

Donkey Kong Anniversary 40 years

End rivets aanpassen: DK rechtop naar beneden vallen en samen met Jumpman en Pauline genieten van verjaardagstaart.

End rivets aanpassen: Happy birthday song afspelen.

In meerdere levels ballonnen laten bewegen en bonuspunten als je een ballon pakt: 1e ballon 100, 2e ballon 300, 3e ballon 500 punten.

Eventueel pie factory omtoveren naar echte taarten en oil can veranderen in een oven?

Geluid aanpassen

Eerst proberen afspelen alternatieve unused Pauline muziekje bij finish van rivets.

Wordt hier gedaan:

```
1913 218A60 LD HL,#608A ; load HL with sound address
1916 360C LD (HL),#0C ; play sound for rivets cleared
1918 3A2962 LD A,(#6229) ; load A with level #
191B 0F RRCA ; roll a right . is this an odd level ?
191C 3802 JR c,#1920 ; Yes, skip next step

191E 3605 LD (HL),#05 ; else play sound for even numbered rivets

1920 23 INC HL ; HL := #608B = sound duration
1921 3603 LD (HL),#03 ; set duration to 3

500: 00 db $00 ; %00000000 0: No music
501: 20 db $20 ; %00100000 1: Music when DK climbs ladder
502: 10 db $10 ; %00010000 2: How high can you get?
503: 05 db $05 ; %00000101 3: Running out of time
504: 06 db $06 ; %00000110 4: Hammer music
505: 12 db $12 ; %00010010 5: Music after beating even-numbered rivet levels
506: 40 db $40 ; %01000000 6: Hammer hit
507: 16 db $16 ; %00010110 7: Music for completing a non-rivet stage
508: 01 db $01 ; %00000001 8: Music for barrel stage
509: 02 db $02 ; %00000010 9: Music for pie factory
50A: 00 db $00 ; %00000000 A: Music (or lack thereof) for elevator stage
50B: 04 db $04 ; %00000100 B: Music for rivet stage
50C: 14 db $14 ; %00010100 C: Music after beating odd-numbered rivet levels
50D: FA db $fa ; %11111010 D: Used when rivet removed
50E: 1E db $1e ; %00011110 E: Music when DK is about to fall in rivet stage
50F: 2D db $2d ; %00101101 F: DK roars
```

De twee muziekjes (oneven en even) zijn #05 en #0C. Dat zijn de entries in de bovenstaande tabel: 505 en 50C.

Deze verwijzen weer naar posities #12 en #14 in de volgende tabel:

```
; Playlist table for $57d songs
; -----
; For instance, $516 will play patterns $14, $83, $fe, etc. until it hits $00.
; FE 85 FE 85 FE 00 plays three gorilla roars
510: 10      db $10          ; How high can you get?
511: 00      db $00
512: 13      db $13          ; Rescued Pauline (even level)
513: 00      db $00
514: 11      db $11          ; Rescued Pauline (odd level)
515: 00      db $00
516: 14      db $14          ; Completed non-rivet stage
517: 83      db $83
518: FE      db $fe
519: 85      db $85
51A: FE      db $fe
51B: 85      db $85
51C: FE      db $fe
51D: 00      db $00
51E: 15      db $15          ; DK about to fall
51F: 00      db $00
```

Dus als je in beide gevallen het unused stukje wilt spelen dan moeten 512 en 514 beiden aangepast worden naar 12:

```
; Pattern $11, Rescued Pauline (odd level)
407: 20 80 DC 98 E0 9A E2 9C 20 90 E4 88 10 E3 90 A4
417: 20 88 DA 8A EA 84 10 E8 8A A4 20 84 E2 82 E2 87
427: 16 E8 88 0A A3 10 E4 82 A0 15 C8 80 2B CA 83 40
437: CB 80 00

; Pattern $12, Rescued Pauline (unused variant)
43A: 10 E8 A4 E8 A4 E6 A2 E6 A2 E4 A0 E4 A0 E2 9C E2
44A: 9C 10 E0 90 A0 E4 88 A0 E4 80 A0 E4 88 A8 20 EA
45A: 96 E8 90 40 F0 80 00

; Pattern $13, Rescued Pauline (even level)
461: 10 E8 A4 E8 A4 E6 A2 E6 A2 E4 A0 E4 A0 E2 9C E2
471: 9C 20 E0 90 10 88 08 A0 A2 20 E4 90 E0 88 20 E2
481: 92 10 8A 08 A2 A4 20 E6 92 E2 8A 20 E8 88 10 82
491: A8 EA 88 A8 E6 82 A4 E8 80 2B 83 40 80 00
```

Deze aanpassing moet doorgevoerd worden in de s_3i_b.bin.

Ja dat gaat goed.

Nu kijken of dat pattern aangepast kan worden. Hoe is zo'n pattern opgebouwd?

Uitleg van de patternopbouw:

```
; Format:  
;   * If the value is $00, end the pattern.  
;   * Else, if bit 7 is clear, the byte is the duration for all following notes until another duration code is read.  
;   * Else, for channel A:  
;     * Bits 0-3 set the note (A, A#, B, C ...)  
;     * Bits 4-5 set the octave (higher values choose lower octaves)  
;     * Bit 6 indicates the next byte is a note to play on channel B.  
;  
; Channel B notes follow the same format, except bits 6-7 are ignored.  
;  
; This format means it is impossible to play a note on channel B without also playing one on channel A,  
; and the channels cannot have separate durations.
```

Notentabel:

0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011
A	A#	B	C	C#	D	D#	E	F	F#	G	G#
0	1	2	3	4	5	6	7	8	9	A	B

Eerste deel van de pattern:

10 E8 A4 E8 A4 E6 A2 E6 A2 E4

Uitschrijven:

10 0001 0000 – duration voor de rest van de notes
E8 1110 1000 – Channel A – Octave 10 – Note 1000 - F
A4 1010 0100 – Channel B – Octave 10 – Note 0100 – C#
E8 1110 1000 – Channel A – Octave 10 – Note 1000 - F
A4 1010 0100 – Channel B – Octave 10 – Note 0100 – C#
E6 1110 0110 – Channel A – Octave 10 – Note 0110 – D#
A2 1010 0010 – Channel B – Octave 10 – Note 0010 - B
E6 1110 0110 – Channel A – Octave 10 – Note 0110 – D#
A2 1010 0010 – Channel B – Octave 10 – Note 0010 - B

Aanpassen bijvoorbeeld E8 naar Octave 00: 11001000 -> C8 en duration langer -> 80:

43A: 40 C8 A4 C8 ...

Ja dat gaat goed.

Andere noten pakken:

43A: 40 C8 A5 C9 A6 ...

Ja dat gaat ook goed.

Eens een octaaf laten oplopen:

43A: 40 C0 81 C2 83 C4 85 C6 87 C8 89 CA 8B

De noten voor Happy Birthday:

Happy Birthday

Traditional

Musical notation for the first line of "Happy Birthday". The key signature is one sharp (F#). The melody starts on F, moves to C7, and then back to F. The lyrics are "Hap - py birth - day to you! Hap - py".

Musical notation for the second line of "Happy Birthday". The key signature is one sharp (F#). The melody continues from the previous line, ending on F. The lyrics are "birth - day to you! Hap - py".

Musical notation for the third line of "Happy Birthday". The key signature changes to two flats (Bb). The melody continues from the previous line, ending on a note with a fermata over it. The lyrics are "birth - day dear (John - ny) Hap - py".

Musical notation for the fourth line of "Happy Birthday". The key signature changes to F/C. The melody continues from the previous line, ending on F. The lyrics are "birth - day to you!".

Zonder timing:

C – C – D – C – F – E → A3 – A3 – A6 – A3 – A9 – A8

C – C – D – C – G – F → A3 – A3 – A6 – A3 – AB – A9

C – C – C+ – A – F – E – D → A3 – A3 – B3 – B0 – A9 – A8 – A6

A# – A# – A – F – G – F → B1 – B1 – B0 – A9 – AB – A9

Raar: de warden proefondervindelijk vastgesteld. Maar komt niet overeen met notentabel.

Tempo eventjes op 18 zetten. De groene half (0C) en de gele dubbel (30).

OB - A3 - A3 - 16 - A6 - A3 - A9 - 2C - A8 -
 OB - A3 - A3 - 16 - A6 - A3 - AB - 2C - A9 -
 OB - A3 - A3 - 16 - B3 - B0 - OB - A9 - A9 - 16 - A8 - 2C - A6 -
 OB - B1 - B1 - 16 - B0 - A9 - AB - 2C - A9

Deze eerste versie klinkt goed. En juiste tempo.

Nu versie v0.01 opgeslagen.

Nu kijken of je er ook B channel noten tussen kan krijgen. Maar welke dan?

Alternatieve muziek:

HAPPY BIRTHDAY TO YOU!

Words and Music by
MILDRED J. HILL and
PATTY S. HILL.

Brightly

The musical score consists of three parts of sheet music. Part 1 starts with a treble clef, a key signature of one sharp (F#), and a common time signature. It features a piano part with bass notes and a vocal part with lyrics. The lyrics are: "Hap-py Birth-day to you, Hap-py Birth-day to you, Hap-py Birth-day, dear *———, Hap-py Birth-day to you!" The piano part includes chords G, C, Am, G, D7, and G. Part 2 continues with the same key signature and time signature. The lyrics are: "Hap-py Birth-day to you, Hap-py Birth-day to you, Hap-py Birth-day, dear *———, Hap-py Birth-day to you!" The piano part includes chords G, D7, G, C, Cm6, G, D7, and G. Part 3 concludes with the same key signature and time signature. The lyrics are: "Hap-py Birth-day to you, Hap-py Birth-day to you, Hap-py Birth-day, dear *———, Hap-py Birth-day to you!" The piano part includes chords G, C, Cm6, G, D7, and G.

Deze klinkt wel goed en is maximaal twee noten.

Ook weer op gaan bouwen:

OB - A6 - A6 - 16 - A8 - A6 - AB - 2C - A9

OB - A6 - A6 - 16 - E8 - A2 - E6 - A2 - EB - A2 - 2C - EA - A3

Eerste stuk is goed !

Volgende:

OB - A6 - A6 - 16 - A8 - A6 - B0 - 2C - AB

OB - A6 - A6 - 16 - E8 - A3 - E6 - A3 - F0 - A3 - 2C - EB - A2

Tweede stuk is goed !

Volgende:

OB - A6 - A6 - 16 - B6 - B2 - AB - A9 - A8

OB - A6 - A6 - 16 - F6 - AB - F2 - A6 - EB - A6 - E9 - A6 - E8 - A3

Derde stuk is goed !

Volgende:

OB - B3 - B3 - 16 - B2 - AB - B1 - 58 - AB

OB - F3 - A7 - F3 - A7 - 16 - F2 - A6 - EB - A6 - F0 - A3 - 58 - EB - A2

Vierde stuk is ook goed !

Nu versie v0.02 opgeslagen.

Nu de High Score tekst en het titelscherm aanpassen.

CREDIT opschuiven:

#36C2: 3F 9F 75 13 22 15 14 19 24 3F

Aanpassen verwijzing:

3655: C2 36 5 ; #36C2 "CREDIT"

en

HIGH SCORE aanpassen naar = 40 YEARS =

#36B2: 80 76 34 10 04 00 10 29 15 11 22 23 10 34 3F

Het titelscherm aanpassen

De KONG letters 1 positie naar beneden verschuiven:

3D59: 05 30 77 05 10 77 02 F1 76 02 D0 76 02 D3 76 ; K

3D68: 05 90 76 05 70 76 01 50 76 01 54 76 05 30 76 ; O

3D77: 05 F0 75 02 D1 75 02 B2 75 05 90 75 ; N

```
3D83: 03 51 75 05 30 75 01 10 75 01 14 75 ; G (part 1)
3D8F: 01 F0 74 01 F2 74 01 F4 74 02 D2 74 ; G (part 2)
3D9B: 00 ; end code
```

En het TM symbool weghalen.

```
081C CD243F CALL #3F24 ; draw TM logo onscreen [patch? orig japanese had 3 NOPs here]
```

Hiermee wordt het stukje extra code dat het TM-symbool tekent ook vrijgemaakt:

#3F24 t/m #3F2F.

Dus op #081C weer 3 NOP's maken.

Dan kan het stuk van #3F00 t/m #3F23 ook vrijgemaakt worden en gebruikt worden voor de teksten in het titelscherm.

```
3687: 00 3F 1E ; #3F00 "(C) 1981"
3689: 09 3F 1F ; #3F09 "NINTENDO OF AMERICA"
```

In #3F00 de copyright notice maken:

© 1981-2021 NINTENDO

```
5C 77 49 4A 10 01 09 08 01 2C 02 00 02 01 10 1E 19 1E 24 15 1E 14 1F 3F
```

En de plek van de checksum ook aanpassen: INTEND: start op 3F10:

```
2441 21103F LD HL,#3F10 ; load HL with ROM area that has NINTENDO written
```

En dan in #3F18 de titel maken:

ANNIVERSARY EDITION

```
2E 77 11 1E 1E 19 26 15 22 23 11 22 29 10 15 14 19 24 19 1F 1E 3F
```

En de verwijzing ernaar toe aanpassen:

3689: 18 3F

Nu nog de Donkey Kong een regel naar beneden verplaatsen.

Wordt hier gedaan:

```

0820 210869 LD HL,#6908 ; load HL with start of kong sprite X pos
0823 0E44 LD C,#44 ; load C with offset to add X
0825 FF RST #38 ; draw kong in new position
0826 210B69 LD HL,#690B ; load HL with start of kong sprite Y pos
0829 0E78 LD C,#78 ; load C with offset to add Y
082B FF RST #38 ; draw kong
082C C9 RET ; return

```

Kan door aanpassen:

0829: 0E80 LD C,#80 ; load C with offset to add Y

Sprites met goudgele 40 over Donkey Kong logo heen:

Invoegen op de plek van de verwijderde aanroep van het tekenen van het TM logo/

We hadden de #081C drie NOP's gemaakt (niet meer TM logo tekenen). Daar kan dan mooi een aanroep naar een stuk additionele code:

081C: C32E3F JP #3F2E ; jump to additional code to display version

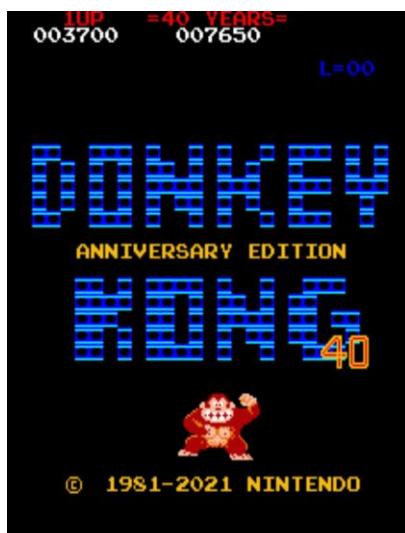
en

```

3F2E 213C3F LD HL,#3F3C ; source - has the information about the barrels
3F31 11A869 LD DE,#69A8 ; destination = sprites
3F34 011000 LD BC,#0010 ; counter is #10
3F37 EDB0 LDIR ; draws the barrels pile
;
3F39 C31F08 JP #081F ; jump back
;
3F3C C0 66 01 A4 ; first sprite - 4
3F40 CF 67 01 A4 ; second sprite - 0
3F44 50 56 0B C8 ; third barrel
3F48 B0 56 0B C8 ; fourth barrel

```

Het titelscherm ziet er nu zo uit:



Nu versie v0.03 opgeslagen.

De bonus items veranderen in verjaardags items.

Gedaan. Het juiste kleurenpalette moet nog wel gekozen worden.

De definitie van de bonus items zit hier:

; bonus items on conveyors

```
3E3C 53 73 0A A0          ; position of hat on pies is 53,A0  
3E40 8B 74 0A F0          ; position of purse on pies is 8B,F0  
3E44 DB 75 0A A0          ; umbrella on the pies is at DB,A0
```

; bonus items for elevators

```
3E48 5B 73 0A C8          ; hat at 5B,C8  
3E4C E3 74 0A 60          ; purse at E3,60  
3E50 1B 75 0A 80          ; umbrella on elevator is 80,1B
```

; bonus items for rivets

```
3E54 DB 73 0A C8          ; hat on rivets at DB,C8  
3E58 93 74 0A F0          ; purse on rivets at 93,F0  
3E5C 33 75 0A 50          ; umbrella on rivets at 33,50
```

De kleur is de derde waarde.

Kleur 01 is aardig. Alleen cadeau kleuren wisselen.

Kleur 02 is leuk voor de feestmuts en eventueel het cadeau.

Kleur 08 is aardig. Alleen feestmuts kleuren wisselen.

Kleur 00 is aardig.

Alle drie de items dezelfde kleurencombinatie geven is wel stijlvol.

Kleur 0A blojven gebruiken en cadeau anders maken.

Nu versie v0.04 opgeslagen.

De High Score aanpassen naar 1981-2021

#36B2: 80 76 01 09 08 01 35 02 00 02 01 3F

Staat niet zo leuk.

De High Score aanpassen naar ANNIVERSARY

#36B2: 80 76 11 1E 1E 19 26 15 22 23 11 22 29 3F

Staat wel leuk. Zo houden.

Nu versie v0.05 opgeslagen.

Nu kijken of de end sequence waarin Donkey Kong valt aangepast kan worden.

Dit begint bij #17B7.

Eerst kijken of Donkey Kong rechtop kan vallen. Het ondersteboven draaien van Donkey Kong wordt hier gedaan:

```
1870 211F3A LD HL,#3A1F      ; start of table data for kong upside down
1873 CD4E00 CALL #004E       ; update kong's sprites
1876 3E03 LD A,#03          ; A := 3
1878 328460 LD (#6084),A    ; play falling sound
187B 218863 LD HL,#6388     ; load HL with end of level counter
187E 34 INC (HL)           ; increase
187F C9 RET                ; return
```

Wellicht op te lossen door de update van de sprites van Donkey Kong niet te doen:

1873 000000 NOP, NOP, NOP

Hij valt nu inderdaad rechtstandig naar beneden. Maar schiet te ver door. Hij moet eigenlijk precies op de ingestorte girders terecht komen.

Dit lijkt hier te gebeuren:

```
1880 210B69 LD HL,#690B      ; load HL with kong start sprite
1883 0E01 LD C,#01          ; load C with 1 pixel to move
1885 FF RST #38            ; move kong
1886 3A1B69 LD A,(#691B)    ; load A with ???
1889 FED0 CP #D0            ; == #D0 ?
188B C0 RET NZ              ; no, return
```

De #D0 in een andere waarde veranderen. Moet ongeveer 2 girders omhoog.

1889 FEC0 CP #C0 ; == #C0 ?

Ja dat is nu goed.

Alleen de animatie van Kong is nu nog vreemd:

Mond wordt een vreemde tandenbende: zorgen standaard mond, maar dan lachend.

Dit wordt hier gezet:

```
188C 3E20 LD A,#20          ; A := #20
188E 321969 LD (#6919),A    ; store into kong's face sprite - kong is now bigmouthed with crazy eyes
```

De sprite #20 aangepast naar het normale hoofd en de mond lachend gemaakt. Ook de sprites @21 en #22 aangepast omdat die voor het rollen van de ogen zorgde. Nu pretoogjes die naar links en naar rechts kijken.

Ook nog de animatie van de sterretjes om het hoofd. Die kunnen verwijderd worden. Dat gebeurd hier:

```
1891 21246A LD HL, #6A24 ; load HL with sprite address used for kong's aching head lines
1894 367F LD (HL), #7F ; set sprite X value
1896 2C INC L ; next
1897 3639 LD (HL), #39 ; set sprite color
1899 2C INC L ; next
189A 3601 LD (HL), #01 ; set sprite value
189C 2C INC L ; next
189D 36D8 LD (HL), #D8 ; set sprite Y value
```

Deze kunnen NOP gemaakt worden. Eventueel later deze ruimte gebruiken om bijvoorbeeld te springen naar routine die een taart tekent.

Is goed nu.

Nu versie v0.06 opgeslagen.

Nu de sprites van Jumpman en Pauline naar beneden verplaatsen.

Pauline gebeurt hier:

```
18AB 110400 LD DE, #0004 ; load counters
18AE 012802 LD BC, #0228 ; load counters
18B1 210369 LD HL, #6903 ; set sprite girl table data Y position
18B4 CD3D00 CALL #003D ; move the girl down
```

Dit is een move down: aanpassen Y waarde, X-waarde blijft hetzelfde. Daar hebben we niet zoveel aan. Moeten overigens ook veel verder naar beneden. Wellicht handiger om dat gewoon opnieuw te zetten.

#6900 is X-positie van Pauline

#6903 is Y-positie van Pauline

Dan dus doen:

```
18AB 210069 LD HL, #6900 ; load HL with X-position Pauline sprite
18AE 3E60 LD A, #60 ; new X-position Pauline
18B0 77 LD (HL), A ; set X-position
18B1 210369 LD HL, #6903 ; load HL with Y-position Pauline sprite
18B4 3EC0 LD A, #C0 ; new Y-position Pauline
18B6 77 LD (HL), A ; set Y-position
```

Dat zou precies moeten passen. Klopt, maar het werkt niet goed omdat Pauline uit twee sprites bestaat en ze allebei aangepast moeten worden. Dus veranderen:

18AB 3E60	LD	A,#60	; new X-position Pauline
18AD 210069	LD	HL, #6900	; load HL with X-position Pauline sprite 1
18B0 77	LD	(HL),A	; set X-position Pauline sprite 1
18B1 210469	LD	HL, #6904	; load HL with X-position Pauline sprite 2
18B4 C34C3F	JP	#3F4C	; jump to additional code – jump to label AAAA

en

3F4C 77	LD	(HL),A	; set X-position Pauline sprite 2 – label AAAA
;			
3F4D 210369	LD	HL, #6903	; load HL with Y-position Pauline sprite 1
3F50 3EC3	LD	A,#C3	; new Y-position Pauline sprite 1
3F52 77	LD	(HL),A	; set Y-position Pauline sprite 1
;			
3F53 210769	LD	HL, #6907	; load HL with Y-position Pauline sprite 2
3F56 3ED3	LD	A,#D3	; new Y-position Pauline sprite 2
3F58 77	LD	(HL),A	; set Y-position Pauline sprite 2
;			
3F59 C3B718	JP	#18B7	; jump back

Gaat nog niet helemaal goed. Als jumpman aan linkerkant, dan staat de bovenkant van Pauline de verkeerde kant op te kijken. Dus de sprite aanpassen naar de variant die naar rechts kijkt. Maar die is er niet. De sprite wordt gedraaid op een of andere manier. Hoe werkt dat? Kijken bij code wanneer jumpman van helft wisselt in het rivets screen. Ja alle sprite graphics bestaan ook in de gespiegelde variant van #80. Dus sprite graphic wordt dan: #80? Klopt niet moet #10 zijn.

3F4C 77	LD	(HL),A	; set X-position Pauline sprite 2 – label AAAA
;			
3F4D 210369	LD	HL, #6903	; load HL with Y-position Pauline sprite 1
3F50 3EC3	LD	A,#C3	; new Y-position Pauline sprite 1
3F52 77	LD	(HL),A	; set Y-position Pauline sprite 1
;			
3F53 210769	LD	HL, #6907	; load HL with Y-position Pauline sprite 2
3F56 3ED3	LD	A,#D3	; new Y-position Pauline sprite 2
3F58 77	LD	(HL),A	; set Y-position Pauline sprite 2
;			
3F59 210169	LD	HL, #6901	; load HL with sprite graphic Pauline sprite 1
3F5C 3E90	LD	A,#10	; new sprite graphic for Pauline sprite 1
3F5E 77	LD	(HL),A	; set Y-position Pauline sprite 1
;			
3F5F C3B718	JP	#18B7	; jump back

Werkt goed.

Jumpman ook verplaatsen.

Dat gebeurt hier:

```
18EE 3E50 LD A,#50 ; A := #50
18F0 324F69 LD (#694F),A ; store into mario sprite Y value
18F3 3E00 LD A,#00 ; A := 0
18F5 324D69 LD (#694D),A ; store into mario sprite value
18F8 3E9F LD A,#9F ; A := #9F
18FA 324C69 LD (#694C),A ; set mario sprite X value at #9F
18FD 3A0362 LD A,(#6203) ; load A with mario X position
1900 FE80 CP #80 ; < 80 ?
1902 D20F19 JP NC,#190F ; yes, skip next 4 steps

1905 3E80 LD A,#80 ; A := #80
1907 324D69 LD (#694D),A ; store into mario sprite value
190A 3E5F LD A,#5F ; A := #5F
190C 324C69 LD (#694C),A ; store into mario sprite X value
```

Jumpman bestaat maar uit 1 sprite. Dus hier de juiste X en Y invullen. Wordt altijd al naar links kijkend neergezet en dat moet nu ook:

```
18EE 3ED0 LD A,#D0 ; A := D3, Y-position jumpman
18F8 3E9B LD A,#9B ; A := 9B, X-position jumpman
1905 3E00 LD A,#00 ; A:= 00, Y-position jumpman
190A 3E9B LD A,#9B ; A := 9B, X-position jumpman
```

De heart sprite nu niet nodig. Die wordt hier getekend:

```
1923 21236A LD HL,#6A23 ; load HL with heart sprite
1926 3640 LD (HL),#40 ; set heart sprite Y position
1928 2B DEC HL ; decrement HL
1929 3609 LD (HL),#09 ; set heart sprite color
192B 2B DEC HL ; decrement HL
192C 3676 LD (HL),#76 ; set heart sprite
192E 2B DEC HL ; decrement HL
192F 368F LD (HL),#8F ; set heart sprite X position
```

Deze allemaal NOP maken.

Daarnaast ook nog een stukje hier:

```

1934 fe80 CP #80 ; is mario on the left side of the screen?
1936 d0 RET nc ; yes, return

1937 3E6f LD A,#6F ; else A := #6F
1939 32206A LD (#6A20),A ; store A into heart sprite X position

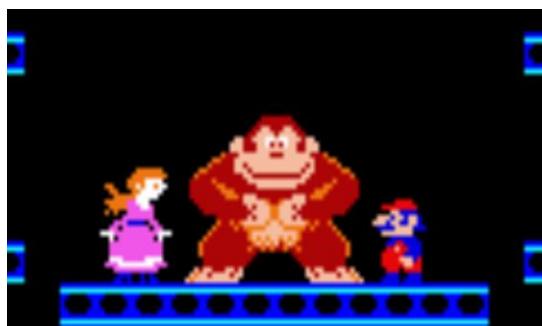
```

Deze aanpassing van de heart sprite ook niet doen:

1937-193B NOP's maken.

Gaat goed.

Ziet er nu zo uit:



Nu versie v0.07 opgeslagen.

Heart sprite weer terug:

```

1923 21236A LD HL,#6A23 ; load HL with heart sprite
1926 3640 LD (HL),#40 ; set heart sprite Y position
1928 2B DEC HL ; decrement HL
1929 3609 LD (HL),#09 ; set heart sprite color
192B 2B DEC HL ; decrement HL
192C 3676 LD (HL),#76 ; set heart sprite
192E 2B DEC HL ; decrement HL
192f 368f LD (HL),#8f ; set heart sprite X position

```

Maar dan vullen met sprite met feesthoedje: #68 op positie (80, B4) en kleur 0A.

Ziet er nu zo uit:



Nu versie v0.08 opgeslagen.

Jumpman wordt later dan Pauline naast Kong geplaatst. Liever dat dat tegelijkertijd gaat. Het verplaatsen van jumpman toevoegen aan de additionele code voor het verplaatsen van Pauline.

3F5F C3B718 JP #18B7 ; jump back

vervangen door:

```
3F5F 3ED0 LD A,#D0 ; A := #D0
3F61 324F69 LD (#694F),A ; store into mario sprite Y value
3F64 3E00 LD A,#00 ; A := 0
3F66 324D69 LD (#694D),A ; store into mario sprite value
3F69 3E9B LD A,#9B ; A := #9B
3F6B 324C69 LD (#694C),A ; set mario sprite X value at #9F
;
3F6E C3B718 JP #18B7 ; jump back
```

Dat gaat goed. Dan kan het originele stukje die dat doet leeggemaakt worden: #18EE-190E. Daar kan later eventueel code voor het tekenen van de 40 tussengevoegd worden.

Gaat goed.

Nu versie v0.09 opgeslagen.

Kijken of we het programma zo kunnen maken dat bij starten als eerste het titelscherm getoond wordt. Niet in de volgorde aanpassen. Dat is eerder geprobeerd, maar dat ging niet goed. Maar de fase wordt aangeduid met een variabele, misschien deze bij het starten anders initialiseren.

Wordt bepaald door de GameMode2: die wordt hier gezet:

```
01E2 3E01 LD A,#01 ; A := 1
01E4 32827D LD (REG_FLIPSCREEN),A ; store into flip screen setting
01E7 320560 LD (GameMode1),A ; store into game mode 1
01EA 322762 LD (#6227),A ; initialize screen to 1 (girders)
01ED AF XOR A ; A := 0
01EE 320A60 LD (GameMode2),A ; store into game mode 2
```

En hier is te zien dat introscherm overeenkomt met GameMode2 = 06:

```

0746 7E      LD      A,(HL)      ; else load A with game mode2
0747 EF      RST     #28       ; jump based on A

0748 79 07      0      ; #0779      ; clear screen, set color palettes, draw attract mode text and high score table,
                           ; [continued] increase game mode2, clear sprites, ; draw "1UP" on screen , draws number of coins needed for play
074A 63 07      1      ; #0763      ;
074C 3C 12      2      ; #123C      ; set initial mario sprite position and draw remaining lives and level
074E 77 19      3      ; #1977      ; set artificial input for demo play [change to #197A to enable playing in demo part 1/2]
0750 7C 12      4      ; #127C      ; handle mario dying animations
0752 C3 07      5      ; #07C3      ; clears the screen and sprites and increase game mode2
0754 CB 07      6      ; #07CB      ; handle intro splash screen ?
0756 4B 08      7      ; #084B      ; counts down a timer then resets game mode2 to 0

```

Eventueel aanpassen:

```

01ED C3983F    JP      #3F98      ; jump to additional code – jump to label AAAA
01F0 00          NOP

```

en

```

3F98 3E06      LD      A,#06      ; A := 6 – label AAAA
3F9A 320A60    LD      (GameMode2),A ; store into game mode 2
3F9D C3F101    JP      #01F1      ; jump back

```

Ja dat werkt goed. Leuk !

Nu versie v0.10 opgeslagen.

De bonus items veranderen in cadeau's. Dus thema: cadeau's en ballonnen verzamelen.

Cadeau sprite en ballon sprite aangemaakt. Nu kijken naar het meest geschikte kleurenpalet.

Gedaan. Het juiste kleurenpalette moet nog wel gekozen worden.

De definitie van de bonus items zit hier:

```

; bonus items on conveyors

3E3C 53 73 0A A0      ; position of hat on pies is 53,A0
3E40 8B 74 0A F0      ; position of purse on pies is 8B,F0
3E44 DB 75 0A A0      ; umbrella on the pies is at DB,A0

; bonus items for elevators

3E48 5B 73 0A C8      ; hat at 5B,C8
3E4C E3 74 0A 60      ; purse at E3,60
3E50 1B 75 0A 80      ; umbrella on elevator is 80,1B

; bonus items for rivets

3E54 DB 73 0A C8      ; hat on rivets at DB,C8
3E58 93 74 0A F0      ; purse on rivets at 93,F0
3E5C 33 75 0A 50      ; umbrella on rivets at 33,50

```

De kleur is de derde waarde.

~~Kleur 00: 73 is goed: paars met rode strik – in rivets OK – in pies NOK~~

~~Kleur 01: 75 is goed: rood met gele strik – in rivets NOK~~

~~Kleur 02: 74 is goed; blauw met rode strik – in rivets OK – in pies OK – in springs OK~~

~~Kleur 03: 73 is goed: blauw met donker gele strik in rivets NOK~~

~~Kleur 07: 75 is goed: roze met rode strik – in rivets OK – in pies OK – in springs OK~~

~~Kleur 07: 73 is goed: donkerrood met witte strik – in rivets OK – in pies ?OK – in springs OK~~

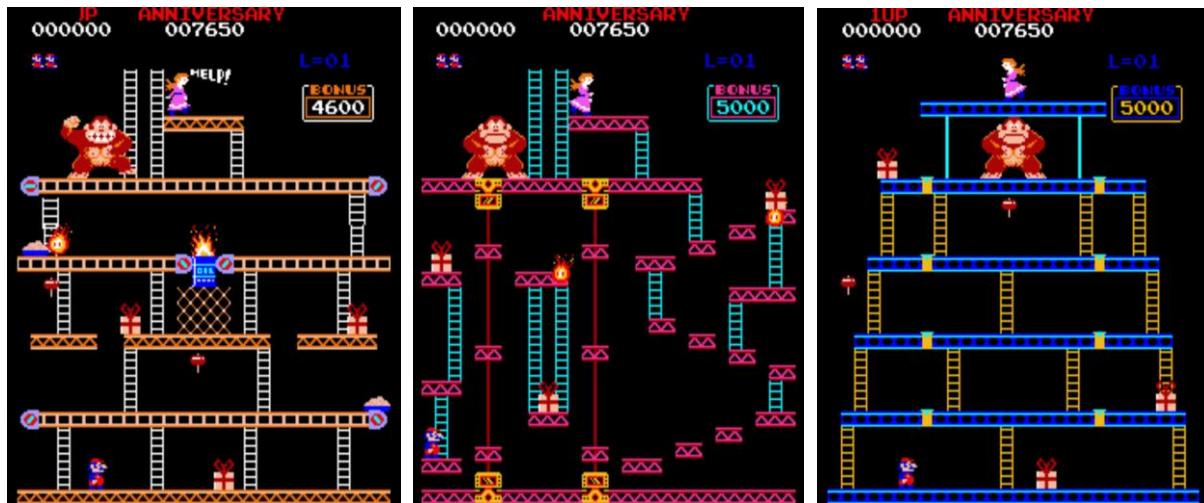
~~Kleur 08: 75 is goed: creme met donkerrode strik – in rivets NOK~~

~~Kleur 0B: 75 is goed: oranje met blauwe strik – in rivets OK – in pies OK – in pies NOK~~

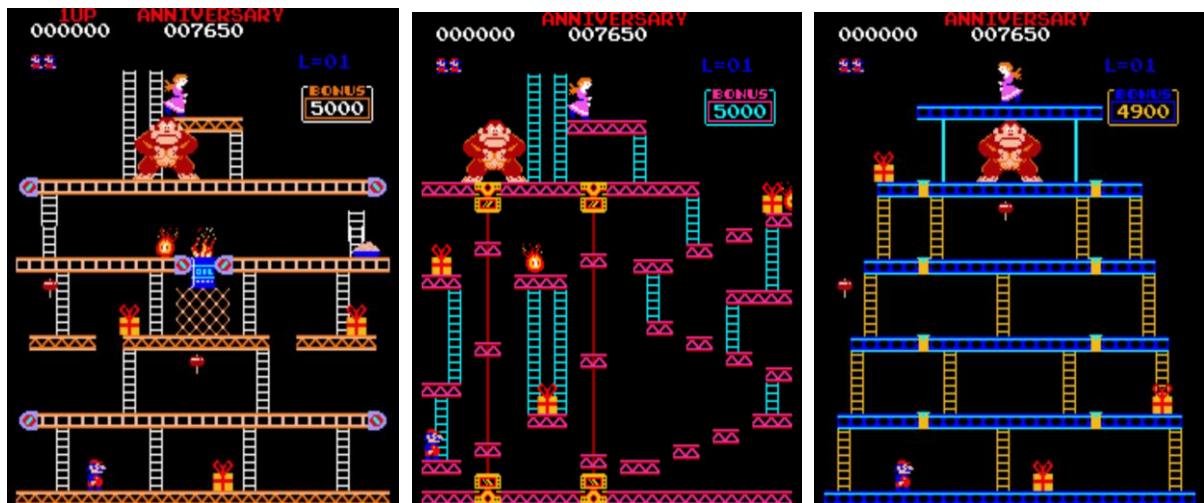
~~Kleur 0D: 75 is goed: knalgeel met rode strik – in rivets OK – in pies OK – in springs OK~~

Volgorde keuze: 07/75, 0D/75 of 02/74. De laatste valt toch nog af: niet mooi met springs.

07/75:



0D/75:



Maakt niet zo heel veel verschil. Gaan voor de primaire keur i.p.v. creme: dus voor 0D/75.

[Nu versie v0.11 opgeslagen.](#)

Nu kijken drie verschillende cadeau's (qua vorm). Ja een smallere en een bredere ervan gemaakt:



[Nu versie v0.12 opgeslagen.](#)

Pauline een cadeau laten geven. Eventueel nog de 40 weergeven. En eventueel vuurwerk sparkles weergeven. Eerst daarvoor kijken naar hoe de ogen bewogen worden. Daar is een animatie loop voor nodig. Daar dan deze zaken aan toevoegen. Tekenen van de additionele sprites kan afgekeken worden van Barrelopalooza waar de Pauline items ook getekend worden in een end-sequence. Dat werd zo gedaan:

```
2276 21483E LD HL,#3E48 ; load HL with table data for bonus items on elevators
2279 110C6A LD DE,#6A0C ; load DE with sprite destination
227C 010C00 LD BC,#000C ; 3 items x 4 bytes = 12 bytes (#0C)
227F EDB0 LDIR ; draw bonus item sprites
```

Eigenlijk gewoon het tekenen van de bonus items (zijn er drie) doen. Maar dan nu aanpassen met een eigen data adres. En misschien wanneer meer zaken tekenen, dan de counter ook ophogen. Eerst vaststellen welke sprites gebruiken: bijvoorbeeld barrels en dan vullen met default en plaatsen. Daarna de juiste sprites, kleuren en plekken invullen.

Eigenlijki tegelijk met het feesthoedje doen. Dat is echter een stuk code voor het weergeven van de heart sprite. Wellicht deze code vervangen door de bovenstaande code en dan het hoedje meenemen in de data.

Heart sprite tekenen wordt hier gedaan:

```
1923 21236A LD HL,#6A23 ; load HL with heart sprite
1926 3640 LD (HL),#40 ; set heart sprite Y position
1928 2B DEC HL ; decrement HL
1929 3609 LD (HL),#09 ; set heart sprite color
192B 2B DEC HL ; decrement HL
192C 3676 LD (HL),#76 ; set heart sprite
192E 2B DEC HL ; decrement HL
192f 368f LD (HL),#8f ; set heart sprite X position
```

Vervangen door code voor het tekenen van meerdere sprites:

```
1923 21713F LD HL,#3F71 ; load HL with table data for end sequence sprites
1926 118069 LD DE,#6980 ; load DE with sprite destination
1929 011400 LD BC,#0014 ; 5 sprites x 4 bytes = 20 bytes (#14)
192C EDB0 LDIR ; draw sprites
192E-1930 3x NOP
```

De barrel sprites gebruiken: deze sprites starten bij #6980.

We willen sprites gebruiken: feesthoed, pie, kaarsjes, cadeau en ballon: dus 5x4 = 20 bytes (#14).

Beginnen met initiële data:

```
3F71: 80 68 0A B4 ; sprite data for party hat
3F75: 40 4B 08 80 ; sprite data for pie
3F79: 60 4B 08 80 ; sprite data for candles
3F7D: 80 73 0D 80 ; sprite data for present
3F81: A0 69 0D 80 ; sprite data for balloon
```

De sprites zijn heel even te zien. Worden weer weggehaald. Daarom even debuggen hoe dat komt. Watchpoint zetten op #6980 en monitoren 6980 t/m 6994. Blijkbaar wordt de animatie gedaan door steeds alle sprites te wissen en opnieuw te tekenen (regel #30E5). De 5 sprites van de barrels daarvan uitsluiten.

```

; arrive here from #1615 when rivets cleared
; clears all sprites for firefoxes, hammers and bonus items

30BD 215069 LD HL,#6950 ; load HL with start of hammers
30C0 0602 LD B,#02 ; B := 2
30C2 CDE430 CALL #30E4 ; clear hammers ?
30C5 2E80 LD L,#80 ; L := #80
30C7 060A LD B,#0A ; B := #A
30C9 CDE430 CALL #30E4 ; clear barrels ?
30CC 2EB8 LD L,#B8 ; L := #B8
30CE 060B LD B,#0B ; B := #B
30D0 CDE430 CALL #30E4 ; clear firefoxes ?
30D3 210C6A LD HL,#6A0C ; load HL with start of bonus items
30D6 0605 LD B,#05 ; B := 5
30D8 C3E430 JP #30E4 ; clear bonus items

```

Komt misschien door #30C5 t/m #30CB? Testen door #30C9 aan te passen naar 3xNOP. Ja dan blijven ze wel staan. Alleen feesthoed, cadeau en ballon zijn zichtbaar. De pie en de kaarsjes niet. Kan kloppen: sprite nummer moet 4B zijn. Gaat dan wel goed.

Permanently adjust:

```
30C9 C3853F JP #3F85 ; jump to additional code – jump to label AAAA
```

and

```

3F85 3A2762 LD A,(#6227) ; load A with screen nr – label AAAA
3F88 FE04 CP #04 ; is this rivets?
3F8A 2803 JR Z,#BBBB ; skip clearing barrels – jump to label BBBB
3F8C CDE430 CALL #30E4 ; clear barrels
3F8F C3CC30 JP #30CC ; jump back – label BBBB

```

That goes well now. Try with barrels screen or if that also goes well (barrels disappear at the end of the screen). That also goes well.

Now the sprites are positioned:

```

3F71: 80 68 0A B4 ; sprite data for party hat
3F75: 40 4B 08 80 ; sprite data for pie
3F79: 60 4B 08 80 ; sprite data for candles
3F7D: 6A 6A 0D C8 ; sprite data for small present
3F81: 98 69 00 BD ; sprite data for red balloon

```

The pies are still not done. Use these for the numbers 40.

Nu versie v0.13 opgeslagen.

Display the numbers 40 and position them:

```
3F71: 80 68 0A B4 ; sprite data for party hat  
3F75: 78 66 01 98 ; sprite data for 4  
3F79: 88 67 01 98 ; sprite data for 0  
3F7D: 6A 6A 0D C8 ; sprite data for small present  
3F81: 98 69 00 BD ; sprite data for red balloon
```

Ziet er nu goed uit:



Nu versie v0.14 opgeslagen.

Kijken of vuurwerk kan? In de originele worden de ogen gedraaid en worden er een soort vonkjes omheen weergegeven. Die zijn uitgezet? Dat weer aanzetten met sprites van vuurwerk?

De animatie van de sterretjes om het hoofd. Dat gebeurt hier:

```
1891 21246A LD HL,#6A24 ; load HL with sprite address used for kong's aching head lines  
1894 367F LD (HL),#7F ; set sprite X value  
1896 2C INC L ; next  
1897 3639 LD (HL),#39 ; set sprite color  
1899 2C INC L ; next  
189A 3601 LD (HL),#01 ; set sprite value  
189C 2C INC L ; next  
189D 36D8 LD (HL),#D8 ; set sprite Y value
```

Deze zijn NOP gemaakt. Wordt denk ik maar één keer gedaan. Dus hoe krijg je dan de animatie (andere kleur, of andere positie). Even onderzoeken waar de ogen bewogen worden.

Het rollen van de ogen wordt hier gedaan:

```

18D8 211969 LD HL,#6919 ; load HL with ???
18DB 46 LD B,(HL) ; load B with this value
18DC CBA8 RES 5,B ; clear bit 5 of B
18DE AF XOR A ; A := 0
18DF CD0930 CALL #3009 ; ???
18E2 F620 OR #20 ; turn on bit 5
18E4 77 LD (HL),A ; store result

```

Getest door #18E4 te vervangen door een NOP. Dan geen rollende ogen meer.

Dus dit stuk code wordt meerdere malen doorlopen. Er is verderop een stuk vrijgemaakt: #18EE t/m #190E.

Dat was het stuk voor het verplaatsen van jumpman, maar wordt niet meer gebruikt. Gebruiken om de positie van het vuurwerk te varieren.

Eerst vuurwerk zichtbaar maken:

```

1891 21246A LD HL,#6A24 ; load HL with sprite address used for sparkle
1894 3650 LD (HL),#50 ; set sprite X value
1896 2C INC L ; next
1897 3663 LD (HL),#63 ; set sprite value
1899 2C INC L ; next
189A 3601 LD (HL),#01 ; set sprite color
189C 2C INC L ; next
189D 3698 LD (HL),#98 ; set sprite Y value

```

De grap is echter dat deze al beweegt. Dat wordt hier gedaan:

```

18D1 21256A LD HL,#6A25 ; load HL with ???
18D4 7E LD A,(HL) ; get value
18D5 EE80 XOR #80 ; toggle bit 7
18D7 77 LD (HL),A ; store result

```

Maar als je daar de X positie van maakt dan wordt hij verplaatst. Testen. Ja dat werkt ! Dan de juiste toggle doen: XOR met E0.

Wordt dan:

```

18D1 21256A LD HL,#6A25 ; load HL with X-position of sparkle
18D4 7E LD A,(HL) ; get value
18D5 EEE0 XOR #E0 ; toggle bit 7,6 and 5
18D7 77 LD (HL),A ; store result

```

Het verspringen eindigt echter iets eerder en dan blijft de sparkle sprite rechts staan. Kan dat voorkomen worden?

Het verspringen eindigt wanneer de bonus time in de score verwerkt wordt. Dat is dus in het volgende stukje in de end sequence.

Daar dan dus iets tussenvoegen. Daartoe het vrije stuk tussen #18EE t/m #190E gebruiken. Dat kan door op #18EE een onconditionele jump te doen naar #190F. Dan wordt het stuk overgeslagen en kun je daar iets anders tussen plaatsen.

```
18EE C30F19 JP      #190F      ; skip free code, so free space can be reused
```

en

```
193D C3F118 JP      #18F1      ; jump to additonal code – jump to label AAAA
```

en

```
18F1 21256A LD      HL,#6A25    ; load HL with sprite address used for sparkle
```

```
18F4 3664   LD      (HL),#64    ; set sprite value to empty sprite
```

```
18F6 2A2A62   LD     HL,(#622A)  ; load HL with contents of #622A.
```

```
18F9 C34019 JP      #1940      ; jump back
```

Ja, dat werkt nu goed !

Nu versie v0.15 opgeslagen.

Tabel met mogelijk aanpassingen voor deze anniversary edition:

- Zwevende ballon die extra 800 punten geeft wanneer gepakt
- De elevators op de even levels andersom laten lopen met aanpassingen in veld
- Eventueel de elevators schokkerig laten lopen
- De oil can in pies afwisselend links, midden, rechts plaatsen (random of gekoppeld aan level)
- Eventueel firefoxes in pies aan random kant laten spawnnen
- Rivets girders op verschillende afstanden (random, of gekoppeld aan level) `

Omdraaien van de elevators. Eerst testen wat daarvoor aangepast moet worden. Eenvoudigste start is het omdraaien van de data tabel:

```
; used in elevators, called from #11EC  
; used for elevator sprites
```

```
3E64: 37 F4
```

```
3E66: 37 C0
```

```
3E68: 37 8C          ; elevators on left all have X value of 37
```

```
3E6A: 77 70
```

```
3E6C: 77 A4
```

```
3E6E: 77 D8          ; elevators on right all have X value of 77
```

Dus dan 37 wordt 77 en 77 wordt 37.

Dat gaat goed. Dan starten de elevators omgedraaid. Alleen als de elevator de bovenkant of onderkant bereikt gaat hij niet goed verder en draait de boel weer terug. Dus deze overgangen ook aanpassen.

De eerste zit hier:

```
27BA DD360377 LD (IX+#03),#77 ; set X position to right side of elevators  
27BE DD360D04 LD (IX+#0D),#04 ; set direction to down
```

De waarde aanpassen van 77 naar 37.

De andere zit hier:

```
27F4 DD360001 LD (IX+#00),#01 ; make elevator active  
27F8 DD360337 LD (IX+#03),#37 ; set X position to left side shaft  
27FC DD3605F8 LD (IX+#05),#F8 ; set Y position to bottom of shaft  
2800 DD360D08 LD (IX+#0D),#08 ; set direction to up  
2804 3634 LD (HL),#34 ; reset elevator counter to #34
```

De waarde aanpassen van 37 naar 77.

Ja dat gaat ook goed.

Alleen jumpman beweegt nog de verkeerde kant op. Deze checks ook aanpassen.

Die zitten hier:

```
274F 3A0362 LD A,(#6203) ; load A with mario's X position. eg 37 for first, 75 for second  
2752 FE2C CP #2C ; position < left edge of first elevator ?  
2754 DA6627 JP C,#2766 ; yes, jump ahead  
  
2757 FE43 CP #43 ; else is position < right edge of first elevator ?  
2759 DA6F27 JP C,#276F ; yes, jump ahead for first elevator checks  
  
275C FE6C CP #6C ; else is position < left edge of second elevator?  
275E DA6627 JP C,#2766 ; yes, jump ahead  
  
2761 FE83 CP #83 ; else is position < right edge of second elevator ?  
2763 DA8727 JP C,#2787 ; yes, jump ahead for second elevator checks
```

De waarden 2C en 43 omwisselen met 6C en 83. Werkt wel, maar jumpman wordt altijd op zijn kont verplaatst. Beetje vreemd. Terugdraaien

Misschien omdraaien door de jumps aan te passen. Dus de waarde 276F en 2787 omdraaien. Ja dat gaat wel goed. Dus nu omgedraaid.

Deze wijzigingen dan dus level afhankelijk maken. Ene set voor de even levels en de andere set voor de oneven levels.

Maar eerst ruimte vrijmaken zodat deze aanpassingen geplaatst kunnen worden. Het introschermpje waarin Donkey Kong de ladder beklimt verwijderen. Met daarbij ook de datavelden die niet meer nodig zijn.

Intro scherm overslaan door:

```
0A76 210A60 LD HL,GameMode2  
0A79 34 INC (HL) ; increase game mode2 (to 8?)  
0A7A C9 RET
```

Het hele stuk van #0A7B t/m #0BD9 komt hiermee vrij. Ruimte te gebruiken voor additonele

Ja dat gaat goed.

Nu versie v0.16 opgeslagen.

Nu de elevators afhankelijk van het level maken. Even levels de gewone elevators en de oneven levels de omgekeerde elevators.

Eerst het initialiseren van de elevators. Gaat om de data van de elevators die staat van #3E64 t/m #3E6F:

```
; used in elevators, called from #11EC  
; used for elevator sprites  
  
3E64: 37 F4  
3E66: 37 C0  
3E68: 37 8C ; elevators on left all have X value of 37  
  
3E6A: 77 70  
3E6C: 77 A4  
3E6E: 77 D8 ; elevators on right all have X value of 77
```

Deze wordt hier gebruikt:

```
10B7 21643E LD HL,#3E64 ; start of table data  
10BA 110366 LD DE,#6603 ; Destination sprite ? X positions ?  
10BD 010E06 LD BC,#060E ; Counter = #06, offset = #0E  
10C0 CDEC11 CALL #11EC ; set items from data table
```

Het hele blok wordt gekopieerd middels de generieke code op #11EC die steeds 2 data elementen kopieert. Handigste is om een kopie van het datablok te maken en afhankelijk van het level het ene of het andere te doen.

10B7 C3870A JP #0A87 ; jump to additional code – jump to label AAAAA

en

0A7B: 77 F4
0A7D: 77 C0
0A7F: 77 8C ; elevators on left all have X value of 37
;
0A81: 37 70
0A83: 37 A4
0A85: 37 D8 ; elevators on right all have X value of 77
;
0A87 3A2962 LD A,(#6229) ; load A with level # - label AAAAA
0A8A OF RRCA ; roll a right . is this an odd level ?
0A8B 3806 JR C,#0A93 ; yes, skip next steps
;
0A8D 21643E LD HL,#3E64 ; start of original table data for even levels
0A90 C3BA10 JP #10BA ; jump back
;
0A93 217B0A LD HL,#0A7B ; start of adapted table data for odd levels
0A96 C3BA10 JP #10BA ; jump back

En de originele data tabel herstellen.

Steeds testen met level 1 (odd), dus naar beneden blijven bewegen. Klopt.

Dan de eerste elevator overgang aanpassen:

27BA DD360377 LD (IX+#03),#77 ; set X position to right side of elevators
27BE DD360D04 LD (IX+#0D),#04 ; set direction to down

Wordt:

27BA C3990A JP #0A99 ; jump to additional code – jump to label AAAAA
27BD 00 NOP

en

0A99 3A2962 LD A,(#6229) ; load A with level # - label AAAAA
0A9C OF RRCA ; roll a right . is this an odd level ?
0A9D 3807 JR C,#0AA6 ; yes, skip next steps
;
0A9F DD360377 LD (IX+#03),#77 ; set original X-position for even levels
0AA3 C3BE27 JP #27BE ; jump back
;
0AA6 DD360337 LD (IX+#03),#37 ; set adapted X-position for odd levels
0AAA C3BE27 JP #27BE ; jump back

Werkt goed.

Dan de tweede elevator overgang aanpassen:

```
27F4 DD360001 LD (IX+#00),#01 ; make elevator active
27F8 DD360337 LD (IX+#03),#37 ; set X position to left side shaft
27FC DD3605F8 LD (IX+#05),#F8 ; set Y position to bottom of shaft
2800 DD360D08 LD (IX+#0D),#08 ; set direction to up
2804 3634 LD (HL),#34 ; reset elevator counter to #34
```

Zelfde manier van oplossen:

Wordt:

```
27F8 C3AD0A JP #0AAD ; jump to additional code – jump to label AAAA
27FB 00 NOP
```

en

```
0AAD 3A2962 LD A,(#6229) ; load A with level # - label AAAA
0AB0 0F RRCA ; roll a right . is this an odd level ?
0AB1 3807 JR C,#0AA6 ; yes, skip next steps
;
0AB3 DD360337 LD (IX+#03),#37 ; set original X-position for even levels
0AB7 C3FC27 JP #27FC ; jump back
;
0ABA DD360377 LD (IX+#03),#77 ; set adapted X-position for odd levels
0ABE C3FC27 JP #27FC ; jump back
```

Werkt goed.

En dan nog jumpman ook goed verplaatsen wanneer hij op een elevator staat.

```
274F 3A0362 LD A,(#6203) ; load A with mario's X position. eg 37 for first, 75 for second
2752 FE2C CP #2C ; position < left edge of first elevator ?
2754 DA6627 JP C,#2766 ; yes, jump ahead

2757 FE43 CP #43 ; else is position < right edge of first elevator ?
2759 DA6F27 JP C,#276F ; yes, jump ahead for first elevator checks

275C FE6C CP #6C ; else is position < left edge of second elevator?
275E DA6627 JP C,#2766 ; yes, jump ahead

2761 FE83 CP #83 ; else is position < right edge of second elevator ?
2763 DA8727 JP C,#2787 ; yes, jump ahead for second elevator checks
```

```
2759 C3C10A      JP    #0AC1 ; jump to additional code – jump to label AAAA  
...  
2763 C3CB0A      JP    #0ACB      ; jump to additional code – jump to label BBBB
```

en

```
0AC1 3A2962      LD    A,(#6229)   ; load A with level # - label AAAA  
0AC4 0F          RRCA           ; roll a right, is this an odd level ?  
0AC5 DA8727      JP    C,#2787    ; yes, jump back to alternate location  
0AC8 C36F27      JP    #276F     ; no, jump back to original location  
;  
0ACB 3A2962      LD    A,(#6229)   ; load A with level # - label BBBB  
0ACE 0F          RRCA           ; roll a right, is this an odd level ?  
0ACF DA6F27      JP    C,#276F    ; yes, jump back to alternate location  
0AD2 C38727      JP    #2787    ; no, jump back to original location
```

Dat werkt. Nu verder gaan testen met level 5 en/of 6.

Aanpassen start level:

```
095E 01 65 3A 01 00 00 00      ; #3A65 is start of table data for screens/levels
```

Als starten L=05 dan dit aanpassen naar 05 en 73 3A.

Gaat niet goed. Als je doorspringt naar de platformen in het midden dan wordt jumpman daar ook verplaatst.

Snap het al. De conditionele jumps zijn vervangen door onconditionele jumps. Dat is niet goed.

Aanpassen:

```
2759 DAC10A      JP    C,#0AC1    ; jump to additional code – jump to label AAAA  
...  
2763 DACB0A      JP    C,#0ACB    ; jump to additional code – jump to label BBBB
```

Level 5 (omgekeerde elevators) en level 6 (normale elevators) werken beiden goed.

Nu versie v0.17 opgeslagen.

De ‘Staal’-loop kan in de omgekeerde variant niet gedaan worden. Is inderdaad wel een vreemde weg die niemand neemt, maar kan nu sowieso niet. Het eerste platform omhoog doen.

Dat is deze girder:

```
3C7B: 02 9F E8 88 E8 ; floating girder where the right side elevator gets off
```

Moet dan qua Y-positie naar D8:

3C7B: 02 9F D8 88 D8 ; floating girder where the right side elevator gets off

Dat gaat goed.

Extra uitdaging door het bonus-item van het middelste platform beneden (de originele hat) te verplaatsen naar het zojuist verplaatste platform.

De items van elevators:

```
; bonus items for elevators

3E48 5B 73 0A C8 ; hat at 5B,C8
3E4C E3 74 0A 60 ; purse at E3,60
3E50 1B 75 0A 80 ; umbrella on elevator is 80,1B
```

Dus dan (kleur was al aangepast naar 0D):

3E48 93 73 0D D0 ; hat

Gaat goed.

Nu versie v0.18 opgeslagen.

Dan nu kijken voor variaties op het pies screen. De oil can afhankelijk van het level naar links of rechts of nog in het midden.

Eerst kijken hoe oil can te verplaatsen en hoe het spawnen aan te passen. Daarna kan gekeken worden naar de exacte locaties en het schuiven van de rest van het veld.

De positie van de oil can wordt bepaald door de volgende data table:

```
3E00 27 49 0C F0 ; oil can for girders
3E04 7F 49 0C 88 ; oil can for conveyors ?
```

Verschuiven door de waarde in #3E04 aan te passen. Bijvoorbeeld naar 3F. Ja dan is de oil can verplaatst. Alleen het vuurtje brand nog op de verkeerde plek en de fireballs spawnen op de verkeerde plek.

Het vlammetje wordt hier geïnitialiseerd:

```
1049 21FA3D LD HL,#3DFA ; load HL with start of table data
104C CDFA11 CALL #11FA ; set fireball sprite
```

Met de volgende data tabel:

3DFA: 7F 40 01 78 02 00 ; initial data for conveyors to release a fire ?

Kijken of de eerste waarde in de tabel de X-positie is: 7F moet dan 3F worden.

Ja dat werkt.

Het spawnen van de fireballs wordt gestuurd door de volgende data table:

```
; table data refeernced in #
; controls the positions of fires coming out of the oil can on the conveyors

3AAC:  80 7B 78 76 74 73 72 71 70 70 6F 6F 6F 70 70 71 72 73 74 75 76 77 78
3AC3:  AA           ; end code
```

Die moet dan worden:

3AAC: 40 3B 38 36 34 33 32 31 30 30 2F 2F 2F 30 30 31 32 33 34 35 36 37 38

Dat gaat niet goed. Blijkbaar zijn dit de Y-posities.

De spawn vindt hier plaats voor rechts:

```
3496  DD36037E  LD      (IX+#03),#7E    ; Initialize X position to #7E
```

en hier plaats voor links:

```
34AC  DD360380  LD      (IX+#03),#80
```

Aanpassen naar 3E en 40 ?

Ja dat werkt.

Dan moeten de twee conveyor belts ook aangepast worden (los van het tekenen van de banden en de eindpunten). Zodat de pies eindigen bij de nieuwe oil can.

Dat lijkt hier te gebeuren:

```
25C0  7C          LD      A,H        ; load A with pie X position
25C1  FE80        CP      #80        ; is the pie in the center fire?
25C3  CAD625     JP      Z,#25D6    ; yes, skip ahead
```

De 80 aanpassen naar 40.

Ja dat gaat ook goed.

Dan gaan bepalen hoe het veld er dan uit komt te zien en op welke plek de oil-can geplaatst gaat worden (zowel linkse als rechtse variant). Dan het veld tekenen (onder de variabele delen) en de twee variabele delen afhankelijk van even of neven level tekenen. Daarna bovenstaande aanpassingen afhankelijk van het level maken.

De oilcan op de positie #50 (links) en #B0 (rechts plaatsen).

Alleen de twee girders van de bovenste conveyor hoeven dan maar aangepast te worden. Er hoeven geen trappen aangepast te worden.

Kijken drie verschillende velden: links, midden, rechts. Midden eigenlijk als eerste (op level 3) en dan steeds midden op alle odd levels en voor de even levels afwisselen tussen links en rechts.

Level 3: 0011 : midden

Level 4: 0100 : links

Level 5: 0101 : midden

Level 6: 0110 : rechts

Level 7: 0111 : midden

Level 8: 1000 : links

Level 9: 1001 : midden

Level 10: 1010 : rechts

Dus bit 0 gelijk aan 1 dan midden, anders bit 1 gelijk aan 0 dan links, anders rechts.

Eerst de oil can maar eens plaatsen:

Wordt door deze data tabel gedefinieerd:

```
3E00 27 49 0C F0      ; oil can for girders  
3E04 7F 49 0C 88      ; oil can for conveyors ?
```

Twee extra tabellen maken en dan de keuze bij het tekenen maken voor de juiste data tabel.

Dit wordt hier aangeroepen:

```
104F 21043E LD HL,#3E04      ; set up copy job from table in #3E04 = oil can sprite  
1052 11FC69 LD DE,#69FC      ; destination is #69FC = sprite  
1055 010400 LD BC,#0004      ; four bytes to copy  
1058 EDB0 LDIR              ; draw oil can
```

Aanpassingen:

```
104F CDDDOA CALL #0ADD      ; call additional code – jump to label AAAA
```

en

```
0AD5 4F 49 0C 88          ; oil can for conveyors left side  
0AD9 AF 49 0C 88          ; oil can for conveyors right side  
;  
0ADD 3A2962 LD A,(#6229)    ; load A with level # - label AAAA  
0AE0 0F RRCA              ; roll a right . is this an odd level ?
```

```

0AE1 380B      JR     C, #0AEE      ; yes, use middle – jump to label BBBB
;
0AE3 0F        RRCA
0AE4 3804      JR     C, #0AEA      ; roll a right . determine left or right
; use right – jump to label CCCC
;
0AE6 21D50A    LD     HL, #0AD5      ; load HL with data table oil can left
0AE9 C9        RET
;
0AEA 21D90A    LD     HL, #0AD9      ; load HL with data table oil can right – label CCCC
0AED C9        RET
;
0AEE 21043E    LD     HL, #3E04      ; load HL with data table oil can middle – label BBBB
0AF1 C9        RET

```

Ja dat werkt goed.

Dan nu het vlammetje ook verplaatsen op basis van level:

Het vlammetje wordt hier geïnitialiseerd:

```

1049 21FA3D    LD     HL, #3DFA      ; load HL with start of table data
104C CDFA11    CALL   #11FA      ; set fireball sprite

```

Met de volgende data tabel:

```
3DFA: 7F 40 01 78 02 00      ; initial data for conveyors to release a fire ?
```

Twee extra tabellen maken en dan de keuze bij het tekenen maken voor de juiste data tabel.

Aanpassingen:

```
1049 CDFFEOA    CALL   #0AFE      ; call additional code – jump to label AAAA
```

en

```

0AF2 4F 40 01 78 02 00      ; fire for conveyors left side
0AF8 AF 40 01 78 02 00      ; fire for conveyors right side
;
0AFE 3A2962    LD     A, (#6229)  ; load A with level # - label AAAA
0B01 0F        RRCA
0B02 380B      JR     C, #0AEE      ; yes, use middle – jump to label BBBB
;
0B04 0F        RRCA
0B05 3804      JR     C, #0AEA      ; roll a right . determine left or right
; use right – jump to label CCCC
;
0B07 21F20A    LD     HL, #0AF2      ; load HL with data table fire left
0B0A C9        RET

```

```

;
0B0B 21F80A      LD   HL, #0AF8      ; load HL with data table fire right – label CCCC
0B0E C9          RET
;
0B0F 21FA3D      LD   HL, #3DFA      ; load HL with data table oil can middle – label BBBB
0B12 C9          RET

```

Dat werkt goed.

Dan nu het spawnen van de fireballs aanpassen:

De spawn vindt hier plaats voor rechts:

```
3496 DD36037E LD   (IX+#03),#7E      ; Initialize X position to #7E
```

en hier plaats voor links:

```
34AC DD360380 LD   (IX+#03),#80
```

Rechts aanpassen en links zelfde waarde geven. Origineel was 80 en 7E. Daartussen gaan zitten met 7F. Dan beide kanten dezelfde code laten gebruiken.

```

3496 CD130B      CALL  #0B13      ; jump to additional code – jump to AAAA
3499 00          NOP

```

En

```

34AC CD130B      CALL  #0B13      ; jump to additional code – jump to AAAA
34AF 00          NOP

```

en

```

0B13 3A2962      LD   A,(#6229)    ; load A with level # - label AAAA
0B16 0F           RRCA            ; roll a right . is this an odd level ?
0B17 380D      JR   C, #0B26      ; yes, use middle – jump to label BBBB
;
0B19 0F           RRCA            ; roll a right . determine left or right
0B1A 3805      JR   C, #0B21      ; use right – jump to label CCCC
;
0B1C DD36034F      LD   (IX+#03),#4F  ; initialize X-position left to #4F
0B20 C9          RET
;
0B21 DD3603AF      LD   (IX+#03),#AF  ; initialize X-position right to #AF – label CCCC
0B25 C9          RET
;
0B26 DD36037F      LD   (IX+#03),#7F  ; initialize X-position middle to 7F – label BBBB
0B2A C9          RET

```

Ja dat werkt prima.

Nu versie v0.19 opgeslagen.

Nu de pies ook laten verdwijnen bij de oil can.

```
25C0 7C LD A,H ; load A with pie X position
25C1 FE80 CP #80 ; is the pie in the center fire?
25C3 CAD625 JP Z,#25D6 ; yes, skip ahead
```

Aanpassen:

```
25C0 CD2B0B CALL #0B2B ; jump to additional code
```

en

```
0B2B 3A2962 LD A,(#6229) ; load A with level # - label AAAA
0B2E OF RRCA ; roll a right . is this an odd level ?
0B2F 380B JR C, #0B3C ; yes, use middle – jump to label BBBB
;
0B31 OF RRCA ; roll a right . determine left or right
0B32 3804 JR C, #0B38 ; use right – jump to label CCCC
;
0B34 7C LD A,H ; load A with pie X position
0B35 FE50 CP #50 ; is the pie in the left fire?
0B37 C9 RET ;
;
0B38 7C LD A,H ; load A with pie X position – label CCCC
0B39 FEBO CP #B0 ; is the pie in the right fire?
0B3B C9 RET ;
;
0B3C 7C LD A,H ; load A with pie X position – label BBBB
0B3D FE80 CP #80 ; is the pie in the middle fire?
0B3F C9 RET
```

Ja dat werkt goed.

De draaipunten van de bovenste conveyors moeten ook aangepast worden (diegene naast de oil can).

Definitie daarvan zit hier:

```

; the 6 conveyor pulleys

3E24: 17 50 00 5C          ; 50 = edge of conveyor pulley
3E28: E7 D0 00 5C          ; D0 = edge of conveyor pulley inverted
3E2C: 8C 50 00 84
3E30: 73 D0 00 84
3E34: 17 50 00 D4
3E38: E7 D0 00 D4

```

En deze wordt hier gebruikt:

```

1065 21243E LD HL,#3E24      ; set source table data
1068 11E469 LD DE,#69E4      ; set destination RAM sprites
106B 011800 LD BC,#0018      ; set counter
106E EDB0 LDIR                ; draw pulleys

```

Dit tekent alle draaipunten in één keer. Misschien zo laten en dan in een apart stukje code daarna de X-waarde van de twee draaipunten aanpassen.

De twee draaipunten zijn: #3E2C en #3E30. Als dit gekopieerd wordt dan is de mapping van het begin van de tabel #3E24=#69E4. X-waardes herberekenen:

```

#3E2C -> #69EC
#3E30 -> #69F0

```

Aanpassingen:

```

106B CD400B CALL #0B40      ; Jump to additional code – jump to label AAAA
106E 0000 NOP, NOP

```

en

```

0B40 011800 LD BC,#0018      ; set counter – label AAAA
0B43 EDB0 LDIR                ; draw pulleys
;
0B45 3A2962 LD A,(#6229)      ; load A with level #
0B48 0F RRCA                  ; roll a right . is this an odd level ?
0B49 380B JR C, #0B56        ; yes, use middle – jump to label BBBB
;
0B4B 0F RRCA                  ; roll a right . determine left or right
0B4C 3804 JR C, #0B52        ; use right – jump to label CCCC
;
0B4E 3E43 LD A,#43            ; load A with X-position for oil can left
0B50 1806 JR #0B58            ; reposition pulleys jump to label DDDD
;
0B52 3EA3 LD A,#A3            ; load A with X-position for oil can right – label CCCC
0B54 1802 JR #0B58            ; reposition pulleys jump to label DDDD

```

```

;
0B56 3E73      LD    A,#73      ; load A with X-position for middle– label BBBB
;
0B58 32F069      LD    (#69F0),A   ; reposition left pulley – label DDDD
;
0B5B 0619      LD    B,#19      ; load B with distance left and right pulley
0B5D 80      ADD   A,B      ; calculate position right pulley
0B5E 32EC69      LD    (#69EC),A   ; reposition right pulley
;
0B61 C9      RET

```

Gaat goed.

Nu nog de girder als één geheel tekenen en daarin dan op vier plaatsen een gat maken. Dat zou kunnen door de bovenstaande code uit te breiden. Ook kan het raster uit de definitie verwijderd worden.

```

; table data for screen 2 conveyors
; 135 bytes long

3B5D: 06 8F 90 70 90 ; central patch of XXX's
3B62: 06 8F 98 70 98 ; central patch of XXX's
3B67: 06 8F A0 70 A0 ; central patch of XXX's
3B6C: 00 63 18 63 58 ; kong's ladder (right)

```

Het raster verwijderen kan door te starten met tekenen vanaf #3B6C. Dit wordt hier aangeroepen:

```

0CDF 115D3B  LD    DE,#3B5D      ; load DE with start of table data for conveyors
0CE2 21867D  LD    HL,REG_PALETTE_A    ; load HL with palette bank selector
0CE5 3601  LD    (HL),#01      ; set palette bank selector

```

Aanpassen:

```
0CDF 116C3B  LD    DE,#3B6C      ; load DE with start of table data for conveyors
```

Dat werkt.

De girders samenvoegen betreft de volgende:

```

3BC1: 03 F7 80 88 80 ; top right conveyor next to oil can
3BC6: 03 77 80 08 80 ; top left conveyor next to oil can

```

Aanpassen:

```
3BC1: 03 F7 80 08 80 ; top conveyor
```

En de rest opschuiven.

Dat werkt.

[Nu versie v0.20 opgeslagen.](#)

Fireballs achter de oil can fire langs laten gaan is niet eens zo gek. Dan worden ze niet links of rechts vastgezet. Voor jumpman is de kort loopband wel een lastigere hindernis.

Mario op de lopende band gaat ook nog niet goed. Moet ook aangepast worden naar bewegen tot de oil can. Nu altijd naar het midden van het scherm.

In ieder geval niet zo mooi qua design omdat de oil-can deels de trap raakt. En daardoor jumpman deels achter het draaipunt verdwijnt. Daarom de twee trappen in het midden allebei één plek naar het midden verplaatsen:

```
3B71:  00 63 80 63 A8 ; center ladder to left of oil can fire  
3B76:  00 63 D0 63 F8 ; bottom level ladder #2 of 4  
3B7B:  00 53 18 53 58 ; kong's ladder (left)  
3B80:  00 53 A8 53 D0 ; ladder under the hat  
3B85:  00 9B 80 9B A8 ; center ladder to right of oil can fire
```

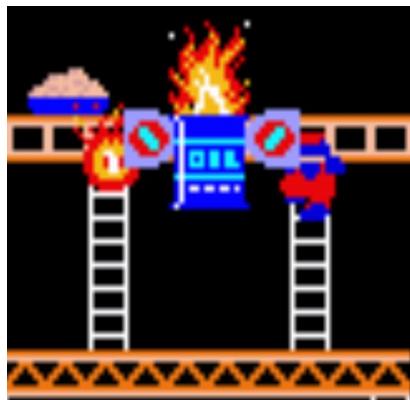
Het betreft dan de twee center ladders:

Aanpassen:

3B72: 6B

3B86: 93

Nee dat gaat ook niet goed omdat dan jumpman deels achter het draaipunt verdwijnt:



Wordt allemaal wel erg veel code nog om dit goed te krijgen: jumpman schuiven en ladders ook level afhankelijk maken.

Daarom kijken naar alternatief waarbij de oil-can een niveau lager staat.

[Terug naar versie v0.18.](#)

Dan eerste effect testen door de firballs te verplaatsen.

Het spawnen van de fireballs wordt gestuurd door de volgende data table:

```
; table data referenced in #
; controls the positions of fires coming out of the oil can on the conveyors

3AAC:  80 7B 78 76 74 73 72 71 70 70 6F 6F 6F 70 70 71 72 73 74 75 76 77 78
3AC3:  AA          ; end code
```

Die moet dan worden:

3AAC: A0 9B 98 96 94 93 92 91 90 90 8F 8F 8F 90 90 91 92 93 94 95 96 97 98

Niet goed, moet nog lager:

3AAC: A8 A3 A0 9E 9C 9B 9A 99 98 98 97 97 97 98 98 99 9A 9B 9C 9D 9E 9F A0

Dit gaat wel goed. Een stuk lastiger is dit, omdat de fireballs veel meer de nadruk op beneden leggen.

Dus deze gaan uitwerken. Op de oneven levels de oude situatie en op de even levels de verplaatste oil can (28 lager , Y-positie hoger).

Eerst de oil can maar eens plaatsen:

Wordt door deze data tabel gedefinieerd:

```
3E00  27 49 0C F0      ; oil can for girders
3E04  7F 49 0C 88      ; oil can for conveyors ?
```

Een extra tabel maken en dan de keuze bij het tekenen maken voor de juiste data tabel.

Dit wordt hier aangeroepen:

```
104F  21043E  LD    HL,#3E04      ; set up copy job from table in #3E04 = oil can sprite
1052  11FC69  LD    DE,#69FC      ; destination is #69FC = sprite
1055  010400  LD    BC,#0004      ; four bytes to copy
1058  EDB0    LDIR           ; draw oil can
```

Aanpassingen:

104F CDD90A CALL #0AD9 ; call additional code – jump to label AAAA

en

```

0AD5 7F 49 0C B0          ; oil can for alternative conveyors
;
0AD9 3A2962      LD A,(#6229)   ; load A with level # - label AAAA
0ADC 0F          RRCA          ; roll a right . is this an odd level ?
0ADD 3804      JR C,#0A93    ; yes, skip next steps
;
0ADF 21D50A      LD HL,#0AD5   ; start of original table data for even levels
0AE2 C9          RET           ; jump back
;
0AE3 21043E      LD HL, #3E04   ; start of adapted table data for odd levels
0AE6 C9          RET           ; jump back

```

Dat werkt goed.

Het vlammetje ook naar beneden verplaatsen:

Het vlammetje wordt hier geïnitialiseerd:

```

1049 21FA3D      LD     HL, #3DFA      ; load HL with start of table data
104C CDFA11      CALL  #11FA       ; set fireball sprite

```

Met de volgende data tabel:

```

3DFA: 7F 40 01 78 02 00      ; initial data for conveyors to release a fire ?

```

Een extra tabel maken en dan de keuze bij het tekenen maken voor de juiste data tabel.

Aanpassingen:

```

1049 CDED0A      CALL  #0AED      ; call additional code – jump to label AAAA

```

en

```

0AE7 7F 40 01 A0 02 00      ; alternative data table for release of fires
;
0AED 3A2962      LD A,(#6229)   ; load A with level # - label AAAA
0AF0 0F          RRCA          ; roll a right . is this an odd level ?
0AF1 3804      JR C,#0AF7    ; yes, skip next steps
;
0AF3 21E70A      LD HL,#0AE7   ; start of original table data for even levels
0AF6 C9          RET           ; jump back
;
0AF7 21FA3D      LD HL, #3DFA   ; start of adapted table data for odd levels
0AF8 C9          RET           ; jump back

```

Dat werkt goed.

Het spawnen van de fireballs aanpassen.

De data tabel met de Y-posities voor het spawnen wordt hier geladen:

```
3487 21AC3A LD HL, #3AAC ; load HL with start of table data
348A 3A0362 LD A, (#6203) ; \ Jump if Mario is on left side of the screen
348D CB7F BIT 7, A ; |
348F CAA834 JP Z, #34A8 ; /
```

Een extra tabel maken en dan de keuze bij het tekenen maken voor de juiste data tabel.

Aanpassingen:

```
3487 CD130B CALL #0B13 ; call additional code – jump to label AAAA
```

en

```
0AFB A8 A3 A0 9E 9C 9B 9A 99 98 98 97 97 97 98 98 99 9A 9B 9C 9D 9E 9F A0 AA
;
0B13 3A2962 LD A, (#6229) ; load A with level # - label AAAA
0B16 0F RRCA ; roll a right . is this an odd level ?
0B17 3804 JR C, #0B1D ; yes, skip next steps
;
0B19 21FB0A LD HL, #0AFB ; start of original table data for even levels
0B1C C9 RET ; jump back
;
0B1D 21AC3A LD HL, #3AAC ; start of adapted table data for odd levels
0B20 C9 RET ; jump back
```

Dat gaat goed.

Nu versie v0.21 opgeslagen.

De hamers aanpassen. De onderste hamer zit gedeeltelijk voor de oil can en de fire. Dus die iets naar links verplaatsen. De andere naar het onderste niveau, waar je er in nood meer aan hebt.

Dit is de definitie van de hamers voor pies:

```
3E10: 23 8D 7B B4 ; for conveyors
```

Wordt dan:

```
3E10: 8B DC 6B B4 ; for conveyors.
```

Maar dan de hamer wel precies boven het bonus item. Daarom de onderste bonus item verplaatsen.

Bonus items definitie wordt dan:

3E3C: 53 73 0D A0

3E40: BB 74 0D F0

3E44: DB 75 0D A0

Dat is nu goed.

Nu versie v0.22 opgeslagen.

Zo maken dat de middelste girder waar de oil can naartoe verplaatst wordt ook gebroken is, dan kunnen de fireballs daar niet van links naar rechts doorsteken. Wellicht dan wel nodig dat de fireballs aan beide kanten spawnen. En dan de kruisen aanpassen zodat de pies 'daarin verdwijnen'.

De kruisen definitie:

```
3B5D: 06 8F 90 70 90 ; central patch of XXX's
3B62: 06 8F 98 70 98 ; central patch of XXX's
3B67: 06 8F A0 70 A0 ; central patch of XXX's
3B6C: 00 63 18 63 58 ; kong's ladder (right)
3B71: 00 63 80 63 A8 ; center ladder to left of oil can fire
```

De kruisen naar boven helpt niet zo heel veel. Is alleen maar raar. Dan anders gewoon laten staan. Maar niet zo mooi bij de verplaatsde oil can. Daarom eventjes weghalen en de rest van de definitie opschuiven.

Dan de girder opsplitsen:

```
3BCB: 02 A7 A8 50 A8 ; center ledge
```

Deze eventjes aan de rechterkant korter maken: 02 70 A8 50 A8.

Een een extra tussenvoegen: 02 A7 A8 88 A8.

Ja dat gaat goed. Is leuk.

De fireballs random aan beide zijden laten spawnen. Is nu gebaseerd op de X-positie van jumpman:

```
348A 3A0362 LD A,(#6203) ; \ Jump if Mario is on left side of the screen,
348D CB7F BIT 7,A ; |
348F CAA834 JP Z,#34A8 ; /
```

Random kant spawnen van de fireballs:

```
348A 3A1960 LD A,(RngTimer2) ; Determine side to spawn fireball
348D CB47 BIT 0,A ; If timer value is even . . .
```

348F CAA834 JP Z,#34A8 ; . . . jump and spawn the fireball at left side

Ja dat gaat goed.

Nu versie v0.23 opgeslagen.

Nu alleen nog een soort van verdwijn-hok sprite plaatsen waar de pies in verdwijnen.

Zelfde als de oil can maken maar dan kleiner. Sprite #6B hiervoor gebruikt.

Als sprite een barrel sprite hiervoor gebruiken. De eerste barrel sprite is #6980.

De sprite tekenen vanuit de code voor het plaatsen van de oil can:

0AD5 7F 49 0C B0		; oil can for alternative conveyors
;		
0AD9 3A2962	LD A,(#6229)	; load A with level # - label AAAA
0ADC 0F	RRCA	; roll a right . is this an odd level ?
0ADD 3804	JR C,#0A93	; yes, skip next steps
;		
0ADF 21D50A	LD HL,#0AD5	; start of original table data for even levels
0AE2 C9	RET	; jump back
;		
0AE3 21043E	LD HL, #3E04	; start of adapted table data for odd levels
0AE6 C9	RET	; jump back

Aanpassen:

0ADF CD210B CALL #0B21 ; jump to additional code – jump to label AAAA

en

0B21 218069	LD HL,#6980	; load HL with barrel sprite
0B24 367F	LD (HL),#7F	; set pipe sprite X position
0B26 23	INC HL	; next
0B27 366B	LD (HL),#6B	; set pipe sprite
0B29 23	INC HL	; next
0B2A 360C	LD (HL),#0C	; set pipe sprite color
0B2C 23	INC HL	; next
0B2D 3680	LD (HL),#80	; set pipe sprite Y position
;		
0B2F 21D50A	LD HL, #0AD5	; start of adapted table data for odd levels
0B32 C9	RET	; jump back

Met de barrel sprite schuiven de pies ervoor langs en liever er achterlangs. Of echt een soort opening maken. Opening gemaakt.

Nu versie v0.24 opgeslagen.

Kijken eventueel of je de sprite ook naar vorn kunt krijgen. Vlammetje is sprite locatie: #6A28. Dus dat is 6A28, 6A29, 6A2A en 6A2B. Wordt 6A2C ook gebruikt? Ja scoring. Wordt 6A30 ook gebruikt? Ja, scoring. Wordt 6A34 ook gebruikt? Nee. Kijken of dat werkt.

```
0B21 21346A      LD      HL,#6A34      ; load HL with free sprite
```

Ja schuift er nu achter. Dan kijken voor oil can achtige sprite. Ja dat is mooi zo.

Nu versie v0.15 opgeslagen.

Er is ook nog een probleem met Game Over in dit scherm. De oil can en het vlammetje erboven blijven staan en staan door de Game Over melding.

Dus bij het plaatsen van de Game Over melding, moeten de oil can sprite en de vlam sprite verwijderd worden. Dat zijn de sprites: #69FC (oil can) en #6A28 (vlam). Deze beiden een X-waarde van 0 geven: dan buiten beeld geplaatst.

Game Over tekst wordt twee keer geplaatst: Player I en Player II:

```
1322 CD2618  CALL  #1826      ; clear an area of the screen  
1325 110003  LD    DE,#0300      ; load task data for text #0 "GAME OVER"  
1328 CD9F30  CALL  #309F      ; insert task to draw text  
132B 210960  LD    HL,WaitTimerMSB ; load HL with timer  
132E 36C0    LD    (HL),#C0      ; set timer to #C0
```

En

```
136A 110003  LD    DE,#0300      ; load task data for text #0 "GAME OVER"  
136D CD9F30  CALL  #309F      ; insert task to draw text  
1370 21D376  LD    HL,#76D3      ; load HL with screen address ???  
1373 CD2618  CALL  #1826      ; clear an area of the screen  
1376 210960  LD    HL,WaitTimerMSB ; load HL with timer  
1379 36C0    LD    (HL),#C0      ; set timer to #C0
```

Aanpassen:

```
132B CD330B  CALL  #0B33      ; call additional code
```

en

```
1376 CD330B  CALL  #0B33      ; call additional code
```

en

```

0B33 210960      LD      HL,WaitTimerMSB      ; load HL with timer
;
0B36 3A2762      LD      A,(#6227)      ; load A with screen number
0B39 FE02        CP      #02          ; are we on the conveyors?
0B3B C0          RET     NZ          ; no, return
;
0B3C 3A2962      LD      A,(#6229)      ; load A with level #
0B3F OF          RRCA
0B40 D8          RET     C           ; yes, return
;
0B41 3E00        LD      A,#00          ; A:=#00
0B43 32FC69      LD      (#69FC),A      ; store in X-position oil can sprite
0B46 32286A      LD      (#6A28),A      ; store in X-position fire sprite
0B49 C9          RET

```

Dit gaat goed voor pies level 6 (met verschoven oil can) voor zowel player I als player II.

Dit gaat goed voor pies level 5 (met originele oil can) voor zowel player I als player II.

Dit gaat goed voor barrels level 6 voor zowel player I als player II.

Nu versie v0.26 opgeslagen.

Dan nu nadenken wat te doen met het rivets level. Wellicht mogelijk om de spacing tussen de girders aan te passen? Lastig want dan moeten er al snel ook zeven ladders mee aangepast worden.

En wat te doen met het barrels level? Wellicht de hamers op een andere plek hangen afhankelijk even/oneven level. En misschien een gat in de girders op twee verschillende plekken afhankelijk even/oneven level. En/of misschien een ladder verwijderen? In ieder geval ook bij barrels drie bonus items weergeven en de afhandeling hiervan doen.

Plaatsen van de hamers voor barrels gebeurt hier:

```

1000 210C3E      LD      HL,#3E0C      ; load HL with table data for hammers on girders
1003 CDA611      CALL    #11A6      ; ???

```

Met de bijbehorende data tabel:

```

3E0C: 24 64      ; position of top hammer for girders. 24 is X, 64 is Y
3E0E: BB C0      ; bottom hammer for girders at BB, C0

```

Aanpassen:

```
1000 CD4E0B      CALL    #0B4E      ; call additional code – call label AAAA
```

en

```
0B4A 24 C0 BB 64      ; alternative hammer location girders
;
```

```

0B4E 3A2962      LD     A,(#6229)    ; load A with level # - label AAAA
0B51 0F          RRCA
0B52 3804      JR     C,#0AE3    ; roll a right . is this an odd level ?
;
0B54 214A0B      LD     HL,#0B4A    ; start of adapted table data for even levels
0B57 C9          RET
;
0B58 210C3E      LD     HL, #3E0C    ; start of original table data for odd levels
0B5B C9          RET
;
```

Omdat bij de hogere levels steeds 3 barrels screens gespeeld moeten worden, de aanpassingen niet van het level laten afhangen, maar random maken. Verschillende hamer posities, verschillende bonus item posities, verschillende plekken van gaten en verschillende trappen verwijderen.

Level afhankelijk maken door A niet te laden met het level, maar met de RngTimer2:

```
0B4E 3A1960      LD     A,(#RngTimer 2)    ; load A with level # - label AAAA
```

Bonus items plaatsen op verschillende plekken. Voorbeeld definitie:

```
; bonus items on conveyors
```

```

3E3C 53 73 0A A0      ; position of hat on pies is 53,A0
3E40 8B 74 0A F0      ; position of purse on pies is 8B,F0
3E44 DB 75 0A A0      ; umbrella on the pies is at DB,A0
;
```

En voorbeeld activeren van de bonus items:

```

1076 213C3E      LD     HL,#3E3C    ; load HL with table data for bonus items on conveyors
1079 110C6A      LD     DE,#6A0C    ; load DE with sprite destination
107C 010C00      LD     BC,#000C    ; 3 items x 4 bytes = 12 bytes (#0C)
107F EDB0        LDIR
;
```

Dus twee definities maken en dan de bovenstaande code uitbreiden met test op RngTimer2 en dan op basis daarvan HL laden met het juiste adres en dan de bonus items aanmaken.

```
1000 CD660B      CALL   #0B66    ; call additional code – call label AAAA
```

en

```

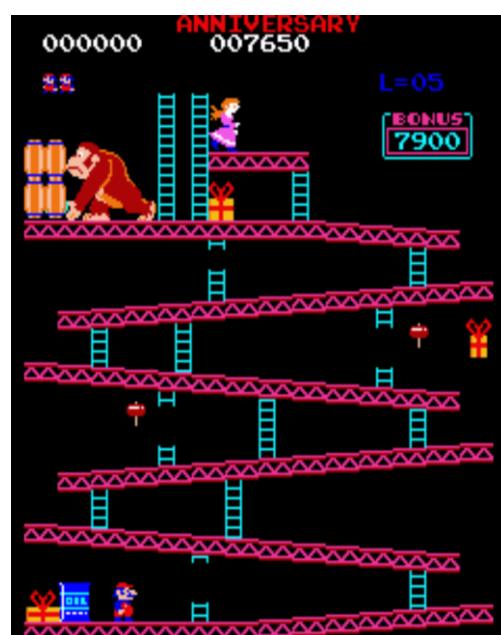
0B4A 44 A0 CB 80    ; alternative hammer location girders
;
0B4E 24 73 0D B3    ; position of presents on girders
0B52 DB 74 0D 92
0B56 DA 75 0D D4
;
0B5A 6D 73 0D 4C    ; alternative location of presents on girders
;
```

```

0B5E 18 74 0D F0
0B62 E8 75 0D 84
;
0B66 3A1960 LD A,(RnGTimer2) ; load A with RngTimer 2 - label AAAA
0B69 0F RRCA ; roll a right. is this an odd number ?
0B6A 3805 JR C,#0B71 ; yes, skip next steps
;
0B6C 214E0B LD HL,#0B4E ; start of adapted table data for even rng
0B6F 1803 JR #0B74 ; skip next step
;
0B71 215A0B LD HL, #0B5A ; start of original table data for odd rng
;
0B74 110C6A LD DE,#6A0C ; load DE with sprite destination
0B77 010C00 LD BC,#000C ; 3 items x 4 bytes = 12 bytes (#0C)
0B7A EDB0 LDIR ; draw bonus item sprites
;
0B7C 0F RRCA ; roll a right again. is this an odd number ?
0B7D 3804 JR C,#0B83 ; yes, skip next steps
;
0B7F 214A0B LD HL,#0B4A ; start of adapted table data for even rng
0B82 C9 RET ; jump back
;
0B83 210C3E LD HL, #3EOC ; start of original table data for odd rng
0B86 C9 RET ; jump back

```

De locaties van de presents:



Nu versie v0.27 opgeslagen.

Dan nu nog een gat in de girders maken op een random plek (keuze uit twee).

Dat is al eerder gedaan en kan op de volgende wijze:

```
102C 211276 LD HL,#7612 ; video memory address to clear  
102F 3E32 LD A,#10 ; load a with I character  
1031 77 LD (HL),A ; clear memory address  
1032 23 INC HL ; next memory address, one lower  
1033 77 LD (HL),A ; clear memory address
```

Dat dan aan de bovenste code toevoegen:

...

```
0B74 110C6A LD DE,#6A0C ; load DE with sprite destination  
0B77 010C00 LD BC,#000C ; 3 items x 4 bytes = 12 bytes (#0C)  
0B7A EDB0 LDIR ; draw bonus item sprites  
;  
0B7C OF RRCA ; roll a right again. is this an odd number ?  
0B7D 3805 JR C,#0B84 ; yes, skip next steps  
;  
0B7F 213276 LD HL,#7632 ; video memory to clear for even rng  
0B82 1803 JR #0B87 ; skip next step  
;  
0B84 211576 LD HL,#7615 ; video memory to clear for odd rng  
;  
0B87 47 LD B,A ; save a for later  
0B88 3E10 LD A,#10 ; load a with space character  
0B8A 77 LD (HL),A ; clear memory address  
0B8B 23 INC HL ; next memory address, one lower  
0B8C 77 LD (HL),A ; clear memory address  
0B8D 78 LD A,B ; restore a  
;  
0B8E OF RRCA ; roll a right again. is this an odd number ?  
0B8F 3804 JR C,#0B95 ; yes, skip next steps  
;  
0B91 214A0B LD HL,#0B4A ; start of adapted table data for even rng  
0B94 C9 RET ; jump back  
;  
0B95 210C3E LD HL, #3EOC ; start of original table data for odd rng  
0B98 C9 RET ; jump back
```

Dat werkt. Nog wel testen of de rng een beetje goed random is en anders nog andere bits of andere RngTimer of FrameCounter gebruiken.

Nu versie v0.28 opgeslagen.

Getest met 15 games en redelijk gelijke verdeling. Dus rng zo houden.

Nu nadenken wat te doen voor rivets. Wat zou kunnen is standaard een afwijkende girder afstand en dan even en oneven een gespiegelde hamer configuratie en items positie. Dan zou je het patroon andersom kunnen/moeten doen.

In ieder geval de girder met de bovenste hamer één naar beneden:

```
3CD1: 05 B7 30 48 30 ; girder above kong  
3CD6: 05 CF 58 30 58 ; girder kong stands on  
3CDB: 05 D7 80 28 80 ; level 4 girder  
3CEO: 05 DF A8 20 A8 ; level 3 girder  
3CE5: 05 E7 D0 18 D0 ; level 2 girder  
3CEA: 05 EF F8 10 F8 ; bottom level girder
```

Dat is de level 4 girder. Deze wordt dan:

```
3CDB: 05 D7 88 28 88 ; level 4 girder
```

Level 3 girder ook naar beneden en de level 2 girder naar boven:

```
3CEO: 05 DF B0 20 B0 ; level 3 girder  
3CE5: 05 E7 C8 18 C8 ; level 2 girder
```

Dan de ladders vanaf beneden goed gaan maken.

De onderste drie ladders: de bovenkant omhoog:

3C90: 00 7B D0 7B F8 ; bottom center ladder	-> 00 7B C8 7B F8
3CC7: 00 1B D0 1B F8 ; bottom left ladder	-> 00 1B C8 1B F8
3CCC: 00 E3 D0 E3 F8 ; bottom right ladder	-> 00 E3 C8 E3 F8

Daarna de tweede laag vier ladders: bovenkant omlaag, onderkant omhoog:

3CB3: 00 23 A8 23 D0 ; level 2 ladder left side	-> 00 23 B0 23 C8
3CB8: 00 5B A8 5B D0 ; level 2 ladder #2 of 4	-> 00 5B B0 5B C8
3CBD: 00 A3 A8 A3 D0 ; level 2 ladder #3 of 4	-> 00 A3 B0 A3 C8
3CC2: 00 DB A8 DB D0 ; level 2 ladder right side	-> 00 DB B0 DB C8

Daarna de derde laag drie ladders: bovenkant omlaag, onderkant omlaag:

3C8B: 00 7B 80 7B A8 ; center ladder level 3	-> 00 7B 88 7B B0
3CA9: 00 2B 80 2B A8 ; level 3 ladder left side	-> 00 2B 88 2B B0
3CAE: 00 D3 80 D3 A8 ; level 3 ladder right side	-> 00 D3 88 D3 B0

Daarna de vierde laag vier ladders: onderkant omlaag:

3C95: 00 33 58 33 80 ; top left ladder	-> 00 33 58 33 88
3C9A: 00 53 58 53 80 ; top left ladder (right side)	-> 00 53 58 53 88
3C9F: 00 AB 58 AB 80 ; top right ladder (left side)	-> 00 AB 58 AB 88
3CA4: 00 CB 58 CB 80 ; top right ladder	-> 00 CB 58 CB 88

Ziet er nu zo uit:



Wellicht mooier en interessanter om te spelen als de tweede girder naar beneden verplaatst wordt, zodat de ruimte tussen 1-2 en 2-3 omgedraaid wordt. Dus level 2 girder 2 keer naar beneden:

3CE5: 05 E7 E0 18 E0 ; level 2 girder

En de ladders dan ook aanpassen:

De onderste drie ladders: de bovenkant twee omlaag:

3C90: 00 7B D0 7B F8 ; bottom center ladder	-> 00 7B E0 7B F8
3CC7: 00 1B D0 1B F8 ; bottom left ladder	-> 00 1B E0 1B F8
3CCC: 00 E3 D0 E3 F8 ; bottom right ladder	-> 00 E3 E0 E3 F8

Daarna de tweede laag vier ladders: bovenkant omlaag, onderkant twee omhoog:

3CB3: 00 23 A8 23 D0 ; level 2 ladder left side	-> 00 23 B0 23 E0
3CB8: 00 5B A8 5B D0 ; level 2 ladder #2 of 4	-> 00 5B B0 5B E0
3CBD: 00 A3 A8 A3 D0 ; level 2 ladder #3 of 4	-> 00 A3 B0 A3 E0
3CC2: 00 DB A8 DB D0 ; level 2 ladder right side	-> 00 DB B0 DB E0

Ziet er nu zo uit:



Nu nog aanpassen: 1) plaats van de rivets 2) verwijderen rivets, 3) plaats spawnen fireballs, 4) plaats hamers en plaats bonus items.

De rivets worden hier gedefinieerd:

```
; start of table data for rivets used above  
; these are addresses in video RAM for the rivets
```

0D17	CA	76	; #76CA
0D19	CF	76	; #76CF
0D1B	D4	76	; #76D4
0D1D	D9	76	; #76D9
0D1F	2A	75	; #752A
0D21	2F	75	; #752F
0D23	34	75	; #7534
0D25	39	75	; #7539

De bovenste rivets op de vierde rij zijn nog goed.

De rivets op de derde rij moeten één naar beneden.

De rivets op de tweede rij moeten één naar beneden.

De rivets op de eerste rij moeten twee naar beneden.

```

0D17 CA 76      ; #76CA      ; left 4th row
0D19 D0 76      ; #76D0      ; left 3rd row
0D1B D5 76      ; #76D5      ; left 2nd row
0D1D DB 76      ; #76DB      ; left 1st row
0D1F 2A 75      ; #752A      ; right 4th row
0D21 30 75      ; #7530      ; right 3rd row
0D23 35 75      ; #7535      ; right 2nd row
0D25 3B 75      ; #753B      ; right 1st row

```

De rivets zitten nu op de goede plaats. Nog niet helemaal duidelijk hoe het detecteren van het verwijderen van de rivets precies werkt. Werkt voor sommige wel en voor sommige niet.

Eerst het spawnen van de fireballs aanpassen aan de nieuwe hoogte van de girders.

```

; Possible X and Y positions to spawn a fireball on the right side of the screen
; First value is X position, 2nd value is Y position

; 3AC4: EE F0 ; bottom, right
; 3AC6: DB A0 ; middle, right
; 3AC8: E6 C8 ; 2nd from bottom, right
; 3ACA: D6 78 ; 2nd from top, right
; 3ACC: EB F0 ; unused?
; 3ACE: DB A0 ; unused?
; 3AD0: E6 C8 ; unused?
; 3AD2: E6 C8 ; unused?

; Possible X and Y positions to spawn a fireball on the left side of the screen
; First value is X position, 2nd value is Y position

; 3AD4: 1B C8 ; 2nd from bottom, left
; 3AD6: 23 A0 ; middle, left
; 3AD8: 2B 78 ; 2nd from top, left
; 3ADA: 12 F0 ; bottom, left
; 3ADC: 1B C8 ; unused?
; 3ADE: 23 A0 ; unused?
; 3AE0: 12 F0 ; unused?
; 3AE2: 1B C8 ; unused?

```

Aanpassen:

3AC4: EE F0 ; bottom, right	-> EE F0
3AC6: DB A0 ; middle, right	-> D8 A8
3AC8: E6 C8 ; 2nd from bottom, right	-> E6 D8
3ACA: D6 78 ; 2nd from top, right	-> D6 80

3AD4: 1B C8 ; 2nd from bottom, left	-> 1B D8
3AD6: 23 A0 ; middle, left	-> 23 A8

3AD8: 2B 78 ; 2nd from top, left -> 2B 80
3ADA: 12 F0 ; bottom, left -> 12 F0

Nu versie v0.29 opgeslagen.

Presents aanpassen. Alleen het rechter present moet naar beneden verschoven worden. De andere staan op de bovenste en de onderste girder die niet verschoven zijn.

```
; bonus items for rivets

3E54  DB 73 0A C8      ; hat on rivets at DB,C8
3E58  93 74 0A F0      ; purse on rivets at 93,F0
3E5C  33 75 0A 50      ; umbrella on rivets at 33,50
```

Aanpassen:

3E54 DB 73 0D D8 (kleur was al aangepast).

Hamers aanpassen. Beide hamers moeten naar beneden.

```
3E14:  1B 8C 7C 64      ; for rivets
```

Aanpassen:

3E14: 1B 94 7C 6C ; for rivets

Nu versie v0.30 opgeslagen.

Nu gaan kijken hoe het verwijderen van de rivets gedaan wordt. In ieder geval in eerste instantie gebaseerd op de X-waarde (linker rij rivets of rechter rij rivets). Daarna moet ook iets met de Y-positie gedaan worden. Is niet duidelijk hoe dat precies werkt.

De code daarvoor begint bij #1A33.

Eerst maar eens middels een debug bekijken wat er precies gedaan wordt met de Y-positie van jumpman. De Y-positie wordt in #1A55 ingelezen. Daar een watchpoint zetten. Gedaan, maar die code is ongelooflijk ingewikkeld. Checkt op een aantal bits aan de linkerkant om zodoende de positie te bepalen en te checken welke rivet overgestoken wordt. Maar dat werkt wel bij een gelijkelijke verdeling van de girders, maar niet meer voor de nieuwe situatie met verschoven girders.

Wellicht te vervangen door check code die we zelf maken: checken op vier Y-positie ranges en dat vertalen naar de juiste index in de rivets array.

Vanaf #6292 t/m #6299 is er een tabel waarin de status van de rivets bijgehouden wordt. Aan het einde van de check code moet HL het adres van de betreffende rivet hebben. De volgorde van de rivets in de tabel maakt eigenlijk niet zo heel veel uit. Als maar consequent dezelfde volgorde aangehouden wordt.

Mario Y-positie ranges op de girders met rivets:

Top rivets: 50 (staan) – 42 (sprong)
Derde rivets: 80 (staan) – 72 (sprong)
Tweede rivets: A8 (staan) – 9A (sprong)
Onderste rivets: D8 (staan) – CA (sprong)

Vanaf #1A55 t/m 1A83 zit de check op de rivets (welke en of al verwijderd). Als te laag dan return en als al verwijderd dan return. Anders HL vullen met index in rivet status tabel (#6292-#6299) en verder gaan op #1A84.

```
1A55 3A0562 LD    A,#6205      ; load A with Mario's Y position
1A58 FEDA   CP    #DA          ; compare with #DA. is mario too low to go over a rivet?
1A5A D0     RET   NC          ; yes, return
;
1A5B 0600 LD    B,#00          ; initialize table index
;
; Check bottom level rivets
;
1A5D FECA   CP    #CA          ; compare with #CA (when jumping)
1A5F 3807 JR    C,#1A68      ; higher, no bottom level rivet – jump to label AAAA
1A61 FED9   CP    #D9          ; compare with #D9 (when standing)
1A63 3003 JR    NC,1A68      ; lower, no bottom level rivet – jump to label AAAA
1A65 C39A0B JP    #0B9A      ; yes, bottom level rivet – jump to label EEEE
;
1A68 04     INC   B           ; increase table index – label AAAA
;
; Check second level rivets
;
1A69 FE9A   CP    #9A          ; compare with #9A (when jumping)
1A6B 3807 JR    C,#1A74      ; higher, no second level rivet – jump to label BBBB
1A6D FEA9   CP    #A9          ; compare with #A9 (when standing)
1A6F 3003 JR    NC,1A74      ; lower, no second level rivet – jump to label BBBB
1A71 C39A0B JP    #0B9A      ; yes, second level rivet – jump to label EEEE
;
1A74 04     INC   B           ; increase table index – label BBBB
;
; Check third level rivets
;
1A75 FE72   CP    #72          ; compare with #72 (when jumping)
1A77 3807 JR    C,#1A80      ; higher, no third level rivet – jump to label CCCC
1A79 FE81   CP    #81          ; compare with #81 (when standing)
1A7B 3003 JR    NC,1A80      ; lower, no third level rivet – jump to label CCCC
1A7D C39A0B JP    #0B9A      ; yes, third level rivet – jump to label EEEE
;
1A80 C3990B JP    #0B99      ; jump to additional code – jump to label XXXX – label CCCC
1A83 00     NOP            ;
```

en

```
0B99 04      INC   B          ; increase table index – label XXXX
;
0B9A 219262  LD    HL,#6292   ; load HL with start of table of rivets – label EEEE
;
0B9D 0E00      LD    C,#00     ; load C with left index
0B9F 3A0362  LD    A,(#6203) ; load A with jumpmans X position
0BA2 FE80      CP    #80     ; left rivets?
0BA4 3802      JR    C       ; yes skip next steps
0BA6 0E04      LD    C,#04     ; load C with right index
;
0BA8 79      LD    A,C     ; load A with leftright index
0BA9 80      ADD   A,B     ; add table index
0BAA 85      ADD   A,L     ; add #92
0BAB 6F      LD    L,A     ; copy to L
;
0BAC 7E      LD    A,(HL)   ; get the status of the rivet jumpman is crossing
0BAD A7      AND   A       ; has this rivet already been traversed?
0BAE C8      RET   Z       ; yes, return
;
0BAF C3841A  JP    #1A84   ; process rivet traversed
```

De rivets moeten nog wel goed weggehaald worden. Nu niet de juiste verwijderd.

Het wissen moet ook aangepast worden. Scheiden links en rechts en daarna op basis van index een offset berekenen en dan het videomemory op basis van index bepalen en rivet wissen.

Nieuwe code om te wissen kan vanaf #1A8A geplaatst worden. Er is ruimte t/m #1A99. Daar begint de rest van de afhandeling qua bijwerken van de score en geluid.

```
0BB2 0C 08 04 01      ; table with video offset
```

en

```
1A8A 21B20B  LD    HL,#0BB2   ; load HL with table with video intervals
1A8D 78      LD    A,B     ; load A with table index
1A8E 85      ADD   A,L     ; add table index to HL
1A8F 6F      LD    L,A     ; load L with A
1A90 46      LD    B,(HL)   ; load B with the correct video offset
;
1A91 210076  LD    HL,#7600   ; load HL with video start address for left (top rivet)
1A94 79      LD    A,C     ; load A with leftright index.
1A95 FE00      CP    #00     ; traversed left rivet?
1A97 2803      JR    Z,XX   ; yes, skip next step
1A99 210075  LD    HL,#7500   ; load HL with video start address for right
;
1A9C 78      LD    A,B     ; load A with the video offset
```

```
1A9D 85      ADD   A,L          ; add video offset to HL  
1A9E 6F      LD    L,A  
;  
1A9F 000000  NOP,NOP,NOP
```

Dat werkt allemaal.

Nu versie v0.31 opgeslagen.

Dan nu rivets zo maken dat op de oneven levels de originele plaatsing van de items en hamers is en op de even levels het omgekeerde (soort van afdwingen omgekeerd patroon).

Plaatsen van de hamers voor rivets gebeurt hier:

```
113D 21143E  LD    HL,#3E14      ; load HL with start of table data for hammer locations  
1140 CDA611  CALL  #11A6      ; draw the hammers
```

De reeds aangepaste definitie van de hamers voor rivets:

3E14: 1B 94 7C 6C ; for rivets

Aanpassen:

```
113D CDBA0B  CALL  #0BBA      ; call additional code – call label AAAA
```

en

```
0BB6 E3 94 79 6C      ; alternative hammer location rivets  
;  
0BBA 3A2962  LD    A,(#6229)    ; load A with level # - label AAAA  
0BBD 0F      RRCA          ; roll a right . is this an odd level ?  
0BBE 3804  JR    C,#0BC4      ; yes, skip next steps  
;  
0BC0 21B60B  LD    HL,#0BB6    ; start of adapted table data for even levels  
0BC3 C9      RET             ; jump back  
;  
0BC4 21143E  LD    HL, #3E14    ; start of original table data for odd levels  
0BC7 C9      RET             ; jump back
```

Hetzelfde doen voor de cadeau's.

Aangepaste bonus items voor rivets:

3E54: DB 73 0D D8

3E58: 93 74 0D F0

3E5C: 33 75 0D 50

Alternatieve definitie maken en die laden.

Die doen in het laatste stuk waar het logo staat. Logo verwijderen en vrijmaken.

3FC8: 23 73 0D D8
3FCC: 63 74 0D F0
3FD0: CB 75 0D 50

Plaatsen van de items voor rivets gebeurt hier:

```
1143 21543E LD HL,#3E54 ; load HL with start of bonus items for rivets
1146 110C6A LD DE,#6A0C ; set destination sprite address
1149 010C00 LD BC,#000C ; set counter to #C bytes to copy
114C EDB0 LDIR ; draw purse, umbrella, hat to screen
```

Aanpassen:

1143 CDC80B JP #0BC8 ; jump to additional code

en

```
0BC8 3A2962 LD A,(#6229) ; load A with level # - label AAAA
0BCB 0F RRCA ; roll a right . is this an odd level ?
0BCC 3804 JR C,#0BD2 ; yes, skip next steps
;
0BCE 21C83F LD HL,#3FC8 ; start of adapted table data for even levels
0BD1 C9 RET ; jump back
;
0BD2 21543E LD HL, #3E54 ; start of original table data for odd levels
0BD5 C9 RET ; jump back
```

Nu versie v0.32 opgeslagen.

Nu nog jumpman laten starten op een andere locatie.

Waar wordt dat geregeld? Niet zo snel te vinden in de code. Eventjes debuggen met watchpoint op 6203 (jumpmans X-position). Gevonden, wordt hier gedaan:

```

; set initial mario sprite position and draw remaining lives and level

123C DF      RST    #18          ; count down WaitTimerMSB and only continue when 0
123D 3A2762 LD     A,(#6227)   ; load a with screen number
1240 fe03 CP     #03          ; is this the elevators?
1242 0116E0 LD     BC,#e016   ; B := #E0, C := #16. used for X,Y coordinates
1245 cA4B12 JP     Z,#124B   ; if elevators skip next step

1248 013FF0 LD     BC,#F03F   ; else load alternate coordinates for elevators

124B DD210062 LD     IX,#6200   ; set IX to mario sprite array
124F 214C69 LD     HL,#694C   ; load HL with address for mario sprite X value
1252 DD360001 LD     (IX+#00),#01 ; turn on sprite
1256 DD7103 LD     (IX+#03),C  ; store X position

```

Uitzondering voor elevators. En daar dan ook een uitzondering voor rivets even levels bij maken.

Aanpassingen:

```

1240 013FF0 LD     BC,#F03F   ; load BC with default Y and X positions jumpman
1243 C3D43F JP     #3FD4      ; jump to additional code – jump to label AAAA
1246-124A 5xNOP

```

en

```

3FD4 FE03 CP     #03          ; is this the elevators? – label AAAA
3FD6 2006 JR     NZ,#3FDE   ; no, skip next steps – jump to label BBBB
;
3FD8 0116E0 LD     BC,#E016   ; load BC with alternative positions for elevators
3FDB C34B12 JP     #124B      ; jump back
;
3FDE FE04 CP     #04          ; is this the rivets? – label BBBB
3FE0 C24B12 JP     NZ,#124B   ; no, use default position, jump back
;
3FE3 3A2962 LD     A,(#6229)   ; load A with level #
3FE6 0F RRCA
3FE7 DA4B12 JP     C,#124B   ; yes, use default position, jump back
;
3FEA 01C1F0 LD     BC,F0C1   ; load BC with alternate positions for even rivets
3FED C34B12 JP     #124B      ; jump back

```

Gaat goed. Omgekeerd veld is leuk.

Nu versie v0.33 opgeslagen.

Grappige constatering. Je kan vanaf een elevator springen naar de rand en als je dat goed doet, dan wordt je teruggekaatst en land je weer op de elevator eronder. Een cadeau ophangen zodat je daarnaartoe kan springen?

De items van de elevator (geen verschil even/oneven levels):

3E48: 93 73 0D D0

3E4C: E3 74 0D 60

3E50: 16 75 0D A4

Door deze aanpassing is het cadeau nu zwevend. Wel lastig te pakken. Moet precieze X,Y combinatie zijn. Wellicht iets makkelijker te maken door ook Y-1 en Y+1 toe te staan voor pakken van een item?

Eerst eventjes testen met elevator die naar beneden beweegt. Ja daar kan het ook mee. De vraag is of het ook werkt met langzamere elevators. Komen al in level 02 voor. Eventjes testen. Ja dat maakt niet uit. Snelheid van de elevators is op alle levels hetzelfde.

Dit is wel een hele leuke toevoeging. Je zou die sprite dan in een ballon kunnen veranderen. Ballon is sprite 69. Kleurcode 00 = rode balon, kleurcode 02 = blauwe ballon, kleurcode 08 = oranje ballon.

De test voor de Y-waarde bij de bonus item pickup zit hier:

```
19ED 3A0562 LD A, (#6205) ; load A with Mario's Y position
19F0 2C INC L
19F1 2C INC L
19F2 2C INC L ; get HL to point to Y position of bonus item
19F3 BE CP (HL) ; are they equal?
19F4 C0 RET NZ ; no, return from this test
```

Aanpassen:

```
19F0 C30D38 JP #380D ; jump to additional code – jump to label AAAA
19F3 0000 NOP,NOP
```

en

```
380D 2C INC L ; label AAAA
380E 2C INC L
380F 2C INC L ; get HL to point to Y position of bonus item
3810 BE CP (HL) ; are they equal?
3811 2809 JR Z,381C ; yes, process item – jump to label BBBB
3813 3D DEC A ; check larger area
3814 BE CP (HL) ; are they equal?
3815 2805 JR Z,381C ; yes, process item – jump to label BBBB
3817 3D DEC A ; check larger area
3818 BE CP (HL) ; are they equal?
```

```

3819 2801    JR      Z,381C      ; yes, process item – jump to label BBBB
381B C9      RET                 ; return, no bonus item picked up
;
381C C3F519   JP      #19F5      ; jump back to process item – label BBBB

```

Hier voor ruimte vrijmaken: #380D t/m 3849

Gaat nog niet goed. Misschien meer testen doen. Testen in een loopje en dan B als loop variabele om het aantal testen (omvang van de testrange) te kunnen varieren.

```

380D 2C      INC     L          ; label AAAA
380E 2C      INC     L
380F 2C      INC     L          ; get HL to point to Y position of bonus item
;
3810 0608    LD      B,#08      ; load B with range to check
;
3812 BE      CP      (HL)      ; are they equal? – label BBBB
3813 CAF519   JP      Z,19F5      ; yes, process item – jump back
3816 3C      INC     A          ; increment to next Y position in range
;
3817 10F9      DJNZ    #3812      ; next B – jump to label BBBB
;
3819 C9      RET                 ; return, no bonus item picked up

```

Gaat niet goed. Pakte de ballon niet op. Uit debugging blijkt dat het oppakken gecheckt wordt op een bit van de sprite value.

Maar nu sprite 69 (0110 1001) gebruikt i.p.v. 75 (0111 0101). Heeft inderdaad een andere waarde voor bit 3. Dus voor de ballon een spritepositie gebruiken met een 1 op bit 3. En omdat we de sprite willen animeren, moet het een vrij blok zijn met 4 sprites (eindigend op 00, 01, 10 en 11).

Eerst even testen als we de ballon sprite kopiëren naar de oorspronkelijke positie 75. Werkt het pakken dan wel? Ja dan werkt het wel.

Ballon sprites op:

```

#54: 0101 0100
#55: 0101 0101
#56: 0101 0110
#57: 0101 0111

```

Plaatsing ballonnen op andere velden:

Definities bonus items van barrels stage.

```

OB4E: 24 73 0D B3
OB52: DB 74 0D 92
OB56: E8 54 02 C6

```

0B5A: 6D 73 0D 4C

0B5E: 18 74 0D F0

0B62: E8 54 02 84

Definitie bonus items van pies stage:

3E3C: 53 73 0D A0

3E40: BB 74 0D F0

3E44: A0 54 02 6B

Definities bonus items rivets:

3E54: DB 73 0D D8

3E58: 93 74 0D F0

3E5C: 16 54 02 60

3FC8: 23 73 0D D8

3FCC: 63 74 0D F0

3FD0: E8 54 02 77

Nu versie v0.34 opgeslagen.

De tekst HOW HIGH CAN YOU GET ? veranderen in CELEBRATE A PARTY !!

Net zo lang: Aanpassen op #36CC:

5E 77 13 15 1C 15 12 22 11 24 15 10 11 10 20 11 22 24 29 10 36 3F

COME, JOIN THE PARTY !!

5E 77 13 1F 1D 15 43 10 1A 1F 19 1E 10 24 18 15 10 20 11 22 24 29 10 36 3F

De meteraanduiding weghalen.

Dat gebeurt hier:

```

0C54 DD2AA863 LD IX,(#63A8) ; load IX with screen VRAM address to draw number of meters
0C58 4F LD C,A ; C := A, used for offset
0C59 0600 LD B,#00 ; B := 0
0C5B 09 ADD HL,BC ; add offset
0C5C 7E LD A,(HL) ; get table data
0C5D DD7760 LD (IX+#60),A ; write to screen
0C60 23 INC HL ; next
0C61 7E LD A,(HL) ; get data
0C62 DD7740 LD (IX+#40),A ; write to screen
0C65 23 INC HL ; next
0C66 7E LD A,(HL) ; get table data
0C67 DD7720 LD (IX+#20),A ; write to screen
0C6A DD36E08B LD (IX-#20),#8B ; write "m" to screen
0C6E C1 POP BC ; restore BC
0C6F DDE5 PUSH IX ; transfer IX to HL (part 1/2)
0C71 E1 POP HL ; transfer IX to HL (part 2/2)
0C72 11FCFF LD DE,#FFFC ; load offset for next screen location
0C75 10 ADD HL,BC ; add offset

```

De vier 'write to screen' in NOP's veranderen:

```

0C5D 000000 NOP, NOP, NOP
0C62 000000 NOP, NOP, NOP
0C67 000000 NOP, NOP, NOP
0C6A 000000 NOP, NOP, NOP

```

Nu versie v0.35 opgeslagen.

De yell van Pauline aanpassen naar Join!

Sprites zijn aangepast.

Nu versie v0.36 opgeslagen.

De ballonnen op het titelscherm van kleur veranderen:

```

3F3C C0 66 01 A4 ; first sprite - 4
3F40 CF 67 01 A4 ; second sprite - 0
3F44 50 56 02 C8 ; left balloon
3F48 B0 56 02 C8 ; right balloon

```

En de ballon in de end sequence ook van kleur veranderen:

```

3F71: 80 68 0A B4 ; sprite data for party hat
3F75: 78 66 01 98 ; sprite data for 4
3F79: 88 67 01 98 ; sprite data for 0
3F7D: 6A 6A 0D C8 ; sprite data for small present
3F81: 98 69 02 BD ; sprite data for blue balloon

```

Ja dat is leuker.

De rechter vuurwerk sparkle sprite iets naar links verplaatsen.

```
1891 21246A LD HL,#6A24 ; load HL with sprite address used for sparkle
1894 3650 LD (HL),#50 ; set sprite X value
1896 2C INC L ; next
1897 3663 LD (HL),#63 ; set sprite value
1899 2C INC L ; next
189A 3601 LD (HL),#01 ; set sprite color
189C 2C INC L ; next
189D 3698 LD (HL),#98 ; set sprite Y value
```

Lastig want is één sprite en naar de andere kant d.m.v. een bit flip met een XOR. Dus links ook aanpassen. Klein beetje schuiven. Met waarde 4F voor de sprite X value gaat het goed:

```
1894 364F LD (HL),#4F ; set sprite X value
```

Zien er nu zo uit:



Nu versie v0.36 opgeslagen.

Dan nu kijken toevoegen animatie voor de ballonnen.

```
#54: 0101 0100
#55: 0101 0101
#56: 0101 0110
#57: 0101 0111
```

De ballon is altijd het derde bonus item. Sprite value van deze zit op:

Laten afhangen van bits in de FrameCounter.

Op #19C2 is er in de main routine nog een ‘slot’ vrij voor een aanvullende routine.

```
19C2 CD1A38 CALL #381A ; call additional routine – call label AAAA
```

en

```
381A 3A1A60 LD A,(FrameCounter) ; load A with framecounter – label AAAA
;
381D CB7F BIT 7,A ; check bit 7 of framecounter
381F 3A156A LD A,(#6A15) ; load A with sprite value balloon
;
3822 2804 JR Z,#BBBB ; if bit 7 = 0 – skip next step – jump to label BBBB
3824 E6FD AND #FD ; clear bit 1 of sprite value
3826 1802 JR #XXXX ; skip next step
;
3828 F602 OR #02 ; set bit 1 of sprite value – label BBBB
;
382A 32156A LD (#6A15),A ; store sprite value balloon
;
382D 3A1A60 LD A,(FrameCounter) ; load A with framecounter – label AAAA
;
3830 CB77 BIT 6,A ; check bit 6 of framecounter
3832 3A156A LD A,(#6A15) ; load A with sprite value balloon
;
3835 2804 JR Z,#CCCC ; if bit 6 = 0 – skip next step – jump to label CCCC
3837 E6FE AND #FE ; clear bit 0 of sprite value
3839 1802 JR #XXXX ; skip next step
;
383B F601 OR #01 ; set bit 0 of sprite value – label CCCC
;
383D 32156A LD (#6A15),A ; store sprite value balloon
;
3840 C9 RET ; return
```

Gaat goed, maar moet sneller. Lager bits van de framecounter testen.

```
381D CB77 BIT 6,A
3830 CB67 BIT 4,A
```

Gaat goed. Is erg leuk.

Nu versie v0.37 opgeslagen.

Heart sprite veranderen in de tekst PARTY?. De gebroken heart sprite veranderen in vraagtekens.

De kleur van de heart sprite wordt hier gezet:

```
1714 3609 LD (HL),#09 ; set heart sprite color
```

Aanpassen:

```
1714 3601 LD (HL),#01 ; set heart sprite color
```

De attract mode aanpassen zodat de attract mode ook de ballon gaat pakken.

```
; data used in sub below for attract mode movement  
; first byte is movement, second is duration
```

```
21D1 80 FE ; jump  
21D3 01 C0 ; run right  
21D5 04 50 ; up = climb ladder  
21D7 02 10 ; run left  
21D9 82 60 ; jump left  
21DB 02 10 ; run left  
21DD 82 CA ; jump left  
21DF 01 10 ; run right  
21E1 81 FF ; jump right (gets hammer)  
21E3 02 38 ; run left  
21E5 01 80 ; run right - mario dies falling over right edge  
21E7 02 FF ; run left  
21E9 04 80 ; up  
21EB 04 60 ; up  
21ED 80 ; ?
```

```
80  
FE 01 ; run right  
80 04 ; climb ladder  
18 01 ; run right  
10 81 ; jump right (grab balloon)  
48 02 ; run left  
10 82 ; jump left (grab hammer)  
FF 02 ; run left  
FF 00 ; stand still  
80 02 ; run left  
20 81 ; jump right  
10 82 ; jump left  
FF 01 ; run right (keep going right until end of girder)  
FF 01 ; run right
```

Een aantal keren testen met invinciblility uit. Ja gaat goed.

Nu versie v0.38 opgeslagen.

Wellicht een idee om na de L=01 ook het screen aan te geven:

L=01-1 / L=01-2 / L=02-1 / L=02-2 / L=02-3 / L=03-1 / L=03-2 / L=03-3 / L=03-4 / ... / ...

Waar wordt de L=xx getekend? Dat gebeurt hier.

```
06D7 210375 LD HL,#7503 ; load HL with screen location for "L="
06DA 361C LD (HL),#1C ; draw "L"
06DC 21E374 LD HL,#74E3 ; next location
06DF 3634 LD (HL),#34 ; draw "="
06E1 3A2962 LD A,(#6229) ; load A with level #
06E4 fe64 CP #64 ; level < #64 (100 decimal) ?
06E6 3805 JR c,#06Ed ; yes, skip next 2 steps

06E8 3E63 LD A,#63 ; otherwise A := #63 (99 decimal)
06Ea 322962 LD (#6229),A ; store into level #

06Ed 010Aff LD BC,#ff0A ; B: = #FF, C := #0A (10 decimal)

06f0 04 INC b ; increment B
06f1 91 SUB c ; subtract 10 decimal
06f2 d2f006 JP NC,#06f0 ; not carry, loop again (counts tens)

06f5 81 ADD A,C ; add 10 back to A to get a number from 0 to 9
06f6 32A374 LD (#74A3),A ; draw level to screen (low byte)
06f9 78 LD A,b ; load a with b (number of tens)
06fa 32C374 LD (#74C3),A ; draw level to screen (high byte)
06fd c9 RET ; return
```

622E is het aantal Goofy Kongs dat getekend moet worden: dat is een indicatie voor het schermnummer binnen een level.

Aanpassing:

```
06FA C3B438 JP #38B4 ; jump to additional code – jump to label AAAA
en
38B4 32C374 LD (#74C3),A ; draw level to screen (high byte) – label AAAA
;
38B7 3E2C LD A,#2C ; load A with "-"
38B9 328374 LD (#7483),A ; draw hyphen
;
38BC 3A2E62 LD A,(#622E) ; load A with number of Goofy Kongs to draw
38BF 3C INC A ; increment A
38C0 326374 LD (#7463),A ; draw screen in level number
;
38C3 C9 RET ; return
```

Eventjes testen met start vanaf begin.

```
095E 01 65 3A 01 00 00 00 ; #3A65 is start of table data for screens/levels
```

Gaat niet helemaal goed. Geeft een getal te hoog op. Dus die increment moet weg. Kan door 38BF NOP te maken. Ja, gaat dan wel goed.

Goofy Kong plaatje aanpassen. Een bordje met 40 in handen geven:



Gemaakt en ziet er goed uit.

Nu versie v0.39 opgeslagen.

Beter idee voor de tekst 'Party!!' en de vraagtekens aan einde van de boards. Een rode ballon weergeven en die laten spatten.

Sprites aangepast. Alleen de kleuren nog goedzetten.

De kleur van de heart sprite wordt hier gezet:

```
1714 3609 LD (HL),#09 ; set heart sprite color
```

Aanpassen:

```
1714 3600 LD (HL),#00 ; set heart sprite color
```

Ja dat ziet er goed uit. En ook erg leuk. Leuker dan de tekst. Dit is veel aansprekender.

Bij de stapel Goofy Kongs ook nog twee ballonnen tekenen?

Kan met deze code:

```

38C4 21D538      LD    HL,#38D5      ; source - has the information about the barrels
38C7 11A869      LD    DE,#69A8      ; destination = sprites
38CA 010800      LD    BC,#0008      ; counter is #08
38CD EDB0        LDIR             ; draws the balloons
;
38CF 210960      LD    HL,WaitTimerMSB ; load HL with timer to wait
;
38D2 C38B0C       JP    #0C8B        ; jump back
;
38D5 50 56 02 D8 ; left balloon
38D9 B0 56 02 D8 ; right balloon

```

En dat dan aanroepen vanuit het tekenen van het How High scherm:

```

0C82 110703      LD    DE,#0307      ; load task data for text #7 "HOW HIGH CAN YOU GET?"
0C85 CD9F30       CALL   #309F        ; insert task to draw text
0C88 210960      LD    HL,WaitTimerMSB ; load HL with timer to wait
0C8B 36A0        LD    (HL),#A0      ; set timer for #A0 units
0C8D 23          INC   HL            ; HL := GameMode2
0C8E 34          INC   (HL)         ;
0C8F 34          INC   (HL)         ; increase game mode twice - starts game
0C90 C9          RET               ; return

0C88 C3C438      JP    #38C4        ; jump to additional code

```

Ja dat ziet er leuk uit.

Nu versie v0.40 opgeslagen.

Kijken of bij de even pies levels de spawn van de fireballs aangepast kan worden.

Eerdere aanpassing:

```
3487 CD130B      CALL   #0B13        ; call additional code – jump to label AAAA
```

en

```

0AFB A8 A3 A0 9E 9C 9B 9A 99 98 98 97 97 97 98 98 99 9A 9B 9C 9D 9E 9F A0 AA
;
0B13 3A2962      LD    A,(#6229)    ; load A with level # - label AAAA
0B16 0F          RRCA             ; roll a right . is this an odd level ?
0B17 3804      JR    C,#0B1D      ; yes, skip next steps
;
0B19 21FB0A      LD    HL,#0AFB    ; start of adapted table data for even levels
0B1C C9          RET               ; jump back
;
0B1D 21AC3A      LD    HL, #3AAC    ; start of original table data for odd levels
0B20 C9          RET               ; jump back

```

Dat betekent dat er een extra random keuze moet komen bij even levels voor een derde spawn tabel met Y-posities (de boog die de fireballs beschrijven bij het spawnen).

De positie (het adres) van de positie in de tabel wordt opgeslagen in IX+1A (register L) en IX+1B (register H). Dus individueel per fireball wordt dit onthouden. Dus je kan in één level als het goed is meerdere tabellen door elkaar gebruiken. De random kant wordt bepaalt door RngTimer2, dus wellicht de random hoogte spawn laten bepalen door RngTimer1.

Aanpassen:

```
0B19 C3F538 JP #38F5 ; jump to additional code – jump to label AAAA
```

en

```
38DD 80 7B 78 76 74 73 72 71 70 70 6F 6F 6F 70 70 71 72 73 74 75 76 77 78 AA  
;  
38F5 3A1860 LD A,(RngTimer1) ; Determine level to spawn fireball – label AAAA  
38F8 CB47 BIT 0,A ; If timer value is even ...  
3F8A 2806 JR Z,#3F92 ; ... skip next steps – jump to label BBBB  
;  
3F8C 21DD38 LD HL,#38DD ; start of adapted table data variant 2  
3F8F C31C0B JP #0B1C ; jump back  
;  
3F92 21FB0A LD HL, #0AFB ; start of Adapted table data variant 1 – label BBBB  
3F95 C31C0B JP #0B1C ; jump back
```

Testen met level L=05 en level L=06.

Aanpassen start level:

```
095E 01 65 3A 01 00 00 00 ; #3A65 is start of table data for screens/levels
```

Als starten L=05 dan dit aanpassen naar 05 en 73 3A.

Ja dat gaat goed. Nu alleen de spawn boog nog aanpassen, zodat het lijkt alsof de vuurbal wel vanuit de onderste oil can gespawned wordt.

Aangepast naar:

```
38DD 90 88 80 74 72 70 6E 6C 6A 6A 69 69 68 69 69 6A 6C 6E 70 72 74 76 78 AA.
```

Geeft een mooie boog door het blauwe vat bovenin.

Nu versie v0.41 opgeslagen.

De tekst die Pauline roeft aanpassen naar Come!. Gedaan.

Nu versie v0.42 opgeslagen.

Nu de volgende zaken testen:

Verschillende boards op moeilijkheid vanaf level 5:

- Barrels is lastig, maar door steering wordt het wel makkelijker. Zelfde als anders, alleen hamers op ander plek en gat. Niet echt een probleem voor de gevorderde DK speler.
- Pies met spawn op lagere level is lastig, maar wachten en beneden hameren. Hopen dat ze aan een gunstige kant spawnen (link/rechts, beneden/boven) en dat er een weg vrij komt.
- Elevators is niet echt moeilijk. De omgekeerde elevator is wel wennen wanneer je meteen wilt wpringen. Maar de weg is niet echt heel moeilijk.
- Rivets is wel lastiger. Het normale bord is lastiger door de langere ladders. Het gespiegelde bord is lastig omdat je er aan moet wennen. Daarnaast gaan de fireballs best snel naar rechts. Dan waarschijnlijk strategie, dat je ze rechts laat spawnen (snel naar links lopen). Mogelijk afhankelijk van de plek waar de fireball spawned.
- De ballonnen hangen op interessante plekken en zijn niet heel eenvoudig te pakken. Zeker niet in elevators en rivets.

Doorlopen van de velden en juiste levelnr/boardnr weergave.

- Ja dat gaat allemaal goed. Goofy Kongs goed gestapeld en de boardnr's worden goed weergegeven.

Player I en Player II werking:

- Ja dat gaat goed.

Killscreen:

- Level L=22 geeft gewoon een Kill Screen zoals het hoort.

Nu de final versie v1.00 opgeslagen.

De ballonnen soms nog moeilijk te pakken. Wellicht makkelijker door de range aan te passen?

Dat is het stukje code:

Gaat nog niet goed. Misschien meer testen doen. Testen in een loopje en dan B als loop variabele om het aantal testen (omvang van de testrange) te kunnen varieren.

```
380D 2C      INC    L          ; label AAAA
380E 2C      INC    L
380F 2C      INC    L          ; get HL to point to Y position of bonus item
```

```

;
3810 0608 LD B,#08 ; load B with range to check
;
3812 BE CP (HL) ; are they equal? – label BBBB
3813 CAF519 JP Z,19F5 ; yes, process item – jump back
3816 3C INC A ; increment to next Y position in range
;
3817 10F9 DJNZ #3812 ; next B – jump to label BBBB
;
3819 C9 RET ; return, no bonus item picked up

```

Kijken of dit nog steeds zo is en of de range groter gemaakt kan worden. Met name in elevators en rivets iets makkelijker maken.

Range aangepast naar 10. Testen met de verschillende velden.

Aanpassen start level:

095E 01 65 3A 01 00 00 00 ; #3A65 is start of table data for screens/levels

Als starten L=05 dan dit aanpassen naar 05 en 73 3A.

Invincibility aan.

Ballonnen in barrels iets lager hangen. Daardoor iets makkelijker te pakken:

OB4E: 24 73 0D B3

OB52: DB 74 0D 92

OB56: E8 54 02 C8 – Y was C6

OB5A: 6D 73 0D 4C

OB5E: 18 74 0D F0

OB62: E8 54 02 86 – Y was 84

Bij de pies gaat het goed. Beide velden hetzelfde.

Bij de elevators gaat het goed. Beide velden hetzelfde. Maar omhoog met de elevator is het eenvoudiger om de ballon te pakken. Omlaag is het lastiger om weer op de elevator terug te landen.

Bij de rivets de ballon links hangt goed, maar de ballon rechts iets lager hangen.

3E54: DB 73 0D D8

3E58: 93 74 0D F0

3E5C: 16 54 02 60

3FC8: 23 73 0D D8

3FCC: 63 74 0D F0

3FD0: E8 54 02 79 – Y was 77

Nu weer de aanpassingen terugzetten:

Level/boards overzicht en startlevel L=01. En invincibility weer uit.

[Nu de final versie v1.01 opgeslagen.](#)