

# ZUHANÁS

verzió: 2023. január 9.

első kiadás: 2022. december 31.

## TVC ügyességi játék 32K, 64k és 64k+ gépekre

Készítette: Bertók Zsolt – BerySoft, 2022

Ez egy ügyességi játék **1 vagy 2 játékos** számára, amit a 2020-ban, **Commodore VIC-20**-ra megjelent, **FALL** című játék ihletett.

## A játék leírása

A TVC-s **ZUHANÁS** játék célja, hogy a sziklahasadékban zuhanó, közben a sziklafalon magát fékező játékos minél messzebbre jusson anélkül, hogy a sziklafalon feltűnő körfűrészek kárt tennének benne. Ennek érdekében a játékos a körfűrész elöl át tud rugaszkodni a szemben levő sziklafalra, és vissza.

A játékot lehet egyedül, **1 játékos** módban játszani, vagy egymással is versenyezve, **2 játékos** módban.

## A FRISSÍTETT VERZIÓ ÚJDONSÁGAI

- intró zene a játék elindulásakor (*Dragonborn – Skyrim theme* részlet)
- kis geg: a játék címfeliratába kapaszkodó játékos figura (Kiss Károly ötlete)
- törmelék animáció a sziklafalon lecsúszó játékosok kezeinél
- digitális hangeffektek ugrásnál és zuhanásnál, szimpla TVC csipogás helyett
- 100, 150 és 200 pontnál sötétedik a háttér
- 120 és 240 pont körül kis lélegzetvételnyi időt ad a játék
- ugró játékos sprite, bár ez a gyors ugrás miatt nem is igazán észlelhető
- a körfűrészek forognak tovább lezuhanás után is, új játék indítására várva
- gyorsabb sziklafal *scrollozás* (mármint *kódban gyorsabb*)
- az **ALT** (*Játékos 1*) és a **CTRL** (*Játékos 2*) gombokkal is lehet ugrani
- *cursor-megszakítás* figyelése helyett annak szimulálása *hang-megszakításból*
- ékezet nélküli játéknév (zuhanás.cas helyett zuhanas.cas)

## IRÁNYÍTÁS

A **JÁTEKOS 1** ugrás: **1. joystick TŰZ**, vagy **RETURN**, vagy **ALT** gombokkal.

A **JÁTEKOS 2** ugrás: **2. joystick TŰZ**, vagy **SZÓKÖZ**, vagy **CRTL** gombokkal.

# A JÁTÉK KELETKEZÉSE

A játék a 2022-es TVC JÁTÉKFEJLESZTŐ VERSENY-re készült, de szó szerint az utolsó pillanatban. Eredetileg egy sokkal összetettebb játékot kezdtem fejleszteni a versenyre, de annak is elég későn, november elején kezdtem neki, és közbejötték magánéleti dolgok is, így annak a befejezése reménytelennek látszott, így azt el is halasztottam 2023-ra.

De a 2022-es verseny december 31-ig kitolt határideje adott némi esélyt arra, hogy mégis induljak a megmérettetésen, bár amikor a határidő kitolásról döntöttünk, akkor még nem volt ilyen gondolatom. Végül, az utolsó 3 napban, napi 4-8 órában készült a játék, amit december 29-én kezdtem el, és december 31-én este fejeztem be.

Az eredeti VIC-20-as játékot már 2 éve ajánlgattam a TVC-s Facebook csoporttagoknak, hogy valaki készítse el TVC-re, de mivel senki nem kapott a lehetőségen, így én lettem az, aki nekiállt. A VIC-20-as játékban a két játékos külön-külön hasadékokban zuhan, de a TVC-n nincs sem gyors karakteres mód, sem 1-2 pixeles függőleges hardver *scrollozási* lehetőség, a szoftveres *scroll* pedig sajnos nem bizonyult elég gyorsnak, ezért gyors áttervezéssel egy helyre kerültek a játékosok. Csak a sziklafalak szélei vannak így is *scrollozva*, és a körfűrészek „*sprite*”-ként vannak mozgatva.

## TECHNIKAI ÉRDEKESSÉGEK

A TVC 4 bites digitális-analóg konverzióra képes, ha a hangjelet letiltjuk. Ekkor az amplitúdó értéknek (hangerő) megfelelő jelszint jelenik meg a TVC hangkimenetén, ilyen módon az amplitúdó-szabályozó áramkör durva, 4 bites digital / analóg konverterként működhet, azaz 4 bites digitális hang lejátszására lesz alkalmas.

A játék a TVC hang-megszakítását használja a játék közben digitális hangeffektek lejátszására. 4 KHz-es mintavételű digitalizált hangeffekteket játszik le a program, amikor a játékos ugrik, és amikor lezuhan. Ez azt jelenti, hogy a TVC ilyenkor másodpercenként durván 4000-szer hívja meg a megszakítást, ami persze át van irányítva egy saját rutinra, ami kiszolgálja ezt, és elvégzi a digitális hangminták lejátszását, ha szükséges. Mivel ez jelentősen megterheli a processzort, ezért a hang-megszakítást kezelő saját rutin nagyon rövidre van megírva, mindössze **13** assembly utasításból áll, és ebből egy a megszakítás nyugtázása, egy másik az **EI** (*Enable Interrupts*), ami engedélyezi újra a megszakításokat, egy harmadik pedig a **RET** utasítás, aminek hatására a program végrehajtása visszatér a megszakításból. Ezenkívül még 3 utasítás növel egy számlálót, kicsit lassabb leírom, hogy miért. Szóval valójában csak **7** assembly utasítás végzi a digitális hangok lejátszását. Persze ehhez elő kell készíteni a hangmintát, hogy azzal már ne kelljen bajlódni, és még a Z80 shadow regisztereket sem árt használni. Ez utóbbi ötletért köszönet *Persa Noel*nek, akivel tavaly, a *Bomberman* készítésekor értekeztünk erről.

A hang-megszakítással kis probléma, hogy amikor engedélyezzük, akkor viszont nincs *cursor-megszakítás*, azaz nem tudjuk megvárni, hogy a függőleges szinkron, azaz a képernyőfrissítés, a megadott pozícióba, alapértelmezetten a képernyő aljára, érjen. Ez miért baj? Mert ennek

segítségével lehet folyamatossá tenni a képfrissítést, hogy minden nagyon simán mozogjon a képernyőn, márpedig itt a sziklafalak és a körfűrészek „scrollozása” miatt ez nagyon kellett. De sajnos TVC-n vagy *cursor-megszakítás* van, vagy *hang-megszakítás*, a kettő együtt nem megy.

Mit lehet akkor tenni? Lehet számolni a *hang-megszakítás*ban, hogy mikor ér el egy *képernyőfrissítésnyi*. Ez a számláló van benne a saját *hang-megszakítás* rutinban. Ha a *cursor-megszakítás*t megvárva indítjuk a *hang-megszakítás*t, akkor ezzel szépen szimulálni tudjuk azt, csak figyelniünk kell, hogy mikor történik egy képernyőfrissítésnyi hang-megszakítás. Ez mennyire egyszerű, és milyen szépen hangzik! Igen ám, de a TVC-n a képernyő másodpercenként  $1000 / 20.096 = 49,76146...$  -szor frissül. Ez nem egy könnyen számolható egész szám. A 4 KHz-es *hang-megszakítás* pedig másodpercenként 4000-szer következik be, de 4000-ben a 49,76146... az bizony nincs meg egész értékszer. Hogy a manóba fogjuk akkor ezt számolni? Na, ezzel eltöltöttem vagy 3 napot, de még egy másik készülő játékom, a Raptor fejlesztésekor. Néha már WinTVC emulátorban jó volt az eredmény, de igazi TVC-n nem. Aztán rájöttem, hogy ha a képfrissítés nem is egész szám, de nyilván egyféle frekvenciaosztó dolgozik a TVC-ben, így muszáj, hogy legyen egy olyan *hang-megszakítás* frekvencia érték, ami pont egész számnyi hányadost ad, ha elosztjuk a képfrissítés értékével. Ezen az úton elindulva aztán már gyorsan meglett ez az érték, így végre szimulálni tudtam a *cursor-megszakítás*t a *hang-megszakítás* rutinomban futó számláló kiolvasásával. Hurrá!

**A *cursor-megszakítás* szimulálása *hang-megszakítás*ból sajnos WinTVC emulátorban nem működik jól, mert az valószínűleg nem kezeli jól a *hang-megszakítás*t. De *tv64emu*-ban, és természetesen igazi TV-Computeren tökéletesen működik.** Ugyanakkor WinTVC-n futtatva sem okoz hibát, csak néha kicsit megröccen, bevillan a képernyő, ennyi az összes látható probléma.

## Köszönetnyilvánítás

Köszönet

- **Major Tamás**nak a *DevTool* fejlesztőeszközért;
- **Persa Noel**nek a hangmegszakítással kapcsolatos információkért és ötletekért;
- **Sáránszki Misine**nek a paletta színállítási ötletért, bár másképp került bele, mint ahogy javasolta;
- **Kiss Károly**nak pedig az első próbaverziók tesztjéért, és címszövegről lógó játékos ötletéért.

Bertók Zsolt / BerySoft, 2022. január 9.