

# Configurable Automate LCDs

Érkezett egy kérés, hogy a csináljak egy összefoglalót erről a szkriptről:

<http://steamcommunity.com/sharedfiles/filedetails/?id=403921671&searchtext=configurable>

Ez a legfantasztikabb szkript/program/valaki más által elkészített dolog amit valaha láttam. Annyi munka van benne (96 ezer karakter!) hogy egy kis segítséget hozzá leírni is megtiszteltetés :D Bocsi ha párszor tévedek a blokkok megnevezésében, de én angolul játszok a játékkal.

Íme egy példavilág, ahol megnézhetitek mi hogy működik és hogy néz ki.

<http://steamcommunity.com/sharedfiles/filedetails/?id=412154340>

## **Amit érdemes tudni róla elsőre:**

Nem kell hozzá semmilyen mod! Csak egy programozható blokkba be kell tölteni a szkriptet és futtatni másodpercenként egy timer blokkal. Ez viszont pont a hátránya is: ha a szkript frissül, új funkciókkal egészül ki, akkor neked kell újratöltened a programozható blokkba, nem cserélődik si sehol automatikusan, ahogy egy moddal lenne.

Az Argument részt hagyd üresen. A szkript minden lefutásakor 1 LCD képernyőjét frissíti. Úgy azonosítja be az LCD-ket, hogy szerepel a nevükben ez a tag: "[LCD]" Tehát például: "LCD kijelző a hídon [LCD]". Azt pedig hogy mit kell kírni a képernyőre, a "Public Title" (Nyilvános szöveg) szövegdobozba beírt parancssor alapján tudja. A betűméret teljesen szabadon változtatható, a legtöbb esetben 0.8 az ideális, de a szkript alkalmazkodó, mindegy mekkora betűméretet használsz.

## **Amire használhatod:**

- Ugró hajtómű töltöttségének kijelzése
- Oxigén nyomás, oxigén farmok kimeneti teljesítményének jelzése és az oxigéntankok szintje
- Sérült blokkok listája és hozzájuk állapotjező sáv (progress bar)! --> [|||||||.....] 50%-os a blokk. [|||||||.....] 75%-os a blokk. stb...
- Kiválasztott/összes cikkek mennyiségének kijelzése a kiválasztott/összes hajókonténerből, szintén progress bar-ral
- Reaktorok, napelemek, akkumulátorok energiaadatairól minden infó

- Blokkok általános állapotjelzése (kikapcsolt, bekapcsolt)
- Ajtók, leszállótalpak, konnektorok állapota
- muníció jelentés, szintén progress bar-ral
- LCD kijelzők összekapcsolása!!
- működik LCD kijelzőkön, széles LCD kijelzőkön, TEXT paneleken, és ezek összes moddolt változatán
- teljes rakomány együttes mennyiségét (térfogatban) kijelezni a hajón, és hogy mennyi hely van még, progress bar-ral
- Blokkok megszámlálása
- Dolgozik, várakozik (Finomító, feldolgozó, ívkemence)
- Mi van egedélyezve és mi nem kilistázva
- Lézer antenna állapota
- GPS koordináták, sebesség, dátum és idő kijelzése
- működik bármelyik szerveren, elvégre ez nem egy mod

## Beüzemelés:

1. Egy időzítő blokkba tedd be a programozható blokkot - FUT parancs - az Argument rész üres maradjon
2. Az időzítő blokk önmagát START-olja
3. Az időzítő blokk 1 mp-en legyen (így másodpercenként 1 kijelző frissül)

## Néhány tipp, segítség:

**1. tipp:** Nyilvánvalóan, ha sok indteraktív LCD kijelzőt akarsz, nem lesz elég a másodpercenkénti 1 frissítés. Nem lenne szerencsés, ha mondjuk 10 másodpercet kéne várod arra hogy megtudd, mennyi a hajó aktuális energiafelhasználása, vagy be vannak-e kapcsolva a hajon a fegyverek, stb.. csak mert már 10 kijelző működik a hajón.

Ahhoz, hogy meggyorsítsd a dolgot, 2 mód van:

**1/1. mód (ajánlott):** több programozható blokkot csinálsz, és mindegyikbe betöltöd a szkriptet. ÉS!! megváltoztatod az egyiknél, hogy ne az [LCD] tagot keresse a kijelzők nevében, hanem mondjuk ezt: [Power LCD]. Fontos, hogy a [ jel legyen az első és a ] jel legyen az utolsó karakter!

A szkriptben keresd meg ezt: 20. sor

```
public static string LCD_TAG = "[LCD]";
```

És írd át erre:

```
public static string LCD_TAG = "[Power LCD]";
```

(ne használj nem angol karaktereket - már ha lehet egyáltalán!!)

Így az egyik szkript az [LCD]-ket fogja frissíteni, másodpercenként 1-et, míg a másik a [Power LCD]-ket. Azért is előnyös ez a módszer, mert így lehet pl. 9 [LCD]-d a kevésbé fontosnak, és 1 [Power LCD]-d a fontosnak, ami másodpercenként fog frissülni.

**1/2. mód (nem ajánlott):** Azt változtatod meg, hogy mennyi kijelzőt frissítsen a szkript 1 lefutása alatt.

Keresd meg ezt: 23. sor

```
public static int PANELS_PER_STEP = 1;
```

írd át erre:

```
public static int PANELS_PER_STEP = 2;
```

Így már nem 1, hanem 2 kijelző fog frissülni 1 lefutással, azaz másodpercenként 2. De ez a módszer nem ajánlott! Egyrészt mert így csak azt éred el, hogy az energia nem 10, hanem 5 másodpercenként fog frissülni, ami még mindig elég rossz, másrészt ez destabilizálja a szkriptet. Minél nagyobb a szám, annál gyakrabban, már a 2-nél is elő-elő fordul. Kiír valami olyat, hogy túl komplex a szkript vagy mit. De csak abból veszed észre hogy nem működik, hogy nem frissülnek a kijelzők. Megnézed a programozható blokkot és lám: ott a hibaüzenet. Ilyenkor be kell lépni a szkriptbe, ellenőrzés, mentés és kilépés. Ekkor újra jó lesz, de ki tudja mikor romlik el megint, és észreveszed-e?

**1/3. mód: (még nem próbáltam, nem tudom hogy működne-e)**

Nem csak 1, hanem több timer blokkal is futtatod a szkriptet. És mindegyik ráküld másodpercenként 1 futást. Nem tudom ez mit okoz :D (Azért írtam hogy csak 2 mód van.)

**2. tipp:** Az LCD kijelzőn az "Offline" felirat áll. Ez egy Space Engineers hiba, nem a szkripté. Nézz el másfelé, aztán vissza. Előbb-útóbb visszajön a kép.

**3. tipp:** Fekete a kijelző: valószínű rossz a parancssor amit beírtál.

**4. tipp:** Tulajdonos!!: a szkript csak azokat az LCD-eket frissíti, amiknek a tulajdonosa egyezik a sajátjával. Erre figyelj oda mindenképp!

## Oké, vágjunk bele. Hogyan kell használni:

Itt egyébként megtalálod a teljes segítséget angolul, amit folyamatosan frissítenek is.

<http://steamcommunity.com/sharedfiles/filedetails/?id=407158161>

Tutorial video: <https://www.youtube.com/watch?v=oopzyQ0t6Dk>

Vannak parancsok, és azokhoz kiegészítések.

A parancsok után egy szóközzel elválasztva jön a első kiegészítés, legtöbbször az hogy a parancs melyik nevű blokkra vonatkozzon. Majd egy újabb szóközzel elválasztva a második kiegészítés, majd a 3.,4.,stb... A ; (**pontosvessző**) jellel a parancsokat tudod elválasztani egymástól.

Parancsjelek az első kiegészítésben: \* [] {} ; G:

\* - jelentése összes

G: jelentése: Group (csoport)

[ ] , { } - szűrés nevekre

Egyszerűbb példákkal bemutatni a parancsjeleket: (parancs: Power - később)

**Power \*** - jelentése: Power parancs az összes hajón található blokkon.

**Power kicsi** - jelentése: Power parancs az összes olyan blokkon, aminek a nevében megtalálható a kicsi szó

**Power [kicsi]** - jelentése: ugyan az mint [] nélkül

**Power {Kicsi reaktor}** - a {} jelekkel lehetőség van több szóra is rákeresni. A szavak szigorú sorrendben vannak és az első szóval kezdődnek. Tehát ezzel a névszűkítéssel Power parancs lesz az összes olyan blokkon, aminek a neve "Kicsi reaktor"-ral kezdődik.

**Power {G: Reaktorok}** - jelentése: a "Reaktorok" csoportban lévő összes blokkra Power parancs.

2 parancs egymás után: (BlockCount - később) pontosvesszővel kell elválasztani.

**Power \*; BlockCount \* reactor** - a "reactor" egy második kiegészítés. Itt a jelentése: BlockCount

paranccsal megszámolhatod az összes blokkot a hajón, és ezek közül a reaktorok számát kiírja a kijelző. Ez a példa először kiírja az energiajellemzőket, aztán hogy mennyi kis reaktor és nagy reaktor van a hajón.

## 1. LCD kijelzők összekapcsolása:

Tegyük fel 2 egymás alatti kijelzőt akarsz összekapcsolni. Az az alap nevük, hogy LCD panel 1 és LCD panel 2.

**[LCD] !LINK:Csoport Szam** - [LCD] - ez szabadon módosítható, pl. [Power LCD]-re (ha átírtad a szkriptet, lsd. fentebb. A csoport ABC betűkkel van jelölve.

**!LINK:A 1** - jelentése: az A LCD csoport 1. tagja (felső). **!LINK:A 2** jelentése: az A LCD csoport 2. tagja (alsó).

Tehát a nevek így fognak kinézni:

felső: **LCD panel 1 [LCD] !LINK:A 1**

alsó: **LCD panel 2 [LCD] !LINK:A 2**

Ha egy másik panelpárost (hármast, négyest, stb...) akarsz összeapcsolni, akkor az A-t kell átírni B-re, majd C-re, D-re, stb., ezzel jelzed az eltérő csoportokat.

**LCD panel 1 [LCD] !LINK:B 1**

**LCD panel 1 [LCD] !LINK:B 2**

**LCD panel 1 [LCD] !LINK:B 3**

## 2. Dokkolt hajók:

A csillag, [], {}, G: névkiválasztás a dokkolt hajókra is érvényes! Ha nem akarsz hogy a dokkolt hajók összezavarják a te LCD kijelzőidet, akkor 2 lehetőség van:

1. (ajánlottabb) minden saját blokkodnak egyedi nevet adsz, és arra szűkítened az parancs névkeresését. Pl. használhatod a saját hajód nevét. Legyen ez: ALICE.

ALICE kis reaktor, ALICE kis konténer, stb. És nem használsz akkor \* jelet, hanem ezt: {ALICE}. Ugyan azt az eredményt éred el, mint csillaggal tehát a parancs minden blokkra rákeres, csak mivel a dokkolt hajón jó eséllyen nincs ALICE első szó, mindig csak a saját hajód értékeit látod. Úrállomásokon nagyon hasznos lehet.

2. minden saját blokkodat rendelt csoportokba, és arra szűkítened az parancs névkeresését. Pl.

létrehozol egy "ALICE power" csoportot, amiben benne van az összes reaktor, napelem, akkumulátor. Vagy egy "ALICE tárolók" amiben minden konténer benn van. És így néz ki egy parancssorod: Power {G:ALICE power}. Ezzel minden, az "Alice power" csoportban lévő modulra lesz érvényes a parancs és csak azokra! Dokkolt hajókra nem. (mint a \*-nál)

amiért kevésbé ajánlom: Nincs rá mód, hogy a csoporton belül tovább szűkítsd a kört. Tehát mindig új és új egyedi csoportokat kell létrehozni ha kitalálsz valamit. Plusz a csoportok nem maradnak meg az építéskor! Ha egy csoport minden blokkja elvész, akkor a csoport nem jö vissza amikor a blokkokat újra létrehozod. Tehát amikor Projektorból betöltesz egy Tervrajzot (Blueprint) és megépíted a hajót, abban egy darab csoport nem lesz! Vagy ha csata közben megsemmisül egy modul, és te utána újraheggeszted azt, (vagy egy projektor a hajón amivel kivetítéd folyamatosan a hajót önmagára) az szintén nem lesz benne újra a csoportban. Egyikben sem. Neked kell manuálisan visszatenni, de mi van ha valamit kihagysz?

A neve viszont mindig megmarad a blokknak. Tehát ha egy csata/ütközés után regenerálsz a hajódat, akkor visszatér a teljes funkcionalitás is. Ezért ajánlom inkább az első módszert.

### 3. Parancs: Inventory

A legösszetettebb parancs, akár sok-sok kiegészítéssel. Segítségével kiíratathatod az egyes cikkek mennyiségét. Akár csak 1 cikket is 1 adott konténerben. Vagy az összeset az egész hajón.

**InventoryX:** Ha valami cikkből 0 van, akkor azt ki sem írja.

**InventoryS:** Ha nem akarsz beleszámítani az érceket az öntvények mennyiségébe. (nem kell +ingot/uranium, elég csak +uranium)

**InventoryXS:** előző kettő együtt

#### Kiegészítések:

Nincs: összes cikk az összes blokkban kírásra kerül

1. kiegészítés: szűkítés névre

További kiegészítések: szűkítés cikktípusra

Vannak a további kiegészítéseknek operátorai +, -, /, : - példákon bemutatom mire valók

#### Fő cikktípusok:

**ore** (ércek)

**ingot** (öntvények?)

**component** (komponensek)  
**ammo** (muníció)  
**tool** (karakter felszerelései, pl. gépfegyver, kézi fúró)  
**oxygen** (az oxigén palack darabszáma)

konkrét cikkjelzések:

**iron** (vas), **gold** (arany), **nickel**, **platinum**, **cobalt**, **magnesium**, **silicon**, **silver** (ezüst), **uranium**  
(mind az ércekre, mint az öntvényekre vonatkozik!)

**scrap** (ingot) (ez a fémtörmelék)

**construction** (magyarul?), **metalgrid** (fémrács), **interiorplate** (belső fémlap?), **steelplate**  
(fémlap), **girder** (fémoszlop?), **smalltube** (kicsi vascső), **largetube** (nagy vascső), **motor**, **display**  
(kijelző), **bpglass** (golyóálló üveg), **computer**, **reactor**, **thruster** (meghajtó egység), **gravgen**  
(gravitációs generátor alkatrész), **medical**, **radio**, **detector**, **explosives** (robbanószer), **solarcell**  
(napelem alkatrész), **powercell** (components) (akkumulátor alkatrész)

**automaticrifleitem** (kézi gépfegyver), **welderitem** (kézi hegesztő), **anglegrinderitem** (kézi vágó),  
**handdrillitem** (tools) (kézi fúró)

**nato\_5p56x45mm** (kézifegyver töltény), **nato\_25x184mm** (golyó), **missile200mm** (ammo)  
(rakéta)

**oxygenbottle** (oxygen) (oxigénpalack)

Példák:

**Inventory \* +all**

+all: kijelezni az összes létező cikktípust (érceket, öntvényeket, komponenseket, oxigén palackokat, muníciót, szerszámokat), \*: a hajó összes blokkjából

**Inventory \* +all -ore**

+all -ore: kijelezni minden cikktípust, de az érceket nem, a hajó összes blokkjából (\*)

**Inventory Cargo +ingot -scrap**

Kijelezni az összes öntvénytípust, kivéve a fémtörmeléket. Csak azoknak a blokkoknak a tartalmát számítsa bele, amiknek a nevében szerepel a Cargo szó.

**Inventory {Red Storage} +ingot:300000**

Kijelezni az összes öntvénytípust úgy, hogy a progress bar maximális értéke 300000 minden egyes öntvénytípusnál. (az számít a 100%-nak) Csak azoknak a blokkoknak a tartalmát számítsa bele, amiknek a neve "Red Storage"-el kezdődik.

**Inventory {Lovegtorony kontener 1} +nato\_25x184mm:240**

Kijelezni a töltényládák számát egy 240-es max értékű progress bar-ral egy hajóhoz kapcsolódó lövegtöröny konténerében.

**Inventory \* +ingot/uranium:3000**

Az a baj, hogy ha annyit írál hogy uranium, akkor abba beleszámítanak az ércek is. (bár a progress bar fölött egy sorral fel van tüntetve, hogy mennyi az érc a kiírt összmenyiségből.) Amit viszont egy reaktor pont nem tud hasznosítani. Az ingot/uranium jelzés pont erre nyújt megoldást: csak a finomított után mennyisége lesz kijelezve, az érceké nem.

**Inventory {Nagy reaktor 1} +ingot/uranium:1000**

Kijelezni egy 1000 maxértékű progress barral, hogy mennyi finomított urán van a Nagy reaktor 1-ben. Bár ez esetben inkább érdemesebb az Amount parancsot választani. - később.

További példák:

**Inventory [STORAGE] +component**

**Inventory {Cargo Container} +all -tool -ammo**

**Inventory {Large Cargo Container 3} +ingot:1000 +component:50**

**Inventory {Oxygen Storage} +oxygen:100**

Az én extrém példám: (14 kiegészítéssel)

**Inventory \* +steelplate:20750 +motor:550 +interiorplate:950 +explosives:600**

**+construction:4700 +thruster:1300 +largetube:1200 +smalltube:1250 +computer:3350**

**+radio:800 +girder:600 +metalgrid:6550 +detector:300 +reactor:600**

Megszámoltam hogy 50 db "nyomtatható" rakétához pontosam mennyi alkatrész kell, majd ezzel íráttam ki egy kijelzőre. Ezzel a progress bar-ral folyamatosan láthatom, hogy még hány rakétát tudok kinomtyatni, esetleg melyik alkatrészből van a legkevesebb (fogyott el), ha a nyomtatás valamiért nem volt teljes.

## 4. Parancs: InvList

Hasonlít az Inventory parancsra. Csak ott kategóriába vannak szedve a cikktípusok (Component summary (Kompoensek mennyisége), Ores summary... (Ércek mennyisége))

Ennél viszont nincsennek feltüntetve a kategóriát, hanem egyben van minden.

**InvListX**: hansoló az InventoryX-hez.

**InvListS**: hansoló az InventoryS-hez



InvListXS: hansoló az InventoryXS-hez

## 5. Parancs: Missing

Csak azok a cikkek jelennek meg, aminek a mennyisége egy beállított minimum kvótánál kevesebb!

Példák:

**Missing \* +component:50 +ingot:100 +ammo:10**

Ha a hajó összes blokkjában együttesen (\*) a komponensek, öntvények és muníciótípusok mennyisége sorban 50, 100, 10 alá csökken, akkor az megjelenik az LCD kijelzőkön.

**Missing {G:Kontenerek} +ore:2000**

Ha a "Kontenerek" nevű csoport blokkjaiban az összes érc közül valamelyik (akár mindegyik) mennyisége 2000 alá csökken, akkor az adott érc megjelenik az LCD kijelzőkön.

## 6. Parancs: Cargo

Megjeleníteni a konténerek tárolókapacitását. Csak a konténerekét!

**CargoX:** Kizárólag a töltöttségi %-ot mutatja, progress bar nélkül.

Kiegészítések:

Nincs: összes konténert a hajón/állomáson kiírja

1. kiegészítés: szűrés a konténer nevére

Példák:

**Cargo**

**Cargo kicsi**

**Cargo {kicsi bal kontener 1}**

## 7. Parancs: CargoAll

Nem csak a konténerek tárolókapacitását veszi bele az összesítésbe, hanem minden blokkét amiben van hely. (Finomító, Feldolgozó, Hegesztő, stb..)

**CargoAllX:** Kizárólag a töltöttségi %-ot mutatja, progress bar nélkül.

## 8. Parancs: Power

Az energiaforgalomban szerepet játszó blokkok statisztikáit jeleníti meg 3 szeparált csoportban (reaktorok, akkumulátorok, napelemek). Működik moddolt blokkokkal is.

A napelemeknél egy elméleti (játékbeli) maximumot vesz alapul a progress bar-nál.

**PowerX:** Progress bar nélkül csak a %-ok megjelenítése.

### Kiegészítések:

Nincs: reaktorok, akkumulátorok, napelemek, mindent kiír

1. kiegészítés: NÉV alapján szűrés

### Példák:

**Power**

**Power {Kis Reaktor}**

**Power {Nagy Reaktor}**

**Power {Akkumulátor 2}**

## 9. Parancs: PowerStored

Megjeleníteni az akkumulátorok tárolt energiamennyiségét.

**PowerStoredX:** nincs progress bar, csak %.

### Kiegészítések:

Nincs: összes akkumulátort beleszámítja

1. kiegészítés: NÉV alapján szűrés

## 10. Parancs: PowerSummary

Megjeleníteni az összes energialeadási értéket a 3 energia-blokkcsoportnak.

**PowerSummaryX**: nincs progress bar, csak %.

### Kiegészítések:

Nincs: reaktorok, akkumulátorok, napelemek, mindegyik energialeadását kiírja

1. kiegészítés: NÉV alapján szűrés

## 11. Parancs: Charge

Az ugróhajtóművek töltöttségi szintjét kijelezni progress bar-ral.

### Kiegészítések:

Nincs: nem ír ki semmit (ezt írja a leírás, de kipróbáltam és működik, kiír mindent, csak még szerintem nem javították ki a leírást)

1. kiegészítés: NÉV alapján szűrés

### Példák:

**Charge**

**Charge {ugrohajtomu}**

**Charge {G:ugrohajtomuvek 1}**

## 12. Parancs: Damage

Minden olyan blokknak megjeleníti a sebzettségi értékét progress bar-ral, ami a K menüben is látható. (Tehát a páncélblokkokét, vezetékeket, elosztókét, stb. nem) Az ép blokkok állapotát (100%) nem jeleníti meg, csak a sebzeteket.

**DamageX**: progress bar nélkül, csak %

### Kiegészítések:

Nincs: minden blokkot figyel a hajón/állomáson

1. kiegészítés: NÉV alapján szűrés

Példák:

**Damage**

**Damage {Large Thruster}**

**Damage [IMPORTANT]**

## 13. Parancs: BlockCount

Megjeleníteni a kiválasztott blokkok számát. Separálható blokk típusonként a további kiegészítésekben.

Kiegészítések:

Nincs: nem ír ki semmit!

1. kiegészítés: NÉV alapján szűrés, de ha csak ennyi van, akkor nem ír ki semmit

További kiegészítések: típus szerinti blokk szűrés.

Blokk típusok:

**reactor**, **thruster** (hajtómű), **container** (konténer), **refinery** (finomító), **assembler** (feldolgozó), **solar** (napelem), **battery** (akkumulátor), **beacon** (jelzőfény), **antenna**, **gyro** (giroszkóp), **sensor** (szenzor), **connector** (konnektor), **spotlight** (reflektor), **interlight** (lámpa), **landgear** (landolótalp), **program** (programozható blokk), **timer** (időzítő), **rotor**, **piston** (szelep), **projector** (projektor), **merge** (összekapcsoló blokk), **sound** (hangszóró), **collector** (begyűjtő), **door** (ajtó), **spheregrav** (gömb alakú gravitációs generátor), **gravgen** (gravitációs generátor), **drill** (fúrófej), **grinder** (fűrész), **welder** (hegesztő), **gatlingturret** (gatling torony), **missileturret** (rakétatorony), **interiorturret** (belső torony), **gatlinggun** (gépfegyver kis hajókra), **reloadlauncher** (újratölthető rakétavető kis hajókra), **launcher** (rakétavető nagy hajókra), **mass** (mesterséges tömeg), **warhead** (robbanófej), **light** (fény: spotlight + interlight), **control** (irányító panel), **cockpit** (pilótafülke), **medi** (elsősegély állomás), **remote** (távírányító), **button** (gomb), **camera** (kamera), **detector** (ércdetektor), **laserantenna** (lézerantenna), **airvent** (légkeverő), **oxytank** (oxigéntank), **oxygenerator** (oxigéngenerátor), **sorter** (irányító), **oxyfarm** (oxigén farm), **cryo** (kriókamra), **jumpdrive** (ugróhajtómű)

A Hangárajtó egybe van véve a sima ajtókkal.

Példák:

**BlockCount \* reactor**

**BlockCount [BACKUP] reactor**

**BlockCount \* refinery,assembler**

**BlockCount \* thruster gyro reactor,solar,battery**

## 14. Parancs: EnabledCount

Kiírja a számát a bekapcsolt blokkoknak.

Kiegészítések:

Nincs: nem ír ki semmit!

1. kiegészítés: NÉV alapján szűrés, de ha csak ennyi van, akkor nem ír ki semmit

További kiegészítések: típus szerinti blokkszűrés (a típusok azok mint a BlockCount-nál)

Példák:

[b]EnabledCount \* reactor[/b]

[b]EnabledCount \* assembler[/b]

## 15. Parancs: ProdCount

Kiírni a számát a dolgozó blokkoknak.

Kiegészítések:

Nincs: nem ír ki semmit!

1. kiegészítés: NÉV alapján szűrés, de ha csak ennyi van, akkor nem ír ki semmit

További kiegészítések: típus szerinti blokkszűrés (a típusok azok mint a BlockCount-nál)

Példa:

**ProdCount \* refinery,assembler**

## 16. Parancs: Working (Ugróhajómű!)

Megjeleníteni bizonyos blokkok aktuális állapotát.

Amiket kiír: ON/OFF (bekapcsolva/kikapcsolva), IDLE/WORK (várakozik/dolgozik - finomító, ívkemence, feldolgozó), LOCK/UNLOCK/READY (rögzítve, rögzítetlen, készen áll - a landolótalpaknál és konnektoroknál), OPEN/CLOSED (nyitva, zárva - ajtóknál)

#### Kiegészítések:

Nincs: nem ír ki semmit!

1. kiegészítés: NÉV alapján szűrés, de ha csak ennyi van, akkor nem ír ki semmit

További kiegészítések: típus szerinti blokkszűrés (a típusok azok mint a BlockCount-nál)

A további kiegészítésnek van egy operátora, a **kettőspont** :. Segítségével az állapotokra tudsz szűrni - példák.

**WorkingX**: kizárólag az ON/OFF állapotokat jeleníti meg!

#### Például:

**WorkingX \* assembler:on** Ha egy feldolgozó be van kapcsolva, akkor megjelenik a kijelzőn a neve és hogy ON. Máskor nem!

A további kiegészítések vagy szóközzel vagy vesszővel választhatók el. A Hangárajtó egybe van véve a sima ajtókkal. Ha egy 5 tagú hangárajtó csoportot akarsz megjeleníteni úgy, hogy nem íratatod ki 5-ször, hogy nyitva, akkor szűrj rá csak az egyikre név alapján. (Hangárajtó 1)

#### Példák:

**Working \* missileurret,gatlingurret,interturret**

**Working \* refinery:off refinery:idle**

Ha a finomító ki van kapcsolva az egész hajón/állomáson (\*), vagy várakozik, akkor megjelenik a neve kijelzőn és hogy OFF vagy IDLE. Jó kis figyelmeztető :) Kombinálhatod egy kis Missing paranccsal is, és akkor már a leállítás előtt észreveheted, ha kifogyóban van valamelyik finomító.

**Working \* door**

Kiírja az összes ajtó nevét, és hogy nyitva (OPEN) vagy zárva (CLOSED) van-e.

**Working \* door:open**

Ha egy ajtó nyitva van, akkor megjelenik a kijelzőn a neve és hogy OPEN. Ha zárva van, akkor

még a neve sincs kiírva.

**Working {G:Allomas Feldolgozok} assembler:idle assembler:off**

Akkor jöhet nagyon jól, ha egy állomás összekapcsolódik egy hajóval. Ilyenkor is csak az "Allomas Feldolgozok" csoportban lévő feldolgozókra keres rá a Working parancs, és megjeleníti azok nevét és státuszát, amelyek váraognak (IDLE) vagy ki vannak kapcsolva (OFF).

**Working Red refinery, assembler**

Kiírja azoknak a finomítóknak és feldolgozóknak a nevét és státuszát (ON, OFF, WORK, IDLE) amiknek a nevében szerepel a "Red" szó.

## 17. Parancs: Details (Projektörök!!)

A név alapján kiválasztott blokkokról jelenít meg információkat. (pontosan nem tudom miket ír ki még nem használtam)

Például: (a leírásban) Projector details showing needed parts. (A projektör helyzete azt mutatja, hogy szükségesek még részletek.) - nyugi ezt én sem értem :D. Talán: nincs befejezve még a tervrajz amit vetít?

Kiegészítések:

Nincs: nincs semmi a kijelzőn

1. kiegészítés: NÉV alapján szűrés

Példák:

**Details {Projector 2}**

**Details {G:Akkumulatorok}**

## 18. Parancs: Amount

Olyan mint egy BlockCount + Working kombó.

Amire való: olyan blokkoknál amibe csak 1 féle cikk tehető, pl. rakétavető a 200mm-es rakéta, vagy reaktorba az uránium.

Reaktornál a neve után kiírja:

Ha üres, akkor EMPTY.

Ha be van kapcsolva és van benne urán, akkor egy progress bar, aminek a max. értéke automatikusan a reaktor max. térfogata. És így jelzi ki mennyi urán van benne, mennyire van telítve.

**AmountX:** Nincs progress bar, csak %.

Rakétavető Toronynál pl.:

Ha üres, akkor EMPTY.

Ha be van kapcsolva és van benne lőszer, akkor 1/6, 2/6, 3/6, stb.. mert hogy 6 db 200mm-es rakétatároló doboz fér bele. (1 rakétatárolóban 4 rakéta van)

Gatling Toronynál

EMPTY, vagy 1/24, 2/24, 3/24, stb... 24 db NATO\_25x184mm-es láda fér bele. (1 láda olyan 3-4 másodperces folyamatos tüzelést biztosít.)

Oxigén generátornál, a kis és nagy manuális rakétavetőnél és a gépfegyvernél jön még jól.

Kiegészítések:

Nincs: kiírja az összes torony, gépfegyver, reaktor és oxigén generátor adatát.

1. kiegészítés: NÉV alapján szűrés

További kiegészítések: típus szerinti blokkzűrés (a típusok azok mint a BlockCount-nál)

Példák:

**Amount {G:Weapons} launcher,gatlinggun**

**Amount {Small Reactor} reactor**

**AmountX {G:Turrets} missileturret,gatlingturret**

## 19. Parancs: Oxygen

Megjeleníti a nyomásértéket a légkeverőknél (azt is ha lyukas a hajó (LEAKING)), átlagos outputját az oxigénfarmoknak, és átlagos töltöttségét az oxigén tankoknak.

Megjegyzés: Ha nem szeretnél semmit sem kiírni amikor minden rendben van (összes



légkeverő nyomásértéke progress bar-ral), csak egy figyelmeztető LCD kijelzőt akarsz akkor, ha lyukas a hajó, akkor használd a Working parancsot a leak szűréssel a 2. kiegészítésben.

**Working \* airvent:leak**

**Working {G:Gephaz} aivent:leak**

Kiegészítésel:

Nincs: összes légkeverő, oxigénfarm-és tank állapota

1. kiegészítés: NÉV alapján szűrés

Pélák:

**Oxygen**

**Oxygen Vent**

**Oxygen {Oxygen Tank}**

**Oxygen {Oxygen Farm}**

**Oxygen {G:Gephaz legkeverok}**

## 20. Parancs: Echo

Egy egysoros szöveg kiírása (balra zártan), vagy egyszerűen csak egy üres sor beszúrása.

Kiegészítések:

Nincs: üres sor

1. kiegészítés: A szöveg amit ki akarsz íratni.

Például:

**Echo**

üres sor

**Echo MMaster's Automatic LCD**

**Echo Small;Power Small;Echo Large;Power Large**

**Power {Nagy Reaktor};Echo;Echo;BlockCount \* reactor**

Kiírja a Nagy Reaktor kezdetű blokkok energiaadatait, aztán 2 üres sor és kiírja a reaktorok számát.

## 21. Parancs: Center

Hasonlít az Echo-ra, csak itt a szöveg a sor közepén(!!) fog kiíródni.

## 22. Parancs: Text

A privát szöveg részbe írt szöveget jeleníti meg a kijelzőn.

Példa:

**Text;Power \* reactor**

**TextLCD**: egy másik LCD kijelző szövegét jeleníti meg a kijelződön. NAGYON HASZNOS amikor egy másik szkripttel kiíratasz dolgokat egy másik LCD kijelzőre, te pedig azt kombinálni akarod az adott LCD kijelző szövegével.

Példa:

**TextLCD {Masik LCD}**

## 23. Parancs: Time

Megjeleníti az aktuális időt. Már magában a parancsban is eszközölhetsz változtatást operátorokkal:

**Time-2** - aktuális idő -2 óra

**Time+1** - aktuális idő +1 óra

**Time+0.5** - aktuális idő +0.5 óra (30 perc)

Kiegészítések:

Nincs: megjeleníteni az aktuális időt

1. kiegészítés: egy szöveg amit az idő előtt kiír

Példák:

**Time**

**Time MMaster's Text Panel Time:**

**Time+1 Ido plusz 1 ora:**

## 24. Parancs: Date

(angol formátumban, tehát: nap/hó/év)

Megjeleníteni az aktuális dátumot. Itt is eszközölhetsz változtatást a parancsban operátorokkal, de az órában lesz értendő!:

**Date-2** - aktuális dátum -2 óra (tehát 2 órával később vált a nap)

**Date+0.5** - aktuális dátum +0.5 óra (30 perc)

Kiegészítések:

Nincs: megjeleníteni az aktuális dátumot

1. kiegészítés: egy szöveg amit a dátum előtt kiír

Példák:

**Date**

**Date Aktualis datum:**

ez jelenik meg a kijelzőn: Aktualis datum: 06/09/2015

**Date+1 Datum plusz 1 ora:**

## 25. Parancs: Pos

Megjeleníteni a pozíciódat egy 1 soros szöveggént.

**PosXYZ**: 3 külön sorban kiírni a pozíciót.

**PosGPS**: GPS formátumban kiírni a pozíciót.

Kiegészítések:

Nincs: hely megjelenítése az LCD kijelzőn

1. kiegészítés: megjeleníteni annak a blokknak a helyét amit a NÉVszűrés ELŐSZÖR megtalál.  
(Tehát csak 1 pozíciót ír ki!)

Példák:

**Pos**

**PosXYZ**

**Pos {Kis Reaktor 1}**

**PosXYZ {Pilotafulke}**

**PosGPS {Pilotafulke}**

## 26. Parancs: Speed

Megjeleníteni a hajó aktuális sebességét. Sok esetben pontatlan lehet!

Remélem tudtam segíteni :)

laci4321





