issue 92 - March/April/May 2000

MSX Computer & Club Webmagazine

Atmosphere

The preface to the second edition of our Webmagazine.

Manuel Bilderbeek

The wolf and the seven notes

Having trouble programming the FM-Pac in Moonblaster? This time Wolf's magic-box shows some hints. The fact that composing is more than just putting notes will be made clear in this article.

Maarten van Strien

Realms of Adventure

Jochen Bilderbeek reviews the latest UMAX game.

Jochen Bilderbeek

Multiplication on a Z80

ML-programmers on a Z80 don't have the luxery of a built in multiplication instruction and will have to write one themselves. This article shows some of the few possibilities. And maybe even the fastest one.

David Heremans

De Maiskoek/Bits and Pieces

Short news — mainly in Dutch, sorry about that! — small advertisements and the column of Parcellus can be found here.

The LP keyboard interface

A review of the LP keyboard interface, used to connect a pc keyboard to your MSX.

Pierre Gielen

Metal Gear save file faking

In this article Bas Wijnen explains you how you can fool around with the Metal Gear savegame files. It's not only useful to learn how to cheat, but the assembly programmer may learn something from this also.

Bas Wijnen

Wammes' kolom [nl]

Een illustratie hoe het vroeger ging op de redactie...

Wammes Witkop

Find It

There are many types of puzzle games released on MSX, but this type actually is really new for on our computers.

Albert Beevendorp

Big computer meeting 2000

A report of the Tilburg Fair 2000, in which Anne also tells you about his dwellings with his Japanese guests.

Anne de Raad

23 steps to high resolution on MSX1

Our exploration trip into the depths of high resolution on the MSX1-videomodes continues as a our guide Antti Silvast reveals a bit of his source code.

Antti Silvast

previous:
MCCW Contents

MCCW issue 92, March/April/May 2000
Back to contents

next:
The wolf and the seven notes

Preface 92

Atmosphere

It was quite nice to hear from Henk van de Kamer — the man that handles the technical details behind the server on the Aktu site — that our new MCCW site was visited quite well on the first few days after the release! The 30th of March we had more visitors than the site of Linkonline, which means it was the most popular site on the webserver of Aktu that day! That day we had 100 visitors. Let's see what the statistics are for a longer period, as soon as I get those from Henk.

I am telling you this because it really encourages us to continue with MCCW. I'm already receiving quite some articles, but still not as much as I would like. So if you are interested in writing some nice interesting articles: please be my guest and contact me! As you noticed, we released this issue quite late, which was mainly caused by a lack of articles...

It was April again, so there was also a MSX fair in Tilburg again, as always. The fair was on April the 15th — as you can read in our previous issue — and it was the first fair where MCCW also had a booth. I was very happy with the positive reactions we received via e-mail, so I expected also a lot of response on the fair. But unfortunately there were only a few people that said it was a very nice thing, our MCCW. Actually, there was hardly any interest in our booth. This can mean two things: either everyone had already seen it, or it was totally unknown, looking like a pc-thingie and therefore uninteresting. It can even be a mixture of those things also. But, it could also have been caused by the fact that our booth was not manned all the time, we were quite busy talking to other people at the various booths...

Ah well, I look at it on a positive way anyway: we had no — absolutely none! — negative reactions at all, except for one person that remarked that the magazine is a bit too short. The reasons for that were mentioned in the preface of the first issue. Anyway, in this new issue we continue with the thread we started in our first issue. So expect second parts of our courses and some other interesting articles. We even have some reviews now, a category which was missing in the previous issue. Next time we hope to have more reviews.

There is one other thing I should say here: from half of June until the end of September, I will be in Japan. This is an internship for the work I'm doing to graduate in Physics here at the University of Nijmegen. This means I will be a little bit harder to reach, but I hope I will be able to read e-mail as normal and use `ssh` — secure shell, a way of logging in with encryption on remote computers — to generate the magazine. If I would get into trouble with this — maybe my Japanese boss won't let me do it... —, Maarten ter Huurne will generate the HTML, he promised me. Actually he is now the co-editor-in-chief. Thanks Maarten! Without you we would never have MCCW as it is now!

The fact that I will be on the other side of the world for three months does not mean you shouldn't hand in new articles! So please help us keeping MCCW up for at least one year and send mail to our — hopefully — well-known address `mccw@aktu.nl`!

Okay, this leaves me wishing you a lot of fun/entertainment/education with this second issue!

A short note from Maarten ter Huurne:

Manuel arrived safely in Japan and established an e-mail connection from there. However, he doesn't have much time for MCCW now and he can't access his university Unix account. So I finished the last bits of this issue and I'll be doing most of the editorial work for issue 93.

Feedback about MCCW is welcome at `mccw@aktu.nl`. If you tell us you like it, it will motivate us. If you tell us why you don't like certain parts, it will help us improve MCCW. So don't be shy, write us what you think!

Manuel Bilderbeek

previous:
MCCW Contents

MSX Computer & Club Webmagazine
issue 92, March/April/May 2000

next:
The wolf and the seven notes

part 2 - the magic box opens

The wolf and the seven notes

I use to ask myself what music I prefer... Dragonslayer 6 or SD-Snatcher. Same gametype, another era but both fun. I can't give the answer, only the fact that the music can't be compared with eachother. Both have great music, but the design differs. That design is the main point of the course this time.

Maarten van Strien

Below is a small scheme with the kind of music game companies did on MSX:

accent on sounddesign	allround	accent on composition
Konami with SCC	Compile with FM-Pac	Falcom with FM-Pac
Microcabin with FM-Pac		Konami with PSG

The left column shows both Microcabin and Konami's SCC. The introduction of the SCC chip was by then ofcourse great news, at last there was a new sound and things which were impossible on the PSG could finally be done: sounddesign. The same happened with Microcabin. While we were annoyed with boring hardware voices and not less boring drums, Microcabin laughed and showed us a total new sound on the same chip. Some parts of Xak 3 show that the main thing in the music wasn't melody, but sounddesign. The same story applies for SD-Snatcher, several parts are based on the atmosphere in the sounddesign and not really on melody. There's nothing wrong with this by the way, and there are parts which do have good themes!

When we take a look at some other Konami titles — Maze of Galious, Usas, Metal Gear etc. — then we notice that Konami offers us some great melodic material. Konami had to do so, the PSG was limited! If there's no way for good sounddesign, then make sure you make a good melody instead, a melody which sticks into your head. That was no problem for Konami, as we all know; especially the three titles above offer us some of the greatest PSG themes on MSX. Also in the right column is Falcom, especially with their masterpiece Dragonslayer 6. The music from DS6 is thematically perfect, yet not really difficult when speaking of production quality, like Microcabin. Falcom, with DS6, avoided the sounddesign concept and emphasised on melody.

What we learn from this is that these companies were conscious of the limits and possibilities of the chip they composed for. If a concept couldn't be realised they wiped it from the drawing board. This sentence is extremely important! Composing is making choices, and sometimes you have to say "NO!".

In practice
This whole story was only an intermezzo for the main goal for today: how to compose and produce for the FM-Pac using Moonblaster. Below are a few standard procedures of mine. But, understand that it is not the effect of the procedures that counts, but the reason why the procedure is here anyhow. I also had to choose back then, I also did the research, *that's* why the procedures are here.

The basic rules:

- quality assurance! If it's not good, *wipe it!*
- if you can do better, do so!
- if you can't do better, stop! It's no use to continue...
- if something is not possible from the beginning, don't try it anyway.
- if something is not good: never make a musicdisk from it!
- not 40 bad tunes, rather four good ones!

A little information on the first point: this is a real important one! The problem is: how do you know — second item — if you can do better? When can you say something is good or not? The answer is rather simple: if you ever hear something on the same system you're working on which sounds better than your product — whether it is on FM-Pac or soundcanvas — then that means there *is* a way to do better. The only way to get to that level is that you *want* yourself to be there too. Dare to say "NO" to something which isn't good enough. Actually, this is the only rule you must remember, the example below is just an example, like there are many examples possible. Every situation is new, *that's* the challenge!

Step one
Ok, the journey begins at the "set start voices" menu where you set all instruments 1 on 1. So channel 1 has instrument 1, channel 2 has instrument 2 etc. The advantage of this is that you never have to go to this part of the program anymore, especially when you're not sure yet about the set-up of your voices, which usualy happens when you start a new song.

Step two
We'll continue with the set-up. As we only have 6 channels — using the 6 melody + 5 drums mode — we have to figure out what we're going to use and what we want in the future. Anybody who does this well is well ahead! Anyhow, the question remains: how to set-up only 6 channels... Take notice of the following points:

- Make sure the melody always has an echo channel. If the melody consists of two lines, pick the upper line for the echo channel. We choose a melody based on one line with one echo channel. The echo channel has to be detuned to the original melody channel, in normal circumstances a detune value of +1. If you use two echo channels, the second detune has to be -1.
There's another little rule here for the echo channels: how many echo steps do we use. When using the

average speed of 18 to 20 stick to two or three steps. Don't use an echo time of only one step, you'll loose the spacyness, the FM-Pac is already pretty dry.

But the choice between two or three steps also deserves a little attention. An echo is something which has to be heard, if an echo coincides with the original direct melody, then the echo effect disappeares for a big part. You have to make sure that the echos fill-up the empty steps of the dry melody. If you already know that your melody consists of eight's notes, i.e. a note, an empty step, a note, an empty step etc., then you'll know that the echo distance should be three steps!

For the die-hards: it's possible to make an echo by writing notes which should be the result of an original dry note which never was given anyhow. And for the true die-hards, it's even possible to make an echo effect *without* loosing channels! The only cost is a software instrument and you do win some channels here. Oh, yeah, the other cost is some year of your life because of the painful puzzle you have to solve as you have to know the melody *before* you start. But this is too heavy for now, let's continue with the article.

- Luckily the bass sounds rather good on the FM-Pac, one channel is good enough. It's very interesting to try some other instruments instead of the acoustic bass everyone uses. Rather try the trumpet, oboe or violin etc. Your music immediately sounds different. This method is very much like that of Microcabin who regulary made this choice also. You also could take the synthesizer as bassinstrument, but as everybody does this already, it's no fun anymore. The vibraphone misses the body in the low frequencies, but if you turn-up the volume enough, it's possible, I'll mention this later on in this article. But all depends on your type of music.

- We have the bass and melody ready — the base — and still three channels left to fill-up. A chord does a lot when it comes to emotion — I'll also mention this further on in the article again — but costs a normal non-MSX composer about three or four channels. And that's a no-go for the FM-Pac! Lets say, one or two channels is enough. With these one or two channels, you have to deal carefully, or actually, with these one or two channels *and* the melody! If you pay a little attention to them, you can let the melodyline be a part of the chord. This doesn't always work, but *if* it works, then you got one extra channel, if your melody consists of two channels, you'll get two extra channels! Imagine: the melody is an [E 4], if you were going to use three chordnotes [C 3][E 3][G 3], then the [E 3] would actually be a double. Wipe the [E 3] and what we get is [C 3][G 3][E 4]. This is a major chord, including the melody. And because the melody had its own echo channel, the overal sound is pretty nice. And we gain that extra channel also!

  Of course you can see what happens: the melody makes you change the two chord notes frequently. Indeed! What I personally used to do is writing a melody which doesn't need much changing in the chordnotes. This method has his advantages and disadvantages: disadvantage is that you have to be able to write simple yet attractive melodies, advantage on the other side is that the melody gets very structured and accessible to the listener, it's all part of efficient FM-Pac composing. How microscopic this all might be, if you use the method well, the sounddesign makes you write a melody which you normaly wouldn't write, because you might not have paid attention to sounddesign before!

  But what it's all about is the quality assurance. If the above method does *not* work for your tune, wipe it! If something else doesn't work either, perhaps you have to change your complete melody. It's quite funny actually when we look at the way a melody is made. You start with the melody and you hope the arrangement will fit. If not, you can change the arrangement as well as the melody. This means the you can end up with a totally different melody compared to the one you begun with! This way of composing where your melody depends on your hardware is not only applicable to MSX composers with FM-Pac. Also the bigger projects — leaders, commercials, company-presentations — have this problem. If someone has no time to hire instrument players — project has to be done yesterday! — and plays the saxophone by him/herself from a synthesizer, that person has to change the melody too when a nasty wavetable-switch shows up. And this isn't just something I made up!

- We still have a channel left. You could use it to accent the bassline. *Not* by doubling the same bass with a detuned one, but, for example, by putting a violin an octave above the bass. With a bit of luck, this violin gets part of the chord, just as the melody did!. Again you use one instrument for two purposes! ..And now the channels are all used!

Step three

Ok, so we didn't compose anything yet, we just made the setup. Before we worry about the balance, we set all instruments at volume 13, don't just let them be on volume 15 by default. At this stage, step one shows up, as you're only dealing with the select voice menu, it is clear how all instruments are balanced: for now volume 13. Now you start to compose. By the time you'll notice that you have to adjust the balance of your voices; use the select voice menu for this. Because you set all the instruments at volume 13, you can really make instruments louder. The best setup has its instruments between volume 11 and 13. Because of these volumes, the drums — which you put at volume 15 ofcourse! — show up better in the mix. And instruments which miss the body of a bass, e.g. the vibraphone, can easily be boosted up by setting it to volume 15.

Emotion

I mentioned the term 'emotions' in chords, maybe it's a nice idea to write some down here. I'll use the — vertical — tracker notation, so you can input them and test hem right away. You might have to raise the octaves a bit... depends on your instrument also!

```
[C 3] [C 4][E 4][G 4]
```
normal happy

```
[C 3] [B 3][E 4][G 4]
```
releaved, relaxed

```
[C 3] [A#3][E 4][G 4]
```
normal, amazed

```
[C 3] [A#3][D 4][G 4]
```
laughing-like, amazed

```
[C 3] [A#3][D 4][F 4]
```
happy amazed

```
[C 3] [C 4][D#4][G 4]
```
normal sad

```
[C 3] [B 3][D#4][G 4] sad, with some extra flavour
[C 3] [A#3][D#4][G 4] angry
```

This are just some examples, very often a chord only has its effect when you put another chord before or after the chord. For example this happy amazed scheme:

```
[C 3] [A#3][D 4][A 4]
[C 3] [A#3][D 4][G 4]
```

Or: happy and releaved amazed:

```
[C 3] [A#3][F 4][A 4]
[C 3] [A#3][D 4][G 4]
[F 2] [A 3][C 4][E 4]
```

Now, what's the use of this? Well, if someone asks you to make the music for an introdemo for a game — with some sort of active storyline — then you can extract the chord from the emotions of the characters playing in the story. Add a melody — don't forget the echo! — and done you are!

Conclusion

You might ask: 'What has this whole story to do with Microcabin and Konami?' Well, technically not very much indeed. It's all about the quality assurance. Konami made melody-type music for the PSG and sounddesign-type music for the SCC. Or simply, basic rule four: 'if something is not possible from the beginning, don't try it anyway'. Sounddesign is just more or less impossible on PSG. Microcabin choose for huge sounddesign in games like Xak 3 and Illusion City. To make the FM-Pac sound huge they used a lot of channels to boost the whole thing up. This meant that there often wasn't room for a melody. If they *would* have inserted a melody, it would have cost quality in the sounddesign. They didn't, and that was right. I said it once: composing is making choices. If you do the right choices always, you don't even need the rules above, you make them yourself!

Any questions...? Feel free to send them to me!

previous:
The wolf and the seven notes

MCCW issue 92, March/April/May 2000
Back to contents

next:
Multiplication on a Z80

the new standard for MSX RPGs?

Realms of Adventure

At the International Computer fair in Tilburg of this year, Umax presented their new RPG: Realms of Adventure. It took a while since The Lost World, but they did it again. And this time it's even better than before!

Jochen Bilderbeek

Directory
What you get
The game
Umax RPG style
Battling evil
Non-linear
Graphics
Music
Conclusion

At the Tilburg fair of 1999, there was already a preview of this game and it looked very cool. Now, a year later, the game is finally finished and everybody can start a new Umax adventure on his MSX. If you enjoyed games like Dragon Slayer 6, Pumpkin Adventure III and The Lost World this new Umax game will not disappoint you.

What you get
The game is delivered in a full colour video-sized box, with some snapshots and this message on the back: 'Prepare to be immersed into a complex world of magic and adventure. Take on the role of Nagash Silverstone and head out into the world, looking for adventure'. That sounds good! The box contains one disk — the Master Disk —, with a nice full colour label on it, two seperate black/white A4 sheets of paper with a map of the land the adventure starts in and a map of the world. Last and certainly not least there is a nice manual with a coloured cover and lots of information on the game. My version was in Dutch, it seems logical to me that there is also a version in English available.

A funny detail: the box has just room for two disks and the manual. There will be more parts — see below — but where do we have to put all these disks then?

The manual is very detailed, it is divided in several parts telling all about the game and its many options. The first part tells the story, which is more detailed than the intro story in the game itself.

After this you can find lots of explanation about the options of the game, about all the menus and a list of weapons and potions which you can find in the game. All this info seems very useful because of the many options and the complexity of the game. In the end there is a small walktrough so you can get started easily.

Who's living there?

The game
Realms of Adventure (RoA) can be played from the disk or it can be installed on harddisk. This can be done easily, and you don't have to have the first disk in the drive at startup for checking the copy protection, as was necessary in PA3. I played RoA from harddisk and after a few seconds of loading and the well-known 'Stichting Sunrise' logo, the title screen appears. After a push on the fire button or spacebar a new screen appears which shows a dragon and the intro story scrolling upwards. After reading the story of Nagash Silverstone going on adventure or a push on the spacebar you can choose between 'New Game' or 'Load Game'. Of course I start with a new game, and after a few words and a view on the village Abernus where the adventure starts, the game can really begin!

Caves with cold streaming water!

Umax RPG style
If you know the previous Umax RPGs I don't have to explain much about what the game looks like, but for those who do not I will explain it anyway.

The game screen is separated in two parts. Most of the screen is occupied by the playing field. On the right side of the screen there is information about your party — the people who you are with and their status — and the amount of money you have. The playing field does not scroll like in e.g. Xak, but it has 'rooms' — like in Metal Gear — which are changed if you walk through the 'edges' of the screen.

As in every other RPG, the player can walk through many lands, villages, caves and forests. When walking around you will face monsters and demons which you can fight to gain money and experience. In RoA you will face monsters randomly: most of them you can't see in the playing field; you just suddenly go into the battle screen where you can fight. In caves, dungeons and castles you can see the monsters. I'll tell you more about fighting later. With a push on the button you will get into the menu. Here are many options accessible. Most, like Spells, Object Scroll and Status are about the same as in other RPGs. However, there are

many extra options in this particular game! Think about mixing herbs e.g., at the sorcerer's. Compared to other games like Pumpkin Adventure III, the status of a character is more detailed. In this game there is also information about the

character's sex, alignment (good or bad) class and mental status. The latter features very nice details; you can be drunk or in love, for example.

One other difference in RoA compared to the other Umax RPGs is that you don't have to walk for hours to get somewhere. Near the most important locations in the game are roadsigns. Just bump into them and you can choose to what other roadsign you want to travel. Very comfortable, just as the fact that you don't have to slay a zillion monsters to get money: in this game you can sell your weapons and other items. New is also that there is a difference between group and party. The party consists of the people walking with you — like in PA3 and The Lost World, or even Dragon Slayer 6 — and the group is the total amount of people working with you. You can only take four extra characters with you on your journeys, but the group can be much larger. Group members not going with you will stay behind in the Inn. You have to decide for yourself if you want to add characters to your group when they ask you to. And of course you also have to assemble the party yourself, making it the most suitable one for your current quest.

Animated graphics in battle mode...

Battling evil
Of course this is an important part of the game, because with fighting you can collect money and experience to get a higher level and become stronger. As said, you enter the battle screen when you run into a monster. On the bottom of this screen are some options like strike (attack), status, scroll, item, spell — use one in the battle — and parry. The latter is also something new: with this one you can exclude a character from the battle, so it can't strike and can't get hit, which can be very handy sometimes. The top part of the screen, the biggest part, is where the monsters you are facing are shown. Below the monster is its number of hit points and with a small arrow you can select the monster you want to hit.

Non-linear
The major difference between RoA and almost all of the other MSX RPGs is that this game is non-linear. The player can make decisions in the game during conversations, when finding objects or facing a monster, which means you are in control of how the story will continue. Every time you play the game the story will be different because you make other decisions. This means the game is very nice to play again, after finishing it, because then you will have more experience in playing the game so you will make better decisions. Of course there is a main storyline in the game, but you can advance in the game in many different ways.

The game consists of many different main quests, only after finishing such a quest you go to the next part of the game. But within these quests there are many different subquests; these can vary from saving a man's daughter or finding an object to saving a village from evil creatures. These subquests don't have to be done, but you can often gain money, reputation (!), objects, weapons, information or something else useful by finishing them. Sometimes someone asks you to do such a quest and sometimes you can do one because you did something special. On the first game disk there is one main quest — part one — and I have discovered about seven subquests.

Hmm, this looks interesting!

Not in the least because of the quests system, it is possible to make a very large game: as far as I heard there will be at least 15 more parts. It's also possible to let Umax know your ideas for quests which they can fit into the game. If you have some, mail Peter Meulendijks: PeterMilldyke@hetnet.nl.

If this lightning spell doesn't kill him...

Graphics
The graphics are just as in the previous Umax RPGs mostly from Peter Meulendijks's hand and are absolutely great. The lands and villages you can walk through look beautiful and the houses, bars and other buildings are very detailed. A difference with e.g. The Lost World are the monsters. In RoA part 1, these are not very big and don't look so nice. However, the monsters you see in the battle screen are animated! Wolves walk from left to right, knights swing their swords and blood splatters when you hit a monster. This is very unique and it is a very cool detail.

Music
When the first music starts you think: 'Hey, I already know this one!'. Most of the themes are from previous Umax RPGs — PA3, The Lost World — because they couldn't find a composer who wanted to make new musics. Officially only the FM-Pac is supported, but since the original compositions are for both FM-Pac and Music Module, the Music Module is used as well. However, since there is no samplekit loaded, you hear a lot of buzzing from the Music Module's sample channel, so it would be best to disconnect the Music Module before playing. A silly thing which could easily have been avoided. There is no Moonsound music yet, but Peter announced that there

might be updates for Moonsound music on the Bussum MSX fair in September. Most of the musics are nice to listen to, but if you have played other Umax RPGs you will recognize most of them and then it can be pretty boring.

Conclusion

A typical Umax game: a great story, nice graphics. Very original is the great amount of details in the status of characters which makes them more real. The quests makes the game nice and fun to play and very varying.

Woods, woods, woods... everywhere woods!

Too bad the monsters are not very big and detailed, I always liked that, but the animations are very cool and original. By the way, Peter wrote that future parts will contain larger monsters. A shame that no one wanted to make new musics for this game, especially for all who have played other Umax RPGs. When writing this the second disk is still not ready. I hope that it will not take too much time between each new part, because it's very irritating when you have finished a part and you can't continue because you don't have the disk.

RoA has lots of good things from RPGs like Pumpkin Adventure, Dragon Slayer 6 and Xak combined and that makes it a perfect game for everyone who likes RPGs like I mentioned before. Absolutely a game you should buy if you are a lover of RPGs.

A last remark: Peter wrote on the MSX mailinglist that the first disk was actually only the master disk and they added some gameplay to get people interested. He said the actual game will start at part 2, continuing where part 1 ended. In part 2 they want to cover the whole first world, which will be about 5 or 6 disks in size. There will also be a new master disk then, because they made some improvements that cannot be patched.

previous:
Realms of Adventure

MCCW issue 92, March/April/May 2000
Back to contents

next:
De Maiskoek/Bits and Pieces

when speed matters

Multiplication on a Z80

Multiplication is one of those 'difficult' operations for a proccessor, one that eats up a lot of transistors and therefore uses a lot of CPU power and takes up much space on the silicium. So it doesn't surprise us that in the early days technicians didn't directly integrate multiplication instructions into the processor. So it wasn't build in in the — at that time — advanced Z80 8-bit CPU. Multiplication was left to be implemented by the programmers which needed it.

David Heremans

Directory

Most programmers won't care about the fact that multiplication wasn't built into the processor. If you use a higher level programming language it will most certainly provide the necessary commands or code libraries which will solve multiplication arithmetics for us. The only people who will worry about the missing CPU instructions are the people who directly are affected: the assembly language programmers. In our MSX case the only ones who really worry about multiplication are demo-programmers; they need a really fast solution for performance reasons. More normal applications can still use a relatively slow multiplication, since once assembled, ML is still the most performant and for the occasional multiplication one won't see the difference between 40 or 140 microseconds. If you need however thousands of multiplications like in realtime 3D algorithms or a realtime spectrum analyzer this 1000 times 100 microseconds will be a difference of 100 milliseconds between possible screen updates. With the fast code you get at maximum 1/40 ms = 25 Hz, so 25 frames per second. With the slow code you will obtain 7 frames per second. This means that your framerate can vary 357 percent depending on the code!

The solutions

Let's have a look at some of the possible solutions for multiplication on a Z80. Remember, just like in real life there is no best solution, it all depends upon the priorities needed for the programming project at hand.

Solution 1

Let's start off easy. Most novice assembly writers will solve multiplication with something that resembles the following code:

```
ML-listing: SOLUT1.ASC

; INPUT: THE VALUES IN REGISTER B EN C
; OUTPUT: HL = B * C
; CHANGES: AF,DE,HL,B
;
              LD HL,0
              LD A,B
              OR A
              RET Z
              LD D,0
              LD E,C
LOOP:         ADD HL,DE
              DJNZ LOOP
              RET
```

This is almost literally the definition of multiplication like most of us have learned in elementary school. 5 times 3 equals 3+3+3+3+3, 8 times 2 equals 2+2+2+2+2+2+2+2 etc. It is clear that this approach can be optimized quite a bit, like making sure that register B contains the smallest value. Extra functionality can be easily implemented like adapting the routine so that it can multiply if there are one or two bytes in two's complement format and accordingly outputting a two's complements value in HL.

Advantages

It's simple, it's short. No more memory used than needed to store the code. And it is very easy to understand for outsiders. If you write code it is always nice if somebody else can read and maintain it later.

Disadvantages

Time needed by this routine can vary a lot. When B is zero it will be very fast. 11+4+4+7 is 26 T-states, on a regular 3.5 MHz Z80 this corresponds to ± 7.5 microseconds. In the worst case however this routine will need 11+4+4+7+7+4+254*(11+12)+11+7+7=5904 T-states, meaning milliseconds! Well allright, not really multiple milliseconds, one multiplication can take about 1.45 millisecond considering this worst case scenario. And don't forget on an MSX there is for every instruction an extra T-state wasted to make it even worse. This means that our Z80's 1.455 milliseconds will become 1.834 milliseconds when run on our beloved MSX.

Solution 2

Like in most assembly written applications you will need to make a choice between space and speed. The previous solution is probably the best choice if you were going for a very short solution, that needs the least amount of memory. If you however have some mapperpages free you can precalc all the information in advance in one *huge* matrix and then do lookups. Let's say you need multiplications where number one can be from zero up until 255 and number two is in the range of zero up until 63. Now we can use 32 kB to precalc all this. Let us say we are going to use the memory from #4000 to #BFFF to store this data. We will calculate the data with the following pseudo-code:

```
Pseudo-listing: PRECALC

for a=0 to 63
for b=0 to 255
solution=a*b
poke #4000+a*256+b,low-byte(solution)
poke #8000+a*256+b,high-byte(solution)
next b,a
```

If we have the data stored in this format we can use the following code to multiply:

```
; IN: L NUMBER ONE (0-255)
;     A NUMBER TWO (0-63)
; OUT: BC IS RESULT
; CHANGES: AF,BC,HL
;
        OR A,#40
        LD H,A
        LD C,(HL)
        ADD A,#40
        LD H,A
        LD B,(HL)
        RET
```

Advantages

Fast, this solution will always, no matter what the input is, use 7+4+7+7+4+7+7=43 T-States. If you want to use two's complement numbers the code doesn't change, just make the needed matrix. Or you could alter your routine to take into account the signs of the numbers. Just introduce a litle overhead to transform both numbers to positive numbers and afterwards change HL to its two's complement when the final result had to be negative. This approach would let you multiply numbers from -63 up until 63 with any other number from -255 up until 255 using the same matrix.

Disadvantages

You need a lot of memory! At least you will have to store the result matrix. If you are going to store a 128×256 matrix your entire view range of 64 kB will be filled and you will have to introduce an extra overhead to use the memmory mapper. Making a matrix for 256×256 posibilities is going to use up 128 kB of memory. So for a regular European MSX2 this is not even possible, because there would be no mapperpage left for the program, stack and system variables!

The perfect compromise

Let's introduce a method of which the speed is constant and which doesn't fill the entire memory range. We still need a lookup table of 512 bytes to do our lookups. How are we going to go about it? Let me show you the code.

First, the lookup table. This one must start at #xx00. Most demo programmers will already have the habit of aligning lookup tables this way. It speeds up the indexing since there is no offset to take into account and all lookups can be calculated using boolean operations on addresses. We did this in the previous example also, it was not by coincidence that the lookup table started at #4000.

```
MULTAB    DB      0, 0, 1, 2, 4, 6, 9, 12, 16, 20
          DB      25, 30, 36, 42, 49, 56, 64, 72, 81, 90
          DB      100, 110, 121, 132, 144, 156, 169, 182, 196, 210
          DB      225, 240, 0, 16, 33, 50, 68, 86, 105, 124
          DB      144, 164, 185, 206, 228, 250, 17, 40, 64, 88
          DB      113, 138, 164, 190, 217, 244, 16, 44, 73, 102
          DB      132, 162, 193, 224, 0, 32, 65, 98, 132, 166
          DB      201, 236, 16, 52, 89, 126, 164, 202, 241, 24
          DB      64, 104, 145, 186, 228, 14, 57, 100, 144, 188
          DB      233, 22, 68, 114, 161, 208, 0, 48, 97, 146
          DB      196, 246, 41, 92, 144, 196, 249, 46, 100, 154
          DB      209, 8, 64, 120, 177, 234, 36, 94, 153, 212
          DB      16, 76, 137, 198, 4, 66, 129, 192

          DB      0, 192, 129, 66, 4, 198, 137, 76, 16, 212
          DB      153, 94, 36, 234, 177, 120, 64, 8, 209, 154
          DB      100, 46, 249, 196, 144, 92, 41, 246, 196, 146
          DB      97, 48, 0, 208, 161, 114, 68, 22, 233, 188
          DB      144, 100, 57, 14, 228, 186, 145, 104, 64, 24
          DB      241, 202, 164, 126, 89, 52, 16, 236, 201, 166
          DB      132, 98, 65, 32, 0, 224, 193, 162, 132, 102
          DB      73, 44, 16, 244, 217, 190, 164, 138, 113, 88
          DB      64, 40, 17, 250, 228, 206, 185, 164, 144, 124
          DB      105, 86, 68, 50, 33, 16, 0, 240, 225, 210
          DB      196, 182, 169, 156, 144, 132, 121, 110, 100, 90
          DB      81, 72, 64, 56, 49, 42, 36, 30, 25, 20
          DB      16, 12, 9, 6, 4, 2, 1, 0

          DB      0, 0, 0, 0, 0, 0, 0, 0, 0, 0
          DB      0, 0, 0, 0, 0, 0, 0, 0, 0, 0
          DB      0, 0, 0, 0, 0, 0, 0, 0, 0, 0
          DB      0, 0, 1, 1, 1, 1, 1, 1, 1, 1
          DB      1, 1, 1, 1, 1, 1, 2, 2, 2, 2
          DB      2, 2, 2, 2, 2, 2, 3, 3, 3, 3
          DB      3, 3, 3, 3, 4, 4, 4, 4, 4, 4
          DB      4, 4, 5, 5, 5, 5, 5, 5, 5, 6
          DB      6, 6, 6, 6, 6, 7, 7, 7, 7, 7
          DB      7, 8, 8, 8, 8, 8, 9, 9, 9, 9
          DB      9, 9, 10, 10, 10, 10, 10, 11, 11, 11
          DB      11, 12, 12, 12, 12, 12, 13, 13, 13, 13
          DB      14, 14, 14, 14, 15, 15, 15, 15

          DB      16, 15, 15, 15, 15, 14, 14, 14, 14, 13
          DB      13, 13, 13, 12, 12, 12, 12, 12, 11, 11
          DB      11, 11, 10, 10, 10, 10, 10, 9, 9, 9
          DB      9, 9, 9, 8, 8, 8, 8, 8, 7, 7
          DB      7, 7, 7, 7, 6, 6, 6, 6, 6, 6
          DB      5, 5, 5, 5, 5, 5, 5, 4, 4, 4
          DB      4, 4, 4, 4, 4, 3, 3, 3, 3, 3
          DB      3, 3, 3, 2, 2, 2, 2, 2, 2, 2
```

```
          DB          2, 2, 2, 1, 1, 1, 1, 1, 1, 1
          DB          1, 1, 1, 1, 1, 1, 1, 0, 0, 0
          DB          0, 0, 0, 0, 0, 0, 0, 0, 0, 0
          DB          0, 0, 0, 0, 0, 0, 0, 0, 0, 0
          DB          0, 0, 0, 0, 0, 0, 0, 0
```

The precalced table may seem long and boring — which it is actually — but the code itself is much shorter and cleaner:

ML-listing: `MULTROUT.ASC`

```
; IN : A and D are to be multiplied
; OUT: HL is result
; CHANGES : AF,BC,E,HL
;
          LD          E,A
          SUB         D
          LD          H,MULTAB/256
          LD          L,A
          LD          C,(HL)
          INC         H
          LD          B,(HL)
          LD          A,E
          ADD         A,D
          LD          L,A
          LD          E,(HL)
          DEC         H
          LD          L,(HL)
          LD          H,E
          OR          A
          SBC         HL,BC
```

Some people would by now feel the urge to take paper and pencil and they would try to figure out how this works. If you are one of those people I'll give you a hint and afterwards you will find it much easier to figure out how it works. Ok, here is the hint. MULTAB is made using the folowing pseudo-code:

Pseudo-listing: `PRECALC`

```
for i=0 to 255
if i<128 then a=i else a=i-256
solution=int((a*a)/4)
poke i,low-byte(solution)
poke 256+i,high-byte(solution)
next
```

Those who feel they can tackle the program alone can start working now. For those who didn't find a piece of paper or whose pencils are blunt I will explain the code. Nobody would read this far and then appear too lazy to figure it out himself, right? Now hold on, here we go.

Everybody has once during his scolar career heard about the 'special products'. And we learned that $(a+b)(a-b)=a^2-b^2$ and that $(a+b)^2=a^2+2ab+b^2$. First of all, we make a lookup table for the function $f(x)=(x^2)/4$.

The trick is that now we can say that $a*b=f(a+b)-f(a-b)$. Let's prove that this is true. We are going to work out this function:

$a*b = f(a+b) - f(a-b)$	this is our original function
$a*b = ((a+b)^2)/4 - ((a-b)^2)/4$	we replaced the funtion by it's definition
$a*b = ((a+b)^2 - (a-b)^2)/4$	here we say $a/b -c/b = (a-c)/b$
$a*b = ((a^2+2ab+b^2) - (a^2-2ab+b^2))/4$	we worked out the special products
$a*b = (a^2-2ab+b^2 - a^2+2ab-b^2)/4$	just dropped the brackets that weren't needed. Pay attention to alter the signs if needed, $-(-X)$ becomes $+X$
$a*b = (a^2-a^2 +b^2-b^2 +2ab+2ab)/4$	rearranged the components between the brackets
$a*b = (4ab)/4$	Since $x-x$ can be dropped this simplifies a lot
$a*b = ab$	Voilà, it seems that our initial (complex) function can be reduced to the result of a simple multiplication

Let's use a simple example to make it clearer:

$5*8 = f(5+8)-f(5-8)$

$5*8 = f(13)-f(-3)$

$5*8 = 13*13/4 - (-3)(-3)/4$

$5*8 = 169/4 - 9/4$

$5*8 = 42.25 - 2.25$

$5*8 = 40$

Our pre-built list contains the values for $x^2/4$ for all x's in the range of -128 up until +127. So the constraints for the registers A and D in our program are -128 <= A+D <=127 and -128 <= A-D <=127. If we are sure that A

and D are always between -64 and +63 these conditions will always be true.

A little remark however for those who saw the light. Our lookup table doesn't contain $x^2/4$ but $int(x^2/4)$! So for the case of 3 we don't get 2.25 but 2 as result of the lookup. Rest assured however that this doesn't affect the final result.

I hope that this article has made it clear that, even on a slow CPU, fast solutions exist to what seem to be long and tedious problems. Just as in real life, there are multiple solutions to one and the same problem, every solution having its good and bad sides. For those who were looking for a fast multiplying routine, feel free to use the given solution. I'm waiting to see the results of *your* imagination.

previous:
Multiplication on a Z80

MCCW issue 92, March/April/May 2000
Back to contents

next:
The LP keyboard interface

Note: This article contains text in more than one language.

Page for computer news in general and MSX news in particular

De Maiskoek/Bits and Pieces

Gratis internet

Wanneer mag een persoon of organisatie en als zodanig rechtspersoon eigenlijk de term gratis gebruiken? U komt in de diverse media de laatste tijd regelmatig de term gratis tegen in combinatie met internet. Vaak ook nog eens samen met het woordje 'met'. Alsof je ook zou adverteren met 'nu zonder gratis internet'. Taalkundig heb ik met dat laatste geen probleem, maar of het veel klanten trekt... Laten we eens kijken wat u moet hebben om op het internet bezig te kunnen zijn. Er blijken diverse mogelijkheden te zijn.

De gebruikelijke is dat u uw eigen computer uitrust met een modem, zorgt voor een telefoonlijn en zich aanmeldt bij een provider. U hebt daarvoor de volgende kosten:

EVA player for IDE

The Spanish MSX user Sergio Guerrero is working on a program that will enable your MSX Turob-R or MSX2+ to play EVA files via your IDE interface. If you have only (!) an MSX2+, you can use the Music Module for the sound output. Until now it was only possible to play EVA files on a MSX turbo R with MEGA-SCSI interface. This was the work of the Japanese group ESE Artists Factory, presented in Tilburg in 1995. The EVA fileformat is like MPEG and FLI a digital compressed movie format.

New owners mailinglist

Since a few months, the ownership of the MSX mailinglist has been taken over by Maarten ter Huurne and Wynke Stulemeijer. The former listowner, Wiebe Weikamp, didn't have the time any more to maintain the list. For more

New MSX from ASCII?

The Japanese MSX-group Frontline has obtained ASCII's permission for the development of a new MSX. Currently, Frontline is trying to develop a new VDP without getting in trouble with the owner of the copyright, Yamaha. The Japanese are still looking for contacts all over the world to share their knowledge. Panasonic is most likely to produce the new system.
(source: Ikeda's MSX Print)

Gameboy Emulator for MSX

Patriek Lesparre is working on perfecting a Gameboy emulator for MSX. A lot of games already work, but most of them very slow. However, we think this is a very impressive accomplishment! The emulator runs on MSX2 and up with MSX-DOS 2 and is written fully in assembly. Even sound is emulated through PSG.

information about the MSX mailinglist you can mail to `msx-owner@stack.nl`.

Pentaro Odyssey 2

The sequel to Pentaro Odyssey was released in Tilburg. However, we heard that it contains quite some bugs. It is even mentioned in the manual that there will be enormous bugs in some levels. We hope that Sunrise or the author(s) will improve the game. In any case, it will probably be reviewed in the next issue. We will keep you updated.

Bussum 2000

Also this year there will be an MSX fair in Bussum, replacing the former fair in Zandvoort. There will also be a second edition of the MSX Marathon. The organizers promise improvements: helpers to get your stuff up the stairs, better organization of the Marathon, easier payment. All info can be found on the updated homepage: `http://msxfair.cjb.net`.

www.msx.org renewed

The `http://www.msx.org` homepage has been renewed. The authors are planning to make the biggist general MSX site on the net.

SME3 international PD

At the moment Xelasoft is working on an international version of Super Music Editor 3.0. This version of the program will become public domain and can be downloaded from Xelasoft's homepage called The MSX Plaza.

Superchip

Het is allemaal nog in ontwikkeling, maar vanuit Cambridge kregen wij lucht van een nieuwe efficiëtere chip. Op de universiteit aldaar — geen industrieontwikkeling en dus extra geduld voor een productiemodel — ontwikkelde men een microprocessor die data opslaat als magnetische velden. Verteld wordt dat dit 40000 maal zo efficiënt is. De chip is ook veel kleiner dan nu gebruikelijk: zitten er nu 6,6 miljoen transistors op een vierkante centimeter, zitten er bij de nieuwe chip 5,5 miljard gezellig bij elkaar op dezelfde ruimte. Een server met vier of acht Itaniums in uw mobieltje behoort daarmee tot de reële toekomstmogelijkheden; fysiek tenminste want financieel... De techniek is nog niet volledig uitgebuit, want de capaciteit zal zover nu kan worden gezien nog toenemen tot 250 miljard transistors per vierkante centimeter. WAPpen voor vlooien en luizen komt eraan.

Hartbewaking via internet

The author is planning to make a GFX9000 version too, which should be faster. If the Z380-projects become more succesful he even wants to write a Z380-version. You can check out GEM on its page on the site of TNI: `http://gem.tni.nl`.

FAT16 for IDE

Jon de Schrijder, who is developing the software for the Sunrise IDE interface, recently reported that he has been working on FAT16 support for the IDE interface, together with the Japanese MSX user Okei. He says he finished it the last week of May and is will work on an `IDEFDISK` program in July to make FAT16 partitions on the IDE harddisk. With FAT16 it is possible to use more than 32 MB of diskspace per partition, as is the case on the normal MSX format FAT12. More news about this later!

Moonsound trouble: advice

The output amplifiers of the MoonSound are not short-circuit proof. Therefore, make sure that if you plug the jack (earphone connector) into or pull it out of the MoonSound, the computer and a possible slot expander are switched off. Otherwise there is a risk to blow up the output amplifiers.
In case it happens afterall, you can contact Sunrise for repairs (see their homepage for address info). The repair itself isn't very expensive, only about one euro, but the postage is (for the Netherlands: about 10 Euros), so the repair still costs a lot (11 Euros for people living in The Netherlands!). So here also goes: to prevent is better than to cure.
(source: Sunrise Foundation)

Improved VDP engine for download

On The MSX Plaza is now an enhanced VDP command driver available for download to improve the performance of fMSX 2.x. This code may also be used by other emulator authors as a reference for their own VDP engine.

Liegen op afstand

Wie kent niet het telefoontje naar het café waar mevrouw X vroeg of mijnheer X misschien daar was en de barman richting mijnheer X wat non-verbale signalen gaf, die werden beantwoord met heftig nee-schudden. De barman zei daarop dat hij even zou kijken en deed na enige tijd zijn ogen stijf toe en meldde in de telefoon dat hij hem niet zag. Mannesmann maakt het de smoezenverteller wat gemakkelijker door op www.mannesmann.de de telephone background noise animator beschikbaar te stellen, zodat onze bedrieger zijn smoes met passende achtergrondgeluiden kan vertellen. 'Sorry schat,

I/O

Bij een experiment in de Verenigde Staten van Amerika — waar anders? — is bij een honderdtal hartpatiënten een monitor bij het hart ingebracht. De monitor registreert elke onregelmatigheid en zendt die via internet op naar de beveiligde site van de arts. Deze kan de gegevens die hij zo van de Chronicle, zoals het apparaatje heet, binnenkrijgt eenvoudig bestuderen en op grond daarvan de patiënt uitnodigen voor nader onderzoek of de medicatie aanpassen. Niet alleen is dit gunstig voor patiënten die weten dat de dokter gelijk op de hoogte wordt gebracht indien nodig en als toch naar de dokter moet worden gegaan is die al bij aanvang van het onderzoek op de hoogte van de recentste gegevens en kan sneller en gerichter zijn werk doen. Verder kan hij in noodsituaties beoordelen welke patiënt het snelst aandacht verdient en normale routineonderzoekjes, die meestal kort duren, maar veel tijd vergen voor de patiënt verminderen. Wij vragen ons wel af of dit niet een eerste stap is naar een monitor in elk mens die op alles en nog wat let. Gelijk een berichtje naar de politie als u plotseling doodgaat. Die politie hoeft dan alleen maar te kijken naar de mensen in de buurt met een verhoogde hartslag en zij kennen uw moordenaar misschien al voor u rochelend ter aarde stort. En doordenkend... nog voor uw familie goed en wel is ingelicht is de begrafenisondernemer al bezig met zijn voorbereidingen.

IEEE stoot door naar 10 Gb

Het is nog maar een half jaar geleden dat alles met de 1 gigabit Ethernet verbindingen rondkwam en nu al is IEEE, de standaardenorganisatie op dit gebied, bezig met de norm naar 10 gigabit te krijgen. De OSI 192 standaard levert al snelheden op van 9,6 miljard bits per seconde op, maar de echte 10 Gb/s zal vermoedelijk op de langere termijn iets goedkoper blijken. De hogere aanvangskosten komen doordat deze snelheden niet meer op koper kunnen, maar van bijvoorbeeld glasvezel gebruik moeten maken. Wilt u op de hoogte blijven kijk dan eens op hun site.

Moskou heeft Windows 2000 eerst

Je kon er op wachten, maar zo snel? Windows 2000 is het eerst op de markt verschenen in Moskou en voor zeer schappelijke prijs van slechts drie dollar voor de cd met Windows 2000 Professional. En dat enkele dagen voor de officiële introductiedatum. Het waren illegale kopieën van de gold-versie, de versie die na de beta's geleverd wordt aan OEM-partners en softwareontwikkelaars. Volgens Microsofts marketing manager is 90% van de software in de voormalige Oostbloklanden illegaal.

Resultaten Microsoft goed

we hebben een brandje op kantoor.' of 'Het wordt wat later want ik zit vast in een storm.' U begrijpt al uit de mogelijkheden dat het echt voor noodgevallen gebruikt kan worden, want twee stormen per week die geeneens op het journaal/weerbericht worden genoemd zijn zelfs voor het stereotiepe blondje te veel.

Philips start nieuwe chipfabriek

Maakte Philips enige tijd geleden de indruk de chipproductie te willen verlaten, komt daar nu toch verandering in. Samen met het Franse STMicroelectronics wordt in Crolles een fabriek opgezet, die wafers van 30 cm (12 inch) de nieuwe maat voor chipproductie, zal produceren. Er is een bedrag van 1600 miljoen voor uitgetrokken. Het samenwerkingsverband hoopt hiermee te concurreren met TSMC uit Taiwan, die een relatieve voorsprong heeft omdat die alleen de 8 inch faciliteit hoeft om te bouwen naar 12 inch en zo al dit jaar in staat is te produceren, terwijl Philips pas over twee jaar zo ver is, maar dan wel in een splinternieuwe fabriek. Naar verwachting zal de grote vraag naar 12 inch wafers echter pas rond 2003 op gang komen dus meent Philips precies goed te zitten.

Seagate komt met snelle harddisk

Dat in onze it-wereld alles steeds groter, kleiner en sneller gaat was bekend, maar soms zijn de stappen weer net een slagje groter dan verwacht. Seagate kwam op CeBIT op de proppen met een nieuwe harddisk die draait met 15000 toeren per minuut. Na 5400 — nu de standaard —, 7200 en 10000 hadden wij een volgende stap op 12000 of 12500 verwacht. Het eerste model in deze reeks de Cheetah X15 heeft een opslagcapaciteit van 18 gigabyte en heeft door de hoge rotatiesnelheid een gemiddelde zoektijd van 3,9 milliseconde. Naar keuze kan hij worden geleverd met Ultra 160 SCSI of met 2 GB fibre channel interface. De schijf kan gelukkig gecombineerd worden met de oude 10000 rpm schijven van Seagate en werkt met dezelfde koeling en vraagt ook evenveel energie als de oude modellen. Seagate sluit grotere capaciteiten voor in de toekomst niet uit, maar meent dat gebruikers nu meer behoefte aan snelheid dan opslag hebben. De hoeveelheid geluid die de drive produceert — men koos nogal eens voor 7200 rpm in plaats van 10000 vanwege het lawaai — valt erg mee. Wij hoorden wel wat geruis, maar storend was het niet. Natuurlijk had Seagate niet het slechtste exemplaar op CeBIT neergezet, maar aan de andere kant stond hij 'vrij' en was niet ingebouwd en dat scheelt natuurlijk ook veel.

ADSL voor UT

De TU Twente krijgt in het kader van het Gigaportproject aansluiting op ADLS. Met deze

Wanted: Original Metal Gear, of course incl. box and manual. Mail: j.a.schaap@let.rug.nl.

Who has some cheap MSX stuff for sale/trade? Please mail me! manuel@msxnet.org. You can also see what I already have on my homepage! At the moment, I have a lot of Konami cartridges for trade only... Check it out.

Looking for an MSX2+. No need for it to be original, can be an MSX2 converted to MSX2+. Mail me your price, and what kind of 2+ you might want to sell: Bastiaan@nighttown.com

Wanted: MCCM 66-90 for a nice price. Mail: sandy@generation-msx.nl.

For sale: Philips VS-0080 RGB monitor with SCART and CVBS connectors. It features all knobs that should be on a decent RGB monitor: contrast, brightness, centering and stretching knobs, etc. Condition: fine. I used it for a y ear. Ok, sometimes it takes a little effort to turn the thing on/off... The power button seems to be a little dirty inside. But just push hard and it works fine! Also, it seems that the image flashes once in every 5 minutes or so... But you 'd miss it in an eyeblink. Price: 80 NLG, excluding any shipment. Interested? Mail manuel@msxnet.org.

Who wants to write an article about an interesting MSX topic for this webmagazine? Please contact us! mccw@aktu.nl

We want to continue with the Art Gallery! So please send us your MSX drawings! Thanks!

Lief Maiskoekje zoekt leuke I/O'tjes om samen op het net te staan. Mail even naar mccw@aktu.nl.
If *you* have some little advertisement for this page, you can mail us at mccw@aktu.nl.

Ondanks alle dreiging van een splitsing in twee of drie onderdelen blijft Microsoft goed draaien. Het eerste kwartaal werd afgesloten met 5,6 miljard gulden winst bij een omzet van 13,3 miljard. Vergeleken met het vorige jaar zijn beide cijfers 23% hoger dan het vergelijkbare kwartaal vorig jaar. Toch was er een achterblijven van de omzet naar OEM-klanten. Mogelijk werden door de overgang van 1999 naar 2000 minder computersystemen verkocht en daarmee minder Windows' aan de fabrikanten daarvan. De markt is, nu de wissel eenmaal genomen is, alweer aan het aantrekken en komende resultaten zullen wellicht nog rooskleuriger zijn.

Sony komt met CiXD

In de Vaio-serie brengt Sony een nieuwe mininotebook uit met ingebouwde camera. Het notebook weegt ondanks die camera nog geen kilogram! Wel bleef het bij de prijs inbegrepen diskettestation buiten het notebook. Het beeldscherm is 8,9 inch groot en dat is natuurlijk klein als je het met een normale notebook — laat staan een desktop — vergelijkt, maar fors als je de concurrentie in deze markthoek bekijkt. De verdere uitrusting zal menigeen die al een poosje niet upgradede doen watertanden: 64 MB intern geheugen en een 400 MHz Mobile Pentium II. Verder een 12 GB harddisk en zogenaamde jog-dial knoppen om veelgebruikte functies alsmede de camera te bedienen. Het machientje is vooral bedoeld voor gebruikers die onderweg opnamen willen maken en die van commentaar voorzien via internet op te sturen. De CiXD is biometrisch beveiligd en zo alleen door zijn eigen 'baas' te bedienen.

Systeembus naar 400 MHz

Intel heeft op CeBIT — en enkele dagen daarvoor in Amerika — de nieuwe Willamette getoond. Deze super Pentium draait tussen 1300 en 1500 MHz en is nog steeds een 32-bitsprocessor. Daarmee zal hij in performance toch achterblijven op de komende 800 MHz Itanium. Itanium is de productnaam die gekozen is voor de processor die ontwikkeld werd als Merced en die is volledig 64-bits. Deze ontwikkeling hadden kenners echter al zien aankomen. De Merced-lijn werd voor professioneel zware toepassingen ontwikkeld en de Willamette als volgende stap voor de thuisgebruiker. De kloksnelheid is echter zodanig hoog dat de processor op punten de 64-bitslijn verslaan kan. Een ander nieuwtje is de verhoogde bussnelheid; die ligt bij de getoonde Willamette op 400 MHz. Technisch is er een oplossing gekozen die vergelijkbaar is aan het 2× en 4× bij AGP, de werkelijke bussnelheid ligt nog steeds op 100 MHz, maar voor de gebruiker lijkt het 400 MHz.

Word2000 in Frankrijk uit de handel

Na het uitkomen van Word2000 is in Frankrijk een stortvloed van kritiek over Microsoft uitgestort. De kritiek bedroeg de spellingchecker die rascistische trekjes vertoonde. Men viel ondermeer over 'arabe' dat in tegenstelling tot het vermoeden van de meeste Nederlanders geen inwoner van Saoedi-Arabië mee wordt bedoeld maar een immigrant uit Algerije. Dezen voelen de term als discriminerend, in ieder geval beledigend. Zou het alleen om het normale,

---

Asymetric Digital Subscriber Line zijn snelheden van minstens dertig maal ISDN te halen. Niet alleen de universiteitsgebouwen zullen dit krijgen maar ook de studenten thuis. De snelheid van dataverkeer is richting gebruiker minimaal 2 megabit per seconde, maar als de student dicht bij de centrale zit kan dit oplopen tot 8 Mb/s. Met ADSL zijn videobeelden beeldvullend in tv-kwaliteit weer te geven. Tot de huizen blijven de gewone telefoonkabels liggen, maar in het net zelf zijn veel glasvezelkabels opgenomen. Voor verkeer richting internet is de snelheid lager, maar een halve megabit per seconde is altijd nog haast tien keer zo snel als een 56k-modem.

WAP beschermd tegen virussen

Wij hoorden nog niet van virussen op WAP. WAP, het wireless application protocol, maakt internetten met een geschikte mobiele telefoon mogelijk en heeft naar ons weten nog geen problemen met virussen. F-Secure wil dat voor zijn en levert nu al bescherming tegen dat soort virussen. WAP werkt met speciale versies van internetpagina's of beperkt de hoeveelheid gegevens op normale sites tot voor het kleine 'handy'-schermpje behapbare groottes, maar werkt voor de rest als het normale internet, met alle voordelen en nadelen van dien. Onder de laatste natuurlijk ook virussen; die kon het bedrijf al aanpakken voor normale omstandigheden en nu dus ook voor WAP.

Altavista komt met gratis internet

Gratis met de nodige korrels zout natuurlijk, niet alleen blijft men telefoontikken betalen, maar ook moet een vrij fors entreegeld van rond honderd gulden dokken. Toch kan het aantrekkelijk zijn, want men blijft via Altavista wel verschoond van reclameteksten die via andere providers — de spellingchecker wil dit veranderen in profiteurs — wel via de 'gratis internet'-aanbieding moeten worden geslikt.

Amerikanen surfen onder werktijd

In een ongeloofwaardig bericht werd gemeld dat de Amerikaanse werknemers gemiddeld 21 — ja u leest dat goed: eenentwintig — uur per week onder werktijd websurfend doorbrengen. Het zijn wel niet alle Amerikanen, maar alleen de mensen die op kantoor werken, maar dan nog komt ons het hoeveelheid tijd onwaarschijnlijk voor. Vermoedelijk heeft vrijwel de helft van de kantoorwerkers een pc op zijn bureau staan en aangezien in de VS geen lokale telefoontikken hoeven te worden betaald kost het vrijwel niets om vrijwel constant online te zijn. De 21 uur zegt dus meer over de computerdichtheid op kantoor dan over de gemiddelde surftijd.

Betalen via Telekom

Om websites waar (iets) moet worden betaald voor het geleverde kent men enige methodes, waarvan de creditcard wel de bekendste is. Tegelijk bellen via een betaald 06-nummer is een andere, maar dat heeft praktische bezwaren. Deutsche Telekom biedt nu aan om als bank te fungeren en tegen een maximale prijs van 5 DM per pagina of per minuut kunt u gewoon doorklikken; afrekenen schiet later gewoon via de telefoonrekening.

China neemt gas terug

verkeerd spellen gaan had het misschien niet zo'n vaart gelopen, maar de gebruiker kreeg het al als alternatief voor anti-stress, dat hoewel geen Frans begrip toch vaak gebruikt wordt. De verbeterde versie zal op internet zijn op te halen.

AMD verslaat Intel

De race naar de 1 GHz heeft AMD gewonnen en introduceerde de op die snelheid werkende Athlon net een paar dagen voordat Intel zijn Pentium III op die snelheid kon aankondigen. Niet dat de gewone gebruiker er veel aan heeft want de chips bestaan wel, maar zijn nergens te leveren. Je moet duidelijk meer meebrengen dan een stapel dollars om een van de eerste exemplaren te bemachtigen. Zelfs de modellen die iets langzamer gaan zijn feitelijk niet leverbaar en in de advertenties duiken dan ook nu pas de eerste systemen van 800 MHz pas mondjesmaat op. Geduld is ene schone zaak; het gaat erg snel, maar je moet lang wachten om snel te gaan.

DoS gevaarlijk

Het is een oude naam voor nieuw begrip: Denial of Service. Door 'of' met een kleine letter te spellen zijn kenners gewaarschuwd dat het iets anders betekent. Het begrip houdt in dat een bedrijf niet de diensten kan aanbieden die men er van verwachtte. Het kwam weer in de spotlights te staan toen een aantal crackers — vaak ten onrechte hackers genoemd — kortgeleden Yahoo bestookte met zoveel datastromen dat het systeem daar plat ging. Eén van de routers kan de gegevensstroom niet meer aan en geeft de pijp aan Maarten. Het is weer eens iets anders dan het maken van een virus, maar de ellende die men anderen ermee bezorgt is vergelijkbaar.

CD op stuiver

Dataplay is een klein bedrijf dat — net opgericht — zich bezighoudt met dataopslag. Steve Volk hield zich al bezig met harde schijven voor notebooks in de maten 2,5 en 1,8 inch, maar is nu weer een stap verder gegaan. Op een schijfje zo groot als een Amerikaanse halve-dollarmunt weet hij 500 megabyte te krijgen. Zeg maar bijna een cd. Flash cards zijn natuurlijk sneller en nog kleiner in een apparaat, want daar hoeft geen echte drive bij, maar de kosten geven het systeem toch een goede kans. Een schijfje zal vermoedelijk rond zes dollar gaan kosten en dat is stukken minder dan de flash cards drie tot vier duizend dollar.

Cisco duurste bedrijf

Lange tijd — in it-terminilogie — stond Microsoft bovenaan de lijst met waardevolste bedrijf, maar die rol is nu door Cisco overgenomen. Tegen de huidige beurskoers was de netwerkleverancier eind maart 579,2 miljard dollar waard en dat was op dat moment iets meer — ik wil dat verschil best hebben — dan de 578,2 miljard die Microsoft toen waard was. Die laatste is door alle narigheid over de monopolierechtszaak op dit moment vermoedelijk meer dan een miljard minder dan Cisco waard. Op de derde plaats staat nu General Electronics dat in 1998 door Microsoft werd onttroond. De koers van het Cisco-aandeel was bij deze berekening $79,37 en u kunt dus met een blik op de huidige koers weten

In China neemt men het niet te nauw met de copyrights en dat leidde al tot het volledig apart houden van China bij de DVD-regiocodering. Een weinig steekhoudende maatregel overigens omdat het niet de kopieën van Chinese content is die men vreest, maar juist de content die voor de rest van de wereld is bedoeld. De Chinese regering vreest echter de politieke invloed van internet en zo werd ook het verspreiden van nieuws via internet aan banden gelegd. De internetverbreiding is nog niet groot in China; met negen miljoen aansluitingen heeft nog geen vijf procent toegang, maar de groei is enorm: vorig jaar 400%. De aangekondigde maatregelen moeten vooral het verspreiden van MP3-bestanden beperken. Wil men die en andere videoproducten online aanbieden moet men een vergunning aanvragen. Over controle en mogelijke straffen is momenteel nog niets bekend, maar het is in ieder geval een stap in de goede richting.

Digitale producten BTW-vrij

Onlangs heeft minister Zalm van Financiën besloten dat digitale producten mogen worden verhandeld zonder BTW. Aan de ene kant begrijpelijk vanwege de moeilijke controle, maar aan de andere kant onredelijk voor de overige handel. Als u een programma legaal downloadt via internet is dat meestal al voordeliger omdat de leverancier geen transport- en verpakkingskosten heeft en ook de disk/cd uitspaart. Nu komt er ook nog eens het voordeel van BTW-vrij bij. Zo verdient u uw internetkosten snel terug en het is legaal.

Flexibele camera van Toshiba

Het moest er natuurlijk van komen en als u niet te veel op de kwaliteit lette was het er al. De videocamera's zijn al lang digitaal en de fotocamera's volgden die tendens. De beeldkwaliteit bij de zogenaamde 'still video' moet bij een fotocamera veel groter zijn dan bij de bewegende beelden bij een video-opname. In de Mec100AS van Toshiba gebruikt men MPEG om de beelden te comprimeren en met een voor een (digitale) fotocamera redelijke resolutie kan men nu ook video-opnames maken. De prijs is met ongeveer 2400 gulden niet laag, maar op dit moment wel redelijk gezien de mogelijkheden. Bent u serieus geïnteresseerd moet hij (nog) wel in Japan worden besteld.

GSM vertaalhulp

Misschien handig als je het hard nodig hebt. Virtualplus levert momenteel een vertaalservice; stuur een SMSje met vertaalcode naar het bedrijf en je krijgt per kerende post — lees enkele seconden — een SMSje terug met de vertaling. Werkt voor Nederlands, Frans, Duits, Engels, Spaans en Italiaans. Tot na het examen...

Internet voor post

PTT Post heeft een nieuwe service: u kunt voortaan uw postzegels via internet regelen. Beter gezegd: u kunt uw post frankeren met een speciale digitale postzegel die u via internet kunt bemachtigen. Vervolgens moet u die zelf afdrukken en op de enveloppe plakken, maar direct op de enveloppe drukken kan natuurlijk ook. De postzegel bestaat uit een speciale

of het bedrijf nu nog meer waard is of niet. En voor de goede orde het aandeel Microsoft stond voor deze berekening op $111,69.

Bescherming tegen webaanvallen

Network Associates heeft een programma op de markt gebracht dat kan dienen ter bescherming tegen het platleggen van een website door een vloed aan dataverkeer. Als een server van de website te veel aanvragen krijgt te behandelen, kan die in 'shock' raken en nergens meer op reageren. Om deze 'denial of service' te voorkomen waarschuwt het programma tijdig als er zich zo'n congestie dreigt voor te doen. En... op de markt gebracht doet de makers onrecht, u kunt het gratis downloaden op http://www.nai.com/.

Foutje in Playstation 2

Nu al is de tweede fout in de Playstation 2 van Sony aan het licht gekomen. Of de gebruikers daar echt mee zullen zitten valt nog te bezien want met de in Japan meegeleverde software om DVD's af te spelen is het mogelijk ook de DVD's uit de andere regio's af te spelen. Zeker ook voor Europeanen een interessant gegeven daar wij samen met de Japanners in regio 2 zitten. Maar rent u niet gelijk naar de winkel; niet alleen is de Playstation 2 buiten Japan nog steeds moeilijk te verkrijgen, maar de fout is bij de nu uitgeleverde apparaten 'verbeterd'.

IBM heeft chip om DNA te testen

In de laboratoria van IBM maakte men enige jaren geleden al een 'silicium-kammetje' waarmee de aanwezigheid van bepaalde stoffen kon worden vastgesteld. Een tandje boog als de stof aanwezig was. Hierop voortbordurend heeft men nu de kam laten reageren op bepaalde proteïneparen in het DNA. Met zo'n kammetje verbonden met een elektronische schakeling om het uit te lezen is zo in een draagbaar meetinstrument te maken, maar ook een robot. Deze robots kunnen heel klein zijn, zo klein zelfs dat zij in de bloedbaan kunnen worden ingebracht. De 'nanoprobes' uit StarTrek?

Irridium weg

Het mooie maar dure satellietnetwerk dat onder de naam Irridium rond de aarde cirkelt, maakt 'mobiele' telefonie op bijna de hele aarde — alleen de uiterste polen op grondniveau niet — mogelijk. Het was echter aan de prijzige kant, rond vijftien gulden per minuut en dan moet je wel heel erg nodig iemand moeten bellen om dat te gebruiken. Motorola wil het nog enige tijd aanzien om zaken af te handelen en wil de satellieten dan één voor één naar beneden laten komen. Ze zullen vermoedelijk in de atmosfeer verbranden, maar eventuele brokjes zullen de eenzame oceaanzeiler het leven zuur maken.

No sales-tax werkt averechts

In Nederland zijn via internet geregelde aankopen van BTW vrijgesteld. Dit ook omdat in de VS gekozen is voor de vrijstelling van sales tax bij aankopen via internet. In Nederland betreft dit echter alleen digitale producten, maar in Amerika producten die via internet worden besteld. Dat om te voorkomen dat bepaalde staten geen aanbieders konden hebben omdat

'barcode' die het gemakkelijkst kan worden aangemaakt op de speciale printer die ook nog eens in staat is uw brief te wegen en zo de juiste frankering te regelen. De printer drukt op speciale stickers af die dan op de enveloppe kunnen worden geplakt. De zo gefrankeerde brief kan dan gewoon in de rode bus worden gepost.

Middeleeuwse teksten op volgorde

Teksten uit de (vroege) middeleeuwen werden handmatig gekopieerd en bij dat 'monnikenwerk' werd wel eens van de oorspronkelijke tekst afgeweken. Zo werden soms stukken weggelaten, maar andere keren iets opzettelijk veranderd. Vaak gebeurde het dan ook nog eens door een kopie van een mogelijk gewijzigde kopie te maken. Ben Salemans is kortgeleden gepromoveerd op een onderzoek om met behulp van de computer de diverse kopieën op de juiste tijdsvolgorde te plaatsen. Voor tekstarcheologen een fantastisch hulpmiddel.

Playstation mag worden geëxporteerd

De Playstation 2 van Sony is nog steeds beperkt leverbaar, maar mag tenminste wel vrij worden uitgevoerd. De processor is zo krachtig dat er een raketgeleidingssysteem mee te maken is en daarmee is het een militair belangrijk object en mag als zodanig niet vrij Japan uit gaan. In Japan zijn er inmiddels zo'n 1,4 miljoen verkocht, maar de Europeaan zal tot het najaar moeten wachten om ze in de winkel te kunnen kopen. Sony rekent er op dat in Europa zo'n zes miljoen een baasje kunnen vinden.

Superaanbieding

De fabrikant van supercomputers Cray Research is in 1996 opgekocht door Silicon Graphics voor 700 miljoen dollar. De kroon op het werk zou je menen, omdat Silicon Graphics zelf al zware computersystemen op de markt bracht. De kroon knelde blijkbaar, want na er eerst al op te hebben gezinspeeld, is Cray nu weer afgestoten. De koper is Tera Computer dat zelf ook supercomputers maakt. Dat meervoud is echter nogal voorbarig want Tera Computers leverde tot op heden slechts één systeem af. Met Cray in huis zal dat vermoedelijk veranderen. In ieder geval heeft Tera Computer er met 100 miljoen een superkoopje aan.

Telefoontarieven in België omlaag

In België gaan de telefoontarieven vanaf 15 juni fors omlaag. Waren de tarieven 45 frank voor dal en 120 frank voor piek per uur, gaan ze naar respectievelijk 30 en 60 frank per uur. De tarieven komen daarmee wat beter in overeenstemming met wat in Nederland gevraagd wordt.

USB 2.0 veel sneller

De USB Promotor Group heeft de bètaspecificaties van versie 2 naar de fabrikanten gestuurd. Versie 2 zal vooral veel sneller dan de huidige versie 1.1 zijn en wel veertig maal. Dat betekent dat gegevensoverdracht kan plaatsvinden met 480 Mb per seconde, ruim voldoende voor digitale beelden zodat USB-

alleen die staat een bepaalde belasting eiste. Niemand kocht dan natuurlijk bij een zaak die in zo'n staat was gevestigd. Maar nu hebben de Amerikanen ontdekt dat met name sigaretten en andere rookwaar enorm veel voordeliger kunnen worden aangeschaft door het via internet te bestellen. Lang zal deze situatie wel niet blijven.

Silk versnelt chips

In de laboratoria van IBM heeft men silk — geen zijde — een isolatiemateriaal afkomstig van Dow Chemical in chips weten toe te passen. Het zal nog wel tot midden 2001 duren voordat het massaal in het productieproces zit, maar dan profiteren we ook van een dertig procent sneller werkende chip. Ook de overgang van aluminium naar koper zal dan zijn gerealiseerd en uw computer zal volgend jaar dus weer sneller zijn.

Superdisks

Na Imation heeft nu ook Seagate een superhoge dichtheid op diskette aangekondigd. Zij demonstreerde dat een opslag van 45 miljard bits op een vierkante inch kan. Bits geen bytes! Maar toch zal een op deze techniek gebaseerde diskette van 3,5" 60 gigabyte kunnen bevatten. Dat is weliswaar de helft van wat Imation beweerde te kunnen, maar dit was een demonstratie en dat betekent toch wat dichter bij realisatie.

Topscanner bij Microtek

De Scanmaker X12USL is het nieuwe topmodel van Microtek. Het apparaat levert een scan af met een kleurdiepte van 42-bits. Dit is wel iets minder dan de 44-bits van de Scanmaker 3600, maar in dit geval is de resolutie weer wat hoger. Die is nu 1200×2400 dpi, optisch wel te verstaan en de 3600 haalde maar 600×1200. Dat een A4 zo ongecomprimeerd 1,3 GB kost zal menigeen nog wel even achter de oren doen krabben. Meegeleverd worden de light-versie van Adobe Photoshop en Omnipage. Met de 'eigen' programmatuur is ondermeer scannen met slechts één druk op de knop mogelijk. Blijft de prijs voor al dit moois, die is verbazingwekkend laag met 799 gulden; de Professional-versie is duurder, maar kan ook transparanten aan en kan van automatische documenteninvoer worden voorzien. Voor de goede orde: de Scanmaker 3600 kost maar 249 gulden.

camera's nu een klasse hoger dan webcams kunnen. Bij 30 beeldjes per seconde is er 16 Mb ofwel 2 MB per beeldje beschikbaar. Ofwel 800×600 bij 32 bits kleurdiepte en dat is beter dan uw normale tv en haast 1024×768×24. Achter de Promotor Group staan namen als Compaq, HP, Intel, Lucent, Microsoft, Nec en Philips. Tegen het eind van 2000 is de eerste hardware op de markt te verwachten. Analisten verwachten dat het misschien op termijn wel eens de doodsteek voor SCSI kan zijn.

previous:
Multiplication on a Z80

MSX Computer & Club Webmagazine
issue 92, March/April/May 2000

next:
The LP keyboard interface

experiences with it

The LP keyboard interface

A real MSX fanatic can be recognized by examining his keyboard. It will be worn down by about fifteen years of daily use. None of the impressed letters are visible anymore. Some keys falter and sometimes a wire gets loose on the connector. But hey, whats the alternative? A brand new keyboard! Hardware wizard Leonardo Padial from Madrid, Spain, has designed an external keyboard interface. With this small card, you can simply connect a cheap pc keyboard to your MSX. This will work with any AT or PS/2 keyboard, even if it has special 'Windows' keys. It's not only a good replacement for your old keyboard but can also be a great solution if you want to build your MSX into a pc tower or desktop case. The problem 'how to connect an external keyboard to it', is then solved.

Pierre Gielen

Directory
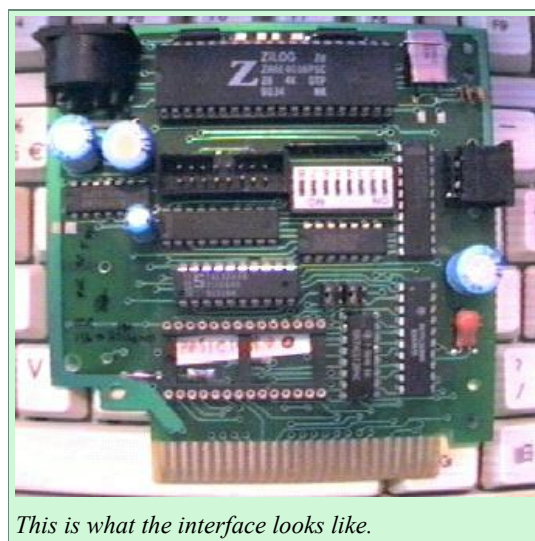
References
1. CHARS.ASM

Building your MSX in a pc case has some advantages: less cables hanging around your computer, it protects vulnerable expansions that are not always built into a cartridge box, a big power supply is provided and there is enough room for a harddisk, cd-rom player, ZIP-drive, etcetera. As you can see on one of the pictures below, a creative person can even build an external MSX cartridge slot into a pc case.

If, however, you don't want all this, you'd better consider buying a second hand MSX keyboard first. Of course that will only be possible if you own an MSX with a seperate keyboard, like the Philips NMS 8250/55/80 or the Sony HB500/700. Keyboards and other spare parts can be found at the bigger MSX fairs. Sometimes a complete second hand MSX computer may even be cheaper than one pc keyboard interface. So if you're looking for a standard MSX, that may be the best choice for you. But if you have a computer that is expanded, like mine, with a 7 MHz clock and an internal megamapper, a replacement will be much harder to find. And if you have an MSX2+ or even an MSX turbo R, you will almost certainly run out of options. In this case, the pc keyboard interface may just be the thing to rescue your computer from a hapless end in the trash can.

Different keys

Connecting a pc keyboard to the MSX is not a trivial task. The first problems are the many differences between the keyboards for both systems. The most obvious one is that the pc keyboard has a lot more keys: `F6` to `F12` for example, `Print Screen`, `PageUp`, `PageDown`, `End`, `Num Lock`, `Scroll Lock`, `Alt`, maybe `Alt-Gr` and if it's a so called 'Windows' or 'Multimedia' keyboard, even a couple of other special keys. The MSX on the other hand, has `Stop`, `Select`, `Code` and `Graph` keys that the pc keyboard doesn't have. In the LP keyboard interface, these MSX specific keys are mapped to the following pc keys:

This is what the interface looks like.

MSX	pc
`Stop`	`Pause/Break`
`Select`	`Scroll Lock`
`Code`	`Left Alt`
`Graph`	`Right Ctrl`

Other 'special' keys on the pc keyboard are left unused in the standard version of the pc keyboard interface. So pressing `PageUp`, `PageDown`, `Num Lock`, `Alt-Gr` (`Right Alt`) or `F6`-`F12` does nothing. I think Leonardo Padial missed a chance here to use at least `F6`-`F10`. Now it is still necessary to press `SHIFT`-`F1` to `SHIFT`-`F5` if you want `F6` to `F10`.

Because `Num Lock` does not work, a thing to remember is also that you cannot use the alternative cursor keys, `Home`, `Del`, etc. of the numerical keypad. For most MSX users, this will not be a problem since on the MSX, the numerical keypad couldn't be used for things other than entering numbers and calculations anyway.

The `Num Lock`, `Caps Lock` and `Scroll Lock` lights on the pc keyboard will not work either. It's a pity that you can't at least see if `Caps Lock` is enabled. Switching the `Caps Lock` light on and off could probably be implemented in the LP keyboard interface, but it isn't yet. This may change in future versions.

If you're not scared of soldering yourself, you could use one of the unused wires in the pc keyboard connector — there is at least one — to make a direct connection to the output for the `Caps Lock` light in your MSX. This also requires a modification in the pc keyboard itself. I have not tried it — yet — so do this at your own risk. After this piece of surgery, the pc keyboard will not be compatible anymore with a pc.

Connecting the interface

The keyboard interface — Leonardo calls it the LP-PCK-V2 — consists of a 10×10 cm small printed circuit board (PCB) that can be inserted into a normal MSX slot. The PCB is not built into a cartridge box, so you should be careful with it.

If you're using it with an MSX turbo R, it is also necessary to connect a flat cable from the keyboard matrix connecter inside the MSX turbo R to a small rectangular connector on the interface. This is needed because of differences in the internal keyboard hardware of the MSX turbo R.

If you want to use the MSX turbo R keyboard socket, one of the eight dip switches on the interface has to be set. I have not tried this, since I do not own an MSX turbo R. The other dip switches are for switching

translation tables, so that the interface can translate several pc keyboards to a number of possible MSX keyboards. Until now, the interface was only available for Spanish keyboards. They are somewhat different from the standard US/International keyboards that are used in the Netherlands. Recently, Leonardo has also finished a Dutch version. Multiple language versions may appear in the future.

The pc keyboard can be connected to one of the two sockets on the board: a big DIN socket for a keyboard with an old AT connector and a small one for keyboards with PS/2 connector.

Micro controller

Translating pc keyboard serial scan codes to MSX keyboard matrix codes is not a simple task. In the interface, it is done by a Z8 micro controller from Zilog. It contains 4 kB of rom, in which an assembler program is burnt. Since the Z8 is mounted in a socket on the circuit board, it can easily be replaced.

All firmware is in the Z8 and, if needed, in an eprom which can also be placed on the circuit board. There is no need for software drivers. Your MSX will immediately recognize the plugged-in interface. This will also work if the interface is in a slot expander, on any MSX, even it it runs at 7 MHz or in R800 mode.

On the interface there is also a socket for a 27C256/512 eprom — erasable programmable rom — or a 32 kB CMOS ram. For the eprom, the dip switch near pin 28 must be in the ON/OFF/ON/OFF position (from left to right). For the ram, it must be in the OFF/ON/OFF/ON position. The eprom — or ram — can contain extensions, like an extra character set. If used with the MSX turbo R for example, it can contain an international character set instead of the original Japanese set. On request, Leonardo can program special character sets into an eprom. Hobbyists who have the proper hardware, can burn their own eproms for the keyboard interface. The example program [1] shows how it can be done. It contains a Spanish character set for the Turbo-R.

Experiences

My experiences with the keyboard interface are mainly positive. I had ordered the interface at the International MSX fair in Tilburg and received it by snail mail a couple of weeks later. It was accompanied by an unclear photocopy and some Spanish instructions. I asked Leonardo about it and learned that this information only concerned a now obsolete version of the interface. A manual for the new version 2 is not necessary, since everything works like it should. Well, almost: one key did not work. It was the backslash ('\') key on the left of the backspace on international keyboards. Without a backslash it is impossible to type in MSX-dos-2 directory names! On Spanish keyboards, this key is apparently placed elsewhere. After contacting Leonardo about this weird problem, he sent me a new Z8 without charge. Great service!

I have not encountered any other hardware or software compatibility problems. Even scanning the MSX keyboard matrix directly by reading and writing to the I/O ports 0A9h and 0AAh of the PPI works perfectly! This means that all software will be able to work with the interface without problems, even if it doesn't use the MSX bios.

The interface mounted in a MSX which is in a pc desktop case.

You can, by the way, keep your old MSX keyboard connected and use the pc-keyboard at the same time. It's up to you to find an application for it.

Special keys

As stated earlier, I found it a little bit disappointing that the extra keys of the pc keyboard are not used. It would have been nice if `F6` to `F10` on the pc keyboard would work like `SHIFT`-`F5` to `SHIFT`-`F6` on the MSX keyboard. I also had to get used to the position of the `Graph` key. Since a pc-keyboard does not have `Code` and `Graph` keys, Leonardo has decided to use `Left-Alt` as `Code` and `Right-Ctrl` as `Graph`. I think it would have made more sense to use `Right-Alt` (`Alt-Gr`) as `Graph` and use `Right-Ctrl` as normal `Ctrl`.

I have asked Leonardo if it was possible to make a new version of the controller software, so that `F6` to `F10` can be used, `Graph` is placed under `Alt-Gr` and `PageUp` and `PageDn` can emulate the MSX key combination `Ctrl`-`CursorUp` and `Ctrl`-`CursorDown`. Some programs, like the popular text editor TED, use these combinations to page up and down. Leonardo has let me know that it is possible, but expensive to produce a single customized version of the interface. At least ten pieces should be ordered to make a customized version possible without extra charge. If you're thinking about buying one, remember to ask if it is possible to get one with a customized translation table.

The interface costs about EUR 66 including postage and packaging. That's not really cheap. But if you decide that a second hand keyboard is not the right solution for you, it's not too much either. If your pc-keyboard breaks down in the future, you can buy a new one for less than EUR 10. Or you can choose to connect a so called 'natural' keyboard or a wireless keyboard to your MSX. And unlike spare parts for the MSX, pc keyboards will probably still be available for a long, long time.

Ordering

The pc interface card can be ordered through the Spanish MSX Club Power Replay. Your contact there is Rafael Corrales de Pulido (replay@meridian.es). Clearly state for which pc-keyboard it must be suitable. In

the Netherlands, we use standard US/International QWERTY keyboard. Belgian, French or German keyboards are different from these. Also don't forget to mention which nationality the MSX keyboard must be. The interface card will be sent to you by mail, normally within two weeks.

For ten or more identical interfaces — that means only one pc keyboard type and only one MSX keyboard type — you can place your order directly with Leonardo Padial (lpadial@teleline.es). In this case, discounts are available and a customized version can be made without extra charge. Contact Leonardo to discuss the possibilities and if you have any technical questions. Besides the keyboard interface, he also produces other hardware extensions for the MSX, like a 16-bits slot expander and of course the Z380 accelerator card that will be at the heart of a brand new modular MSX computer. At the moment, he is working on a video adapter that will be compatible with the standard Yamaha VDP and connects to a VGA-monitor.

More information about MSX hardware can be found on Leonardo Padial's website and on the website of MSX Club Power Replay.

playing in assembly

Metal Gear save file faking

In this article Bas Wijnen explains you how you can fool around with the Metal Gear savegame files. It's not only useful to learn how to cheat, but the assembly programmer may learn something from this also.

Bas Wijnen

Directory

There are many reasons for people to like computers. Playing games is one of them for many people. The feeling that you can finish the game, that you control it and can do everything that is possible with it, is very nice. Another reason to like computers is programming. People who like playing with code, changing it and making new parts to get what they want are called hackers. (Note the difference with crackers. Crackers are those hackers, who break through some sort of security with use of their coding skills.)

Cheating has nothing to do with playing games. It's hacking. If you want to cheat you have to think from the programmers point of view. Imagine you coded the program. Would it in any way be possible to do things that are not meant to be done? This article describes a method, which can always be used if the game uses save files. I shall only write about Metal Gear, but the method is the same for other games.

I shall start with describing the methods for reading save files, finding the desired data — or actually, the location of the data to be changed — and writing the files back. I shall finish with some important locations that I have found.

How to read a save file?
Reading the save file is easy with Metal Gear. The first guess and hope that it is saved in a standard way seems to be incorrect, when reading the file from basic doesn't find an end to it. When you try reading and writing the data, you will find that it is not possible to read more than 255 bytes and that writing 255 bytes is not enough to make a complete save file. This means that, as is usual when you are hacking, assembly language is needed.

Let's just stick to the assumption that it is saved in a standard way. If that is the case — and it seems to be so, since basic can open and read the first part of the file — then the bios routines can be used to read it. So let's try to read the file and store it in memory:

Some values found with the program

Position	description
#000	room
#001	energy
#002	rank-1 (0-3)
#004	weapon in hand (not necessarily in inventory)
#005	item in hand (not necessarily in inventory)
#006	previous room (what's the use of this one?)
#007	maximum energy
#008	radio frequency in BCD
#00A	captives counter, rank up at 5 (and counter resets to 0)
#012	Type of bullet in hand (indeed, you can make a SMG bullet hit like a rocket)
#020	Movement type
#022	y coordinate of the player
#024	x coordinate of the player
#040	maximum bullets in BCD, 2 bytes/weapon, LSB first (0-999)
#050	maximum equipment in BCD, 2 bytes/item, LSB first (0-999)
#090	doors. 0 is open, 1 is closed. It is possible to close even lorries.
#140	weapons: 4 bytes per weapon: +0 = weapon +1 = ammunition (2 bytes) +3 = unused
#160	0 means the weapon is still on the map. 1 mens it isn't

```
                     ML-listing: READ.ASM

        DB    &HFE           ;Header to make it a .bin-file
        DW    &HC000
        DW    EIND
        DW    &HC000

RDHDR:  EQU   &HE1
READ:   EQU   &HE4
MOTOR:  EQU   &HF3
CHPUT:  EQU   &HA2

SIZ1:   EQU   &H11           ;so it breaks after the name
SIZ2:   EQU   &H0301         ;metal gear save file length
BUF1:   EQU   &HCF00
BUF2:   EQU   &HD000
SIZE1:  EQU   &HCEFC
SIZE2:  EQU   &HCEFE

        ORG   &HC000
        CALL  RDHDR          ;Read first header
        DI
        LD    DE,BUF1
        LD    HL,SIZ1
        LD    (MAXSIZ),HL
        CALL  RDTAP          ;Read file-name
        LD    (SIZE1),HL

;print filename to screen
        LD    HL,BUF1+10     ;offset:some bytes before actual name
        LD    B,6
LOOP2:  LD    A,(HL)
        PUSH  HL
        PUSH  BC
        CALL  CHPUT
        POP   BC
        POP   HL
        INC   HL
        DJNZ  LOOP2
        LD    A,13           ;add a return
        CALL  CHPUT
        LD    A,10
        CALL  CHPUT

        CALL  RDHDR
        DI
        LD    DE,BUF2
        LD    HL,SIZ2
        LD    (MAXSIZ),HL
        CALL  RDTAP
        LD    (SIZE2),HL
        RET

RDTAP:  LD    HL,0           ;actual tape-reading
LOOP0:  PUSH  HL
        PUSH  DE
        CALL  READ
        DI
        POP   DE
        POP   HL
        JR    C,STOP
        LD    (DE),A
        INC   DE
        INC   HL
```

#168	equipment:4 bytes per item:
	+0 = item
	+1 = amount (only for rations) (2 bytes)
	+1 = character in screen (only for cards)
	+3 = unused
#1D8	0 means the item is still on the map. 1 means it isn't (some items are on multiple spots)
#229	Shoot gunner. 0 means alive. 1 means dead. 2 means alive with his story told

```
        LD      BC,(MAXSIZ)
        AND     A
        PUSH    HL
        SBC     HL,BC
        POP     HL
        JP      NZ,LOOP0

STOP:   XOR     A
        PUSH    HL
        CALL    MOTOR
        POP     HL
        RET

EIND:

MAXSIZ: DS      2               ;don't put variables in reserved (saved) memory
```

But before we can read the data, we need to know how many bytes we have to read. Just experimenting with SIZE2 gave me &H301. If you run the program on your save file now, it will provide you with the data stored in BUF2. It is very easy to save it to disk, you can just use bsave.

The location of the data

What we want next, is changing the data in a way that it tells that we have all the weapons, energy, or something else. To find the location of the data, we need to make two save files, with only a few differences and compare them. For example, you make a save file in the elevator. You get out, fire exactly one shot and go in again, then you compare the two files. Luckily, the Metal Gear save files are not encryped. If they were, we had to crack the code as well. But now life is easy. Values that can be found rather easily are life, inventory, rank, radio frequency, location etc.

It is possible to check the differeces by hand, but if you have a computer, you can just as well let it do the work. If you move one save file to &HD400 and another to &HD000, then the following code-fragment does the comparison:

Movement modes

Value	Function
0	normal
1	demo (no reaction on controls for a long time, later normal)
2	in elevator (no up/down, block on left wall)
3	no movement at all
4	parachute (straight down)
5	blowing on the rooftop of building no 1
6	no up/down (ladders at the end)
7	climbing (no left/right)

Weapons

Value	Weapon
0	nothing
1	handgun
2	sub machine gun
3	grenade launcher
4	rocket launcher
5	plastic explosives
6	land mine
7	remote controlled missile
8	silencer

Equipment

Value	Item
00	nothing
01	body armour
02	bomb blast suit
03	flashlight
04	infra red goggles
05	gasmask
06	cigarettes
07	mine detector
08	antenna
09	telescope
0A	oxygen cylinder ('BOMBE')
0B	compass
0C	parachute

ML-listing: CHECK.ASM

```
        DB      &HFE
        DW      &HC000
        DW      EIND
        DW      &HC000

CHPUT:  EQU     &HA2

BUFFER: EQU     &HD000
COPY:   EQU     &HD400
SIZE:   EQU     &H0304

        ORG     &HC000
        LD      HL,BUFFER
        LD      DE,COPY
        LD      BC,SIZE

LOOP0:  LD      A,(DE)          ;check all bytes and print them if not equal
        CP      (HL)
        CALL    NZ,PRINT
        INC     HL
        INC     DE
        DEC     BC
        LD      A,B
        OR      C
        JP      NZ,LOOP0
        RET

PRINT:  LD      A,H             ;address
        CALL    PRTHEX
        LD      A,L
        CALL    PRTHEX
        CALL    PRTSPC
        LD      A,(HL)          ;byte 1
        CALL    PRTHEX
        CALL    PRTSPC
        LD      A,(DE)          ;byte 2
        CALL    PRTHEX
        PUSH    AF
        PUSH    DE
        PUSH    BC
        PUSH    HL
        LD      A,13            ;return
        CALL    CHPUT
        LD      A,10
        CALL    CHPUT
        POP     HL
        POP     BC
        POP     DE
        POP     AF
        RET

PRTSPC: PUSH    AF              ;print a space
        PUSH    BC
        PUSH    DE
        PUSH    HL
        LD      A," "
        CALL    CHPUT
        POP     HL
        POP     DE
        POP     BC
```

```
            POP     AF
            RET

PRTHEX: PUSH    HL              ;print a byte in hexadecimal
            PUSH    BC
            PUSH    DE
            PUSH    AF
            PUSH    AF
            RLCA
            RLCA
            RLCA
            RLCA
            LD      B,2
LOOP1:  AND     &H0F
            PUSH    BC
            ADD     A,"0"
            CP      "0"+10
            JR      C,SKIP0
            ADD     A,"A"-"0"-10
SKIP0:  CALL    CHPUT
            POP     BC
            POP     AF
            DJNZ    LOOP1
            POP     DE
            POP     BC
            POP     HL
            RET
EIND:
```

If you tried to find the room, you probably compared the starting room and the first elevator and found that more than one byte has changed. One of them really is the room. Others are the room you came from, position on the screen and what I call the "movement mode". This variable must be there because the allowed movements in the elevator and in a normal screen are totally different.

Write your own save file

When you write your own save files with data you didn't find in known saves, but what you just guessed — where would room 50 be? — don't be surprised if you hang the computer. It will never cause any permanent damage — it's only software —, the game might just hang. But that shouldn't bother you. Just try another value.

Writing the save file must also be done in assembly language, too. That isn't really hard anyway, so let's just do it:

ML-listing: WRITE.ASM

```
            DB      &HFE
            DW      &HC000
            DW      EIND
            DW      &HC000

WRHDR:  EQU     &HEA
WRITE:  EQU     &HED
MOTOR:  EQU     &HF3
BUF1:   EQU     &HCF00
BUF2:   EQU     &HD000
SIZE1:  EQU     &HCEFC          ;make sure this is where read has stored them
SIZE2:  EQU     &HCEFE

            ORG     &HC000
            LD      A,1             ;long header
            CALL    WRHDR
            DI
            LD      HL,BUF1
            LD      DE,(SIZE1)
            CALL    WRTAP
            XOR     A               ;short header
            CALL    WRHDR
            DI
            LD      HL,BUF2
            LD      DE,(SIZE2)
            CALL    WRTAP
            XOR     A
            JP      MOTOR           ;This will return to the caller

WRTAP:  PUSH    HL
            PUSH    DE
            LD      A,(HL)
            CALL    WRITE
            DI
            POP     DE
            POP     HL
            INC     HL
            DEC     DE
            LD      A,E
            OR      D
            JP      NZ,WRTAP
            RET

EIND:
```

Of course you should use this after you made your changes. If you want to be able to do it quick and you have two MSXs, you can connect them. One with the Metal Gear cartridge in it, the other running the code above or something similar. The signal must be ampilfied, for the record signal is not strong enough to be recognised by the other MSX. Just put a tape recorder in between to amplify the signal. This must not be an official computer tape recorder like the one Philips made. They don't output the signal that they are recording, so that doesn't work.

Set it to record and plug in the white plug from the Metal Gear computer and the red one from the other. Do the same with a second tape recorder and the other two

plugs. The 'motor on/off' plugs don't need to be connected. All connections are shown in figure 1. Make sure you give the load command before you give the save command. Load will wait for data, while save just starts writing straight away.

A table of values I found
With the above information you should be able to find all the values yourself, on the left there is a table of some values I found.
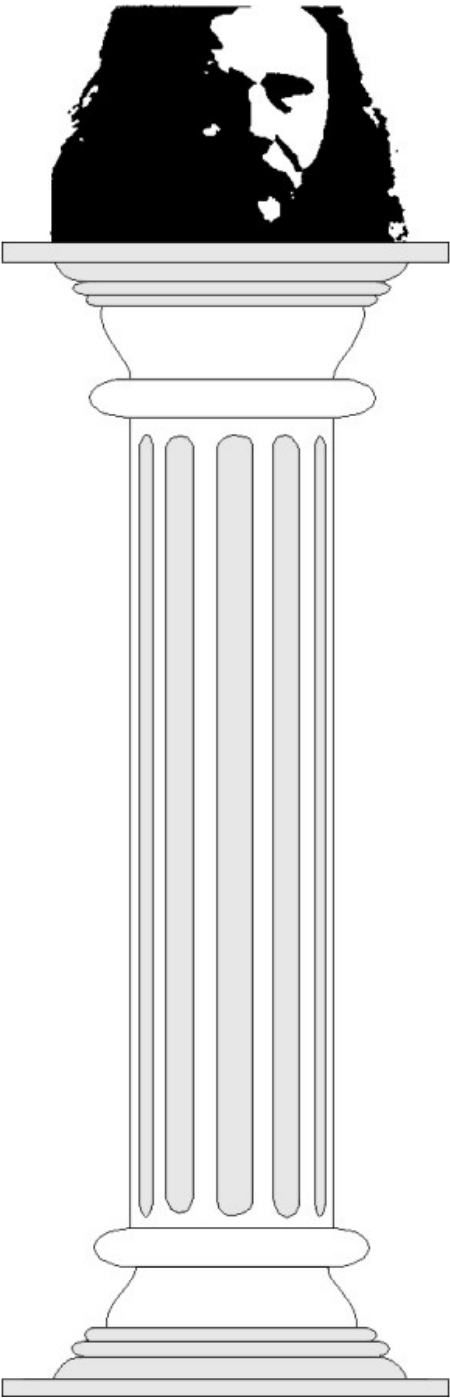
What's the use?
Now the final question: why would you do all this? The entertainment of the game certainly doesn't improve by

Figure 1: *Connecting two MSXs with tape recorders*

cheating. Actually, the game is very boring if you cheat. So why would anyone want to cheat in games? One reason, which goes mostly for children, is that it is an easy way to finish the game. Just start with a large inventory in room #76 — where Metal Gear is — and you'll be finishing within two minutes. If this was the reason you did it, you probably never do it again, since you'll find this is not a very satisfying way of finishing the game.

A very good reason to do it is what I mentioned at the start. You really control the game and you can do anything you want in it. That is what hacking is about in general: controlling the computer. Many people like playing games for that reason. For those people making a cheat in a game is fun. Not because they can cheat with it, but because the making is a puzzle by itself.

previous:
The LP keyboard interface

MSX Computer & Club Webmagazine
issue 92, March/April/May 2000

next:
[nl] Wammes' kolom

previous:
Metal Gear save file faking

MCCW issue 92, March/April/May 2000
Back to contents

next:
Find It

Wammes' kolom

De redactie

En dan niet die van het als een Fenix uit de as der tijden verrijzende MCCW, maar de goede oude MCM-redactie; daar wil ik het deze maand eens over hebben. Immers, het blijkt keer op keer dat ook de huidige MSX-gemeenschap, die elkaar op het Internet treft — met zo nu en dan een beursje — altijd nieuwsgierig blijft naar de tijden van weleer. Bovendien, schreef hij eerlijk, Manuel dringt aan op een stukkie en het is momenteel eigenlijk te druk om er echt voor te gaan zitten. Dan is zo'n "uit de oude doos" bijdrage een snelle en hopelijk afdoende oplossing...

De MCM-redactie heeft op meerdere plekken gezeteld, gedurende de jaren dat het blad bestond. Maar het begon allemaal op een tweetal zolderkamertjes in Amsterdam Nieuw Zuid. Waar ik woonde, boven mijn ouders. Wegens mijn vaders ziekte ben ik daar tamelijk lang blijven hangen, inderdaad.
De eerste maanden was het allemaal nog wel overzichtelijk, ik was er nu eenmaal aan gewend om met ruimte te woekeren. Maar er kwamen steeds meer computers binnen wandelen... en in die tijd had je er ook heel wat nodig. MSX was nu eenmaal ontworpen als algemene standaard, maar dat viel in de praktijk behoorlijk tegen. Allerlei fabrikanten hadden toch zo hun eigen eigenaardigheden, die soms per model verschilden. Tegenwoordig weten we waar die uitzonderingen aan liggen, toen niet. De enige manier om zeker te zijn dat een stukje machinetaal inderdaad op een bepaalde machine werken zou was uitproberen. En dus liep de populatie MSXjes in die zolderkamers al snel behoorlijk uit de klauwen.
Om het allemaal nog een graadje erger te maken, een goede MSX tekstverwerker was er nog niet. Dus het eigenlijke schrijven moest op een — hou je vast — Commodore 64 gebeuren. En dat allemaal met maar één enkele tamelijk kleine werktafel, waar ook nog een aquarium op stond.

Dat aquarium heeft die hectische tijden niet lang overleefd: de salamanders bleken massaal overleden en dat was maar goed ook, zo kon er weer een stukje werkruimte worden geclaimd. Maar daarmee was de koek dan ook wel op. Met regelmaat stonden er naast die tekstverwerker — om die naam niet nogmaals te hoeven noemen — met zijn losse drive, eigen monitor en aparte printer ook nog eens twee of zelfs drie MSXjes naast en op elkaar op dat bureau. De rest van de vloot slingerde door de hele ruimte, 's nachts moest ik regelmatig een stel van mijn bed jagen alvorens tussen de lakens te kunnen kruipen. Het terugvinden van boeken en software werd een asymptotisch uit de hand lopend probleem, de eerste maanden: de stapels werden hoger en hoger en na iedere panische zoekactie was de organisatiegraad weer wat minder. Inderdaad, de tweede wet der thermodynamica begon mijn leven en werken te overheersen.

Maar pas nadat Lies en ik eens midden in de nacht bijna gesmoord werden onder een deel van de bibliotheek — twee hoge stapels boeken en tijdschriften naast mijn bed dat gewoon op de grond lag, waren omgedonderd — kwam ik tot de conclusie dat het nu toch echt onhoudbaar werd. Gelukkig was dat ook het moment dat bij de uitgeverij van MCM wat ruimte vrij kwam, zodat ik kort daarop in een oude keuken kon resideren. Met maar liefst twee werktafels en een rechtstreekse verbinding tussen die tekstverwerker en de zetmachine. Een echte RS-232 kabel van pak-hem-beet twintig meter, 25-aderig uitgevoerd en zwaar afgeschermd was een hele verbetering vergeleken met het acoustische modem dat we tot die tijd gebruikten om teksten over te piepen. Voor hen die dat niet hebben meegemaakt: een acoustisch modem is een apparaat met twee rubber doppen, waar een telefoonhoorn in past. Niks geen aparte kabeltjes, gewoon bellen en de hoorn inklemmen. Als je vervolgens maar niet met de deuren slaat komt de tekst meestal redelijk goed over, maar wel zo langzaam dat je op het scherm nog net mee kan lezen. Dat was knap frustrerend soms, als er een spelfout in de als definitief bedoelde tekst passeerde...

Volgende keer — tenzij ik inspiratie opdoe — verhalen uit de keuken!

Wammes Witkop

previous:
Metal Gear save file faking

MSX Computer & Club Webmagazine
issue 92, March/April/May 2000

next:
Find It

previous:
[nl] Wammes' kolom

MCCW issue 92, March/April/May 2000
Back to contents

next:
Big computer meeting 2000

find the differences

Find It

Remember those two drawings which at first sight look identical? Find It is exactly this type of puzzle game. Albert will tell you in this review if the computer version is as interesting as the one in the newspapers...

Albert Beevendorp

Directory
The game
The graphics
The music
Packaging
Hall of fame
Conclusion

First released at the Tilburg 2000 fair, finally a new type of puzzle game is presented for the MSX2. This puzzle game was made by Delta Soft with music from Surrec. The full name of the game is a bit of a mouth-full: "Find It - Metal Gear Solid Edition". The somewhat original package contains the six game disks, the manual which is in both Dutch and English and an audio-cd. The game can be installed on a harddisk for faster loading and to reduce the disk switching. The audio-cd contains the MoonSound music in the game as the music is MoonSound only.

Figure 1: *Intro screen on MSX2*

Figure 2: *Intro screen in MSX2+ and MSX turbo R version*

Delta Soft currently is working on a patch which enables Find It to use some specific MSX2+ and MSX turbo R features. These features include mainly some pictures which are 'converted' to the MSX2+ screen mode and some minor bug-fixes. The first patch is already released at deltasoft.msx2.com and includes a new intro screen. The patch is done on the disks, so if you want to make the other version as well, copy the disks and start patching on these copies. In figures 1 and 2 you can see the difference in the intro screens.

The game
When Find It is started for the first time, you need to configure some things such as controls and language. These are saved on the first disk or on harddisk and can be changed when you start the game next time. Then a text is printed providing the means to change these settings. After the funny scene with the Delta Soft logo the intro screen appears and a demo will start after a while if nothing is done.

The game itself contains eight levels of four rounds, so there are 32 puzzles. Every puzzle contains two pictures. One picture is the original and the second contains some differences. The number of differences depend on the round you are in. The aim is finding these differences. The faster you do this, the higher your score. After every round you get a password, which you can enter using the password selection at the intro screen. After every level you will also get a small demo starring members of the Metal Gear Solid cast drawn in SD style. These small demos change when playing on another difficulty level — the level selection at the intro screen — to encourage playing on a different difficulty level as well. Find It can be pretty addictive at times, especially when you just cannot find the last difference.

Intro to level 1

The gameplay could be better. The movement of the cursor is not smooth and with the `TAB`-key — used to speed up movement when playing with cursor keys or joystick, similair to Video Graphics — the cursor jumps farther when pressed too long. The problem is that it is hard to pinpoint. You can easily cheat by pressing the `pause`-key — if your MSX has one — during the game, but I have noticed in the MSX2+ and MSX turbo R version this is removed. One minor bug still is that the game simply quits with an error message when attempting to write to a write-protected disk. In my opinion original disks should be write protected. Another minor bug is the fact that there is not enough time to write down the password. I have mailed the programmer about these bugs and I hope there will be a bugfix for these as well.

The graphics
A puzzle game like Find It highly depends on the graphics. Especially for the puzzles the graphics should be good. Many of the puzzles have converted pictures. Because of the — sometimes — poor conversions and the positions and sizes of the differences the puzzles can be pretty hard to solve. I am still busy with a puzzle where I just cannot find the last difference because of this. As said above, you can see the difference between MSX2

and MSX2+ and MSX turbo R version for the intro screen in figures 1 and 2. From those screenshots you can draw your own conclusion, I hope.

Level 1 puzzle

The music
Find It contains MoonSound music only. Even though I think the music is good — many tunes are taken from Metal Gear Solid or Solid Snake and they are arranged pretty well —, there is one disadvantage about it: there will be no music for MSX Music and/or MSX Audio. Because of this, users without MoonSound can enjoy the music using the audio-cd, as many people do have a cd-player nowadays. I hope Delta Soft will continue doing this even when they are going to add music for MSX Music and/or MSX Audio as well as MoonSound music. As the music is partly from Metal Gear Solid, it suits the graphics of the game. However, at many times boss-tunes are pretty loud and boring, so here they are too.

Packaging
The disks, audio-cd and manual are packed in a zippable cd-case, which is very special. As you can see in the photo, these all are labeled in black and white. The original MERCK disks — I have never heard of them — contain the game files. It seems that a big part of the price is spent on the packaging.

Hall of fame
Depending on your total score it can be recorded in the hall of fame when you have finished the last puzzle. The hall of fame is saved on the last disk or on hard disk, so make sure to have both disk 1 and 6 write unprotected (see above). Using this hall of fame you can compete with your friends.

Package and contents

Conclusion
Playing the game is a lot of fun, however sometimes pretty difficult because of the position and sizes of the differences in combination with the sometimes badly converted graphics. The second puzzle is pretty hard as it is in black and white and contains much black. Other puzzles have so few colours you think it is too easy to play, but still it is pretty hard. Delta Soft put a lot of humour in Find It. The music is good — with the condition that there is no music for MSX Music and/or MSX Audio — but can be a bit boring sometimes. The puzzle type is unique for MSX, the small demos are fun and the gameplay is fine. I myself am waiting for the next MSX2+ and MSX turbo R patch for even more fun. If you have a MoonSound and like this kind of puzzle games, you should buy it. If you don't: the cd helps a little bit, but playing without the proper music is a little boring.

previous:
[nl] Wammes' kolom

MSX Computer & Club Webmagazine
issue 92, March/April/May 2000

next:
Big computer meeting 2000

previous:
Find It

MCCW issue 92, March/April/May 2000
Back to contents

next:
23 steps to high resolution on MSX1

Tilburg fair 2000: a report

Big computer meeting 2000

Last year's Tilburg meeting brought little news; it might have been much more enjoyable than the previous years, the new location had to be gotten used to pretty much and a lot of products weren't finished yet or had already been showed to the crowd earlier. Hence I travelled to Tilburg with doubt this year. Was it going to get worse every year from now on? A report.

Anne de Raad

(Translated from Dutch by Laurens Holst.)

For me, the meeting actually started a few days earlier. As soon as I heard from the Japanese Kuniji Ikeda that he was planning to visit Tilburg with some fellow MSX-freaks this year, I immediately invited him to spend a day with me in Hoorn. Civilized, as Japanese are, he accepted my offer and promised to 'keep me informed'. This half word was of course enough for me and I made my plans to calmly visit the fair on Saturday.

To my great surprise I received a mail from Ikeda the Wednesday previous to the fair in which he wrote that he would arrive Thursday at Schiphol. He asked whether Rieks Warendorp Torringa would also be there; since I didn't speak Japanese and he actually didn't speak English either. 'We're with four persons', he also wrote.

Ikeda and the drum system

PPI drum system
That same day the mailman delivered me a strange parcel from Japan. Ikeda sent me his 7 kilo heavy drumsystem which he desired to show at the fair. Gradually I started to realize that I could really expect some Japanese visitors. And not one Japanese, but four!

Since none of the Japanese spoke English, I had to contact Rieks as soon as possible. The only thing I knew about him was that he studied Japanese in Leiden and also lived in Groningen. That's all. Fortunately, there was someone on the Undernet's #msx channel who could provide me with his telephone number. I directly got to speak to Rieks and it appeared that he also had been informed of the visit of the Japanese. He would even collect them from the airport together with Robert Vroemisse.

Ultimately we decided Rieks would go to Hoorn with our Japanese friends to get acquainted and pick up Ikeda's drum system.

Some of the stuff the Japanese brought with them...

Keukenhof
It went differently. Thursday night I received a telephone call from Rieks. He and the Japanese were in his student room since there was no room left in Leiden because of the Keukenhof. Whether I could drive to Leiden to pick up four Japanese and one Rieks. Things went as planned and close to midnight I was talking with Rieks and four Japanese about the upcoming fair and MSX in general.

The suitcases they brought were crammed more with MSX-stuff than with clothes. They for example brought an enormous load of original software with them, including Salamander with an original MSX Salamander telephone-card, the MSX2 version of King's Valley II, Space Manbow, Hydlide 3 MSX2-version and much more. Despite of my pleads they insisted to put the games on for auction at the fair. Hence this happened. A number of MSX users have been made very happy with it.

Kazuhiko Nishi
They also brought an example of the new Japanese MSX MAGAZINE. This magazine is officially supported by ASCII and must be the first impulse to re-introduce the MSX-system in Japan.

Furthermore, the Japanese sold original MSX2+-computers and a MSX turbo R FS A1ST. They also brought nice MSX-merchandise with them like Snatcher Collection Cards. Last but not least everybody could read the letter Kazuhiko Nishi himself wrote to all European MSX users. It seems ASCII still takes MSX seriously.

Fortunately, the fair had more to offer than what Ikeda & co. had brought. The Japanese influence however still remained. A few Dutch clubs have contacted Japanese MSX users and this resulted in a growing offer of Japanese software.

Me (Anne) and Yokoi

Software
Among others the game 'Kyokugen, The Ultimate Conquest' was for sale, a graphically and musically good shoot 'em up, made by M-Kai. This game could be acquired at the Totally Chaos Team booth of Rinus Stoker.

MSX-NBNO has been known for its contacts with the far east. For some time, they publish the Japanese NV-magazine on the Dutch

market. This diskmagazine is released once a month and is edited by Syntax from Japan. The creators of XSW-magazine also had a couple of new Japanese games: CosmoGang De Puzzle, a nice Tetris-clone and Dead Or Live, a beat 'em up game a bit alike Fighter's Ragnarök.

Ghost

The Japanese booth-keepers were housed in a seperate room. There also was Ghost selling his game 'Morning Star'. I especially noted the shoot 'em up's very nice packing.

Fortunately the Dutch softwareproducers also showed their best side. At Sunrise for MSX the first part of Umax's new RPG could be bought: 'Realms of Adventure'. Sunrise also presented 'Pentaro Odyssey 2', a new platformgame of Spanish make. DeltaSoft came up with 'Find It', a nice puzzlegame, greatly inspired on the great Playstation success 'Metal Gear Solid'. Sargon sold 'Randar II', entirely translated to English, TeddyWareZ impressed with their 'SCC Blaffer NT' and their new music-disk 'Tunez II' and Compjoetania TNG had another update for 'Compass'.

The Teddywarez crew and Wolf

Hardware
Next to new software also some cd's could be acquired with MSX-related music. Remco for example sold the 'MSX MP3 Collections' 1 to 4 and MSX-NBNO also had a new part in their series 'Moonsound Greatest'.
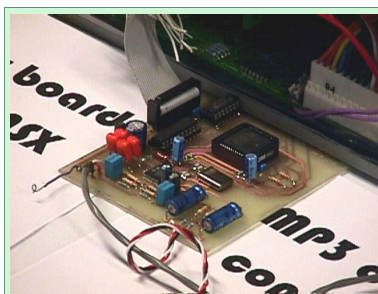
The MP3 decoder by Compjoetania TNG

The meeting was also very interesting for hardwarefreaks. Leonardo Padial Ortiz, from Spain, demonstrated his 'LPE-Z380 cartridge' with the 32 bits Zilog Z380-processor. Sunrise for MSX still had some GFX9000s for sale, as well as the nessecary IDE and RS232-interfaces, which were available as seperate cartridges as well as integrated ones. Compjoetania TNG showed that MP3-files can not only be replayed on pcs. There was a prototype of a hardware MP3-decoder controlled by an MSX.

Of course the necessary devices and printer ribbons could also be found at the fair. Bas Kornalijnslijper of MSX Club West-Friesland once again brought an impressive amount of hardware with him and this time also some fancy 'gadgets' for the Playstation.

Conclusion
I could go on like this for some time more. The Big Computer Meeting 2000 was a success. There were a lot of new developments. The entire day it was pleasantly crowded and the organizing CGV arranged everything very well. The international character of the fair expressed itself very well this year. Next to Belgian, Spanish, German and Swiss booth keepers there also were a nice amount of Japanese this year. In the meantime I heard from Ikeda that they also enjoyed themselves very much. Japanese fairs are very different. On those fairs, there are a lot more lectures about several developments or products. It might have been too bad that there wasn't room for things like that on Tilburg 2000, since one of

The MCCW booth

the Japanese had prepared a speech which he had very much liked to give to us European MSX-users. Maybe next year. Until then!

More pictures can be found on <u>FUNet</u>.

23 steps to high resolution on MSX1

In the previous episode we talked about theories, principles, numbers, mathematics and stuff like that. But, we computer-oriented people are all a bit of practicalist — who usually just tend to get bored to death with anything involving a pen and a paper — so let's get down to business. Let's see what we really can do and even dish out a bit of source code along.

Antti Silvast

Directory

References
1. MAKEINT.BAS
2. INT.GEN
3. FILE.INC
4. INTER.LZH
5. PRETWIST.PAS
6. TW.GEN
7. TWIST.LZH

There's three kinds of effects I've produced with the MSX: pattern based effects, chunky based effects and character based effects. Pattern based effects dump to the *pattern table*, chunky based effects to the *colour table* and character based to the *name table*. In this article I will go more in depth to the truly most spectacular thing you can do on the MSX1 — that is, before you've discovered the character based effects — pattern based effects.

But still on a more general level, somewhere in between the masses of ignorance our demos are achieving worldwide I've sensed a bit of disappointment towards the fact that our effects are all so polite; even running correctly on top of emulators and definitely not stretching the limits of MSX1's hardware features. We'll, we'd sure like to, but there aren't any; at least much of use.

When I divided the effects a moment ago an MSX2-demowatcher would've instantionally pointed out the total absence of sprite and videobeam based effects. Yes, I know they've multiplexed the four — or whatever — sprites C64 has all over again and we have a massive amount of 32 even without multiplexing. But then the MSX-standard features the four sprites per line limit, a limit that you can't as far as I know overcome anyhow at all, since the Z80 is too slow to do almost anything during one scanline. Someone correct me if I'm wrong, I haven't got any numbers one me and haven't ever tried to overcome the four sprites limitation; I did however try to multiplex the 32 sprites from the upper part of the screen to the lower and couldn't get even that working.

And as for the videobeam, we've already proved it is possible to do raster tricks even without a horizontal retrace or some nice memory place updating the current videobeam line: just make a big cpu-loop out of it, it takes about four NOP's for one scanline during the retrace. After the effect truly takes up all cpu-time, the display is all flickery and you'll have to rewrite your music players all over again, but that wouldn't be a real problem if this trick was of any use which it isn't. This is MSX1, we haven't got any scroll registers, just the crummy base registers that all work on at least 6 bits a granularity. The only real use I can think of for this trick is to change the background color (vdp 7) for every row which works fine, but why would you want to since it's just as possible in SCREEN 2? Aside from that, if I never ever was to touch SCREEN 3 again that could prove to have some implementations: improving the resolution, color flickering, maybe using hires-borders in SCREEN 2 and switching the display mode in between the retrace. But alas, I'm not.

Planarity revealed
Let's get to the pattern based effects which truly are one form of art. The display mapping in SCREEN 0, 1 and 2 is planar, that is, 8 pixels of graphics are packed into 8 bits or one byte. Every byte has two colours, but you can change the colours for every 8×1 a tile - see the previous article for more on this.

For a 32-bit-programmer this planarity will seem inconvenient, but for us it's a true blessing and the reason you can make 1×1 dumping effects fast on the MSX. It's all in the numbers: at 16 colours a chunky buffer will take up one byte to store two pixels. A planar buffer takes one byte to store eight pixels so that's four times the speed. Remember, it's all on how fast you can output your effect. Plus we get 1×1 accuracy where chunky pixels can only be four pixels wide on the MSX1. If you're still a bit confused thinking about the locations of different pixels stored in the byte, don't be: we're never going to touch them. Of course it would be dead slow and fiddly to start shifting and masking the byte to alter one individual pixel, but the magic is never to touch anything smaller than one byte, 8 pixels.

If you're not used to planarity, this will take some readaptation. How do you produce a dumping effect that cannot change every pixel? Just make it row based. A truly simple example would be a vertical scroller: you always dump the entire screen row using the same row from the virtual buffer. You won't need to change the way the pixels are packed inside the bytes. A horizontal scroller on the other hand wouldn't be possible in a straightforward manner since for different frames you'd need to alter the positions of the pixels in horizontal direction.

I will now introduce two of my previous effects, the interference rings and the twisting bars. But first let's correct a few misunderstandings. Someone once claimed these effects take up all the cpu-time. Well, they don't, in fact, neither of them outputs the 2 kB we could on one blank; so both of them waste precious time just waiting for the vblank, time that could be used for instance to play a background digital sample. In fact, the day I write an impossible effect for the MSX you'll have my word it isn't running on the MSX at all.

Interference rings $0.00
For the principle on interfering rings see figure 1. Just like we've learned in physics, interfering is summing two waveforms and that's how we create the interference rings: sum two images of rings. Now to optimize it for the pattern based techniques let's set two restrictions: First, the interfering images can only move up and down. Second, both the images will be two-coloured with colours 0 and 1 such that when interfering 0+0=0, 1+0=1,
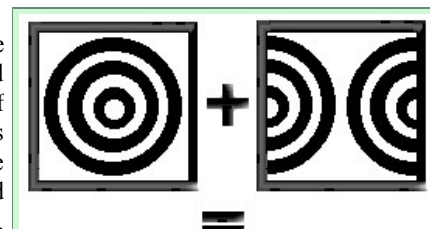
0+1=1, 1+1=0. After that we can store both the images in planar buffers and never have to change the way pixels are packed within the bytes since the images can't move sidewards. And as for our summing-operations, it turns out that XOR has exactly the properties required: 0 XOR 0=0, 1 XOR 0=1, 0 XOR 1=1, 1 XOR 1=0. Since our pixels are packed as individual bits inside a byte, we can XOR an entire of 8 pixels at a time, at the cost of one operation. After we're done XORring, that will be the corresponding screen value, so we can directly output the obtained value. Here's the Z80 inner loop for interfering and outputting:
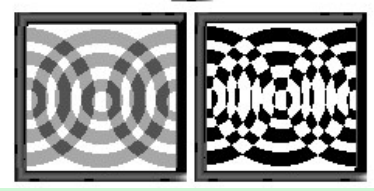

Figure 1: *Interfering two waveforms*

```
; ...
_innerloop:
    ld a,(de) ; de is image 1
    xor (hl)  ; hl is image 2
    out (#98),a ; output to screen
    inc hl
    inc de
    djnz _innerloop ; dec b, loop if > 0
    ; ...
```
ML-listing: `INLOOP.GEN`

Next up, let's think about the size of the thing. Let's say our output window is $x×y$ pixels large where x is dividable by 8. As you recall we can do 8 pixels at a time, so this window actually requires $x/8×y$ outputs. As I said in the previous article, we can only do 2048 outputs per frame, so our formula becomes $x/8×y$=2048. So the window could be for instance 128×128. For this particular effect we will however only use 128×96 for symmetry reasons to get four sets of interference rings on screen: this is what I meant by wasting resources.
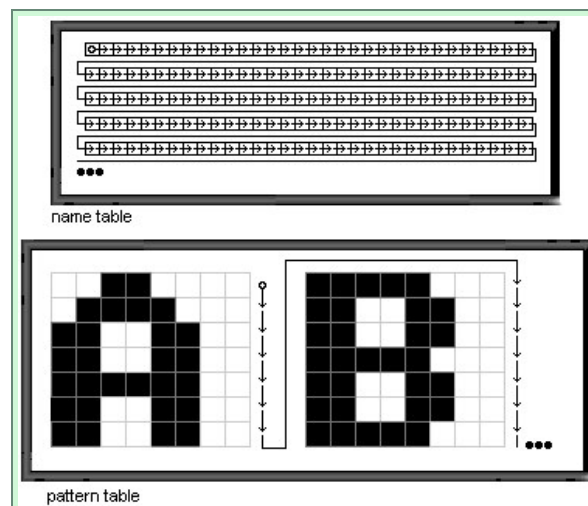

Figure 2: *name table and pattern table as a default*

Last of, we're in a character based screenmode, so we'll have to consider how to characters are organized in the memory. The basic layout goes as seen in figure 2. As you can see, this could be all right if we wanted to output text, but as far as pattern based effects go, it's really clumsy. First the eight horizontal neighbouring pixels are packed into one byte. Then the following byte will be the eight pixels below these. After we're done with the eight rows a character has, we'll have to move eight pixels back up and eight pixels sidewards. How would you implement anything when dealing with an order like this?

Luckily, we don't have to. Enter the pseudolinear mode; It's clear that we cannot change the way the *pattern table* is formed, but what we can do is change the order of the characters via the *name table*. We construct our *name table* to look like figure 3.

Check out the source INT.GEN for an example on how this is actually done.

Now, the eight neighbouring pixels are still packed within one byte and we cannot do anything about it, but it doesn't matter since we never want to touch anything smaller than eight pixels anyway. After that the bytes below each other are stored sequentially in the memory for each column, which is really nice for us. Not as good as a real linear mode, but this is as close as we get on the MSX.

After we're done with these basics, we can start coding the effect. First, let's precalc. I used MSX-BASIC, I know it's slow, but I've really no clue how the graphic commands work on my Borland Pascal 1.0 or even if there are any. If someone actually does, please drop me a mail. Here's the program: [1]. Just type BASIC from the command prompt, then LOAD
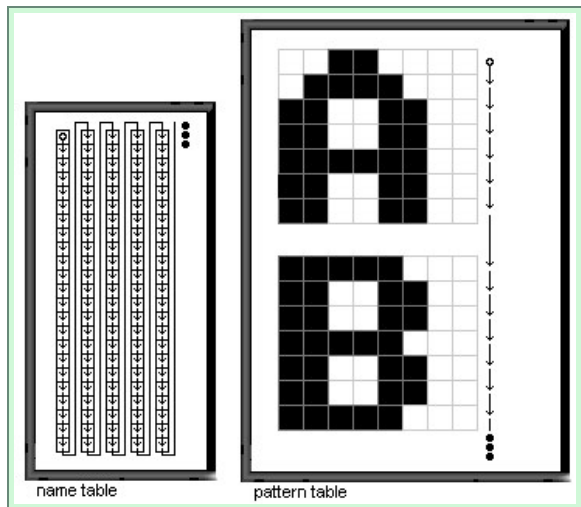

Figure 3: *name table and pattern table in pseudolinear* SCREEN 2

"MAKEINT.BAS" and RUN. The program first draws the circles and then writes the precalced data which will take a while. About the only thing worth of notice is that we write each column twice: this is because it would've been too complicated to check for overflows when actually scrolling and updating two interfering images, so when the display does overflow, it overflows to an other incarnation of the same image.

Now we have the data file `INTER.DAT` and it's time to start writing the real effect. Here's the basic procedure:

1. Load the datafile.
2. Setup screen mode with one *pattern table* and *colour table* for `SCREEN 2`; see the source code.
3. Make an other incarnation of the image from slightly a different position for interfacing. Of course it could, and probably even should, be a completely different image, but I'm going cheap tonight.
4. Setup the *name table* in pseudolinear mode like described above. Use four copies of the same characters.
5. Mix the two images and output.
6. Update scrolls.
7. Go to 5.

Here's the source code, you'll need both these files on your diskette: [2] and [3]. Type `GEN80 INT.GEN` from the command prompt and after that `INT INTER.DAT` to run. And there you have it. Use `SPACE` to quit. For a package of all the files and precompiled executables download this package [4].


Figure 4: *Rotating bar sequence*
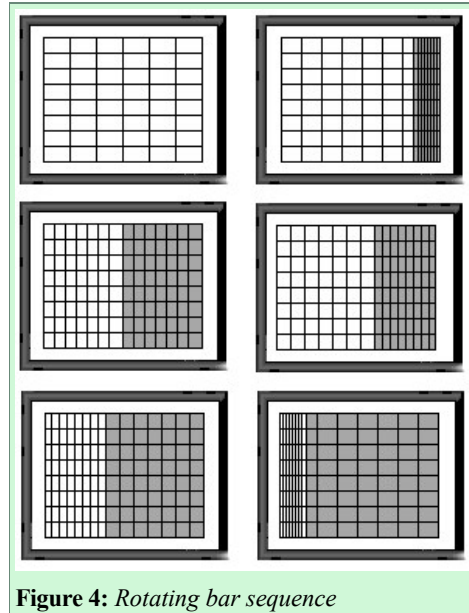
Twisting bars $0.00

Now that we've covered the interference rings, let's get to the twisting bars. Have a look through this sequence of a rotating bar in figure 4. This is just an illustration, in reality we will have a real texture and more frames than just six. The rotation is only pseudo, that is, we only really rotate about the y-axis, the data for different rows remains linear.

Let's have a few restrictions: First, the following face of the bar — marked grey in the picture — should always be the same as the current one, so we'll only need to store the rotation sequence once and then repeat the same set of images for all the four faces. Second, the texture should repeat itself between a constant number of lines, say every eight rows. After that we only need to store the eight rows of the rotation, the rest is always the same as our pseudorotation is vertically linear. This has got nothing to do with speed, it just saves us some valuable memory. If you feel like it, make all the faces look different and have us large a texture you will, the 40 kB or so won't fit that much though.

After this is done with, we'll make the bar twist. The principle is as stated in figure 5. It's really simple: when dumping your bar to the screen, instead of displaying every row using the same amount of rotation, alter it depending on the row. Since we're going to be working with precalced data, this can be done really fast: just display a different row of a precalced animation for each row. The size our effect is determined just as earlier with the interference rings. Let's pick 80×192 and use three copies of the same bar twisting about the screen.

Figure 5: *Principle of twisting bars*
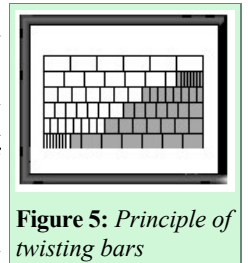
I'd like to go through the dumper inner loop in a bit greater focus since it takes a really clever use of the stack on the Z80. The thing is, if you want to load data sequentially, the stack is by far the fastest way to do it... and you get 16 bits a `POP` so it's faster to load 16 bits sequentially than it is 8 bits! The only drawback is that you'll have disable the interrupts because we're messing with `stack pointer` and a jump to the interrupt handler would absolutely mess everything up. So if you're, for instance, using a music player that updates the notes through v-interrupt, you'll have to divide your dumper to smaller pieces and wait for the retrace in between if the music starts to slow down.

Once again we're operating with the pseudolinear nametable, so our inner loop would look something like this in pseudocode:

Pseudo-listing: `twist.psd`

```
for x:=0 to 9 do
  for y:=0 to 191 do begin
    a:=byte(image[(frame+word(twister[y])) and 63*80+
                  (y and 7)*10+
                  x
                  ]);
    port[$98]:=a;
  end;

frame:=frame+1;
```

Where image is the buffer of prerotated bars and twister has the twisting function that takes the row as a parameter. It's clear to see that our inner loop requires some more precalcs. Here's an improved one:

Pseudo-listing: `twist2.psd`

```
for y:=0 to 191 do
  pre_calc[y]:=twister[y]*80+(y and 7)*10
```

```
   for x:=0 to 9 do begin
     a:=byte(image[word(pre_calc[y])+frame]);
     port[$98]:=a;
   end;
   inc(frame);
end;
frame:=frame-10;
frame:=frame+80;
```

Now it looks real good for us, just two table lookups for 8 pixels. The other 16-bit one looks slow though, but this is where the stack steps in. In Z80 the inner loop looks this simple:

```
; pre_calc already precalced

di            ; no interrupts
ld (spede),sp ; store sp

ld de,(k)     ; k=(frame and 63*80)
ld hl,image
add hl,de
ld d,h
ld e,l        ; de = k+image

ld c,10
_outer:
ld sp,pre_calc
ld b,192
_inner:
pop hl        ; hl = word(pre_calc[y]), y=y+2
add hl,de     ; hl = pre_calc[y]+frame*80+x
ld a,(hl)     ; a=image[hl]
out (#98),a   ; output to screen
djnz _inner

inc de ; x=x+1
dec c
jp nz,_outer

ld sp,(spede)
ei
```

Let's precalc. Since I'm a bit of a good programmer implementing good programming style and since we used BASIC for the previous precalc, we're going for Pascal this time — have I ever mentioned that I truly hate programming theory? — for convenience. Speaking frankly, Pascal is all right for this effect since we won't be needing the graphical commands for anything. Here's the source: [5]. You can find the compiled executable 'pretwist.com' in the package at the end of the paragraph. There isn't really much to explain about the source, just draw the rotating bars, convert the chunky data to planar and output to a file. Note that we output the same buffer twice once more, since we want to prevent overflows. It takes 5 kB's more memory but it's worth the speed.

After that it's just the question of coding the inner loop above and that's about it. Here's the assembly source: [6] and [3]. Once again type GEN80 TW.GEN to compile and then TW TWISTED.DAT to run. Here's a package of all of the above: [7].

Next up
Okay, that's all for now. Be sure the tune in next episode for the following chapter in this ever-mutating demotutorial series: Chunky based effects; how to write just as boring effects as seen on PC-intros and -demos.

Note from the editors:
If you want to see these effects in action, you could download some demos from the Bandwagon page. Antti is too humble to mention this, so we do it instead...

previous:
Big computer meeting 2000

MSX Computer & Club Webmagazine
issue 92, March/April/May 2000

next:
MCCW Contents