

BCMiniTrain

Version 1.0

1. Copyright Notice / Disclaimer

This documentation and the program "BCMiniTrain" copyright © 2003 Christian Baltensperger (cb4yh@yahoo.com). All rights reserved. No part of this documentation or the program "BCMiniTrain" may be reproduced or transmitted in any form or by any means without express written consent from Christian Baltensperger.

Christian Baltensperger reserves the right to revise this documentation and/or the program "BCMiniTrain" without obligation on the part of Christian Baltensperger to provide notification of such revision.

CHRISTIAN BALTENSPERGER MAKES NO REPRESENTATIONS OR WARRANTIES THAT THE PROGRAM "BCMINITRAIN" AND THIS DOCUMENTATION ARE FREE OF ERRORS OR THAT THE PROGRAM "BCMINITRAIN" OR THIS DOCUMENTATION ARE SUITABLE FOR YOUR USE. THE PROGRAM "BCMINITRAIN" AND THIS DOCUMENTATION ARE PROVIDED ON AN "AS IS" BASIS. CHRISTIAN BALTENSPERGER MAKES NO WARRANTIES, TERMS OR CONDITIONS, EXPRESS OR IMPLIED, EITHER IN FACT OR BY OPERATION OF LAW, STATUTORY OR OTHERWISE, INCLUDING WARRANTIES, TERMS, OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND SATISFACTORY QUALITY. TO THE FULL EXTENT ALLOWED BY LAW, CHRISTIAN BALTENSPERGER ALSO EXCLUDES ANY LIABILITY, WHETHER BASED IN CONTRACT OR TORT (INCLUDING NEGLIGENCE), FOR DIRECT, INCIDENTAL, CONSEQUENTIAL, INDIRECT, SPECIAL, OR PUNITIVE DAMAGES OF ANY KIND, OR FOR LOSS OF REVENUE OR PROFITS, LOSS OF BUSINESS, LOSS OF INFORMATION OR DATA, OR OTHER FINANCIAL LOSS ARISING OUT OF OR IN CONNECTION WITH THE PROGRAM "BCMINITRAIN" OR THIS DOCUMENTATION, EVEN IF CHRISTIAN BALTENSPERGER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

HotSync and PalmSource are trademarks of PalmSource, Inc. or its affiliates.

2. Purpose

BCMiniTrain is a program for PalmOS 3.5 (or higher) that is purely meant for fun. It offers you the possibility to build a symbolic model railway on the screen of the handheld and run trains on it. The available building blocks are track elements (straight, curved, forked, and intersection) as well as landscape elements (trees, grass, houses, a lake, and a station). Up to 8 trains can be run on the tracks; the speed and direction can be individually set for each train. Forks can be switched and stop signals can be set and removed. Building the railway and/or the landscape can be done at the same time as trains are running. Up to 255 railway layouts can be stored.

The graphics are in black & white only and support a resolution of 160 by 160 pixels.

3. Minimum Requirements

The minimal operating system required is PalmOS 3.5. The screen resolution must be at least 160 by 160 pixels and the screen must allow tapping input.

4. Trial Version Restrictions

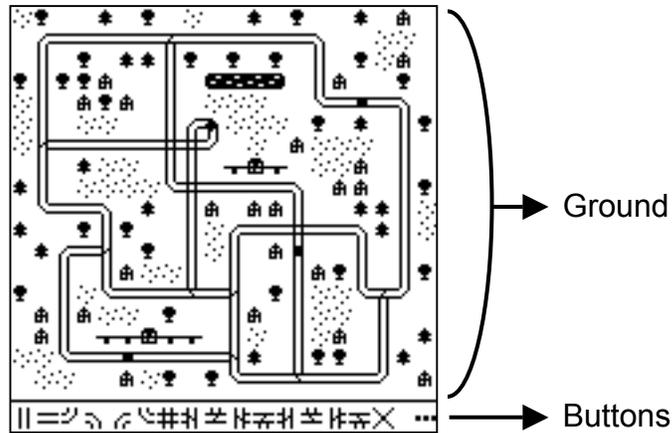
The trial version of BCMiniTrain has two restrictions: (1) no layouts can be stored (see below, "Editors/Storage Button Set"), and (2) when the program is quit and restarted, the last layout is not restored (as with the full version), but instead, the screen is cleared. This means that you can start the program and try all functions of BCMiniTrain, but you have to build a new layout from scratch every time you start the program.

To tell whether you have the trial or the full version of the program, simply watch out for a corresponding alert box when you start the program. If no alert box is displayed, you have the full version. Once the program is running, you can get that information from the "About" pop-up, which you can call using the menu.

5. Screen Layout

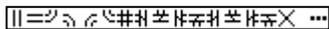
The screen is divided into two parts: the upper part (the majority of the screen) is the "ground" to place tracks and landscape elements on. This area is divided into 20 blocks in the horizontal direction and 18 blocks in the vertical direction, while each block can hold one track or landscape element.

The lower (very narrow) part of the screen is used to display buttons to control the whole program. There are four different button sets; switching from one to the next is done by tapping on the ... button at the far right.



6. Control

BCMiniTrain is controlled entirely by the buttons on the lowest part of the screen. The control mechanisms are described in the following:



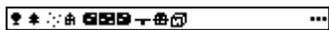
Tracks Button Set

This button set holds all available track elements: straight tracks $\parallel =$, four 90-degree curves $\curvearrowright \curvearrowleft \curvearrowup \curvearrowdown$, one intersection $\#$, eight fork elements $\begin{matrix} \text{---} \\ \diagup \\ \text{---} \end{matrix}$, $\begin{matrix} \text{---} \\ \diagdown \\ \text{---} \end{matrix}$, $\begin{matrix} \text{---} \\ \diagup \\ \text{---} \\ \diagdown \\ \text{---} \end{matrix}$, and one clear element \times . Note that for the fork elements, there are two different versions for each specific combination of directions: In this version $\begin{matrix} \text{---} \\ \diagup \\ \text{---} \end{matrix}$, a train coming from the left will turn to the left (i.e., go upwards), while in this version $\begin{matrix} \text{---} \\ \diagdown \\ \text{---} \end{matrix}$, a train coming from the left will turn to the right (i.e., go downwards). Switching the forks is done using the train control button set (see below under "Train Control Button Set").

To place a track element on the ground, first tap on the button with the desired element: a square is shown around it. Then, tap on the desired position on the ground, and the selected track element will be put there (if another element was on that block before, it will be overwritten with the new element). To remove an element, select the clear button \times and tap on the block whose element you wish to remove.

Note that placing or removing elements onto/from blocks is not possible if a train is currently located on that block. If you still wish to place or remove an element onto/from that block, you have the following choices: if the train is running, simply wait until it has left the block in question; if the train is stopped, you can use the train control button set (see below) to either lift it off the tracks or to run it so it will leave the block.

To switch to the landscape button set, tap on the \dots button.



Landscape Button Set

This button set holds landscape elements: two trees $**$, grass \circ , a house $\#$, three lake elements \square , \square , \square , and a station building with roof elements --- . In addition, this button set holds a randomizer button --- . The lake can be built in different sizes by placing any

number (or none) of center elements  between the two end elements  and . The station roof elements  can be placed to the left and the right of the station building  to construct a station that runs along several track elements (for an example, check the title screenshot of this document).

The landscape elements can be placed in the same manner as track elements, i.e., tap on one of the buttons until a square is shown around it, and then tap on the ground to place the element. To remove a landscape element, use the clear button  found in the tracks button set.

The randomizer button  will place approx. 40 landscape elements (only trees, grass, and houses) in random empty positions. This feature can be used as an inspiration as to where to place the railway tracks. To run this feature, tap the randomizer button  once: since this is an immediate action button, no frame will be shown around it, but instead, the action will start right away. However, before placing the elements a dialog will ask you whether you really want this action (note: there is no "Undo" button). If you want more elements placed, simply tap on the button again. The randomizer can be tapped anytime and any number of times; however, elements will only be placed if the randomizer "hits" on empty blocks (hence, the more elements are already placed on the ground, the less elements are likely to be put by the randomizer).

To switch to the editors/storage button set, tap on the  button.



Editors/Storage Button Set

This button set holds buttons that are needed to edit the ground (scrolling, erasing) and to store and load layouts. All buttons are immediate action buttons (i.e., tapping on them immediately starts the action): scroll    , clear ground , store , browse  , delete , and storage information . All of these actions (except storage information ) can only be performed when no trains are on the tracks (see below under "Train Control Button Set" how to lift trains off the tracks).

The scroll buttons     move the whole ground by one block in the direction in which the arrow faces (unless elements would "fall off the ground": e.g., if elements are placed in the very top row of blocks, the layout cannot be scrolled upwards). This function allows you to begin laying out tracks anywhere on the screen without having to plan the whole layout up front: if your tracks expand too much to one side and reach the edge of the ground, simply scroll the whole ground to create more space.

The clear ground button  removes all elements from the ground. If there are any unsaved modifications (see next paragraph), a dialog will ask you to explicitly confirm the action.

The store button  stores the current layout (*note: this function is disabled in the trial version*). This is only possible if modifications (elements added or removed) were made since the last storage. A completely empty ground cannot be saved. After the layout is saved, a dialog is shown to confirm the action. The maximum number of layouts that can be stored is 255.

With the browse buttons  , you can flip through the saved layouts. If you tap on any of these buttons after changes were made to the layout (since the last store or browse action, that is), you will be warned that by browsing, you will lose these changes, and you can decide whether to go on or not (if you want to preserve the changes, answer

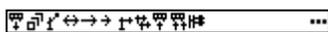
"No" to the dialog question, then store the layout with the store button , and try again). If you continue, the first stored layout is shown, and from then on, tapping on the browse buttons  will show the next or previous one of the stored layouts (while cycling around when the end is reached).

The delete button  removes the last layout browsed to (and hence currently displayed) from the storage. If changes were made since browsing, however, the delete action is not performed. Thus, to delete a stored layout, use the browse buttons  to display it, then immediately tap on the delete button . A dialog will ask you to confirm the delete action. When the layout is deleted from the storage, the ground is not changed, i.e., no elements are removed. But since they are not stored anymore, the layout is considered to contain unsaved changes. Hence, when you browse to a (different) stored layout, you will be asked whether you really want to lose these "changes".

The storage information button  does not perform any actual action, but simply brings up a dialog that shows whether the current layout is stored or not (if it is not, this means that either changes were made or the ground is completely cleared) and how many layouts are stored in total. If the current layout is stored, its sequential number is shown as well (note: the first stored layout has the number 0; if a layout is deleted, the numbers of all layouts that were "above" the deleted one are decreased by one).

Hint: to update a stored layout, proceed as follows: (1) Browse to the desired stored layout. (2) Tap on the delete button : this will remove the layout from the storage, but will leave the elements on the ground. (3) Make the desired modifications to the ground (add and/or remove elements). (4) When done, tap on the store button , and the new layout is stored again.

To switch to the train control button set, tap on the  button.



Train Control Button Set

This button set holds buttons that are needed to run the trains: put , select , go , change direction , faster , slower , stop , stop all , lift , lift all , fork and signal control .

When the put button  is tapped, a frame is shown around it, which means that you can now tap on a track element to put a train, i.e., a small square, there. A train cannot be put on a landscape element or an empty block, only on tracks. Also, just like when placing tracks or landscape elements, a train cannot be put on an element on which another train is already located. A total of 8 trains can be put onto the tracks. Hint: by putting multiple trains in a row, a single train with several wagons can be simulated.

To send a command to a train (run, stop etc.), that train must first be selected, which is indicated by a hollow square (instead of a filled square) representing the train (after a train is put onto the tracks using the put button , that train becomes selected automatically, so you can send commands to it right away). To select a different train, tap on the select button  (if necessary repeatedly) until the desired train is selected.

When the last train is selected, tapping on the select button  once again selects none of the trains (i.e., all of them are represented by a filled square and none of them are accepting commands). Tapping on the button once more selects the first train again.

The only commands for trains that work without selecting a train are stop all  and lift all , which will both be explained shortly.

The go button  starts the selected train. Note that when the train reaches either the end of the track or a track element where a different train is located at, it stops running. However, when the other train has left or the track is extended (by placing new track elements), the train will automatically continue to run, i.e., there is no need to tap on the go button  again.

The buttons change direction , faster , and slower  make the selected train change the direction, go faster, or go slower (the slowest speed is approx. one step every two seconds; the fastest speed is approx. eight steps per second). Note that these three commands can be issued even if the train is stopped. Although in that case, the changing of the direction or a new speed setting cannot be seen immediately, it is applied and will become apparent as soon as the train is starts running again.

The stop button  stops the selected train, the stop all button  stops all trains at once; the lift button  lifts the selected train off the tracks, the lift all button  lifts all trains off the tracks. This function is useful when you want to invoke storage or scrolling actions (for which no trains must be on the tracks).

The fork and signal control button  becomes active when tapped on (i.e., a frame is shown around it). In that state, you can tap (a) on any fork on the ground to change its direction setting and (b) on any horizontal or vertical track element on the ground to set or remove a stop signal on/from it (both forks and stop signals only work if no train is currently located on the referring element).

The two states of a fork are graphically represented as follows: a train coming from the left will turn to the left (i.e., go upwards) if the fork looks like this: . The train will go straight (i.e., go to the right) if the fork looks like this: . Note that it is only one single pixel that makes the difference.

A stop signal is shown like this  or this , depending on the underlying track element (remember: a stop signal can only be set on a vertical or horizontal track element).

When a train reaches a stop signal, it stops until the signal is removed (tap on the stop signal while the fork and signal control button  is active) or the train's direction is changed (tap the change direction button  when the train is selected).

Menu

For normal control, the application's menu is not used. The only item that can be selected from the menu bar ist the displaying of the "About" pop-up.

7. Data Persistence

When BCMiniTrain is quit and started again later, the layout as it was displayed before quitting is restored (unless a hard reset deleting all data was performed in the meantime) (*note: this function is disabled in the trial version*). However, the settings of the trains (position, speed etc.) and of forks, as well as the stop signals are lost: none of the trains are located on the tracks, all forks are set to their default position (the one that's initially set when they're placed onto the ground), and no stop signals are set. Similarly, layouts are stored without the information about trains, fork states, and stop signals.

The stored layouts are backed up during a HotSync operation, i.e., if a hard reset is performed on the device, the next HotSync operation will fetch the stored layouts from the backup storage (typically the Palm's backup directory on the connected PC) and re-install them on the handheld.

If a new version of BCMiniTrain is installed, it is possible that the layouts stored with the older version cannot be read with the new version. In this case, you are informed about the situation when you first start the new version of BCMiniTrain.

8. Technical Notes

The application's creator ID is 'BCMT'. It is registered with PalmSource.

The current layout and other status information is stored in the "unsaved" preferences database (two records: 'BCMT 100' and 'BCMT 200').

The layouts are stored in the database named BCMiniTrainDB (creator ID 'BCMT', type 'DATA'). The "Backup" bit of this database is set.

Memory used:

- The database size of BCMiniTrain (type 'appl') is approx. 34 kilobytes.
- The size of the preferences in the "unsaved" preferences database is 366 bytes.
- The size of the database that contains the stored layouts is 84 bytes (header) plus 380 bytes per stored layout.