found in previous step

Example:

```
sudo lpc21isp -control -controlinv -hex DM15_V31.hex /dev/tty.usbserial-0001 115200
12000
```

- Press the reset button with a paperclip to initiate the flash process.

If the calculator beeps and displays `SERIAL CONSOLE`, reset again until the flash process begins.

- After the flash procedure is done, press the reset button again and disconnect from the computer.
- Close the Terminal application

5.4. Update on Linux

5.4.1. Required Software

- Install the `lpc21isp` tool for firmware update.

  On Debian based systems run

  ```
  sudo apt-get install lpc21isp
  ```

  or run package installer according to the Linux distribution.

5.4.2. Firmware Update

- Find USB Serial Port Name
- Start the flashing utility by

  ```
  sudo lpc21isp <firmware_file> <serial_port> 115200 12000
  ```

  replace `<firmware_file>` with the appropriate firmware filename and `<serial_port>` with Serial Port Name found in the previous step

  Example:

  ```
  sudo lpc21isp DM15_V31.hex /dev/ttyUSB0 115200 12000
  ```

- Wait for message "Synchronizing (ESC to abort)". Press the reset button with a paperclip to initiate the flash process.
  If the calculator beeps and displays `SERIAL CONSOLE`, reset again until the flash process begins.
- After the flash procedure is done, press the reset button again and disconnect from the computer.

Chapter 6. Serial Console

Useful links:

- Encode and decode utility (html with jscript) for Save and Restore State function
- A great HP-15C Simulator by Torsten Manz for Windows, Linux and Mac OS X supports data exchange with the DM15.

6.1. Connection to Serial Console

- Prepare Serial Console program:
  - Console on Windows
  - Console on MacOS
  - Console on Linux
- Activate Serial Console on the calculator

Remove the USB cable from the calculator if not needed. Residual currents of the CP2102 chip can cause shortening of the battery life when left connected for prolonged periods of time after the calculator is turned OFF.
To be completely sure any residual charges are removed, turn the calculator ON/OFF after the USB cable is removed.

6.2. Enable Serial Console Mode

Serial Console Mode is enabled by btn:[ON] + btn:[C], see Special Key Combinations chapter for more details.

6.3. Serial Port Drivers

6.3.1. Windows

Download and install Silicon Labs CP2102 USB Driver from Silicon Labs drivers page/Downloads.

Windows users should NOT use the "Universal" Windows driver, use the older Windows drivers instead, the newer type has installation issues on Windows 10.

6.3.2. MacOS

Driver is part of the system since "Big Sur".

For earlier systems download and install the CP2102 Mac OSX driver from Silicon Labs drivers page/Downloads.

6.3.3. Linux

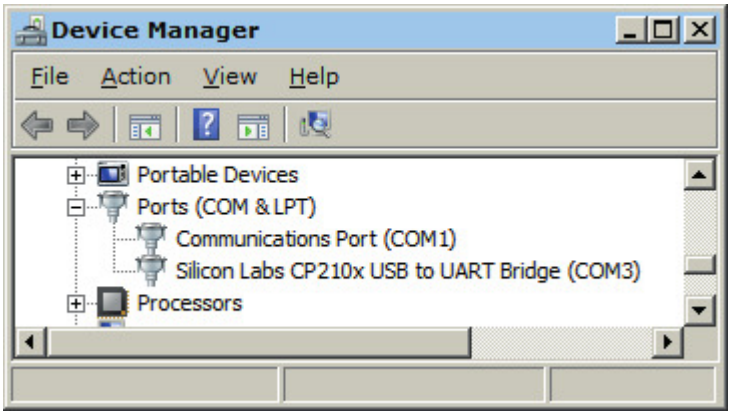
No action needed.

The cp210x driver is part of the kernel build on all major Linux systems.

6.4. Finding Serial Port

6.4.1. Windows

- Install Serial Port driver
- Connect the calculator with a miniUSB cable to the computer.
- Open Device Manager and find the `Silicon Labs CP210x` line,
- Note the name given to the USB Serial Port (here : `COM3`)



6.4.2. MacOS

- Ensure the Serial Port driver is installed.
- Connect the calculator with a miniUSB cable to the computer.
- Find the name of the USB Serial Port

```
ls /dev/tty.usbserial*
```

```
or with Silicon Labs drivers
```

```
ls /dev/tty.SLAB*
```

The port is usually `/dev/tty.usbserial-0001` or `/dev/tty.SLAB_USBtoUART`
If the port isn't present, check the USB connection to calculator.

6.4.3. Linux

- Connect the calculator with a miniUSB cable to the computer.
- In the console run:

```
dmesg
```

  There should be a message near the end like `cp210x converter now attached to ttyUSB0`

  If so, the full path to the USB Serial Port is `/dev/ttyUSB0`

6.5. Console on Windows

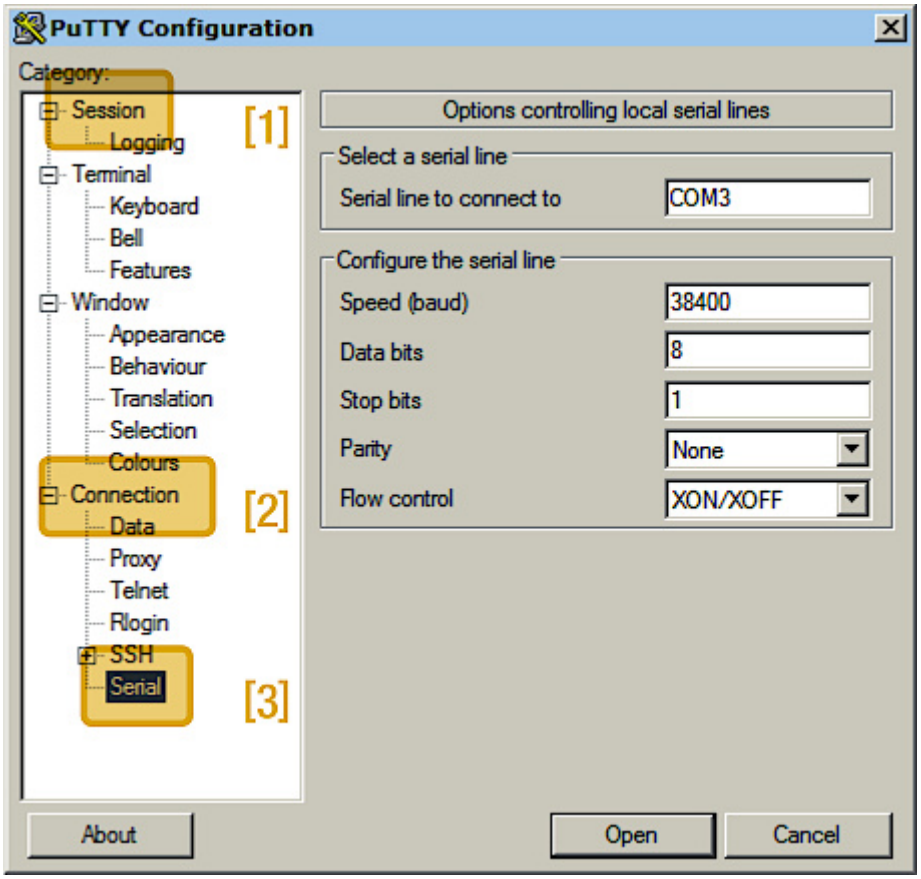
6.5.1. Required Software

Setting up a USB Serial Port connection in Windows

1. Install Serial Port driver
2. Download and install PuTTY (http://www.putty.org/)

6.5.2. Connection

- Connect the calculator with a miniUSB cable to the computer.
- Enable the Serial Console Mode on the calculator: btn:[ON] + btn:[C] see Special Key Combinations chapter for more details.
- Find the Serial Port Name
- Start PuTTY, go to Session [1] and select Serial in the panel on the right; go to Connection [2] → Serial [3] and configure the serial line with the following values:
  - Serial line to connect to : Serial Port Name found in previous step
  - Speed : 38400
  - Data bits : 8
  - Stop bits : 1
  - Parity : None
  - Flow control : XON/XOFF (or none)



- Click "Open" to start monitoring Serial Data.
- The Serial Connection is now established. The monitor window is blank when it starts. Press «?» for help.
- Pressing any button on the calculator ends the Serial Console Mode.

6.6. Console on MacOS

6.6.1. Required Software

- Ensure the serial driver is installed.
- The `screen` utility is preinstalled on MacOS system.

6.6.2. Connection

- Connect the calculator with a miniUSB cable to the computer.
- Enable the Serial Console Mode on the calculator: btn:[ON] + btn:[C] see Special Key Combinations chapter for more details.
- Find the Serial Port Name
- Open the Terminal application (Applications→Utilities→Terminal)
- Start a console session in terminal application:

```
screen <serial_port> 38400,8,n,1
```

replace `<serial_port>` with serial port name found in previous step

Example:

```
screen /dev/tty.usbserial-0001 38400,8,n,1
```

- The Serial Connection is now established. The monitor window is blank when it starts. Press «?» for help.
- Pressing any button on the calculator ends the Serial Console Mode.
- End the console session: press btn:[CONTROL] + btn:[a] then btn:[k] then btn:[y]
- Disconnect the calculator
- Close the Terminal application

6.7. Console on Linux

6.7.1. Required Software

- Install the `putty`.

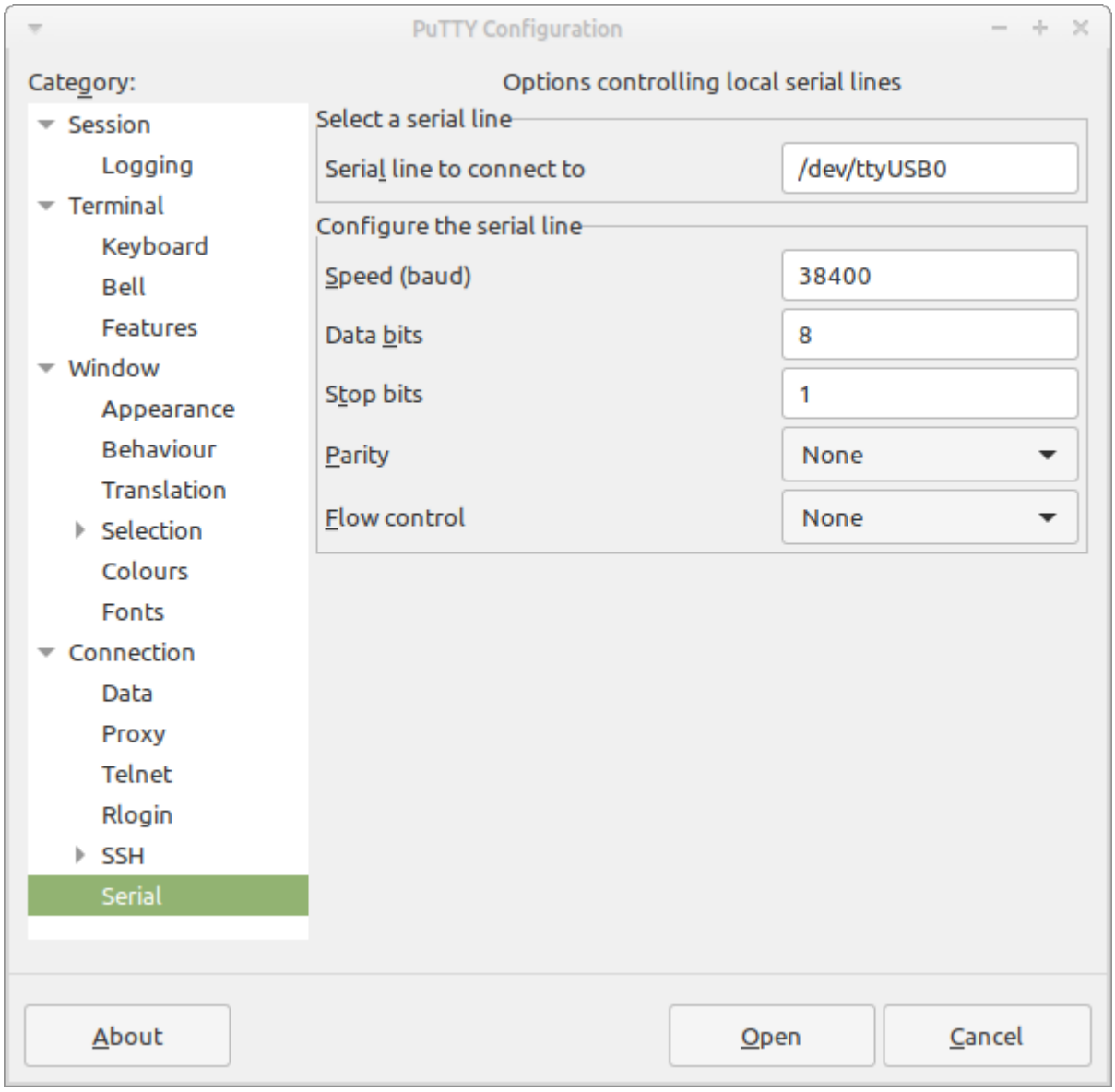
  On Debian based systems run

  ```
  sudo apt-get install putty
  ```

  or run package installer according to the Linux distribution.

6.7.2. Connection

- Connect the calculator with a miniUSB cable to the computer.
- Find the Serial Port Name
- Start PuTTY
- go to "Session" and select Serial in the panel on the right
- go to Connection → Serial and configure the serial line using the following values. The rest should be left to default (check image below):
  - Serial line to connect to: use Serial Port Name found in the previous step
  - Speed : 38400
  - No flow control



- Enable the Serial Console Mode on the calculator: btn:[ON] + btn:[C] see Special Key Combinations chapter for more details.
- Click "Open" to start monitoring Serial Data.
- The Serial Connection is now established. The monitor window is blank when it starts. Press «?» for help.
- Pressing any button on the calculator ends the Serial Console Mode.

Chapter 7. Commands in Serial Console

Once the serial connection to the calculator is established, calculator responds to switching to Serial Console Mode with text:

```
<model> ready
```

where `<model>` is calculator model, e.g. for `DM15_M80` it writes to the serial console

```
DM15_M80 ready
```

This also indicates that the serial console is ready to accept commands.

As a response to `ENTER` in serial console window the calculator responds with **command prompt**, which is
`DM41>>` - for DM41 model, or
`VOYAGER >>` - for other models

A command prompt is also displayed whenever the calculator finishes a command and is ready for the next one.

If an invalid command is entered or `?` command is used, the calculator responds with firmware version and list of available commands, e.g. for `DM15_M80_V31`:

```
VOYAGER >> ?
Firmware DM15_M80_V32

?          Help
p          Key press
s          Dump memory
l          Load memory
ct         Console timeout
bootloader Invoke Bootloader
t          Get time
ts         Set time
td         Update time
kb         Toggle keyboard output to console
d          Toggle display output to console
b          Read battery voltage

Type '<command> ?' for params
VOYAGER >>
```

Next chapters describe several important commands in more details.

7.1. Save and Restore State

Here *state* refers to the complete contents of the calculator, i.e. stack, registers, programs, etc.

Saving calculator memory

- Establish a connection to the Serial Console
- To dump contents of calculator memory, enter:
  `s`
- The memory dump is displayed followed by command prompt
- This dump can be copied, pasted and saved to a text file (including the short line at the top which describes the calculator model) to be later restored to the calculator

Restoring calculator memory

- Establish a connection to the Serial Console
- Engage calculator Restore Mode by entering:
  `l` (lowercase «L»)
- The command prompt changes to:
  `Waiting for data···`
- Copy all characters from the memory dump, including the short line at the top which describes the calculator model
- Paste it in the monitor window
- Monitor window outputs:
  `Read OK`
  and the command prompt is displayed
- Calculator memory is now restored

7.2. Set time and date

Getting calculator internal date and time

- Establish a connection to the Serial Console
- In the serial monitor window, enter:
  `t`
- The monitor window returns date, time and day of the week in format:
  `YYYY-MM-DD HH:MM:SS DDD`

Setting date and time

- Establish a connection to the Serial Console
- The command to set date and time is
  `ts <YYYYMMDD> <HHMMSS>`

for instance for September 22nd of 2016, 6:30 PM, enter the following command:
`ts 20160922 183000`

- Now, date and time are set

Chapter 8. Calculator configuration

Configuration parameters can be directly changed by using special key combinations. Because the different models of the Voyager Series have different button layouts and almost, but not quite, similar configuration possibilities, in this chapter keys are referenced by their row/column place, e.g. btn:[(2-3)] means the key in row 2 and column 3.

See Special Key Combinations for more details of how to invoke this function on a particular calculator model.

8.1. CPU Speed

Invocation: btn:[ON] + btn:[(1-9)]

The calculator supports two CPU speeds:

- Normal - CPU runs at 12MHz
- Fast - CPU runs at 48MHz

This key combination cycles through them.

8.2. LCD Contrast Configuration

Invocation: btn:[ON] + btn:[(1-6)]

This key combination starts LCD contrast configuration.

LCD contrast configuration consists of two parameters:

- param1: LCD brightness
- param2: LCD voltage

The LCD configuration:

- is automatically left without any change after 10 seconds of inactivity.
- can be left without configuration change at any time by btn:[ON] key
- current configuration can be confirmed and saved by btn:[ENTER] key

8.3. LCD Contrast Reset

Invocation: btn:[ON] + btn:[(2-6)]

This key combination resets the LCD contrast configuration to the default values.

8.4. Fonts

Invocation: btn:[ON] + btn:[(1-7)]

The models DM10, DM11, DM12, DM15 and DM16 include several font types.

This key combination cycles through them.

8.5. Annunciator Location

Invocation: btn:[ON] + btn:[(2-2)]

This key combination toggles between location for the Voyager annunciators, displayed either below the digits (default) or above the digits.

Chapter 9. Special Key Combinations

The SwissMicros Voyager calculators have some special key combinations, some are the same as on the HP calculators, some are unique to SwissMicros calculators.

Usage:

1. Turn off calculator
2. Press and hold one of the buttons listed below
3. Press and release btn:[ON] button
4. Release first button pressed

9.1. DM10

Table 1. Key Function Table for DM10

Key	Function description
btn:[√x]	Display time/date for 5 minutes
btn:[e^x]	Benchmark. Evaluate speed compared to original calculator
btn:[%]	Time/date adjustment. Usage: Up btn:[+], Down btn:[−], Next btn:[□], Prev btn:[÷], btn:[ENTER] to Set, btn:[ON] to Cancel
btn:[10^x]	Switch to serial console (see Serial Console for details)
btn:[1/x]	System info. Firmware version, battery voltage, current frequency, etc.
btn:[CHS]	Change LCD contrast. Param1: up btn:[+], down btn:[−], Param2: up btn:[Σ+], down btn:[3]
btn:[EEX]	Reset LCD contrast to default values.
btn:[STO]	Adjust RTC decrement -1h
btn:[RCL]	Adjust RTC increment +1h
btn:[7]	Cycle through available fonts
btn:[9]	Configure CPU speed. Toggle between 12MHz and 48MHz
btn:[÷]	Keyboard self test
btn:[□]	Self test
btn:[−]	Power reset
btn:[+]	Endless test

Key	Function description
btn:[.]	Toggle decimal delimiter
btn:[GTO]	Toggle annunciator position

9.2. DM11

Table 2. Key Function Table for DM11

Key	Function description
btn:[A]	Display time/date for 5 minutes
btn:[B]	Benchmark. Evaluate speed compared to original calculator
btn:[C]	Switch to serial console (see Serial Console for details)
btn:[E]	System info. Firmware version, battery voltage, current frequency, etc.
btn:[CHS]	Change LCD contrast. Param1: up btn:[+], down btn:[−], Param2: up btn:[Σ+], down btn:[3]
btn:[EEX]	Reset LCD contrast to default values.
btn:[STO]	Adjust RTC decrement -1h
btn:[RCL]	Adjust RTC increment +1h
btn:[7]	Cycle through available fonts
btn:[9]	Configure CPU speed. Toggle between 12MHz and 48MHz
btn:[÷]	Keyboard self test
btn:[□]	Self test
btn:[−]	Power reset
btn:[+]	Endless test
btn:[.]	Toggle decimal delimiter
btn:[SST]	Time/date adjustment. Usage: Up btn:[+], Down btn:[−], Next btn:[□], Prev btn:[÷], btn:[ENTER] to Set, btn:[ON] to Cancel
btn:[GTO]	Toggle annunciator position

9.3. DM12

Table 3. Key Function Table for DM12

Key	Function description
btn:[n]	Display time/date for 5 minutes
btn:[i]	Benchmark. Evaluate speed compared to original calculator
btn:[PV]	Switch to serial console (see Serial Console for details)
btn:[FV]	System info. Firmware version, battery voltage, current frequency, etc.
btn:[CH S]	Change LCD contrast. Param1: up btn:[+], down btn:[−], Param2: up btn:[CHS], down btn:[3]
btn:[EE X]	Reset LCD contrast to default values.
btn:[ST O]	Adjust RTC decrement -1h
btn:[RC L]	Adjust RTC increment +1h
btn:[7]	Cycle through available fonts
btn:[9]	Configure CPU speed. Toggle between 12MHz and 48MHz
btn:[÷]	Keyboard self test
btn:[□]	Self test
btn:[−]	Power reset
btn:[+]	Endless test
btn:[.]	Toggle decimal delimiter
btn:[y^x]	Time/date adjustment. Usage: Up btn:[+], Down btn:[−], Next btn:[□], Prev btn:[÷], btn:[ENTER] to Set, btn:[ON] to Cancel
btn:[1/ x]	Toggle annunciator position

9.4. DM15

Table 4. Key Function Table for DM15

Key	Function description
btn:[A]	Display time/date for 5 minutes

Key	Function description
btn:[B]	Benchmark. Evaluate speed compared to original calculator
btn:[C]	Switch to serial console (see Serial Console for details)
btn:[E]	System info. Firmware version, battery voltage, current frequency, etc.
btn:[CHS]	Change LCD contrast. Param1: up btn:[+], down btn:[−], Param2: up btn:[Σ+], down btn:[3]
btn:[EEX]	Reset LCD contrast to default values.
btn:[STO]	Adjust RTC: decrement -1h
btn:[RCL]	Adjust RTC: increment +1h
btn:[7]	Cycle through available fonts
btn:[9]	Configure CPU speed. Toggle between 12MHz and 48MHz
btn:[÷]	Keyboard self test
btn:[□]	Self test
btn:[−]	Power reset
btn:[+]	Endless test
btn:[.]	Toggle decimal delimiter
btn:[SST]	Time/date adjustment. Usage: Up btn:[+], Down btn:[−], Next btn:[□], Prev btn:[÷], btn:[ENTER] to Set, btn:[ON] to Cancel
btn:[GTO]	Toggle annunciator position

9.5. DM16

Table 5. Key Function Table for DM16

Key	Function description
btn:[A]	Display time/date for 5 minutes
btn:[C]	Switch to serial console (see Serial Console for details)
btn:[E]	System info. Firmware version, battery voltage, current frequency, etc.
btn:[F]	Change LCD contrast. Param1: up btn:[+], down btn:[−], Param2: up btn:[CHS], down btn:[3]

Key	Function description
btn:[BIN]	Reset LCD contrast to default values.
btn:[STO]	Adjust RTC decrement -1h
btn:[RCL]	Adjust RTC increment +1h
btn:[7]	Cycle through available fonts
btn:[9]	Configure CPU speed. Toggle between 12MHz and 48MHz
btn:[÷]	Keyboard self test
btn:[□]	Self test
btn:[−]	Power reset
btn:[+]	Endless test
btn:[.]	Toggle decimal delimiter
btn:[GSB]	Time/date adjustment. Usage: Up btn:[+], Down btn:[−], Next btn:[□], Prev btn:[÷], btn:[ENTER] to Set, btn:[ON] to Cancel
btn:[GTO]	Toggle annunciator position

9.6. DM41

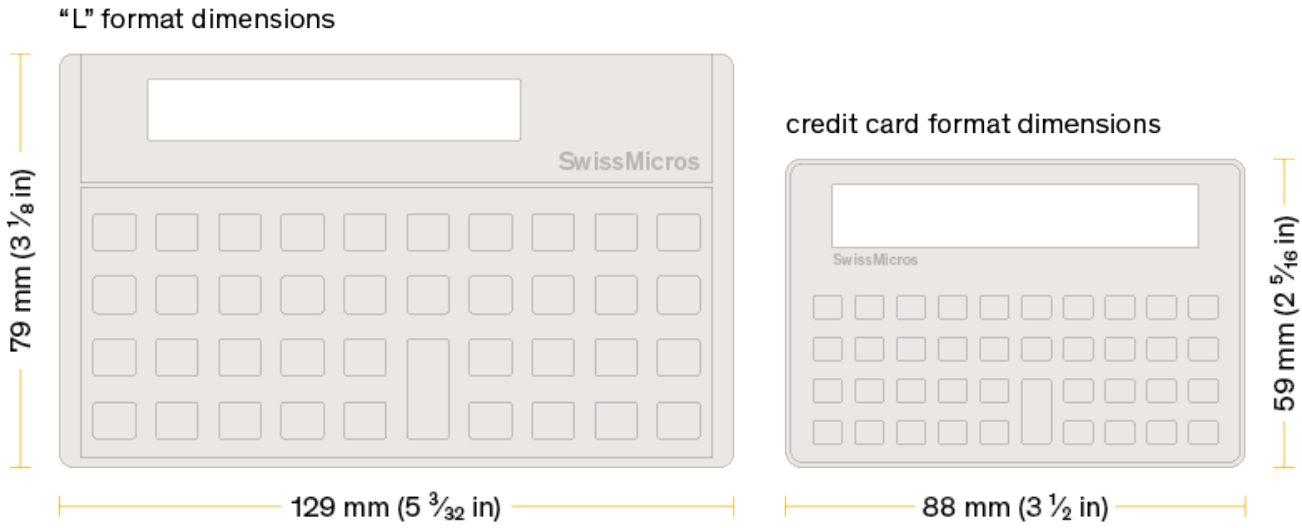
Table 6. Key Function Table for DM41

Key	Function description
btn:[A]	Display time/date for 5 minutes
btn:[C]	Switch to serial console (see Serial Console for details)
btn:[E]	System info. Firmware version, battery voltage, current frequency, etc.
btn:[CHS]	Change LCD contrast. Param1: up btn:[R/S], down btn:[3], Param2: up btn:[.], down btn:[2]
btn:[EEX]	Reset LCD contrast to default values.
btn:[STO]	Adjust RTC decrement -1h
btn:[RCL]	Adjust RTC increment +1h
btn:[8]	Configure CPU speed. Toggle between 12MHz and 48MHz

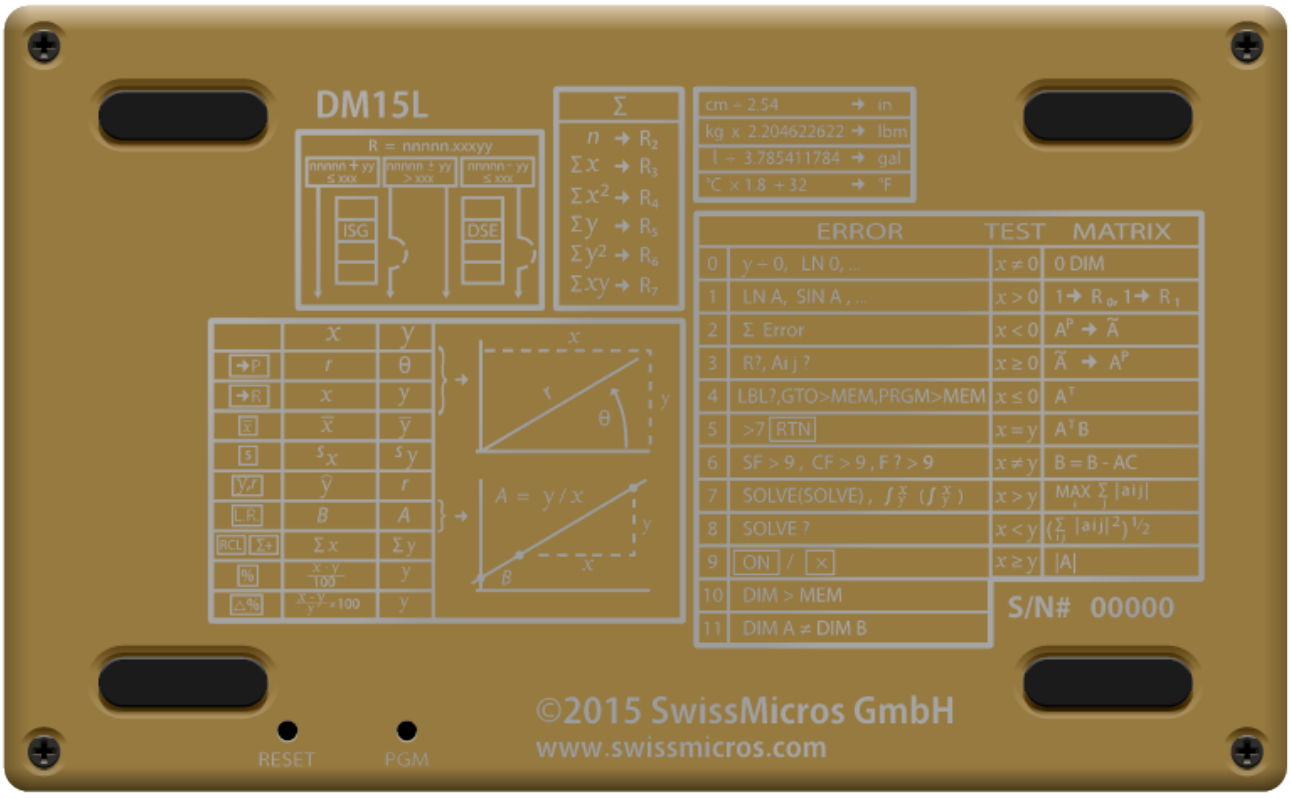
Key	Function description
btn:[9]	Keyboard self test
btn:[x□ y]	Time/date adjustment. Usage: Up btn:[+], Down btn:[−], Next btn:[□], Prev btn:[÷], btn:[ENTER] to Set, btn:[ON] to Cancel
btn:[R□]	Toggle annunciator position

Chapter 10. Illustrations

10.1. Horizontal units size comparison



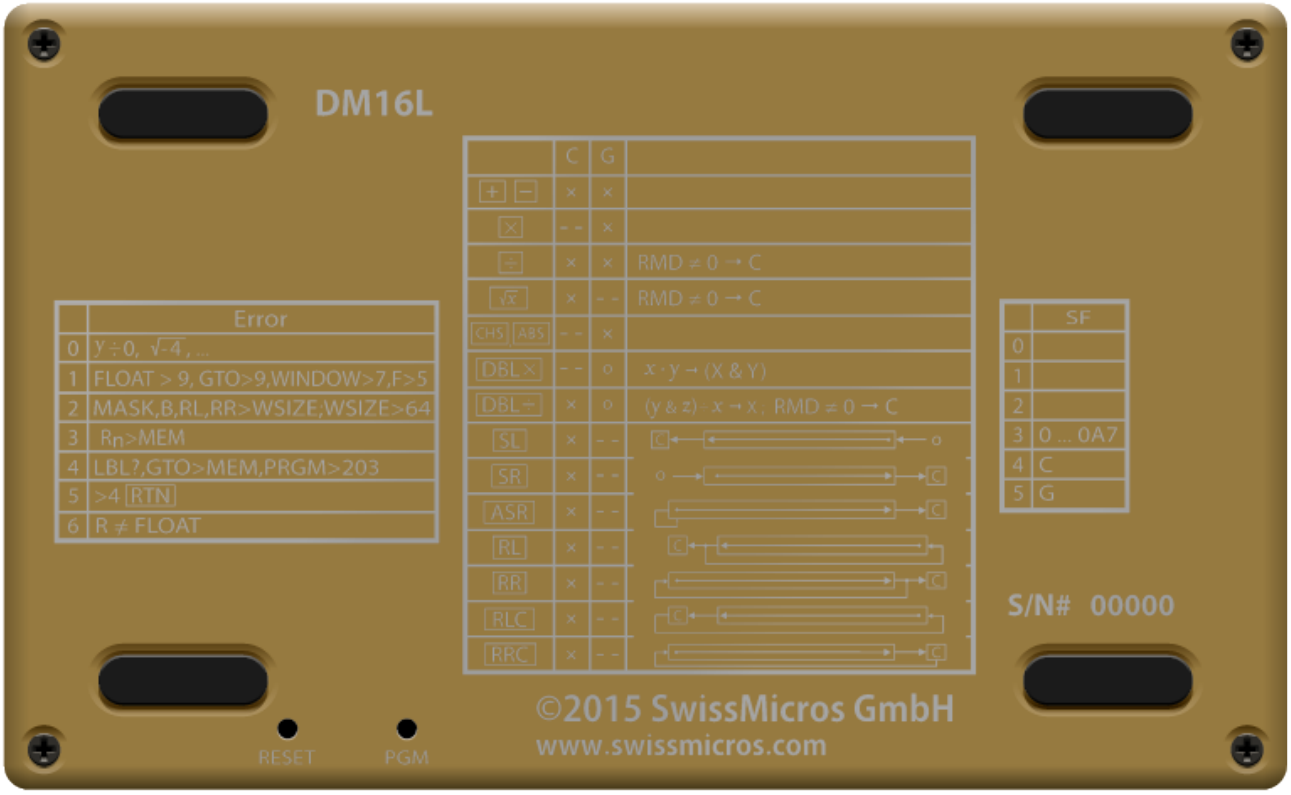
10.2. DM15


DM15L
R = nnnnn.xxxyy
ISG
DSE
Σ
n → R2
Σx → R3
Σx2 → R4
Σy → R5
Σy2 → R6
Σxy → R7
cm ÷ 2.54 → in
kg x 2.204622622 → lbm
l ÷ 3.785411784 → gal
°C x 1.8 +32 → °F
ERROR TEST MATRIX
0 y ÷ 0, LN 0, ... x ≠ 0 0 DIM
1 LN A, SIN A, ... x > 0 1→ R0, 1→ R1
2 Σ Error x < 0 AP → Ã
3 R?, Ai j ? x ≥ 0 Ã → AP
4 LBL?,GTO>MEM,PRGM>MEM x ≤ 0 AT
5 >7 RTN x = y AT B
6 SF > 9 , CF > 9 , F ? > 9 x ≠ y B = B - AC
7 SOLVE(SOLVE) , ∫yx (∫yx) x > y MAX Σj |aij|
8 SOLVE ? x < y (Σj |aij|2)1/2
9 ON / × x ≥ y |A|
10 DIM > MEM
11 DIM A ≠ DIM B
S/N# 00000
©2015 SwissMicros GmbH
www.swissmicros.com
RESET
PGM


15.000 15
RAD
Swiss Micros DM-15
Silver Edition
A B C D E MATRIX FIX SCI ENG SOLVE
LBL HYP DIM (i) I RESULT x≷ DSE ISG
CLEAR
PSE Σ PRGM REG PREFIX RAN # →R →H.MS →RAD R⇔Im
FRAC USER x! ŷ,r L.R. Py,x
ON f g STO RCL ENTER
LST x
TEST

10.3. DM16

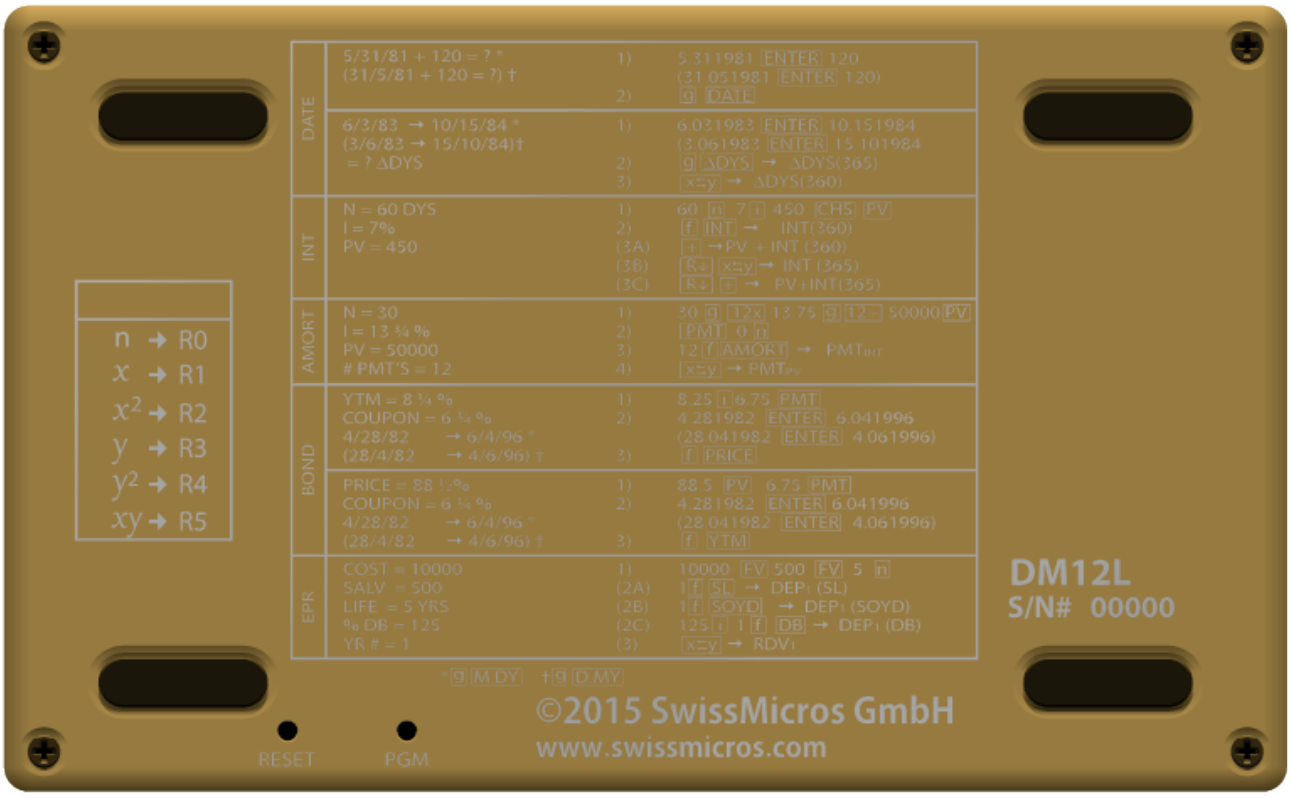

16ABCDEF H
DM 16L
SwissMicros
SL SR RL RR RLn RRn MASKL MASKR RMD XOR
A B C D E F 7 8 9 ÷
LJ ASR RLC RRC RLCn RRCn #B ABS DBLR DBL÷
x≷(i) x≷I SHOW SB CB B? AND
GSB GTO HEX DEC OCT BIN 4 5 6 ×
RTN LBL DSZ ISZ √x 1/x SF CF F? DBL×
(i) I PRGM CLEAR REG PREFIX WINDOW 1'S SET COMPL 2'S UNSGN NOT
R/S SST R↓ x≷y ← ENTER 1 2 3 −
P/R BST R↑ PSE CLx LSTx x≤y x<0 x>y x>0
WSIZE FLOAT MEM STATUS EEX OR
ON f g STO RCL 0 • CHS +
< > x≠y x≠0 x=y x=0


DM16L
Error
0 y÷0, √-4, ...
1 FLOAT > 9, GTO>9,WINDOW>7,F>5
2 MASK,B,RL,RR>WSIZE,WSIZE>64
3 Rn>MEM
4 LBL?,GTO>MEM,PRGM>203
5 >4 RTN
6 R ≠ FLOAT
C G
RMD ≠ 0 → C
RMD ≠ 0 → C
x · y → (X & Y)
(y & z) - x → x, RMD ≠ 0 → C
SF
0
1
2
3 0 ... 0A7
4 C
5 G
S/N# 00000
©2015 SwissMicros GmbH
www.swissmicros.com
RESET
PGM


16ABCDEF H
Swiss Micros DM-16
Silver Edition
SL SR RL RR RLn RRn MASKL MASKR RMD XOR
A B C D E F 7 8 9 ÷
LJ ASR RLC RRC RLCn RRCn #B ABS DBLR DBL÷
x≷(i) x≷I SHOW SB CB B? AND
GSB GTO HEX DEC OCT BIN 4 5 6 ×
RTN LBL DSZ ISZ √x BIT 1/x SF CF F? DBL×
(i) I CLEAR PRGM REG PREFIX WINDOW SET COMPL 1'S 2'S UNSGN NOT
R/S SST R↓ x≷y ← ENTER 1 2 3 −
P/R BST R↑ PSE CLx LSTx x≤y x<0 x>y x>0
WSIZE FLOAT MEM STATUS EEX OR
ON f g STO RCL 0 • CHS +
< > x≠y x≠0 x=y x=0

10.4. DM12

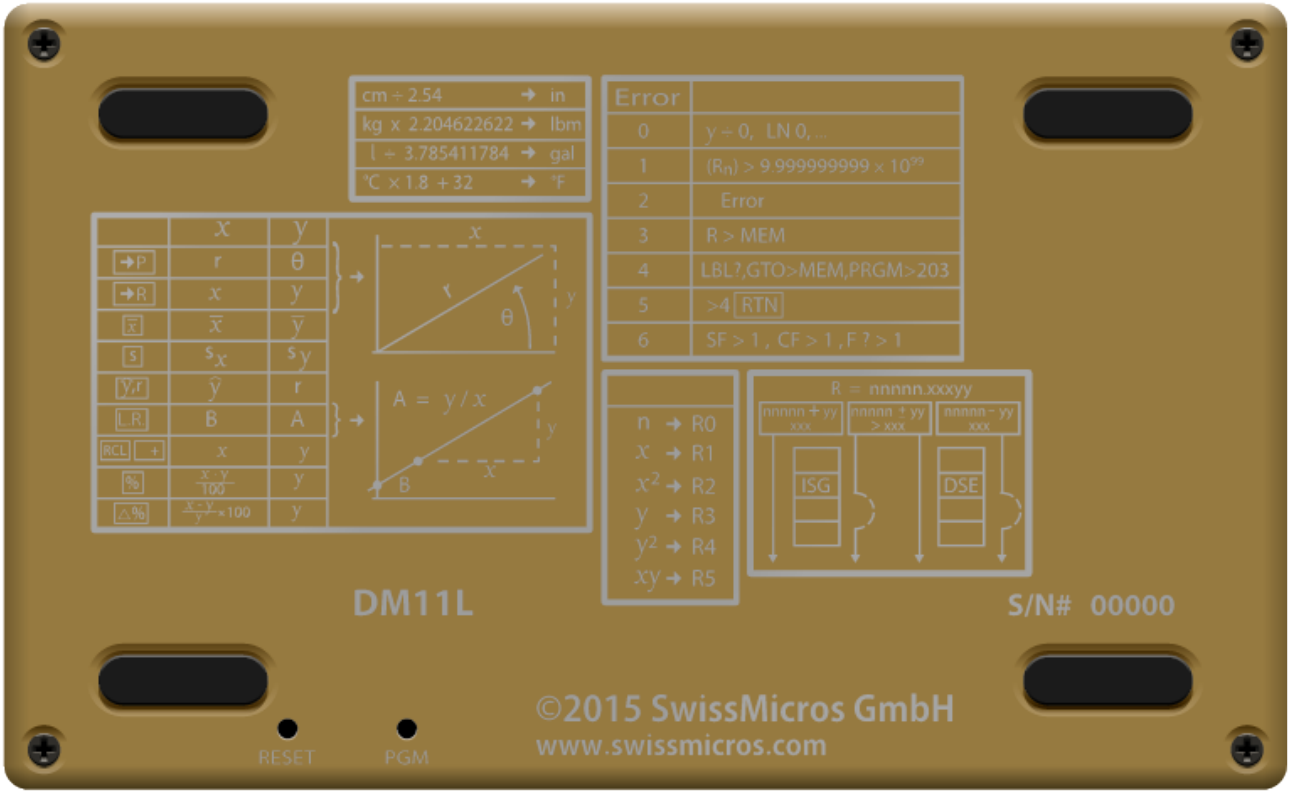

12,000.00
DM 12L
SwissMicros
AMORT INT NPV RND IRR
n i PV PMT FV CHS 7 8 9 ÷
12 x 12÷ CFo CFj Nj DATE BEG END MEM
PRICE BOND YTM SL DEPRECIATION SOYD DB
yx 1/x %T Δ% % EEX 4 5 6 ×
√x ex LN FRAC INTG ΔDYS D.MY M.DY x̄w
P/R Σ PRGM CLEAR FIN REG PREFIX
R/S SST R↓ x≷y CLx 1 2 3 −
PSE BST GTO x≤y x=0 ENTER x̂,r ȳ,r nI
LSTx
ON f g STO RCL 0 • Σ+ +
x̄ s Σ-


DATE
INT
AMORT
BOND
EPR
n → R0
x → R1
x² → R2
y → R3
y² → R4
xy → R5
DM12L
S/N# 00000
RESET
PGM
©2015 SwissMicros GmbH
www.swissmicros.com


1 2,0 0 0.0 0
Swiss Micros DM-12
Silver Edition
AMORT INT NPV RND IRR
BOND
PRICE YTM
DEPRECIATION
SL SOYD DB
CLEAR
P/R Σ PRGM FIN REG PREFIX
ON f g STO RCL
ENTER
LSTx

10.5. DM11


11.000 11
RAD
SwissMicros
DM 11L
A B C D E π FIX SCI ENG x≤y
LBL HYP x≷(i) (i) I →R x≷I DSE ISG x>y
SST GTO SIN COS TAN EEX 4 5 6
BST HYP-1 SIN-1 COS-1 TAN-1 →P SF CF F? x>0
PSE Σ PRGM CLEAR REG PREFIX RAN # Py,x →H.MS →RAD x≠y
R/S GSB R↓ x≷y ← ENTER 1 2 3
P/R RTN R↑ RND CLx Cy,x →H DEG x≠0
FRAC USER x! ŷ,r L.R. x=y
ON f g STO RCL 0 • Σ+ +
INT MEM LSTx x̄ s Σ- x=0


cm ÷ 2.54 → in
kg x 2.204622622 → lbm
l ÷ 3.785411784 → gal
°C x 1.8 + 32 → °F
Error
0 y ÷ 0, LN 0,...
1 (Rn) > 9.999999999 x 10^99
2 Error
3 R > MEM
4 LBL?,GTO>MEM,PRGM>203
5 >4 RTN
6 SF > 1 , CF > 1 , F ? > 1
n → R0
x → R1
x² → R2
y → R3
y² → R4
xy → R5
R = nnnnn.xxxyy
ISG
DSE
DM11L
S/N# 00000
©2015 SwissMicros GmbH
www.swissmicros.com
RESET
PGM


11.000
11
RAD
Swiss Micros DM-11
Silver Edition
ON
f
g
ENTER

10.6. DM41


DM41L
ALPHA
DM 41L
SwissMicros
ON
f
a
ENTER


DM41L
S/N# 00000
EXIT
RESET
PGM
©2015 SwissMicros GmbH
www.swissmicros.com


DM41L
ALPHA
Swiss Micros DM-41

Appendix A: SDK

SwissMicros believes that the user should have full control over the hardware. SwissMicros provides an SDK to get developers started.

The Voyager Series SDK can be downloaded from: dm_lpc111x_sdk.tar.gz.

Appendix B: Matrices in Extended DM15

Solving lin.eq and matrix inversions is limited to matrices with sizes up to 8x8, by J.Fossy Weinzinger

The hp15c was a highlight of its time. It is noteworthy that the original nut code of hp15c is apparently fit for an extension of its RAM. As far as I can say, it work best for all but the inverse of a matrix larger than 8x8. The largest matrix the original hp15c could store was 8x8 - so the algorithm was optimized for that maximum in size.

The reason why the original algorithm fail for larger matrices is that it has to reorder the rows of the matrix under certain circumstances. How the rows are reordered is stored in the signs of the diagonal elements of the matrix. In a register (7 bytes - 14 nibbles) the sign is stored in a nibble as 0 for + or 9 for -. A nibble hold 4 bits - so in the sign is place for additional 3 bits. These remaining 3 bits are used to store the original row offset. This only work up to 8 rows. If a matrix has more than 8 rows, this will not work anymore. :-(

The intermediate step used for matrix division, matrix inverse and determinant of a matrix is called LU decomposition. It is described in the "HP15C advanced functions handbook" - Section 4 - "Using Matrix Operations" - "Understanding the LU Decomposition" on page 96/97 of the original manual from 1982 or on page 82/83 of the reproduction from 2012.

Do not trust the results of matrix division, matrix inverse or determinant of a matrice larger than 8x8.

An easy counter-example:

```
A(i,j) = 0 if j < (n - i + 1);
Otherwise A(i,j) = 1

b(i) = i
1 <= i,j <= n

A*x = b

=>

x(i) == 1 for all 1 <= i <= n

 ----
A * x = b

0 0 0 0 ... 0 1 x(1) = 1
0 0 0 0 ... 1 1 x(2) = 2
...
0 1 1 1 ... 1 1 x(n-1) = n-1
1 1 1 1 ... 1 1 x(n) = n
```

The following program, also available as a dump, will fill pre-dimensioned matrix A, set size of

b, fill b and set result matrix to C.

Matrix testing program

```
* f LBL D                 001-42,21,14
  RCL DIM A               002-45,23,11
  STO 2                   003-   44  2
  g TEST 6  x□y?          004-43,30, 6
  g RTN                   005-   43 32
  f MATRIX 1  R0,R1 := 0 006-42,16, 1
* f LBL 0                 007-42,21, 0
  RCL 2                   008-   45  2
  RCL 0                   009-   45  0
  -                       010-      30
  1                       011-       1
  +                       012-      40
  RCL 1                   013-   45  1
  g TEST 8  x<y?          014-43,30, 8
  g CLx                   015-   43 45
  g TEST 1  x>0?          016-43,30, 1
  1                       017-       1
  USER STO A / USER       018u   44 11
  GTO 0                   019-   22  0
  RCL 2                   020-   45  2
  1                       021-       1
  f DIM B                 022-42,23,12
  f MATRIX 1  R0,R1 := 0 023-42,16, 1
* f LBL 1                 024-42,21, 1
  RCL 0                   025-   45  0
  USER STO B / USER       026u   44 12
  GTO 1                   027-   22  1
  f RESULT C              028-42,26,13
  g RTN                   029-   43 32
```

DM15 dump of the matrix testing program

```
DM15_M1B
00  00000000000000  00000000000000  00000000000000  00b10000000000
04  000000fffff000  0000000000000c  00000000000008  c0b9ab9a000eae
08  00000000000000  2fafcbefbe2280  00000000000040  00000000000000
10  00000000000000  00000000000000  00000000000000  1a000000000000
14  01017980433000  1b2d2d2d2d2d2d  000000000005fb  00000000000000
18  00000000000000  0000000000007f  00000000c00000  00000000000000
bc  00000000000000  01000000000001  00000000000000  00000000000000
f8  00000000000000  00000000000000  00000000000000  0000b28c114bbf
fc  300191ff9bf132  104abff171a278  31faf1fb303200  91ffb27642aa0d
A: 000000fffff000  B: 000000fffffeae  C: c0b9ab9a000eae
M: f000bffffff7fb  N: 00000000000000  G: 04
```

Then use it like this:

```
<n> ENTER f DIM A
f D
RCL MATRIX b
RCL MATRIX A
/
f USER
f MATRIX 1
RCL C
RCL C
...
f USER
```

To do the test:

- Dimension A
- Call D
- Key in the calculation
- Divide b by A
- Check if all elements of C are equal to 1

If 0<n<9 all elements of C will be equal to 1, but if n>8 the algorithm will fail and C will hold totally wrong values. Due to an overflow, the display will blink.

Further investigations lead to the following results:

- Mx 1x1 .. 8x8 give correct results.
- Mx 9x9 .. 11x11 give wrong results.
- Mx 12x12 .. 14x14 never end ⇒ endless loop