User's Manual

MSX2 MSX2+ MSX turbo R MEGA ROM

1. The Story

PRESENT DAY

In a small, quaint town surrounded by dense forests and rolling hills, three childhood pals —Lupo, Dopi, and Dillo— spent their days exploring the local woods, uncovering hidden nooks, and dreaming of grand adventures.

Each had a unique talent: Lupo, the tinkerer, had a knack for solving mechanical puzzles; Dopi, the artist, saw patterns and connections where others saw chaos; and Dillo, the strategist, excelled at planning and problem-solving under pressure.

One fateful evening, while wandering the woods, the trio stumbled upon an old, abandoned arcade machine in a crumbling shed. The machine's screen flickered to life when they approached, displaying the words: "Enter the Puzzle Dimensions and Rewrite the Past." Before they could question what it meant, the screen emitted a bright flash, and they found themselves pulled into the machine.

1987

The friends woke up in a pixelated world filled with vibrant colors and strange mechanical structures. A disembodied voice greeted them, introducing itself as Rav, the guardian of the Puzzle Dimensions. Rav explained that this world was created by an ancient civilization to safeguard the fabric of time. However, the dimensions were now in disarray threatening to unravel everything.

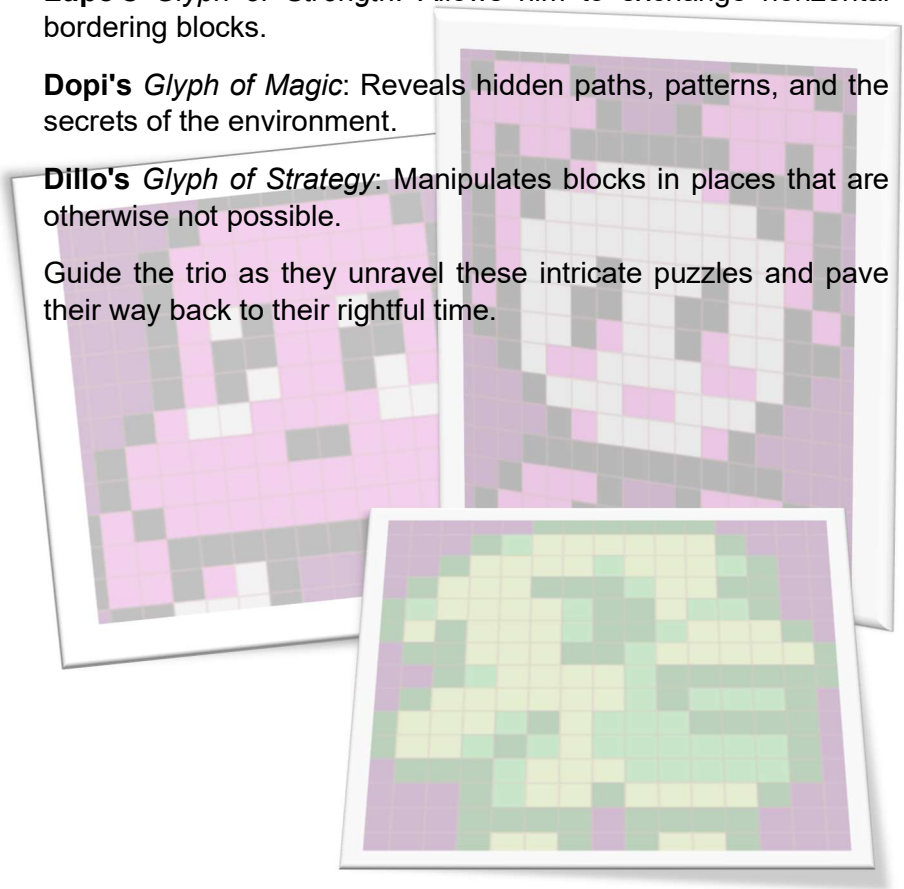
Each of the friends was given a unique "Time Glyph" based on their talents:

Lupo's *Glyph of Strength:* Allows him to exchange horizontal bordering blocks.

Dopi's *Glyph of Magic*: Reveals hidden paths, patterns, and the secrets of the environment.

Dillo's *Glyph of Strategy*: Manipulates blocks in places that are otherwise not possible.

Guide the trio as they unravel these intricate puzzles and pave their way back to their rightful time.



2. Game Play

This game offers game play for 1 player and 2 players. Player 1 is controlled by the keyboard. Player 2 is controlled by a joystick that is plugged in Joystick Port 1. With 2 players, both players play simultaneously.

In case you play this game on an emulator, the mapper type “Konami” is to be selected for the best experience.

The game offers three game play variants.

CAMPAIGN	1 Player
SINGLE PLAYER	1 Player
PLAYER vs PLAYER	2 Players

In all variants your task is to match 3 or more identical blocks, making them disappear. This can be in either the horizontal or vertical direction.

CAMPAIGN

A static level is provided, and the player is tasked to solve the presented puzzle. The level is solved once all blocks are removed. Upon solving the level, the player will progress to the next one until all levels are solved.

In this mode you will play with either Lupo, Dopi or Dillo. Each provides a different type of challenge.

The game is finished, and an end demo is presented when you have solved the last level. In total there are 43 levels.

If your actions have led to an unsolvable situation, you can reset the current level by pressing **F5**. This can be repeated to your liking.

SINGLE PLAYER

A selection screen is shown upon starting of this game play variant. Here you can select to play either with Lupo, Dopi or Dillo, whichever fits best with your personal style.

A dynamic level is presented, in which with a continuous rhythm new blocks are added. Your task is to remove these by matching 3 or more identical blocks. To progress to the next level, a fixed set of matches is to be made where the target value must reach zero.

A match of 3 blocks gives a reduction of 1.

A match of 4 blocks gives a reduction of 2.

A match of 5 blocks gives a reduction of 3.

Etc.

The game is over when a new block can no longer be inserted.

Each time you progress to the next level, a new challenge will be presented. The game mechanics include settings on block insertion speed, more type of blocks and differences in play size area.

PLAYER VS PLAYER

Like the single player mode but offered in player versus player style. A fixed level is presented in which you win by having your opponent lose. This game mode is expanded with an ability bar.

Upon starting a selection screen is shown.

PAL
Both players can individually select to play with either with Lupo, Dopi or Dillo.

SPEED
Selection of new block insertion speed. Options offered range from 0 to 4, where option 0 is the slowest and option 4 is the fastest. This allows you to set the speed to your skill level.

LEVEL
Selection of amount of different type of blocks. Options offered range from 0 to 4, where option 0 corresponds to four types of blocks, and option 4 to the highest number of different type of blocks. This allows you to vary the challenge level.

FILL
Prefills the loaded level with blocks. Options offered range from 0 to 4, where option 0 gives an empty starting area, option 1 corresponds with twenty percent fill. The higher option choices provide more filling of the game area. The amount is the same for both players, but the blocks are different. This allows you to start more challenging and provides quicker access to desired abilities.

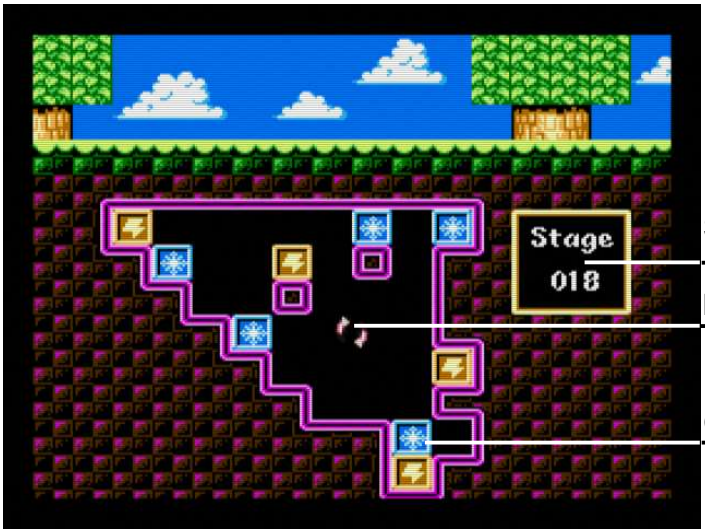
GAME PLAY

Player 1 is allocated to the left side of the screen and player 2 to the right side of the screen.

The offered special abilities are dependent on your PAL selection. An overview of these abilities is given in Chapter 6. You progress through your special abilities by matching blocks. Using an ability will reset your abilities to zero. All abilities can be used indefinitely. Some abilities are applied to your own play area, where others will directly affect the other player.

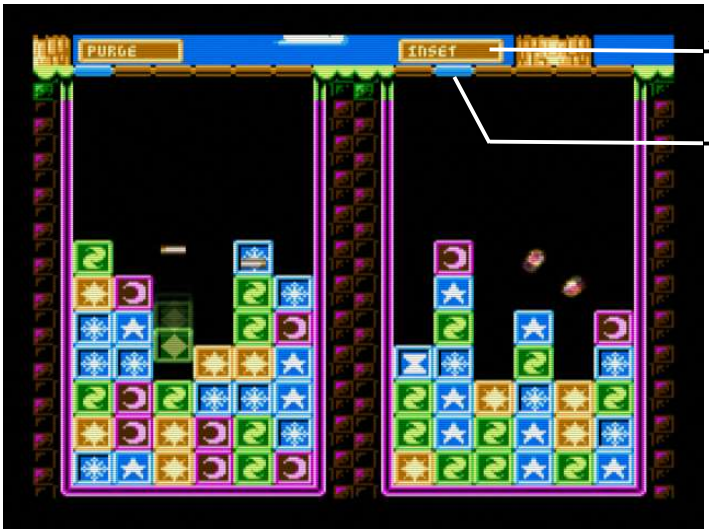
3. The Screen

CAMPAIGN



Stage indicator	Displays the current stage.
Player cursor	Stage dependent.
Game block	Object to be matched and removed.

PLAYER vs PLAYER



Ability name — Displays the name of the ability, changes when progressing through the ability bar or when used.

Ability bar — Graphical representation. Blue indicates an activatable ability, brown indicates an inactive one.

4. Controls

GENERIC

Key	**F1**	Pauses the game
Key	**F5**	Reset current level in Campaign mode
Key	**STOP**	Return to the main screen
Key	**SELECT***	Switches VDP frequency between 50 or 60 Hz.

* Game play speed does not change. Music is affected, offering the original commercial MSX era experience. All songs are intended for listening at 60 Hz.

PLAYER 1

Cursor	**UP**	moves up
Cursor	**DOWN**	moves down
Cursor	**LEFT**	moves left
Cursor	**RIGHT**	moves right
Key	**SPACE**	triggers block swap
Key	**M or N**	triggers special ability in Player vs Player mode

PLAYER 2

Joystick	**UP**	moves upwards
Joystick	**DOWN**	moves downwards
Joystick	**LEFT**	moves left
Joystick	**RIGHT**	moves right
Joystick	**BUTTON A**	triggers block swap
Joystick	**BUTTON B**	triggers special ability in Player vs Player mode

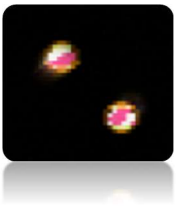
5. Glyphs

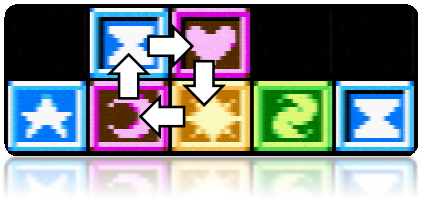
DOPI

Glyph of Strength: Allows him to exchange horizontal bordering blocks.

LUPO

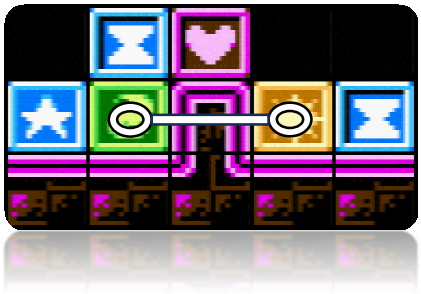
Glyph of Magic: Reveals hidden paths, patterns, and the secrets of the environment.





DILLO

Glyph of Strategy: Manipulates blocks in places that are otherwise not possible.



6. Abilities

In the PLAYER vs PLAYER variant the listed abilities are provided. Some abilities are available to all three PALS whereas others are PAL specific.

ABILITY	NAME	DESCRIPTION
1	Purge	Removes a single block under the player's cursor.
2	Inset	Adds a "?" block to the opponent play area.
3	Swop	Activates the opponent's space key or button A for a specific time.
3	Inverse	Inverses the opponent's controls for a specific time.
3	Conceal	Hides the opponent's player cursor for a specific time.
4	Implant	Inserts a full line of blocks in the opponent's play area.
5	Indelible	Inserts a non-movable and indestructible block in the opponent's play area.
6	Extricate	Removes a horizontal line of blocks under the player's cursor.
6	Unblock	Removes all blocks under the player's cursor.

7. Credits

Main programmer	**Sander Niessen**
Music Designer	**Hans Cnossen**
Graphic Designer	**Balloon Cart**
Supporting Graphics	**Sander Niessen**
Sound Effects	**Hans Cnossen**

The team behind Puzzle Pals wishes to express their gratitude and appreciation to:

Fam. Niessen	Giving all the love and time to pursue this project.
Fam. Cnossen	Who barely recognized me as I got lost in making music and losing track of time – thanks for putting up with me.
Shadow	Connecting Hans and Sander.
Chaos	Coding support, specifically on memory management and music integration.
Huey	Creating the outstanding Trilotracker and revising TMUcompile on short notice.
Pat	Play testing, discovering bugs and giving helpful advice.

MRC forums	Providing a trove of coding knowledge. Special mentions towards, but not limited to, Nyyrikki, Grauw, Santiontanon, Artrag.
openMSX Team	Creating a wonderful MSX emulator.
The Engineer	Ideation on SAT relocation during the Beuningen 2024 fair.
Jorito	Offering help whenever required.
Outer Heaven	Providing graphical feedback and generic input.
TFH	Offering great online service, in both web and WhatsApp groups. Here the idea originated of implementing a Player vs Player mode.
ITCH.IO	Giving Balloon Cart world exposure. This game would not have happened without those graphics.
ChatGPT	Minor graphical creation and story line support.
Jannone	Online PC to MSX graphical convertor
Fibberish	Providing the font as used in the game
Andreas Jönsson	Offering a flexible font to bitmap conversion tool. (www.AngelCode.com)
Sjoerd Mastijn	Creating Sjasm.

Personal notes by Sander

This game was developed with the idea that it could have been released in 1987. As such, many of the design choices were made to reflect that era. These include limitations on music to only PSG and MSX-MUSIC, a typical blocky game-style appearance, and the use of a 1 Megabit (128 KB) ROM.

All code is written in assembly language. Development began around May 2023 and served as a starting point in many aspects. Existing knowledge of Z80 coding was very limited, and there was only a rudimentary understanding of the V9938. This game was an attempt to further develop those skills. For example, almost all sprites are multicolor, achieved by using the OR functionality. Efforts were made to broaden the scope by incorporating magnified sprites, which are rarely seen in games, if at all. The credits section utilizes the YMMM command purely for experimentation. The font in the introduction demo and ACT parts is displayed on screen using the PSET command, allowing for easy color customization. Most routines use referrers, though some (perhaps too many) variables are still hardcoded. The use of BIOS routines is minimal.

The main screen scroll is inspired by *Contra* as released on the NES system, a feature not commonly seen on the MSX system. The small color animation was suggested by Hans. During development, some inspiration was drawn from *Puzznic* and *Tetris Attack*, though not extensively. A direct copy approach was never considered, as the goal was to develop a unique game rather than create a clone or imitation. Code was added for experimental purposes—such as functionality to invert gravity and add bridge tiles—but this utility-based approach was ultimately dropped to remain closer to a traditional matching game design.

The majority of the sprite design was created using Microsoft Excel. By combining Conditional Formatting for cell coloring with a logical operator command to mimic the OR function, sprite development became an efficient process.

The credits were kept brief, as only a small team worked on this game. The credits animation was designed using Microsoft PowerPoint.

The background story was initially drafted by ChatGPT and then refined by us to better suit our vision. There was a concept to include pictures in the game, but this was abandoned due to ROM size limitations.

In the player-vs-player mode, efforts were made to add excitement through the implementation of an ability bar. This feature was clearly inspired by the upgrade systems in countless shmups, but here it was used to encourage interaction between players. Each PAL was given a slightly different ability bar to add variety and offer a more strategic selection process. There are relatively few two-player games for the MSX system, and we hope this game helps to fill that gap.

Everything came together in early November 2024 when Hans joined the project to contribute music and sound effects. His work elevated the game to a whole new level—I'm still amazed by what he managed to achieve with the good old PSG and MSX-MUSIC.

I hope you enjoy playing this game as much as I enjoyed developing it.

Personal Notes by Hans

Composing music, particularly for the MSX, had been on hold for years. I kept hesitating: no time, new software to learn, and worst of all, doubt—could I still do it?

Then, late last year, Roman messaged me. Sander, who was working on this game, needed a composer. Roman asked if it was something I'd be interested in. I jumped at the opportunity. At last, I had a clear goal and the push I needed to start composing again. Thank you, Roman!

Luckily, I haven't been idle when it comes to music. I've been taking weekly piano lessons for years, mainly playing jazz classics by Duke Ellington, Gershwin or Stevie Wonder. Although I'm a mediocre pianist, I've learned a lot from this. What I love about jazz is the richness of its chords and its endless possibilities for variation. That inspiration found its way into a few of the tracks in this game.

The game already looked very promising, both in gameplay and graphics, which motivated me to do my best. Using Moonblaster wasn't an option, which meant I had to learn TriloTracker. At first, its complexity was intimidating, but it turned out to be much easier than I'd expected. With the manual and example tracks, I quickly got the hang of it. TriloTracker even became a favorite of mine, offering more flexibility—like building jazzy solos. And I haven't discovered everything yet.

Over the past few months, I thoroughly enjoyed creating the music for this game. It felt like the good old days—spending every free moment behind the computer and losing track of time. Sadly, there's far less time now than when I was younger. My wife and kids had never seen me in this "zone." My 10-year-old daughter would often say, "Dad, you really need to step away from that computer!"

Much of the music in this game came to life at the piano while trying out some chords or a bassline, and even while humming on my bike with a dictaphone running.

I've always preferred melodic music for game music over ambient sounds. Something you should be able to hum along to. Almost all good music on the MSX is melodic. So that’s what it had to be for this game as well. To me, music is about content and execution. The content—the story—must be authentic and interesting. It needs to stand on its own, even if the execution is simple or resources are limited. The constraints of PSG, MSX-MUSIC, or SCC demand this. The composers at Konami and Microcabin understood that perfectly.

This is the beauty of MSX: its limitations challenge programmers, designers, and composers to be creative. I tried to embrace that.

Collaborating with Sander was a joy; we exchanged ideas that improved the game overall. I hope you enjoy the game and its music. I certainly had a blast creating it! And no, there’s no laser harp this time—sorry!