Issue 6.0

Date: 20th September 2012

FPSC Complete Script Syntax List

By Ched80

Contents

1 Scope

This document attempts to provide the user with a complete list of the *.fpi commands available within FPSC x9 v1.20. The intention is that this list is used as a reference when writing and/or debugging scripts.

Section 2 provides the reader with a basic explanation of FPSC FPI script creation and interpretation and also provides some information on the use of variables and tips for debugging scripts.

Section 3 describes all the possible FPI script commands available within FPSC and it is written with the assumption that the reader already has some experience of producing FPSC scripts.

Section 4 contains the appendices that support this document.

The document is arranged into sections containing all the related conditions and actions for a particular feature within FPSC (such as waypoints or HUDs). There will be some over-lap concerning where certain conditions and actions should be and the sections are only provided as a guide.

If you need to quick search for a particular condition or action please refer to the alphabetical lists in the appendices.

Thanks to 2Beastmode4u, Bond1, Code Master, Conjured Entertainment, Flatlander, Hockeykid, Nickydude, Omegamer and Scene Commander for help during the creation of this document.

1.1 Revision List

Issue	Date	Changes
1.0	22nd February 2011	First Issue.
2.0	9th June 2011	Updated to capture majority of parameter ranges. Script list updated to include syntax. Variable list and explanation added. Tutorials on scripting and debugging added. Following conditions added: *plrpreventedselectinggun*, *plrusingaction*, *shotdamagetype.* Following actions added: *aisetviewrange*, *disablecinematichandsbutton*, *disablefirebutton*, *getcinematichands*, *hideshaddow*, *huduservar*, *lrayset*, *lrayactive*, *bloomactive*, *lrdebugdeactive*, *preventplrselectinggun*, *removecurrentweapon*, *startcinematichands*, *updategui, win.* Corrected some errors.
3.0	3rd October 2011	Improved example code to aid understanding. All gunspec.txt and flakspec.txt commands added. Re-organised some commands into more appropriate sections. Re-organised animation list. Following actions added: *setentityspeed, setanimatespeed.* Corrected various errors.

Issue	Date	Changes
4.0	20th April 2012	Removed gunspec and flakspec parts to create a separate document as this one grew too big. Moved GUI variable commands to variable section. **Following conditions added**: *airequal, airgreater, airlesser, currentweapon, currentweaponjammed, damageby, damagetimegreater, entityfloorequal, entityfloorhigher, entityfloorlower, entityishigher, entityislower, firemode, flashing, isentityimmune, isentitynotimmune, mouseclick, notonrader, onradar, plrcrouching, plrfloorequal, plrisimmune, plrjumping, plrlastfired, plrnotcrouched, plrnotjumping, plrnotrunning, plrnotusingreload, plrnotzoomed, plronground, plrrunning, plrusingreload, plrweaponidle, plrweaponnotidle, plrweaponsequal, plrweaponsgreater, plrweaponslesser, plrzoomed, radarequal, radargreater, sameflooraspl r, soundnotplaying, soundplaying, spawnsgreater, spawnsleft, weaponinslot.* **Following actions added**: *addair, addrawtext, addrawvar, airon, armdec, arminc, armon, arrowkeys, camrotationon, changehudalpha, compassoff, compasson, compassspin, cos, 16ompass, 16ompass, createbulletsplashsound, crosshair, crouchkey, cullevenifimmobile, cullmode, cullmodi, cullranger, debugcursor, debugtext, debugvar, emitflash, emitforce, entityaccuracymult, entitycam, entitydamagemult, entitysetimmune, entrotatex, entrotatey, entrotatez, fogstart, fogend, forcedamageon, freeemplacement, freefromplr, giveplrweapon, globalnoair, hide, hidelimb, holster, instantdrown, jumpkey, lastcam, linktoplr, linkxrotation, linkyrotation, linkzrotation, lockemplacement, lockslot, logicburst, moveplrx, moveplry, moveplrz, needlespin, offsetanglex, offsetangley, offsetanglez, peekkey, playercam, playerstrength, playfullvideo, playgunanimation, plraccuracymult, plraction, plralwaysrun, plrcamoffseton, plrcamoffsetx, plrcamoffsety, plrcamoffsetz, plrdamagemult, plrdeath, plrdeathbounce, plrdeathspeed, plrforcemove, plrlastfired, plroffsetanglex, plroffsetangley, plroffsetanglez, plrpickon, plrpickrange, plrrotatex, plrrotatey, plrsetimmune, plrspeedmod, plrstrength, plrwobble, presetgunanimation, radaron, radarrange, radarx, radary, randomize, removeplrweapon, resetglobalsonreload, rotateblip, runkey, scale, scalehudx, scalehudy, scalelimb, setair, setairmax, setairtime, setairx, setairy, setarm, setarmx, setarmy, setcamoffsetx, setcamoffsety, setcamoffsetz, setcamrotx, setcamroty, setcamrotz, setdrowntime, setflashblue, setflashgreen, setflashrange, setflashred, setforcedamage, setisobjective, setlistkey, setmaxweapons, setnoairdamage, setobjectivemode, setobjectivex, setobjectivey, setposteffect, setvarrnd, show, showlimb, sin, swapplrweapon, swaptoalt, walkkey, watercurrent, waterflow, weapontoslot, wireframe*. Added new internal variables. Corrected various errors.

Issue	Date	Changes
5.0	20th Sept 2012	Updated to v120 **Following conditions added**: *waterequal, watergreater*, *waterison, waterlesser.* **Following actions added**: deletevideotexture, *flashlight, flashlightblue, flashlightgreen, flashlightrange, flashlightred, makevideotexture, moveback, pausevideotexture, resumevideotexture, setbulletcoloff, setvideotexturespeed, setvideotexturevolume, stopvideotexture, usevideotexture* Corrected various errors.

2 Script Basics

2.1 Script Creation and Interpretation

To be able to interpret *.fpi scripts (called scripts from herein) it is essential to understand how they are constructed and how they are interpreted by the FPSC engine. A script is a set of basic instructions that elements within FPSC follow to create the game effects and mechanics you want and every script consists of the following three elements:

- Comment or remark lines,
- Description line,
- Condition and action lines.

Comment or remark lines (called remark lines from herein) are ignored by the FPSC engine and are used to help the reader remember/understand what the script is trying to achieve.

The description line is used by FPSC simply to describe what the script does. FPSC does not perform any actions based on this line.

Condition and action lines make up the bulk of scripts and are directly used by the FPSC engine to generate various effects and mechanics through the game.

An example script "spawndestroy.fpi" created by Uman is shown below as an example.

```
;Artificial Intelligence Script
;Header
desc = Umans load and unload any entity withinrange (excludes
player and explodables they dont repawn)
;Triggers
:
state=0,plrdistwithin=900,activated=1:state=1,loopsound=$0,spawnon
:state=1:state=2,setalphafade=0,setframe=0
:state=2:incframe=0,incalphafade=100
:state=2,frameatend=0:state=3
:state=3,alphafadeequal=100,plrdistfurther=1200:settartget,destroy
;End of Script
```

The first thing to understand is that the **green** lines are remark lines, the **orange** line is the description line and the **red** and **blue** lines are the condition and action lines.

Remark lines are created by making the first character in the line a semi-colon [;]. You can write anything after the semi-colon as it is ignored by FPSC.

The description line is defined by starting the line with [desc =]. Again you can write anything here as none of the text is actions by the FPSC engine.

The rest of the script consists of conditions (**red** text) and actions (**blue** text) and these form the instructions which FPSC obeys. The way in which conditions and actions are written (called the syntax from herein) in the script must be done

correctly otherwise the instructions will not be carried out by FPSC as you want them to.

The correct syntax for every condition and action within FPSC is specified within section 3, but in order to create working scripts it is also necessary to understand how all the conditions and actions are linked together.

All script instructions consist of a set of questions (conditions) followed by a set of orders (actions). In order for the actions to be carried out, all of the conditions in the same instruction line must be satisfied.

In order to initiate an instruction, the script line MUST start with a colon ":". This tells the FPSC engine to take notice of the text following the ":". Importantly it also tells FPSC that the next text will be a condition. There can be as many conditions as you like, but each one must be separated by a comma ",".

After the last condition another colon ":" is used to inform FPSC that the following text will be an action. Like conditions, there can be as many actions in each instruction line as you wish; and each action must be separated with a comma ",".

Before looking at an example, it should also be noted that scripts are written from the point of view of the entity to which the script is attached.

Let's look at an example script instruction line from the script on the previous page:

```
:state=3,alphafadeequal=100,plrdistfurther=1200:settartget,destroy
```

In order to decipher the line above, we'll first explain what each element in the line is doing.

Syntax	Description
`state=3`	This condition tests the fpi's internal 'state' variable value. In this case, if the state variable equals '3' then the test is TRUE, if the variable is any other value then the test is FALSE.
`alphafadeequal=100`	This condition tests the entity's 'alpha fade' or transparency variable value. In this case, if the alpha fade variable is equal to '100' then the test is TRUE, if the variable is any other value then the test is FALSE.
`plrdistfurther=1200`	This condition tests the distance between the player and the entity. In this case, if the player is more than 1,200 units away from the entity then the test is TRUE, if the player is closer or exactly 1,200 units away than this then the test is FALSE.
`settarget`	This action sets the entity's possible target (determined automatically by the FPSC engine) as the entity's actual target.
`destroy`	This action destroys the entity – removing it from the game universe.

The line above can be translated as follows:

If:

- …the internal state variable is equal to 3 and…
- …the entity's alpha fade variable is equal to 100 and…
- …the player is over 1,200 units away from the entity.

So if all of the conditions are TRUE:

- …set the entity's possible target as its actual target and…
- …destroy the entity.

Don't worry if the actions and conditions in this example didn't make complete sense; the explanation of the conditions and actions is the purpose of the rest of the document. The important things to understand here are:

- All script instructions MUST start with a colon ":"
- All subsequent conditions MUST be separated with a comma ","
- The last condition MUST be followed with a colon ":"
- All subsequent actions MUST be separated with a comma ","
- There MUST be at least one condition and one action in an instruction line.

2.2 Variables

The majority of script conditions and actions will allow the use of variables to define their parameters. There are two types of variable within FPSC: *user* variables and *internal* variables.

User variables are those created specifically for your game, whilst internal variables are those fundamental to FPSC and are used in every FPSC game.

To use a user variable with a script condition or action, the user variable name should be preceded by the '%' character. For example:

```
activated=%MyActivationVariable:state=1
```

The "`activated=`" condition would be true if the activation state of the entity equals the value of MyActivationVariable.

To use an internal variable with a script condition or action, the internal variable name should just needs to be stated. For example:

```
healthgreater=$PH:state=1
```

The "`healthgreater=`" condition would be true if the health of the entity is greater than the player's health.

See Annex 4.5 for a list of internal variables available.

2.3 Debugging Techniques

When writing and developing scripts it is invaluable to understand which part of the script the AI is within and what value the user variables are during game testing.

The first thing to add when creating a new and complex script is to set up fpgcrawtext as well as a simple numerical HUD as shown below:

```
;Set up FPGC Raw Text
:state=0:fpgcrawtextx=50,fpgcrawtexty=20
:state=0:fpgcrawtextfont=verdana,fpgcrawtextsize=22
:state=0:fpgcrawtextr=255,fpgcrawtextg=255,fpgcrawtextb=255
;Declare Variables
;TODO
;Set up Numerical HUD
:state=0:hudreset,hudx=50,hudy=10,hudsizex=16,hudsizey=16
:state=0:huduservar=TestVar,hudname=TestVarHUD
:state=0:hudimagefine=gamescore\huds\numeric.tga
:state=0:hudtype=6,hudmake=numeric
:state=0:state=1
```

Adding these features at the top of your script will enable the script author to quickly add debugging messages so that the status of the script can be understood during game testing. The author must remember to remove these debugging features once the script is working correctly.

An example of a work-in-progress script is shown below, so that the debugging messages can be seen.

```
;Set up FPGC Raw Text
:state=0:fpgcrawtextx=50,fpgcrawtexty=20
:state=0:fpgcrawtextfont=verdana,fpgcrawtextsize=22
:state=0:fpgcrawtextr=255,fpgcrawtextg=255,fpgcrawtextb=255
;Setup HUDs
:state=0:hudreset,hudx=50,hudy=90,hudimagefine=gamecore\huds\TestT
ubes\PressEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=d
isplay
:state=0:state=1

;START
:activated=10:state=10,fpgcrawtext=Auto Opening
:activated=20:state=20,fpgcrawtext=Door Active 20
:activated=30:state=30,fpgcrawtext=Door Active 30
:activated=40:state=40,fpgcrawtext=Door Active 40
:activated=50:state=50,fpgcrawtext=Door Active 50
:activated=60:state=60,fpgcrawtext=Door Active 60
:activated=70:state=70,fpgcrawtext=Door Active 70
:activated=666:destroy

;Auto Open/Close
:state=10,anywithin=75:state=11,setframe=0,sound=$0,activate=11
:state=11:incframe=0
:state=11,frameatend=0:state=12,coloff
:state=12,anyfurther=100:state=13,sound=$1,colon
:state=13:decframe=0
```

```
:state=13,frameatstart=0:state=10,setframe=0
```

In the example above, text would be displayed on screen indicating what activation state the entity (in this case, a door) had been set to.

The second thing the developer can do to aid script debugging is to switch ON the entity information, accessed via "File" – "Preferences" and then selecting the "Show Entity Game Information" check box. This feature displays the following information over the entity within the test game as shown in the figure at the bottom:

- The name (AI: (Pistol)),
- The current script being ran (A:people\chase10.fpi),
- The state value within the script (S:1),
- The health (H:100),
- The frame of the current animation being played (FRAME:0\0),
- The global variable count (G:0,0,0),
- The local variable count (L:0,0,0),

If the entity is a character, the following additional info is shown:

- The lives left (L:1),
- The waypoint state value (WS:999),
- The lost target count and the position of the current target (LT:0(550,542,-640)),
- The ray cast count (RCC:0),
- The angle of rotation around the y-axis (ry=246.157730103),

If the character is carrying a weapon, the following additional info is shown:

- The current weapon ammo and clip ammo (WA:0\9999),

Cross referencing this information, with you script can help you to focus on the area of the scrip that may be causing the problem.

3 Scripting Conditions and Actions

This section contains a complete list of all script commands, their operating ranges and an example of where the command is used.

To make it easier for the reader the commands are ground in sections relevant to specific parts of FPSC. The commands are split up as follows:

- **General & Miscellaneous**: General commands relating to all entities.
- **Variables & Timers**: Commands applicable to variables and timers.
- **Trigger Zones**: Commands applicable to trigger zones.
- **NPCs (Enemies, Allies & Neutrals)**: Commands applicable to NPCs
- **Waypoints & Markers**: Commands applicable to NPC movement.
- **Animations**: Commands applicable to entity animation.
- **Dark AI**: Specific Dark AI commands.
- **Player**: Commands applicable to the player and their weapons.
- **Multi-Player**: Commands applicable to multi-player games.
- **Cameras**: Commands applicable to the camera.
- **Lights & Lighting**: Commands applicable to light entities and general lighting within the level.
- **HUDs**: Commands applicable to HUDs, including the compass and radar, and text.
- **GUI x9**: Commands applicable to the GUI x9.
- **Level Navigation & Level Setup**: Commands applicable to the whole level.
- **Water**: Commands applicable to the water entity.
- **Media**: Commands applicable to media (sounds, music, video, etc.)
- **Game Controls**: Commands applicable to the overall game.
- **Setup.ini**: Commands applicable to the setup.ini.

3.1 Generic and Miscellaneous

The following conditions and actions are applicable to all general entities.

3.1.1 CONDITIONS

3.1.1.1 *ACTIVATED=X*

Description: Is true when the activation value of the entity is equal to X.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:activated=1:state=1`

3.1.1.2 *ALPHAFADEEQUAL=X*

Description: Is true when the alpha fade/tranparency value of the entity equals X.

Range: X = 0 to 100

Example: `:alphafadeequal=0:destroy`

3.1.1.3 *ALWAYS*

Description: Is permanently true.

Example: `:always:shootplr`

3.1.1.4 *ANYFURTHER=X*

Description: Is true when any entity is further away than X units (100 units = 1 segment).

Range: X 0 to 540,000 (maximum distance in level)

Example: `:anyfurther=200:decalphade=100`

3.1.1.5 *ANYWITHIN=X*

Description: Is true when any entity is within X units (100 units = 1 segment).

Range: X > 0.

Example: `:anywithin=50:activateifused`

3.1.1.6 *CANTAKE*

Description: Is true if the entity can be taken by the player.

Example: `:cantake:playertake`

3.1.1.7 *DAMAGEBY=X*

Description: Is true when the entity has been damaged by the weapon defined, the slot number defined, or the text "collision", "melee", "indirect", "flak" or "falling".

Range: X = File path of the weapon,

X = The weapon slot number,

X = "collision" – True if entity was damaged by a collision event (such as a crate falling on them),

X = "falling" – True if entity was damaged by falling,

X = "flak" – True if entity was damaged by flak,

X = "indirect" – True if entity was damaged by an indirect event (such as a barrel exploding),

X = "melee" – True if entity was damaged by a melee attack,

Examples: `:damageby=ww2\colt45:state=4`

`:damageby=5:state=5`

`:damageby=indirect:state=6`

3.1.1.8 *DAMAGETIMEGREATER=X*

Description: This is true if the time since the entity was last damaged is greater than X milliseconds.

Range: X = Any value in milliseconds.

Example: `:damagetimegreater=4000:state=1`

3.1.1.9 *ENTITYFLOOREQUAL=X*

Description: This is true if the entity is on floor X.

Range: X = 1 to 20

Example: `:entityfloorequal=2:state=10`

3.1.1.10 *ENTITYFLOORHIGHER=X*

Description: This is true if the entity is X floors higher than the player.

Range: X = 1 to 20

Example: `:entityfloorhigher=3:state=10`

3.1.1.11 *ENTITYFLOORLOWER=X*

Description: This is true if the entity is X floors lower than the player.

Range: X = 1 to 20

Example: `:entityfloorlower=3:state=10`

3.1.1.12 *ENTITYISHIGHER=X*

Description: This is true if the entity is X units higher than the player.

Range: X = 1 to 2000

Example: `:entityishigher=300:state=10`

3.1.1.13 *ENTITYISLOWER=X*

Description: This is true if the entity is X units lower than the player.

Range: X = 1 to 2000

Example: `:entityislower=300:state=10`

3.1.1.14 *FLASHING=X*

Description: Is true if the entity is flashing and X is 1 or if the entity is flashing and X is 0.

Range: X = 0 – True if the entity is not flashing,

X = 1 – True if the entity is flashing.

Example: `:flashing=1:state=1`

3.1.1.15 *ISENTITYIMMUNE*

Description: Is true when the entity is immune to damage.

Example: `:isentityimmune:state=4`

3.1.1.16 *ISENTITYNOTIMMUNE*

Description: Is true when the entity is not immune to damage.

Example: `:isentitynotimmune:state=5`

3.1.1.17 *NEVER*

Description: Is never true.

Example: `:never:destroy`

3.1.1.18 *NORAYCASTUP=X Y*

Description: Is true if a ray cast between X and Y within the vertical axis fails to collide with an entity.

Range: X = 0 to any value in units,

Y = 0 to any value in units.

Example: `:noraycastup=10 20:state=1`

3.1.1.19 *NOTONRADAR*

Description: Is true if the entity is not on the radar.

Example: `:notonradar:state=1`

3.1.1.20 *OBJECTIVERANGEGREATER=X*

Description: Is true if the objective's range is set at more than X units.

Range: X = Any value.

Example: `:objectiverangegreater=500:state=1`

3.1.1.21 *OBJECTIVERANGELESSER=X*

Description: Is true if the objective's range is set at less than X units.

Range: X = Any value.

Example: :**objectiverangelesser**=500:state=1

3.1.1.22 *ONRADAR=X*

Description: Is true if the entity is on the radar and X is 1 or if the entity is not on the radar and X is 0.

Range: X = 0 – True if the entity is not on the radar,

X = 1 – True if the entity is on the radar.

Example: :**onradar**=1:state=1

3.1.1.23 *RADAREQUAL=X Y*

Description: Is true if the entity's range on the radar is equal to X segments and Y is 1 or is not equal to X segments and Y is 0.

Range: X = 0 – 40 (segments),

Y = 0 – checks if range is not equal to X,

Y = 1 – checks if range is equal to X.

Example: :**radarequal**=5 1:state=50

3.1.1.24 *RADARGREATER=X Y*

Description: Is true if the entity's range on the radar is greater than X segments and Y is 1 or is less than X segments and Y is 0.

Range: X = 0 – 40 (segments),

Y = 0 – checks if less than range,

Y = 1 – checks if greater than range.

Example: :**radargreater**=5 1:state=50

3.1.1.25 *RANDOM=X*

Description: Is true when a random number between 0 and X equals 0. I.e. random=1 gives a 50% chance of being true, whereas random=100 gives a 1% chance of being true.

Range: X = 1 to anything, although setting above 1,000 will make this conditions very unlikely to be true.

Example: :**random**=1:incstate

3.1.1.26 *RAYCAST=X Y*

Description: Is true if a ray cast between X and Y in front of the entity collides with another entity.

Range: X = 0 to any value in units,

Y = 0 to any value in units.

Example: :**raycast**=10 20:state=1

3.1.1.27 *RAYCASTBACK=X Y*

Description: Is true if a ray cast between X and Y behind the entity collides with another entity.

Range: X = 0 to any value in units,

Y = 0 to any value in units.

Example: :**raycastback**=10 20:state=1

3.1.1.28 *RAYCASTUP=X Y*

Description: Is true if a ray cast between X and Y within the vertical axis manages to collide with an entity.

Range: X = 0 to any value in units,

Y = 0 to any value in units.

Example: :**raycastup**=10 20:state=1

3.1.1.29 *SAMEFLOORASPLR*

Description: This is true if the entity is on the same floor as the player.

Example: :**sameflooraspl**r:state=1

3.1.1.30 *SHADERVARIABLE=X*

Description: Is true when the FX shader variable for the entity is set to X. This condition is not officially supported or documented and is only available for experimental use only.

Range: X = 1 to 4

Example: :**shadervariable**=1:state=1

3.1.1.31 *SHADERVARIABLEGREATER=X*

Description: Is true when the FX shader variable for the entity is greater than X. This condition is not officially supported or documented and is only available for experimental use only.

Range: X = Any value.

Example: :**shadervariablegreater**=1:state=1

3.1.1.32 *SHADERVARIABLELESS=X*

Description: Is true when the FX shader variable for the entity is less than X. This condition is not officially supported or documented and is only available for experimental use only.

Range: X = Any value.

Example: :**shadervariableless**=1:state=1

3.1.1.33 *SOUNDNOTPLAYING*

Description: Is true if the entity is not playing a sound.

Example: :**soundnotplaying**:state=1

3.1.1.34 *SOUNDPLAYING*

Description: Is true if the entity is playing a sound.

Example: :**soundplaying**:state=1

3.1.1.35 *SPAWNSGREATER=X Y*

Description: Is true when the entity has more than X spawns remaining and Y is 1 or the entity has less than X spawns remaining and Y is 0.

Range: X = Any value up to the maximum spawn number of the entity,

Y = 0 – True if the entity has less than X spawns remaining,

Y = 1 – True if the entity has more than X spawns remaining.

Example: :**spawnsgreater**=4 1:state=4

3.1.1.36 *SPAWNSLEFT=X Y*

Description: Is true when the entity has X spawns remaining and Y is 1 or the entity does not have X spawns remaining and Y is 0.

Range: X = Any value up to the maximum spawn number of the entity,

Y = 0 – True if the entity does not have X spawns remaining,

Y = 1 – True if the entity has X spawns remaining.

Example: :**spawnsleft**=4 1:state=4

3.1.1.37 *STATE=X*

Description: Is true when the FPI state is equal to X

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: :**state**=0:shootplr

3.1.1.38 *UNDERWATER=X*

Description: Is true in the following conditions:

- When the entity is under the water and X=1.
- When the entity is above the water and X=0

Example: :**underwater**:explode

3.1.2 ACTIONS

3.1.2.1 *ACTIVATE=X*

Description: Sets the activation value of the entity to X.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:state=0:activate=1`

3.1.2.2 *ACTIVATEIFUSED=X*

Description: Activates the entity nominated in the entity's "If Used" field and sets the activated entity's activation value to X.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:state=0:activateifused=1`

3.1.2.3 *ACTIVATEIFUSEDNEAR=X*

Description: Activates the entity nominated in the entity's "If Used" field IF THE "If Used" entity is within 80 units (100 units = 1 segment) and sets the activated entity's activation value to X.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:state=0:activateifusednear=1`

3.1.2.4 *ACTIVATETARGET=X*

Description: Activates the entity's target entity and sets the activated entity's activation value to X.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:state=0:activatetarget=1`

3.1.2.5 *ALTTEXTURE=X*

Description: Switches the texture of the entity as defined during the level editor.

Range: X = 0 normal texture

X = 1 alternative texture

Example: `:state=0:alttexture=1`

3.1.2.6 *CARRYALL*

Description: This actions is design to be used by Lifts and allows the lift entity to carry all entities within the segment it occupies. Collision is switched OFF for the lift entity and the carried entities when this is called.

Example: `:state=0:carryall`

3.1.2.7 *COLOFF*

Description: Disables collision detection for the entity.

Example: `:state=0:coloff`

3.1.2.8 *COLON*

Description: Enables collision detection for the entity.

Example: `:state=0:`**`colon`**

3.1.2.9 *CULLEVENIFIMMOBILE=X*

Description: This sets the entity such that it will be 'culled' even if set to immobile when X=1.

Range: X = 0 – If entity is immobile it will not be culled,

X = 1 – If entity is immobile it will be culled.

Example: `:state=0:`**`cullevenifimmobile`**

3.1.2.10 *CULLMODE=X*

Description: This defines the mode of 'culling' to be used on the entity by the engine.

Range: X = 0 – standard entity culling,

X = 1 – entity culling based on entity size and calculated cull range (default)

Example: `:state=0:`**`cullmode`**`=1`

3.1.2.11 *CULLMODI=X*

Description: This is a multiplier of the calculated entity cull distance (default is 400) based on entity size.

Range: X = Any value.

Example: `:state=0:`**`cullmodi`**`=2`

3.1.2.12 *CULLRANGE=X*

Description: This defines the range at which the entity will be culled in units.

Range: X = Any value.

Example: `:state=0:`**`cullrange`**`=300`

3.1.2.13 *DECALPHAFADE=X*

Description: This decreases the alpha fade/transparency value of the entity. X is defined as the target alpha fade value.

Range: X = 0 to 100

Example: `:state=0:`**`decalphafade`**`=0`

3.1.2.14 *DECSHADERVARIABLE=X*

Description: This action decreases the shader variable by X. This action is not officially supported or documented and is only available for experimental use only.

Range: X = Any value.

Example: `:state=0:`**`decshadervariable`**`=1`

3.1.2.15 *DESTROY*

Description: This destroys the entity and removes it from the level. If re-spawn parameters have been set, these are now triggered.

Example: `:state=0:destroy`

3.1.2.16 *EMITFLASH*

Description: This causes the entity to emit a flash. The flash is defined by the setflashred, setflashblue, setflashgreen and setflashrange commands.

Example: `:state=0:emitflash`

3.1.2.17 *EMITFORCE=X*

Description: This causes the entity to emit a force of X as a pulse.

Range: X = Any value.

Example: `:state=0:emitforce=70`

3.1.2.18 *ENTITYDAMGEMULT=X*

Description: This modifies the damage done to the entity by X percent.

Range: X = Any value.

Example: `:state=0:entitydamagemult=50`

3.1.2.19 *ENTROTATEX=X*

Description: This rotates the entity around its x-axis by X degrees.

Range: X = -360° to 360°.

Example: `:state=1:entrotatex=5`

3.1.2.20 *ENTROTATEY=X*

Description: This rotates the entity around its y-axis by X degrees.

Range: X = -360° to 360°.

Example: `:state=1:entrotatey=5`

3.1.2.21 *ENTROTATEZ=X*

Description: This rotates the entity around its z-axis by X degrees.

Range: X = -360° to 360°.

Example: `:state=1:entrotatez=5`

3.1.2.22 *EXPLODE*

Description: This causes the entity to explode allowing it to cause damage. It does not destroy the entity.

Example: `:state=0:explode`

3.1.2.23 *FLOATRATE=X*

Description: This defines the height to which the entity floats. This action will cause the entity to oscillate up and down from its original position to a height of 2X units. The rate oscillation can not be changed using script commands.

Range: X = 0 to 2,000 (above 2,000 is possible, but this places entity over 20 segments high).

Example: `:state=0:floatrate=1`

3.1.2.24 *FLOORLOGIC*

Description: Prevents entity from leaving y (up-down) position.

Example: `:state=0:floorlogic`

3.1.2.25 *FORCEDAMAGEON=X*

Description: This defines whether the entity affected by force damage or not.

Range: X = 1 – entity is affected by force damage (the default for all entities),

X ≠ 1 - entity is unaffected by force damage.

Example: `:state=0:emitforce=70`

3.1.2.26 *FREEFROMPLR*

Description: This releases the entity from the player's camera.

Example: `:state=0:freefromplr`

3.1.2.27 *HIDE*

Description: This hides the entity from being displayed.

Example: `:state=0:hide`

3.1.2.28 *HIDESHADOW=X*

Description: This switches the entity's shadow parameter ON or OFF

Range: X = 0 – Switch shadow ON,

X = 1 – Switch shadow OFF.

Example: `:state=1:hideshadow=1`

3.1.2.29 *INCALPHAFADE=X*

Description: This increases the alpha fade/transparency value of the entity. X is defined as the target alpha fade value.

Range: X = 0 to 100

Example: `:state=0:incalphafade=100`

3.1.2.30 *INCSHADERVARIABLE=X*

Description: This action increases the shader variable by X. This action is not officially supported or documented and is only available for experimental use only.

Range: X = Any value.

Example: `:state=0:incshadervariable=1`

3.1.2.31 *INCSTATE=X*

Description: This increases the state value of the FPI by X.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:random=1:incstate=1`

3.1.2.32 *ISALTAMMO*

Description: This sets the ammo entity to provide alternative fire ammo.

Example: `:state=1:isaltammo`

3.1.2.33 *LINKTOPLR*

Description: This attaches the entity to the player's camera and uses the player's camera offsets.

Example: `:state=0:linktoplr`

3.1.2.34 *LINKXROTATION=X*

Description: This links the entities rotation around the x-axis to that of the player's x-axis rotation.

Range: X = 0 – Entity's rotation around x-axis is not linked to the player,

X = 1 – Entity's rotation around x-axis is linked to the player's.

Example: `:state=1:linkxrotation=1`

3.1.2.35 *LINKYROTATION=X*

Description: This links the entities rotation around the y-axis to that of the player's y-axis rotation.

Range: X = 0 – Entity's rotation around y-axis is not linked to the player,

X = 1 – Entity's rotation around y-axis is linked to the player's.

Example: `:state=1:linkyrotation=1`

3.1.2.36 *LINKZROTATION=X*

Description: This links the entities rotation around the z-axis to that of the player's z-axis rotation.

Range: X = 0 – Entity's rotation around z-axis is not linked to the player,

X = 1 – Entity's rotation around z-axis is linked to the player's.

Example: `:state=1:linkzrotation=1`

3.1.2.37 *LOGICBURST*

Description: This effectively sets the entity to 'always active' for a few moments (approximately 100 ms).

Example: `:state=0:`**`logicburst`**

3.1.2.38 *NOBULLETCOL*

Description: This switches the bullet collision for the entity OFF.

Example: `:state=1:`**`nobulletcol`**

3.1.2.39 *NONE*

Description: This performs no action.

Example: `:state=0:`**`none`**

3.1.2.40 *NOGRAVITY*

Description: This switches off the effect of gravity on the entity.

Example: `:state=0:`**`nogravity`**

3.1.2.41 *OFFSETANGLEX=X*

Description: This sets the entity's rotation around the x-axis (horizontal) by a fixed angle.

Range: X = -360 to 360

Example: `:state=1:o`**`ffsetanglex`**`=15`

3.1.2.42 *OFFSETANGLEY=X*

Description: This sets the entity's rotation around the y-axis (vertical) by a fixed angle.

Range: X = -360 to 360

Example: `:state=1:o`**`ffsetangley`**`=15`

3.1.2.43 *OFFSETANGLEZ=X*

Description: This sets the entity's rotation around the z-axis (line-of-sight) by a fixed angle.

Range: X = -360 to 360

Example: `:state=1:o`**`ffsetanglez`**`=15`

3.1.2.44 *RANDOMIZE*

Description: This sets the current seed for the random command to the timer. This can improve the randomness of the random features.

Example: `:state=0:`**`randomize`**

3.1.2.45 *ROTATEIY=X*

Description: This sets the entity to rotate around it's Y-axis by an increment of X degrees. For example, if the entity is already rotated 45° about the Y-axis, using *rotateiy=45* will rotate the entity to 90°. If the entity is set to always active, the rotation will be smooth. If the entity is not set to always active, the rotation will only be smooth if the player is close to the entity.

Range: X = -360 to 360 (larger angles can be used, but are not necessary)

Example: `:state=0:rotateiy=180`

3.1.2.46 *RUNDECAL=X*

Description: Sets the decal mode for the entity to X. If particles are disabled, the decal mode is set to 2.

Range: X = 1 - plays decal once and rotates it to constantly facing the player,

X = 2 – plays decal once and keeps the angle it was set at,

X = 3 – loops the decal and rotates it to constantly face the player,

X = 4 – loops the decal and keeps the angle it was set at,

X = 5 – plays the decal once facing up,

X = 6 – loops the decal facing up,

X = 7 – Sets the decal to use the decal particle settings.

Example: `:state=1:rundecal=3`

3.1.2.47 *RUNFPI=FilePath*

Description: Starts and runs a new *.fpi defined by X. The child.fpi runs in place of the parent.fpi, not in parallel. To return to the parent.fpi, the parent.fpi must be called by the child.fpi using "runfpi="

Range: FilePath = File path to new *.fpi.

Example: `:state=10:runfpi=MyGameScripts\shoot.fpi`

3.1.2.48 *RUNFPIDEFAULT=X*

Description: This stars and runs a new *.fpi as defined in the APPEAR, MAIN, DESTROY and SHOOT fields in the entity properties deinfed in the level editor. The new.fpi runs in place of the old.fpi, not in parallel. To return to the old.fpi, the "runfpidefaul=" actions must be called stating the correct value for X.

Range: X = 0 – run appear script,

X = 1 – run main script,

X = 2 – run destroy script,

X = 3 – run shoot script

Example: `:state=0:runfpidefault=1`

3.1.2.49 *SELECTSHADERVARIABLE=X*

Description: This action selects the shader variable. This action is not officially supported or documented and is only available for experimental use only.

Range: X = 1 to 4

Example: `:state=0:selectshadervariable=1`

3.1.2.50 *SETALPHAFADE=X*

Description: This sets the alpha fade/transparency property of the entity to X.

Range: X = 0 to 100

Example: `:state=0:setalphafade=100`

3.1.2.51 *SETBULLETCOLOFF=X*

Description: This sets whether the entity can be hit by bullets or not. This command is similar to nobulletcol, but setting X to zero switches bullet collision back on.

Range: X = 0 – entity can be hit by bullets,

X = 1 – entity cannot be hit by bullets.

Example: `:state=0:setbulletcoloff=1`

3.1.2.52 *SETENTITYIMMUNE=X*

Description: This sets whether the entity is immune to damage or not.

Range: X = 0 – entity can receive damage,

X = 1 – entity is immune to damage.

Example: `:state=0:setentityimmune=1`

3.1.2.53 *SETENTITYSPEED=X*

Description: This sets the entity's speed to X.

Range: X = Any value.

Example: `:state=0:setentityspeed=150`

3.1.2.54 *SETFLASHBLUE=X*

Description: This sets the amount of blue light in the flash.

Range: X = 0 - 255.

Example: `:state=0:setflashblue=225`

3.1.2.55 *SETFLASHGREEN=X*

Description: This sets the amount of green light in the flash.

Range: X = 0 - 255.

Example: `:state=0:setflashgreen=225`

3.1.2.56 *SETFLASHRANGE=X*

Description: This sets the range, in units, of light used for the flash. The default range is 600.

Range: X = Any value.

Example: `:state=0:setflashrange=150`

3.1.2.57 *SETFLASHRED=X*

Description: This sets the amount of red light in the flash.

Range: X = 0 - 255.

Example: `:state=0:setflashred=225`

3.1.2.58 *SETFORCEDAMAGE=X*

Description: This sets the damage the next force burst will cause.

Range: X = Any value.

Example: `:state=0:setforcedamage=225`

3.1.2.59 *SETIFUSED=EntityName*

Description: This sets the entity's "If Used" parameter to X.

Range: EntityName = Name of entity to activate if used.

Example: `:state=0:setifused=ExplodingBarrel`

3.1.2.60 *SETISOBJECTIVE*

Description: This makes the entity an objective to which a direction marker will point to. This is not the same as the IsObjective property within the FPSC editor.

Example: `:state=0:setisobjective`

3.1.2.61 *SETSHADERVARIABLE=X*

Description: This action sets the selected shader variable to X. This action is not officially supported or documented and is only available for experimental use only.

Range: X = Any value

Example: `:state=0:setshadervariable=1`

3.1.2.62 *SETOBJECTIVERANGE=X*

Description: This action sets the distance, in units, at which the objective marker appers/disappears.

Range: X = Any value in units.

Example: `:state=0:setobjectiverange=500`

3.1.2.63 *SETTARGET*

Description: This sets the entity's possible target to its actual target.

Example: `:state=0:settarget`

3.1.2.64 *SETTARGETNAME=EntityName*

Description: This sets the entity's target to X.

Range: EntityName = Target entity's name

Example: `:state=0:settargetname=EmptyBox`

3.1.2.65 *SETUSEKEY=EntityName*

Description: This sets the entity's "Use Key" parameter to X.

Range: EntityName = Key name

Example: `:state=0:settusekey=BigRustyKey`

3.1.2.66 *SHAPEDECAL*

Description: This sets the decal size to the size of the entity.

Example: `:state=0:shapedecal`

3.1.2.67 *SHOW*

Description: This displays the entity on the screen (if visible).

Example: `:state=0:show`

3.1.2.68 *SPAWNOFF*

Description: This sets the entity's "Spawn at Start" parameter to NO.

Example: `:state=0:spawnoff`

3.1.2.69 *SPAWNON*

Description: This sets the entity's "Spawn at Start" parameter to YES.

Example: `:state=0:spawnon`

3.1.2.70 *SPINRATE=X*

Description: This sets the entity's spin rate to X.

Range: X = 0 to any value, although setting higher than current FPS makes the entity appear strange.

Example: `:state=0:spinrate=1`

3.1.2.71 *STATE=X*

Description: This sets the FPI state value to X.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:state=0:state=1`

3.1.2.72 *TRIGGERFORCE=X*

Description: This creates a force sphere around the entity and applied it to all entities around it. All entities within the 'blast radius' are damaged and have a force applied to them.

The blast radius (in units) is calculated as: *BlastRadius = 4 * X*.

The force applied to the affected entities is calculated as:

*Force = 75 * (BlastRadius-DistanceFromBlast) / BlastRadius*

The damage applied to the affected entities is calculated as:

Damage = Force / 50

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:triggerforce=200`

3.2 Variables and Timers

These conditions and actions are applicable specifically to variables (internal, global or local) and timers (global or entity based).

3.2.1 CONDITIONS

3.2.1.1 *ETIMERGREATER=X*

Description: Is true if the entity's timer is greater than X milliseconds. This timer cannot be seen by other entities.

Range: X > 0 (no upper limit)

Example: `:etimergreater=1000:state=1`

3.2.1.2 *SVAREQUAL=VarName X*

Description: Is true if the set-up variable VarName equals X.

Range: X = 0 to any value.

Example: `:svarequal=Difficulty Easy:activated=1:`

3.2.1.3 *TIMERGREATER=X*

Description: Is true if the game's timer is greater than X milliseconds. This timer can be seen by any entity.

Range: X >0 (no upper limit)

Example: `:timergreater=1000:state=1`

3.2.1.4 *VAREQUAL=X and VAREQUAL=VarName X*

Description: Is true if the variable defined by "VarName" equals X. If no "VarName" is provided, the current variable, defined by *globalvar* or *localvar,* is tested.

Range: VarName = Variable name (global or local)

X = $3.4 \times 10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example 1: `:varequal=PrimaryObjective 2:state=1`

Example 2:
```
:always:globvar=1
:varequal=5:state=1
```

3.2.1.5 *VARGREATER=X and VARGREATER=VarName X*

Description: Is true if the variable defined by "VarName" is greater than X. If no "VarName" is provided, the current set variable, defined by *globalvar* or *localvar,* is tested.

Range: VarName = Variable name (global or local)

X = $3.4 \times 10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example 1: `:vargreater=PrimaryObjective 2:state=1`

Example 2:
```
:always:globvar=1
:vargreater=5:state=1
```

3.2.1.6 *VARLESS=X and VARLESS=VarName X*

Description: Is true if the variable defined by "VarName" is less than . If no X "VarName" is provided, the current set variable, defined by *globalvar* or *localvar,* is tested.

Range: VarName = Variable name (global or local)

X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example 1: `:varless=PrimaryObjective 2:state=1`

Example 2: `:always:globvar=1`

`:varless=5:state=1`

3.2.1.7 *VARNOTEQUAL=X and VARNOTEQUAL= VarName X*

Description: Is true if the variable defined by "VarName" is not equal to X. If no "VarName" is provided, the current set variable, defined by *globalvar* or *localvar,* is tested.

Range: VarName = Variable name (global or local)

X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example 1: `:varnotequal=PrimaryObjective 2:state=1`

Example 2: `:always:globvar=1`

`:varnotequal=5:state=1`

3.2.2 ACTIONS

3.2.2.1 *ADDVAR=VarName Y*

Description: This adds Y to variable called VarName. The variable can be global or local.

Range: VarName = Variable name (internal, global or local)

Y = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:addvar=KillCount 1`

3.2.2.2 *COS=VarName Degrees*

Description: This sets the variable VarName to the cosine of Degrees.

Range: VarName = Any valid local or global variable.

Degrees = -360° to 360°

Example: `:state=0:cos=Height 180`

3.2.2.3 *DECVAR=X*

Description: This reduces the current set variable by X. To use this action with named created using *dimvar*, use the *setvar* action prior to this action.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example:
```
:always:setvar=HealthPacks
:state=0:decvar=1
```

3.2.2.4 *DIMLOCALVAR=VarName*

Description: This creates a local variable that can only be accessed by the FPI script.

Range: VarName = Your variable name.

Example: `:state=0:dimlocalvar=PassWordAttempts`

3.2.2.5 *DIMVAR=VarName*

Description: This creates a global variable that can only be accessed by all FPI scripts.

Range: VarName = Your variable name.

Example: `:state=0:dimvar=KillCount`

3.2.2.6 *DIVVAR=VarName Y*

Description: This divides the variable called VarName by Y. the variable can be local or global.

Range: VarName = Variable name (internal, global or local)

Y = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=1:divvar=1 10`

3.2.2.7 *ETIMERSTART*

Description: This starts the entity's own timer. This timer cannot be seen by any other entity.

Example: `:state=0:etimerstart`

3.2.2.8 *GLOBALVAR=X*

Description: This sets the FPI to look at global variable X.

Range: X = 0 to 99, or a named variable

Example: `:state=0:globalvar=1`

3.2.2.9 *INCVAR=X*

Description: This increases the current set variable by X. To use this action with named created using *dimvar*, use the *setvar* action prior to this action.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example:
```
:always:setvar=HealthPacks
:state=0:incvar=1
```

3.2.2.10 *LOCALVAR=X*

Description: This sets the FPI to look at local variable X.

Range: X = 0 to 99, or named variable

Example: `:state=0:localvar=1`

3.2.2.11 *MAKESVAR=X Y Z*

Description: This defines a new set up variable within the setup.ini and provides a default value.

Range: X = Variable name.

Y = Default value.

Z = Setup.ini line name.

Example: `:state=1:makesvar=JumpHeight 50 PlayerJumpHeight`

3.2.2.12 *MODVAR=VarName Y*

Description: This performs a modulus operation to the variable called VarName by Y. the variable can be local or global. The *modvar* action is similar to the *wrapvar* action with the exception that Y defines the 'wrap around' point. For example *modvar=RotateAngle 360* is the same is *wrapvar=RotateAngle*.

Range: VarName = Variable name (internal, global or local)

Y = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:modvar=Seconds 60`

3.2.2.13 *MULVAR=VarName Y*

Description: This multiplies the variable called VarName by Y. the variable can be local or global.

Range: VarName = Variable name (internal, global or local)

Y = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=1:mulvar=$PH 10`

3.2.2.14 *RESETGLOBALSONRELOAD=X*

Description: This defines whether or not the default global variables (numbered 0 to 99) are reset at the start of each level.

Range: X = 0 – default global variables are not reset between levels,

X = 1 – default global variables are not reset between levels.

Example: `:state=0:resetglobalsonreload=0`

3.2.2.15 *SAVESVARS*

Description: This writes all setup variables to the setup.ini.

Example: `:state=1:savesvars`

3.2.2.16 *SETSVARLINE=X Y*

Description: This defines which setup.ini line to write a setup variable to.

Range: X = Setup variable name,

Y = Setup.ini line.

Example: `:state=1:setsvarline=VolMusic MusicVolume`

3.2.2.17 *SETSVARTOGUI=X Y*

Description: This defines a relationship between a GUI component and a setup variable so that the GUI component value is stored within the setup variable.

Range: X = Setup variable name,

Y = GUI component name (choice slider, slider or checkbox)

Example: `:state=1:setsvartogui=MusicVolVar MusicVolSdr`

3.2.2.18 *SETSVARVALUE=X Y*

Description: This sets the value of a setup variable.

Range: X = Setup variable,

Y = Value. (using custom "%" or internal "$" variables will not work)

Example: `:state=1:setsvarvalue=MusicVolVar 50`

3.2.2.19 *SETVAR=Y and SETVAR=VarName Y*

Description: This sets the variable called VarName to Y. the variable can be local or global. If no "VarName" is provided, the current set variable, defined by *globalvar* or *localvar,* is used.

Range: VarName = Variable name (internal, global or local)

Y = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example 1: `:state=0:setvar=PlrHasUberKnife 1`

Example 2: `:always:globvar=1`

`:state=5:setvar=1`

3.2.2.20 *SETVARRND=X and SETVARRND=VarName X*

Description: This either sets the current standard global variable (0-99) or set a custom variable to a random number between 0 and X.

Range: VarName = Variable name (custom variables only),

X = Any value.

Example 1: `:state=0:setvarrnd=HealthBonus 17`

Example 2: `:always:globvar=1`

`:state=5:setvarrnd=10`

3.2.2.21 *SIN=VarName Degrees*

Description: This sets the variable VarName to the sine of Degrees.

Range: VarName = Any valid local or global variable.

Degrees = -360° to 360°

Example: `:state=0:sin=Height 180`

3.2.2.22 *SUBVAR=VarName Y*

Description: This subtracts Y from the variable called VarName. The variable can be global or local.

Range: VarName = Variable name (internal, global or local)

Y = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:subvar=EnemiesLeft 1`

3.2.2.23 *TIMERSTART*

Description: This starts the global timer. This timer can be seen by all entities.

Example: `:activated=1:timerstart`

3.2.2.24 *WRAPVAR=VarName Y*

Description: This performs a modulus operation to the variable called VarName by 360. the variable can be local or global. For example *wrapvar=RotateAngle* will set *RotateAngle* to 5 if it originally equalled 365.

Range: VarName = Variable name (internal, global or local)

Example: `:state=0:wrapvar=$CAX`

3.3 Trigger Zones

These conditions and actions are applicable specifically to trigger zones.

3.3.1 CONDITIONS

3.3.1.1 *ANYKEYWITHINZONE=X*

Description: Is true if the entity defined by the "Use Key" parameter is within the zone and X is 1, or if the "Use Key" entity is not in the zone and X is not 1.

Range: X = 1 – returns true if "Use Key" entity is within the zone.

X ≠ 1 – returns true if "Use Key" entity is not within the zone.

Example: `:anykeywithinzone=1:state=10`

`:anykeywithinzone=0:state=20`

3.3.1.2 *ANYWITHINZONE=X*

Description: Is true if any entity is within the zone and X is 1, or if the no entity is within the zone and X is not 1.

Range: X = 1 – returns true if any entity is within the zone.

X ≠ 1 – returns true if no entities are within the zone.

Example: `:anywithinzone=1:state=10`

`:anywithinzone=0:state=20`

3.3.1.3 *ENTITYWITHINZONE*

Description: This actions is meant to detect an entity within a zone and set it as a target. This action, however, is not coded within the FPSC and so does not work. It is included here in case the action is added at a later date.

3.3.1.4 *PLRWITHINZONE=X*

Description: Is true if the player is within the zone and X is 1, or if the player is not within the zone and X is 0. If the trigger zone is set as an objective then the player's objective is triggered.

Range: X = 1 – returns true if the player is within the zone.

X = 0 – returns true if the player is not within the zone.

Example: `:plrwithinzone=1:state=10`

`:plrwithinzone=0:state=20`

3.3.2 ACTIONS

3.3.2.1 *ACTIVATEALLINZONE=X*

Description: This activates all entities within the zone and set their activation values equal to X.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:state=0:activateallinzone=1`

3.4 NPCs (Enemies, Allies & Neutrals)

These conditions and actions are applicable specifically to Non-Player Characters (NPCs). This section does not cover *Animations*, *Markers*, *Waypoints* or *Dark AI*, all of which have their own section.

3.4.1 CONDITIONS

3.4.1.1 *HASWEAPON=X*

Description: Is true if the character has a weapon and X is 1 or if the character does not have a weapon and X is 0.

Range: X = 1 – returns true if the entity has a weapon.

X = 0 – returns true if the entity does not have a weapon.

Example: `:hasweapon=1:shootplr`

3.4.1.2 *HEADANGLEGREATER=X*

Description: Is true when the angle of the character's head is greater than X degrees. Positive angles represent anti-clockwise movement, negative angles represent clockwise movement.

Range: X = -360 to 360 (larger angles can be used, but are not necessary)

Example: `:headanglegreater=90:resethead`

3.4.1.3 *HEADANGLELESS=X*

Description: Is true when the angle of the character's head is less than X degrees. Positive angles represent anti-clockwise movement, negative angles represent clockwise movement.

Range: X = -360 to 360 (larger angles can be used, but are not necessary)

Example: `:headangleless=90:resethead`

3.4.1.4 *HEALTH=X*

Description: Is true when character's health equals X

Range: X = 0 to maximum entity health

Example: `:health=1:activateifused`

3.4.1.5 *HEALTHGREATER=X*

Description: Is true when character's health is greater than X

Range: X = 0 to maximum entity health

Example: `:healthgreater=10:activateifused`

3.4.1.6 *HEALTHLESS=X*

Description: Is true when character's health is less than X

Range: X = 0 to maximum entity health

Example: `:healthless=10:activateifused`

3.4.1.7 *IFWEAPON=X*

Description: Is true when the weapon being used by the character is ready and X is 1 or if the character's weapon is not ready and X is 0. "Ready" is defined as the character having ammo or the weapon not requiring any realoading. If the character is firing the weapon this condition returns false.

Range: X = 1 – returns true if the character's weapon is ready.

X = 0 – returns true is the character's weapon is not ready.

Example: `:ifweapon=1:shootplr`

3.4.1.8 *LOSETARGET=X*

Description: Is true if the character has lost the target for more than X attempts.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:losetarget=100:state=1`

3.4.1.9 *NEARACTIVATABLE=X*

Description: Is true when an entity is within 70 units and the activation value for the entity is X. If true the activatable entity is set as a possible target.

Range: X = 0 to 100,000,000 (above 100,000,000 causes variable problems)

Example: `:nearactivatable=0:state=1`

3.4.1.10 *NEWWEAPONCANBESEEN*

Description: Is true if the entity can see a better weapon. Any weapons visible are compared with respect to their damage parameter. If true, the weapons is set as a possible target for the entity.

Example: `:newweaponcanbeseen:state=1`

3.4.1.11 *NOISEHEARD*

Description: Is true if the entity can hear a noise. If true the sound is set as a possible target for the entity.

The engine determines this by comparing the strength of the sound to the distance between the entity and the sound using the following test: 10 x SoundStrength > Distance.

Example 1: SoundStrength = 6, Distance = 55

10 x SoundStrength = 60 (> Distance = Yes, sound heard)

Example 2: SoundStrength = 6, Distance = 100

10 x SoundStrength = 60 (> Distance = No, sound not heard)

Any sound heard within 50 units is ignored.

Example: `:noiseheard:state=1`

3.4.1.12 *QUANTITY=X*

Description: Is true when the number of ammo clips currently carried by the entity equals X.

Range: X = 0 to any value.

Example: :**quantity**=1:state=1

3.4.1.13 *RATEOFFIRE*

Description: Is true if an entity's "Rate of fire" parameter is ≥ 100, or is randomly true if the following formula equals 0: RandomValue x (100-RateOfFire), where the RandomValue can be any fraction between 0 and 1.

Example: :**rateoffire**:state=1

3.4.1.14 *SHOTDAMAGE=X*

Description: Is true if the shot damage received by an entity exceeds X. The source of the shot damage is also set as a possible target for the entity.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: :**shotdamage**=50:state=1

3.4.1.15 *SHOTDAMAGETYPE=X*

Description: Is true if the limb type shot was equal to X. The source of the shot damage is also set as a possible target for the entity.

Range: X = 1 – Head

X = 2 – Body

X = 3 – Left Arm

X = 4 – Right Arm

X = 5 – Left Leg

X = 6 – Right Leg

Example: :**shotdamagetype**=1:state=1

3.4.1.16 *SPEED=X*

Description: Is true if the entity's speed equals X.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: :**speed**=10:state=1

3.4.1.17 *VELOCITYGREATER=X*

Description: Is true if the speed of the entity in any of the three axes is greater than X. This will not return true if the root-sum-square of the entity's speed (RSS speed) is greater than X and the individual axes speeds are less than X.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: :**velocitygreater**=10:state=1

3.4.2 ACTIONS

3.4.2.1 *ADDHEALTH=X*

Description: This adds X amount of health to the entity.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:addhealth=1`

3.4.2.2 *BLOODSPLASH=X*

Description: This splashes a blood-scorch around the entity in 6 different directions with a distance of X units.

Range: X = 0 to 540,000 (maximum distance possible in a level)

Example: `:state=0:bloodsplash=50`

3.4.2.3 *BLOODSPURT=X*

Description: This causes blood to spurt out of the entity and collide with the level. X defines where the blood spurs from.

Range: X = 1 – Last damaged position,

X = 2 – Head position,

X ≥ 3 – Centre of model, blood falls vertically down.

Example: `:state=0:bloodspurt=1`

3.4.2.4 *CHOOSESTRAFE*

Description: This randomly sets the entity to strafe left or right.

Example: `:state=0:choosestrafe`

3.4.2.5 *COLLECTTARGET*

Description: This makes the entity collect the current target (if collectable). If the collectable is a gun, the entity chooses this gun to use.

Example: `:state=0:collecttarget`

3.4.2.6 *ENTITYACCMULT=X*

Description: This modifies the accuracy of the character when firing their weapon by X percent.

Range: X = Any value.

Example: `:state=0:entityaccmult=50`

3.4.2.7 *FORCEBACK*

Description: This action does not perform any action within the engine although there is a reference to it within the code. The action is shown here as a place-holder in case it is used in the future.

3.4.2.8 *FORCEBOUNCE=X*

Description: This forces the entity in the reverse x and z axes if X is non-zero.

Range: X = 0 – no force is applied to the entity.

X > 0 – any force applied to the entity in the x or z axes is reversed and halved.

Example: `:state=0:forcebounce=1`

3.4.2.9 *FORCEFORE*

Description: This action does not perform any action within the engine although there is a reference to it within the code. The action is shown here as a place-holder in case it is used in the future.

3.4.2.10 *FORCELEFT*

Description: This action does not perform any action within the engine although there is a reference to it within the code. The action is shown here as a place-holder in case it is used in the future.

3.4.2.11 *FORCERIGHT*

Description: This action does not perform any action within the engine although there is a reference to it within the code. The action is shown here as a place-holder in case it is used in the future.

3.4.2.12 *FREEZE*

Description: Prevents the entity from moving.

Example: `:state=0:freeze`

3.4.2.13 *HEADSHOT=X*

Description: This defines the height of the entity affected by head-shots. Setting X to 1, assumed the default head height of 43 units.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:headshot=1`

3.4.2.14 *HEADSHOTDAMAGE =X*

Description: This defines the damaged received by an entity if it receives a headshot.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:headshotdamage=99`

3.4.2.15 *HIDELIMB=X*

Description: This hides the character's limb, identified by X. Note that this can only be called as part of the character's appear script.

Range: X = 1 - Head,

X = 2 – Left clavicle,

X = 3 – Right clavicle,

X = 4 – Left foot,

X = 5 – Right foot,

X = 6 – Left thigh,

X = 7 – Right thigh,

X = 8 – Left hand,

X = 9 – Right hand,

X = 10 – Left forearm,

X = 11 – Right forearm,

X = 12 – Left upper arm,

X = 13 – Right upper arm,

X = 14 – Spine base,

X = 15 – Spine middle,

X = 16 – Spine top,

X = 17 – Neck.

Example: `:state=0:hidelimb=1`

3.4.2.16 *HIDESHADDOW=X*

Description: This hides the NPC's 'blob' like shadow if X is set to 1.

Range: X = 1 (if X is set less than 1 the shadow is not hidden.

Example: `:state=0:hideshaddow=1`

3.4.2.17 *LOOKATPLR*

Description: This rotates the entity's head to face the player.

Example: `:state=0:lookatplr`

3.4.2.18 *LOOKATTARGET*

Description: This rotates the entity's head to face the current target.

Example: `:state=0:lookattarget`

3.4.2.19 *MOVEBACK=X*

Description: This moves the entity backwards at the rate of X where X is defined as a percentage of the entity's speed (i.e. if X=50, movement is 50% of entity's speed). Is the percentage results in less than 2 units of movement, the entity is moved 2 units.

Range: X = Any value.

Example: `:state=0:moveback=50`

3.4.2.20 *MOVEFORE=X*

Description: This moves the entity forward at the rate of X where X is defined as a percentage of the entity's speed (i.e. if X=50, movement is 50% of entity's speed). Is the percentage results in less than 2 units of movement, the entity is moved 2 units.

Range: X = Any value.

Example: `:state=0:movefore=50`

3.4.2.21 *MOVEUP=X*

Description: : This moves the entity upwards at the rate of X where X is defined as a percentage of the entity's speed (i.e. if X=50, movement is 50% of entity's speed).

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:moveup=50`

3.4.2.22 *MUTATE=X Y*

Description: : This adjusts the character's limbs by a percentage between X and Y.

Range: X = Any value,

Y = Any value.

Example: `:state=0:mutate=50 100`

3.4.2.23 *NOROTATE=X*

Description: This sets the entity's ability to rotate.

Range: X = 0 – entity can rotate

X > 0 – entity cannot rotate.

Example: `:state=0:norotate=1`

3.4.2.24 *PIVOTRANDOM=X*

Description: This randomly rotates the entity about the Y axis between -X and X degrees.

Range: X = 0 to 360

Example: `:state=0:pivotrandom=45`

3.4.2.25 *RAGDOLL*

Description: This switches the character's ragdoll ability ON. This is only useable with models that have ragdoll ability.

Example: `:state=0:`**`ragdoll`**

3.4.2.26 *RELOADWEAPON*

Description: This reloads the current weapon that the character is carrying. Only works if the character is carrying a weapon that needs reloaded (i.e. if the weapon is full it will not reload).

Example: `:state=0:`**`reloadweapon`**

3.4.2.27 *RESETHEAD=X*

Description: This sets the character's head angle to 0 and sets the character's head-down angle to X.

Range: X = -360 to 360 (larger angles can be used, but are not necessary)

Example: `:state=0:resethead=0`

3.4.2.28 *ROTATEHEAD=X*

Description: This sets the character's head angle to X. Positive angles represent anti-clockwise movement, negative angles represent clockwise movement.

Range: X = -360 to 360 (larger angles can be used, but are not necessary)

Example: `:state=0:rotatehead=45`

3.4.2.29 *ROTATEHEADRANDOM=X*

Description: This sets the character's head angle to a random value between –X and X degrees. Positive angles represent anti-clockwise movement, negative angles represent clockwise movement.

Range: X = -360 to 360 (larger angles can be used, but are not necessary)

Example: `:state=0:rotateheadrandom=90`

3.4.2.30 *ROTATETOPLR*

Description: This sets the character's head angle to 0 and rotates the character to face the player.

Example: `:state=0:rotatetoplr`

3.4.2.31 *ROTATETOTARGET*

Description: This rotates the entity to face towards its target.

Example: `:state=0:rotatetotarget`

3.4.2.32 *ROTATEY=X*

Description: This sets the entity to rotate around it's Y-axis by X degrees.

Range: X = -360 to 360.

Example: `:state=0:rotatey=45`

3.4.2.33 *RUNFORE=X*

Description: This moves the entity forward at the rate of X where X is defined as a percentage of the entity's speed (i.e. if X=50, movement is 50% of entity's speed). Is the percentage results in less than 2 units of movement, the entity is moved 2 units.

Range: X = $3.4 \times 10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:runfore=50`

3.4.2.34 *SCALE=X or SCALE=X Y*

Description: This scales the character by X percent, or between X percent plus a random percentage between 0 and Y.

Range: X = Any value,

Y = Any value.

Examples: `:state=0:scale=95`

`:state=0:scale=95 10`

3.4.2.35 *SCALELIMB=X Y*

Description: This scales the character's limb, identified by X, by Y percent.

Range: X = 1 - Head,

X = 2 – Left clavicle,

X = 3 – Right clavicle,

X = 4 – Left foot,

X = 5 – Right foot,

X = 6 – Left thigh,

X = 7 – Right thigh,

X = 8 – Left hand,

X = 9 – Right hand,

X = 10 – Left forearm,

X = 11 – Right forearm,

X = 12 – Left upper arm,

X = 13 – Right upper arm,

X = 14 – Spine base,

X = 15 – Spine middle,

X = 16 – Spine top,

X = 17 – Neck.

Y = Any value.

Example: `:state=0:scalelimb=1 200`

3.4.2.36 *SETHEALTH=X*

Description: This sets the entity's health value to X.

Range: X = $3.4 \times 10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:sethealth=100`

3.4.2.37 *SHOOTPLR*

Description: This runs the shoot script defined during the character set up.

Example: `:state=0:shootplr`

3.4.2.38 *SHOWLIMB=X*

Description: This displays the character's limb, identified by X.

Range: X = 1 - Head,

X = 2 – Left clavicle,

X = 3 – Right clavicle,

X = 4 – Left foot,

X = 5 – Right foot,

X = 6 – Left thigh,

X = 7 – Right thigh,

X = 8 – Left hand,

X = 9 – Right hand,

X = 10 – Left forearm,

X = 11 – Right forearm,

X = 12 – Left upper arm,

X = 13 – Right upper arm,

X = 14 – Spine base,

X = 15 – Spine middle,

X = 16 – Spine top,

X = 17 – Neck.

Example: `:state=0:`**`showlimb`**`=1`

3.4.2.39 *STRAFE=X*

Description: This makes the character move in a direction defined by X, where X is the rotation angle of the character.

Range: X = -360 to 360.

Example: `:state=0:`**`strafe`**`=90`

3.4.2.40 *SUBHEALTH=X*

Description: This removes X amount of health from the character.

Range: X = 0 to entity's maximum health.

Example: `:state=0:`**`subhealth`**`=10`

3.4.2.41 *SUSPEND*

Description: This deactivates the entity, but does not destroy the entity so that it may behave as a corpse.

Example: `:state=0:`**`suspend`**

3.4.2.42 *TALK=FilePath*

Description: This makes the character talk using the *wav file located by X.

Range: FilePath = File path to *.wav file.

Example: `:state=0:`**`talk`**`=gamecore\speeches\hello.wav`

3.4.2.43 *TALKORDERED=FilePath*

Description: This makes the character talk through a sequence numbered from 2 up to 9 *.wav files played in order, each time the action is called.

Range: FilePath = File path to first *.wav

Example: `:state=0:talkordered=gamecore\speeches\hello2.wav`

3.4.2.44 *TALKRANDOM=FilePath*

Description: This makes the character talk through a sequence numbered from 2 up to 9 *.wav files played in a random order, each time the action is called.

Range: FilePath = File path to first *.wav file

Example: `:state=0:talkrandom=gamecore\speeches\hello2.wav`

3.4.2.45 *USEWEAPON=X*

Description: This makes the character use their weapon. X defines the fire strength of the weapon, if X is 0 then a default value of 100 is used.

Range: X = $3.4x10^{\pm 38}$ (float variable range from Dark Basic Pro)

Example: `:state=0:useweapon=0`

3.5 Waypoints and Markers

These conditions and actions are applicable specifically entities following waypoints and markers.

3.5.1 CONDITIONS

3.5.1.1 *IFMARKER=X*

Description: Is true if there is a marker previously dropped by the entity in the level and X is 1 and if there are no markers dropped by the entity and X is 0.

Range: X = 0 – returns true if no markers have been dropped,

X = 1 – returns true if markers have been dropped.

Example: `:ifmarker=1:resetmarkers`

3.5.1.2 *IFPLRTRAIL=X*

Description: Is true if there is a marker previously dropped by the player in the level and X is 1 and if there are no markers dropped by the player and X is 0.

Range: X = 0 – returns true if no markers have been dropped,

X = 1 – returns true if markers have been dropped.

Example: `:ifplrtrail=1:rotatetoplr`

3.5.1.3 *REACHTARGET*

Description: Is true if the entity is within 5 units of its target.

Example: `:reachtarget:state=1`

3.5.1.4 *WAYPOINTSTATE=X*

Description: Is true when the waypoint state of the entity equals X.

Range: X = 1 - the entity is looking for the nearest waypoint marker to start from.

X = 2 - the entity is following a waypoint line to its current waypoint marker.

X = 3 - the entity has reached the waypoint marker and needs to decide what to do.

X = 4 - the entity has reached the waypoint marker and the path splits off in more than one other direction.

X = 5 - the entity has reached the very end of the current waypoint structure and requires a decision to be made.

X = 999 – the entity has been placed in zero-waypoint mode where the entity simply ignores waypoints.

Example: `:waypointstate=999:state=1`

3.5.2 ACTIONS

3.5.2.1 *DROPMARKER*

Description: This adds a marker to the entity at its current location. Up to 50 markers can be added per entity.

Example: `:state=0:`**`dropmarker`**

3.5.2.2 *FOLLOWPLR*

Description: This makes the entity follow the player. The entity will follow the markers dropped by the player and not simply head towards the player's current location. To make the entity head towards the player, the player's markers must first be reset. It should be noted that the Dark AI action "aifollowplr" is designed to be a much more improved action over this one.

Example: `:state=0:`**`followplr`**

3.5.2.3 *MOVETOTARGET=X*

Description: This makes the entity move towards it's current target. X defines whether the entity walks or runs.

Range: X = 0 – Entity will walk towards the target.

X = 1 – Entity will run towards the target

Example: `:state=0:`**`movetotarget`**`=1`

3.5.2.4 *NEXTMARKER*

Description: This sends the entity to move towards the marker in the trail.

Example: `:reachtarget:`**`nextmarker`**

3.5.2.5 *RESETMARKERS*

Description: This clears all of the markers dropped by the entity.

Example: `:ifmarker=1:`**`resetmarkers`**

3.5.2.6 *WAYPOINTNEXT*

Description: This sets the entity to move towards to next waypoint along a waypoint path.

Example: `:waypointstate=3:`**`waypointnext`**

3.5.2.7 *WAYPOINTPREV*

Description: This sets the entity to move towards to previous waypoint along a waypoint path.

Example: `:waypointstate=3:`**`waypointprev`**

3.5.2.8 *WAYPOINTRANDOM*

Description: This sets the entity to move towards a random waypoint along a waypoint path.

Example: `:waypointstate=3:`**`waypointrandom`**

3.5.2.9 *WAYPOINTREVERSE*

Description: This reverses the order in which the entity moves through the waypoints along a waypoint path.

Example: `:waypointstate=3:`**`waypointreverse`**

3.5.2.10 *WAYPOINTSTART*

Description: This sets the entity to move towards the start of the nearest waypoint path.

Example: `:waypointstate=3:`**`waypointstart`**

3.5.2.11 *WAYPOINTSTOP*

Description: This stops the entity from navigating along a waypoint path.

Example: `:waypointstate=3:`**`waypointstop`**

3.6 Animations

These conditions and actions are applicable specifically to animations.

3.6.1 CONDITIONS

3.6.1.1 *ANIMATIONOVER=X*

Description: Is true when animation X is within 5 frames of its ending.

Range: See animation list.

Example: `:animationover=66:shootplr`

3.6.1.2 *FRAMEATEND=X*

Description: Is true when animation X is at its end.

Range: See animation list.

Example: `:frameatend=1:shootplr`

3.6.1.3 *FRAMEATSTART=X*

Description: Is true when animation X is at the start.

Range: See animation list.

Example: `:frameatstart=1:freeze`

3.6.1.4 *FRAMEBEYOND=X Y*

Description: Is true when animation X is beyond frame Y

Range: X = See animation list.
Y > 0

Example: `:framebeyond=1 5:freeze`

3.6.1.5 *FRAMEWITHIN=X Y*

Description: Is true when animation X is within frame Y

Range: X = See animation list.
Y > 0

Example: `:framewithin=1 5:freeze`

3.6.2 ACTIONS

3.6.2.1 *ADVFRAME=X*

Description: This advances the animation by X% of the remaining frames. For example, if an animation had 10 frames left, advframe=50 would advance the animation by 5 frames.

Range: X = 0 to 100

Example: `:state=0:advframe=20`

3.6.2.2 *ANIMATE=X*

Description: Sets the animation of the entity to X.

Range: X = See animation list.

Example: `:state=0:animate=3`

3.6.2.3 *ANIMATIONNORMAL*

Description: This sets the entity's animation to play forwards.

Example: `:state=0:animationnormal`

3.6.2.4 *ANIMATIONREVERSE*

Description: This sets the entity's animation to play in reverse.

Example: `:state=0:animationnormal`

3.6.2.5 *DECFRAME=X*

Description: This steps the animation X back one frame.

Range: X = See animation list.

Example: `:state=0:decframe=3`

3.6.2.6 *INCFRAME=X*

Description: This steps the animation X forward one frame.

Range: X = See animation list.

Example: `:state=0:incframe=3`

3.6.2.7 *SETANIMATESPEED=X*

Description: This sets the entity's animation speed and set's it independent of the entity's speed.

Range: X = Any value.

Example: `:state=0:setanimatespeed=75`

3.6.2.8 *SETFRAME=X*

Description: This sets the entity's animation X to the start frame.

Range: X = See animation list.

Example: `:state=0:setframe=3`

3.7 DarkAI

These conditions and actions are applicable specifically to the features added as part of the Dark AI add on.

3.7.1 CONDITIONS

3.7.1.1 *AIACTION=X*

Description: Is true if the character's action state is equal to X.

Range: X = 0 – is idle

X = 1 – is following player

X = 2 – is in cover

X = 3 – has stopped moving

X = 4 – has stopped following player

X = 5 – is moving to sound

X = 6 – is moving away from sound

X = 7 – is attacking

X = 8 – is moving to point

X = 9 – is moving to cover based on enemy positions

X = 10 – is moving to nearest cover regardless of enemy positions

Example: `:aiaction=0:movetotarget`

3.7.1.2 *AIATCOVER=X*

Description: Is true if the character is in cover and X is 1 or if the character is exposed and X is 0.

Range: X = 0 – true if exposed

X = 1 – true if in cover

Example: `:aiatcover=0:aimovetocover`

3.7.1.3 *AIATPOINT=X Y*

Description: Is true if the character is within 5 units of point X and Y is 1. A maximum of points can exist within the whole level. Points are global markers that all DAI can see.

Range: X = 0 to 50

Y = 1

Example: `:aiatpoint=1 1:movetotarget`

3.7.1.4 *AICALLED=X*

Description: Is true if this character is called by another character and X is 1 or the character was not called by another character and X is 0.

Range: X = 1 – true if a character called this character,

X = 0 – true if no character called this character.

Example: `:aicalled=1:airespondtocall`

3.7.1.5 *AICALLEDBYPLR=X*

Description: Is true if this character is called by the player and X is 1 or the character was not called by the player and X is 0.

Range: X = 1 – true if the player called this character,

X = 0 – true if the player did not call this character.

Example: `:aicalledbyplr=1:airespondtocall`

3.7.1.6 *AICANSHOOT=X*

Description: Is true if the character can see an enemy and X is 1 or the character cannot see an enemy and X is 0.

Range: X = 1 – true if the character can see an enemy,

X = 0 – true if the character cannot see an enemy.

Example: `:aicanshoot=1:state=1`

3.7.1.7 *AIHASTARGET=X*

Description: Is true if the character has a target and X is 1 or the character does not have a target and X is 0.

Range: X = 1 – true if the character has a target,

X = 0 – true if the character does not have a target.

Example: `:aihastarget=1:state=1`

3.7.1.8 *AIHEARDSOUND=X*

Description: Is true if the character heard a sound within X units.

Range: X = Any value.

Example: `:aiheardsound=100:state=1`

3.7.1.9 *AITARGETDISTFURTHER=X*

Description: Is true if the character's target is further than X units.

Range: X = 0 to 540,000

Example: `:aitargetdistfurther=100:state=1`

3.7.1.10 *AITARGETDISTWITHIN=X*

Description: Is true if the character's target is within than X units.

Range: X = 0 to 540,000

Example: `:aitargetdistwithin=100:state=1`

3.7.1.11 *AITEAM=X*

Description: Is true if the character is on team X.

Range: X = 1 to 20

Example: `:aiteam=1:state=1`

3.7.1.12 *DUCKING=X*

Description: Is true if the character is ducking and X is 1, or if the character is not ducking and X is 0.

Range: X = 0 or 1.

Example: :**ducking**=1:state=1

3.7.1.13 *IDLE=X*

Description: Is true if the character is idle and X is 1, or if the character is not idle and X is 0.

Range: X = 0 or 1

Example: :**idle**=1:state=1

3.7.1.14 *ISDARKAI=X*

Description: Is true if the character has been loaded into the Dark AI system and X is 1, or the character has not been loaded into the Dark AI system and X is 0.

Range: X = 0 or 1

Example: :**isdarkai**=1:state=1

3.7.1.15 *MOVINGBACKWARDS=X*

Description: Is true if the character is moving backwards and X is 1, or if the character is not moving backwards and x is 0.

Range: X = 0 or 1

Example: :**movingbackwards**=1:state=1

3.7.1.16 *MOVINGFORWARDS=X*

Description: Is true if the character is moving forwards and X is 1, or if the character is not moving forwards and X is 0.

Range: X = 0 or 1

Example: :**movingforwards**=1:state=1

3.7.1.17 *RUNNINGFORWARDS=X*

Description: Is true if the character is running forwards and X is 1, or if the character is not running forwards and X is 0.

Range: X = 0 or 1

Example: :**runningforwards**=1:state=1

3.7.1.18 *STRAFINGLEFT=X*

Description: Is true if the character is strafing left and X is 1, or if the character is not strafing left and X is 0.

Range: X = 0 or 1

Example: :**strafingleft**=1:state=1

3.7.1.19 *STRAFINGRIGHT=X*

Description: Is true if the character is strafing right and X is 1, or if the character is not strafing right and X is 0.

Range: X = 0 or 1

Example: `:strafingright=1:state=1`

3.7.2 ACTIONS

3.7.2.1 *ADDAITEAM=X*

Description: This changes the team for which the character fights for to X. The Player's team is 1.

Range: X = 1 to 20

Example: `:state=0:addaiteam=2`

3.7.2.2 *AIACTION=X*

Description: This sets the character's action mode to X.

Range: X = 0 – idle

X = 1 – follow player

X = 2 – in cover

X = 3 – stop moving

X = 4 – stop following player

X = 5 – move to sound

X = 6 – move away from sound

X = 7 - attack

X = 8 – move to point

X = 9 – move to cover based on enemy positions

X = 10 – move to nearest cover regardless of enemy positions

Example: `:state=0:aiaction=0`

3.7.2.3 *AIADDALLY=X Y_1|Y_2|…|Y_n*

Description: This makes teams Y_1 to Y_n allies of team X. All allies share calls.

Range: X, Y_1 to Y_n = 1 to 20

Example: `:state=0:addally=1 2|3|5|20`

3.7.2.4 *AIADDENEMY=X Y_1|Y_2|…|Y_n*

Description: This makes teams Y_1 to Y_n enemies of team X. All enemies are targetable.

Range: X, Y_1 to Y_n = 1 to 20

Example: `: state=0:addenemy=1 2|3|5|20`

3.7.2.5 *AIADDNEUTRAL=X*

Description: This makes teams Y_1 to Y_n neutrals of team X. All neutrals are ignored.

Range: X, Y_1 to Y_n = 1 to 20

Example: `: state=0:addneutral=1 2|3|5|20`

3.7.2.6 *AIADDPOINT=X*

Description: This creates a point/marker of the character's location with the ID number X. A maximum of points can exist within the whole level. Points are global markers that all DAI can see.

Range: X = 0 to 50

Example: `:state=1:aiaddpoint=1`

3.7.2.7 *AIATTACKAWARENESS=X*

Description: This switches the character's auto-attack behaviour ON or OFF. Set this to 0 in order to keep the AI from targeting and trying to attack one another.

Range: X = 1 – auto-attack ON

X = 0 – auto-attack OFF

Example: `:state=0:aiattckawareness=1`

3.7.2.8 *AIAUTOFACTIONOFF=X*

Description: This switches the faction control actions of all the character's ON or OFF.

Range: X = 1 – auto-faction OFF

X = 0 – auto-faction ON

Example: `:state=0:aiautofactionoff=0`

3.7.2.9 *AICALLTEAM=X*

Description: This calls all team members (and allies if auto faction is ON) within X units to the character's position.

Range: X = 0 to 540,000

Example: `:state=1:aicallteam=500`

3.7.2.10 *AICLEARTARGET*

Description: This clear's the character's targets.

Example: `:state=1:aicleartarget`

3.7.2.11 *AIFOLLOWPLR=X*

Description: This makes the character follow the player (if X is 1) or stops the character from following the player (if X is 0).

Range: X = 0 – stop following player

X = 1 – follow player

Example: `:state=1:aifollowplr=1`

3.7.2.12 *AIGOTOPOINT=X*

Description: This makes the character move towards point X, where X is the ID of the point. A maximum of points can exist within the whole level. Points are global markers that all DAI can see. This action may be ignored by the character if they are in a battle.

Range: X = 0 to 50

Example: `:state=1:aigotopoint=4`

3.7.2.13 *AILOOKAROUND=X Y*

Description: This makes the character turn to a random heading between X and Y degrees from it's current heading.

Range: X = -360 to 360.

Example: `:state=1:ailookaround=45 90`

3.7.2.14 *AIMOVEAWAYFROMSOUND*

Description: This makes the character move away from the largest sound source.

Example: `:state=1:aimoveawayfromsound`

3.7.2.15 *AIMOVERANDOM*

Description: This makes the character move around randomly. Must be looped to continue effect.

Example: `:state=1:aimoverandom`

3.7.2.16 *AIMOVETOCOVER=X*

Description: This makes the character move towards cover based on one of two conditions defined by X. A few things need to be considered when using this command:

- If an AI is flanked by enemies it can easily get caught trying to find cover.
- If no "Safe Zone" trigger zones exist in the map and the cover DAI chooses to go to is occupied by another AI, it will not be able to get there because of collision avoidance and usually the AI smart enough to find alternate cover.
- If the DAI is told to move to cover and no cover exists (like an empty room) it will usually run to a wall or corner of the room and stand there hoping it is safe.

Range: X = 0 – move to the nearest cover, but if that cover is occupied, move to the nearest trigger zone named "Safe Zone".

X = 1 – move to the nearest trigger zone named "Safe Zone".

Example: `:state=1:aimovetocover=1`

3.7.2.17 *AIMOVETOTARGET*

Description: This makes the character move towards its current target.

Example: `:state=1:aimovetotarget`

3.7.2.18 *AIMOVETOSOUND*

Description: This moves the character towards the closest sound made.

Example: `:state=1:`**`aimovetosound`**

3.7.2.19 *AIPLRCALLTEAM=X*

Description: This makes the character call all their teammates and allies within a radius of X units.

Range: X = 0 to 540,000

Example: `:state=1:`**`aiplrcallteam`**`=500`

3.7.2.20 *AIREMOVE*

Description: This removes the character from the Dark AI system.

Example: `:state=1:`**`airemove`**

3.7.2.21 *AIRESPONDTOCALL*

Description: This makes the character move towards the location of the call.

Example: `:state=1:`**`airespondtocall`**

3.7.2.22 *AIRESPONDTOPLRCALL*

Description: This makes the character move towards the location where the player made the call.

Example: `: state=1:`**`airespondtoplrcall`**

3.7.2.23 *AIROTATETOSOUND*

Description: This makes the character rotate to face the nearest sound.

Example: `:state=1:`**`airotatetosound`**

3.7.2.24 *AIROTATETOTARGET*

Description: This makes the character rotate to face their target.

Example: `:state=1:`**`airotatetotarget`**

3.7.2.25 *AIROTATEY=X*

Description: This makes the character rotate around its vertical axis to the angle X.

Range: X = -360 to 360

Example: `:state=1:`**`airotatey`**`=45`

3.7.2.26 *AISETEYELEVEL=X*

Description: This sets the character's eye level with an offset of X units from the default eye level.

Range: X = -50 to 50

Example: `:state=1:`**`aiseteyelevel`**`=-50`

3.7.2.27 *AISETMELEEDAMAGE=X*

Description: This sets the character's melee damage to X.

Range: X = Any value.

Example: `:state=1:`**`aisetmeleedamage`**`=50`

3.7.2.28 *AISETSPEED=X*

Description: This sets the character's speed to X.

Range: X = Any percentage.

Example: `:state=1:`**`aisetspeed`**`=100`

3.7.2.29 *AISETTARGET*

Description: This sets the character's possible target to its actual target.

Example: `:state=1:`**`aisettarget`**

3.7.2.30 *AISETVIEWRANGE=X*

Description: This adjusts the distance the character's can see. The default view range for Dark AI characters is 600 units.

Range: X = 0 to 540,000 – View distance in units.

Example: `:state=1:`**`aisetviewrange`**`=1000`

3.7.2.31 *AISTOP*

Description: This stops the character following any path, marker or waypoint.

Example: `:state=1:`**`aistop`**

3.7.2.32 *AIUSEFULLAIM=X*

Description: This switches the character's Dark AI aiming ability ON or OFF.

Range: X = 0 – Don't use Dark AI aiming,

X = 1 – Use Dark AI aiming.

Example: `:state=1:`**`aiusefullaim`**`=1`

3.7.2.33 *AIUSEMELEE=X*

Description: This makes the character use its melee attack. X switches whether blood is shown as a result of the attack or not.

Range: X = 0 – No blood shown.

X = 1 – Show blood.

Example: `:state=1:`**`aiusemelee`**`=1`

3.7.2.34 *ALLOWMOVE*

Description: This action does not perform any action within the engine although there is a reference to it within the code. The action is shown here as a place-holder in case it is used in the future.

3.7.2.35 *ALLYFOLLOW*

Description: This action does not perform any action within the engine although there is a reference to it within the code. The action is shown here as a place-holder in case it is used in the future.

3.7.2.36 *ALWAYSACTIVE=X*

Description: This switches the character's 'always active' parameter. Dark AI characters require the 'always active' parameter to be ON.

Range: X = 0 – Always Active is switched OFF,

X = 1 – Always Active is switched ON.

Example: `:state=1:`**`alwaysactive`**`=0`

3.7.2.37 *SETAIACTIVE=X*

Description: This sets whether Dark AI controls the character or not.

Range: X = 0 – Dark AI does not control the character,

X = 1 – Dark AI controls the character.

Example: `:state=1:`**`setaiactive`**`=1`

3.8 Player

These conditions and actions are applicable specifically to the player. Although there is no real 'player' script, the player's parameters can be modified by other entities.

3.8.1 CONDITIONS

3.8.1.1 *CURRENTWEAPON=X*

Description: Is true if the player's current weapon is from weapon slot X.

Range: X = Any number up to the maximum number of weapon slots.

Example: `:currentweapon=2:state=1`

3.8.1.2 *CURRENTWEAPONJAMMED*

Description: Is true if the player's current weapon is jammed.

Example: `:currentweaponjammed:state=1`

3.8.1.3 *FIREMODE=X*

Description: Is true if the player's current weapon's firemode is normal and X is 0 or if the player's current weapon's firemode is alternative and X is 1.

Range: X = 0 – True if the player's weapon is in normal fire mode,

X = 1 – True if the player's weapon is in alternative fire mode.

Example: `:firemode=1:state=1`

3.8.1.4 *PLAYERASSOCIATED*

Description: This condition does not perform any action within the engine although there is a reference to it within the code. The action is shown here as a place-holder in case it is used in the future.

3.8.1.5 *PLRALIVE=X*

Description: Is true if the player is alive and X is 1 or if the player is dead and X is 0.

Range: X = 0 – True if the player is dead,

X = 1 – True if the player is alive.

Example: `:plralive=1:state=1`

3.8.1.6 *PLRBLOCKING=X*

Description: Is true if the player is performing a blocking action and X is 1 or if the player is not performing a blocking action and X is 0.

Range: X = 0 – True if player is not performing blocking action,

X = 1 – True if player is performing the blocking action.

Example: `:plrblocking=1:state=1`

3.8.1.7 *PLRCANBESEEN*

Description: Is true if the player can be seen by the entity and, if applied to a character, sets the player as the character's possible target.

Example: :**plrcanbeseen**:state=1

3.8.1.8 *PLRCANNOTBESEEN*

Description: Is true if the entity cannot see the player.

Example: :**plrcannotbeseen**:state=1

3.8.1.9 *PLRCROUCHING*

Description: Is true if the player is crouching.

Example: :**plriscrouching**:state=1

3.8.1.10 *PLRDISTFURTHER=X*

Description: Is true if the player is further than X units from the entity.

Range: X = 0 to 540,000

Example: :**plrdistwithin**=50:state=1

3.8.1.11 *PLRDISTWITHIN=X*

Description: Is true if the player is within X units of the entity.

Range: X = 0 to 540,000

Example: :**plrdistwithin**=50:state=1

3.8.1.12 *PLRELEVFURTHER=X*

Description: Is true if the angle of elevation to the player is more than X°.

Range: X = 0° to 90°

Example: :**plrelevfurther**=30:state=1

3.8.1.13 *PLRELEVWITHIN=X*

Description: Is true if the angle of elevation to the player is less than X°.

Range: X = 0° to 90°

Example: :**plrelevwithin**=30:state=1

3.8.1.14 *PLRFACING=X*

Description: Is true if the player is facing the entity within ± X°. If X is 0, then a default value of 35° is used.

Range: X = 0 to 180°

Example: :**plrfacing**=45:state=1

3.8.1.15 *PLRFLOOREQUAL=X*

Description: This is true if the player is on floor X.

Range: X = 1 to 20.

Example: :**plrfloorequal**=4:state=1

3.8.1.16 *PLRHASKEY=X*

Description: Is true if the player has the entity's "Key" and X is 1 or if the player does not have the "Key" and X is 0.

Range: X = 0 – True if the player does not have key,

X = 1 – True if the player has the key.

Example: :**plrhaskey**=1:state=1

3.8.1.17 *PLRHEALTHGREATER=X*

Description: Is true if the player's health is greater than X.

Range: X = 0 to maximum health value.

Example: :**plrhealthgreater**=50:state=1

3.8.1.18 *PLRHEALTHLESS=X*

Description: Is true if the player's health is less than X.

Range: X = 0 to maximum health value.

Example: :**plrhealthless**=1:state=1

3.8.1.19 *PLRHIGHER=X*

Description: Is true if the player is higher than X units above the entity.

Range: X = 0 to 2,000

Example: :**plrhigher**=500:state=1

3.8.1.20 *PLRINGUNSIGHT*

Description: Is true if the entity is facing the player within 5 degrees.

Example: :**plringunsight**:state=1

3.8.1.21 *PLRISIMMUNE*

Description: Is true if the player is immune to damage.

Example: :**plrisimmune**:state=1

3.8.1.22 *PLRISUSINGRELOAD*

Description: Is true if the player is reloading.

Example: :**plrisusingreload**:state=1

3.8.1.23 *PLRJUMPING*

Description: Is true if the player is jumping.

Example: :**plrjumping**:state=1

3.8.1.24 *PLRLASTFIRED=FilePath OR PLRLASTFIRED=X*

Description: Is true if the player last fired the weapon identified by the file path or if the player last fired the weapon in slot X.

Range: FilePath = Any valid weapon file path.

X = Any weapon slot number

Examples: :**plrlastfired**=ww2\colt45:state=1

:**plrlastfired**=4:state=1

3.8.1.25 *PLRONGROUND=X*

Description: Is true if the player is on the ground and X is 1, or if the player is off the ground and X is 0.

Range: X = 0 – True if player is off the ground,

X = 1 – True if player is on the ground.

Example: :**plronground**=1:state=1

3.8.1.26 *PLRNOTCROUCHED*

Description: Is true if the player is not crouching.

Example: :**plrnotcrouched**:state=1

3.8.1.27 *PLRNOTFACING=X*

Description: Is true if the player is not facing the entity within ± X°. If X is 0 then a default range of 35° is used.

Range: X = 0 to 180°

Example: :**plrnotfacing**=45:state=1

3.8.1.28 *PLRNOTJUMPING*

Description: Is true if the player is not jumping.

Example: :**plrnotjumping**:state=1

3.8.1.29 *PLRNOTRUNNING*

Description: Is true if the player is not running.

Example: :**plrnotrunning**:state=1

3.8.1.30 *PLRNOTUSINGRELOAD*

Description: Is true if the player is not reloading their weapon.

Example: :**plrnotusingreload**:state=1

3.8.1.31 *PLRNOTZOOMED*

Description: Is true if the player's weapon is not zoomed in.

Example: :**plrnotzoomed**:state=1

3.8.1.32 *PLRPREVENTEDSELECTINGGUN*

Description: Is true if the player is currently prevented from selecting a weapon.

Example: :**plrpreventedselectinggun**:state=1

3.8.1.33 *PLRRUNNING*

Description: Is true if the player is running.

Example: :**plrrunning**:state=1

3.8.1.34 *PLRUNDERWATER=X*

Description: Is true if the player is under water and X is 1 or if the player is not under water and X is 0.

Range: X = 0 – True if player is above water surface,

X = 1 – True if player is below water surface.

Example: :**plrunderwater**=1:state=1

3.8.1.35 *PLRUSINGACTION=X*

Description: Is true if the player is using the action key (the default key is enter) and X is 1 or if the player is not using the action key and X is 0.

Range: X = 0 – True if player is not using the action key,

X = 1 – True if player is using the action key.

Example: :**plrusingaction**=1:state=1

3.8.1.36 *PLRWEAPONIDLE*

Description: Is true if the player is not 'using' their weapon.

Example: :**plrweaponidle**:state=50

3.8.1.37 *PLRWEAPONNOTIDLE*

Description: Is true if the player is 'using' their weapon.

Example: :**plrweaponnotidle**:state=51

3.8.1.38 *PLRWEAPONSEQUAL=X*

Description: Is true if the player is carrying exactly X number of weapons.

Range: X = Any value.

Example: :**plrweaponsequal**=2:state=1

3.8.1.39 *PLRWEAPONSGREATER=X*

Description: Is true if the player is carrying more than X weapons.

Range: X = Any value.

Example: :**plrweaponsgreater**=3:state=1

3.8.1.40 *PLRWEAPONSLESSER=X*

Description: Is true if the player is carrying less than X weapons.

Range: X = Any value.

Example: :**plrweaponslesser**=6:state=1

3.8.1.41 *PLRWITHINZONE=X*

Description: Is true if the player is within the zone and X is 1, or if the player is not within the zone and X is 0. If the trigger zone is set as an objective then the player's objective is triggered.

Range: X = 1 – returns true if the player is within the zone.

X = 0 – returns true is the player is not within the zone.

Example: `:plrwithinzone=1:state=10`

`:plrwithinzone=0:state=20`

3.8.1.42 *PLRZOOMED*

Description: Is true if the player's weapon is zoomed in.

Example: `:plrzoomed:state=1`

3.8.1.43 *WEAPONINSLOT=X*

Description: Is true if there is a weapon in slot X.

Range: X = Any number up to the maximum number of weapon slots.

Example: `:weaponinslot=6:state=1`

3.8.2 ACTIONS

3.8.2.1 *ARMDEC=X*

Description: This subtracts X from the current armour level.

Range: X = Any value.

Example: `:state=1:armdec=100`

3.8.2.2 *ARMINC=X*

Description: This adds X to the current armour level.

Range: X = Any value.

Example: `:state=1:arminc=100`

3.8.2.3 *ARMON=X*

Description: This defines whether the player is using armour or not. Armour absorbs all damage before player's heath is reduced.

Range: X = 0 – no body armour – player takes all damage,

X = 1 – body armour used – armour absorbs damage if armour level is high enough. An HUD is also displayed.

Example: `:state=1:disablefirebutton=1`

3.8.2.4 *ASSOCIATEPLAYER*

Description: this associates the player with the entity. Used for riding lifts, etc.

Example: `:state=1:associateplayer`

3.8.2.5 *DISABLECINEMATICHANDSBUTTON=X*

Description: This sets whether the blocking button can be used or not.

Range: X = 0 – blocking button is enabled,

X = 1 – blocking button is disabled.

Example: `:state=1:disablecinematichandsbutton=1`

3.8.2.6 *DISABLEFIREBUTTON=X*

Description: This sets whether the fire button can be used or not.

Range: X = 0 – fire button is enabled,

X = 1 – fire button is disabled.

Example: `:state=1:disablefirebutton=1`

3.8.2.7 *FREEEMPLACEMENT*

Description: This frees the player from a fixed 45° firing arc.

Example: `:state=1:freeemplacement`

3.8.2.8 *GETCINEMATICHANDS=GunName GunFolder*

Description: This sets the player's weapon by defining the gun name and the folder in which the gun exists. A placeholder entity must exist within the level for this command to work. The place holder entity must be dynamic and must not spawn at the start of the level.

Range: GunName = the name of the gun,

GunFolder = the file path to the gun from the generic guns folder.

Example: `:state=1:getcinematichands=colt45 ww2\colt45`

3.8.2.9 *GIVEPLRWEAPON=X*

Description: This adds the gun specified to the player and makes it the current weapon if the player has a spare weapon slot. The weapon named much exist within the level or the player will not be given the weapon.

Range: X = the file path to the gun from the generic guns folder.

Examples: `:state=1:giveplrweapon=colt45 ww2\colt45`

3.8.2.10 *HIDEPLRWEAPON*

Description: This removes the player's current weapon from view.

Example: `:state=1:hideplrweapon`

3.8.2.11 *LOCKEMPLACEMENT*

Description: This locks the player to a 45° firing arc.

Example: `:state=1:lockemplacement`

3.8.2.12 *LOCKSLOT=X Y*

Description: This sets whether the player's weapon slots are available or not.

Range: X = Weapon slot number (1-9).

Y = 0 – Weapon slot is active,

Y = 1 – Weapon slot is locked.

Example: `:state=1:lockslot=4 1`

3.8.2.13 *MOVEPLRX=X*

Description: This adds X to the player's current speed in the x-axis (East-West) by X.

Range: X = Any value.

Example: `:state=1:moveplrx=5`

3.8.2.14 *MOVEPLRY=X*

Description: This adds X to the player's current speed in the y-axis (vertical) by X.

Range: X = Any value.

Example: `:state=1:moveplry=5`

3.8.2.15 *MOVEPLRZ=X*

Description: This adds X to the player's current speed in the z-axis (North-South) by X.

Range: X = Any value.

Example: `:state=1:moveplrz=5`

3.8.2.16 *NEWJUMPHEIGHT=X*

Description: This sets the players jump height to X.

Range: X = 0 to 2,000

Example: `:state=1:newjumpheight=100`

3.8.2.17 *PLAYERDROP*

Description: This causes the player to drop the entity.

Example: `:state=1:playerdrop`

3.8.2.18 *PLAYERDROPCURRENT*

Description: This causes the player to drop the entity if it is currently being used.

Example: `:state=1:playerdropcurrent`

3.8.2.19 *PLAYERSTRENGTH=X*

Description: This sets multiplies the player's current strength by X percent.

Range: X = Any value.

Example: `:state=1:playerstrength=150`

3.8.2.20 *PLAYERTAKE*

Description: This makes the player collect the entity. The entity's main script will continue to run if the entity is defined as Always Active.

Example: `:state=1:playertake`

3.8.2.21 *PLRACCURACYMULT=X*

Description: This modifies the accuracy of the player when firing their weapon by X percent.

Range: X = Any value.

Example: `:state=0:plraccuracymult=50`

3.8.2.22 *PLRACTION=X*

Description: This forces the player to force one of 11 actions as described below.

Range: X = 1 – Fires current weapon,

X = 2 – Zooms current weapon,

X = 3 – Reloads current weapon,

X = 4 – Crouches player,

X = 5 – Jumps players,

X = 6 – Peeks player left,

X = 7 – Peeks player rights,

X = 8 – Forces "Use" key,

X = 9 – Forces left mouse click,

X = 10 - Forces right mouse click,

X = 11 – Jams current weapon of player.

Example: `:state=1:plraction=5`

3.8.2.23 *PLRADDHEALTH=X*

Description: This adds X amount of health to the player. The FPSC engine will prevent the player's health exceeding the maximum. Negative values can be used to reduce the player's health.

Range: X = Any number.

Example: `:state=1:plraddhealth=1`

3.8.2.24 *PLRALWAYSRUN=X*

Description: This defines is the player will always run when moved.

Range: X = 0 – player will only run when run key is pressed,

X = 1 – player will always run when moved.

Example: `:state=1:plralwaysrun=1`

3.8.2.25 *PLRDAMAGEMULT=X*

Description: This multiplies the damage caused by the play by X. Values less than 1 reduce the damage caused by the player and values greater than 1 increase the damage caused by the player.

Range: X = 0 to any number.

Example: `:state=1:plrdamagemult=1.6`

3.8.2.26 *PLRDEATH=X*

Description: This sets how the player falls when dead.

Range: X = 1 – falls left,
X = 2 – falls right,
X = 3 – falls forward,
X = 4 – falls backward,
X = 5 – falls in a random direction.

Example: `:state=1:plrdeath=5`

3.8.2.27 *PLRDEATHBOUNCE=X*

Description: This sets the amount of shake applied to the camera when the player hits the floor when dead.

Range: X = Any value.

Example: `:state=1:plrdeathbounce=50`

3.8.2.28 *PLRDEATHSPEED=X*

Description: This sets the speed at which the player falls to the floor when dead.

Range: X = Any value.

Example: `:state=1:plrdeathspeed=50`

3.8.2.29 *PLRDISABLE=X*

Description: This disables the mouse-look and gun for X milliseconds. Setting X to 0 will re-enable the mouse-look and gun.

Range: X = Any number (can not be negative)

Example: `:state=1:plrdisable=100`

3.8.2.30 *PLRFORCEMOVE=X*

Description: This forces the player to move forwards or backwards.

Range: X > 0 – Moves player forwards,
X = 0 – Disables auto move,
X < 0 – Moves player backwards.

Example: `:state=1:plrforcemove=1`

3.8.2.31 *PLRFREEZE=X*

Description: This prevents the player from moving for X milliseconds. Setting X to 0 will re-enable the player.

Range: X = Any number (can not be negative)

Example: `:state=1:plrfreeze=100`

3.8.2.32 *PLRMOVEDOWN=X*

Description: This moves the player in the vertical axis down by X units.

Range: X = -2,000 to 2,000 (2,000 is the maximum map height, negative values move up)

Example: `:state=1:plrmovedown=1`

3.8.2.33 *PLRMOVEEAST=X*

Description: This moves the player in the x-axis, left by X units.

Range: X = -4,000 to 4,000 (4,000 is the maximum map width , negative values move west)

Example: `:state=1:plrmoveeast=1`

3.8.2.34 *PLRMOVEIFUSED*

Description: This transports the player to the entity's "If Used" entity.

Example: `:state=1:plrmoveifused`

3.8.2.35 *PLRMOVENORTH=X*

Description: This moves the player in the z-axis axis forward by X units.

Range: X = -4,000 to 4,000 (4,000 is the maximum map lentgh, negative values move south)

Example: `:state=1:plrmovenorth=1`

3.8.2.36 *PLRMOVESOUTH=X*

Description: This moves the player in the z-axis axis backwards by X units.

Range: X = -4,000 to 4,000 (4,000 is the maximum map lentgh, negative values move south)

Example: `:state=1:plrmovesouth=1`

3.8.2.37 *PLRMOVETO=X Y*

Description: This transports the player to the location of entity X, where X is the entity number OR the entity name. The Y parameter sets whether the player's rotation is maintained or reset to 0.

Range: X = 0 to maximum entity number, or the entity's name.

Y = 0 – Resets player's rotation to 0°.

Y = 1 – Keeps player's rotation.

Examples: `:state=1:plrmoveto=10 0`

`:state=1:plrmoveto=AlternativeEntrance 1`

3.8.2.38 *PLRMOVEUP=X*

Description: This moves the player in the vertical axis up by X units.

Range: X = -2,000 to 2,000 (2,000 is the maximum map height, negative values move down)

Example: `:state=1:plrmoveup=1`

3.8.2.39 *PLRMOVEWEST=X*

Description: This moves the player in the x-axis axis right by X units.

Range: X = -4,000 to 4,000 (4,000 is the maximum map width , negative values move east)

Example: `:state=1:plrmovewest=1`

3.8.2.40 *PLROFFSETANGLEX=X*

Description: This sets the player's camera rotation around the x-axis (horizontal) by a fixed angle.

Range: X = -360 to 360

Example: `:state=1:plroffsetanglex=15`

3.8.2.41 *PLROFFSETANGLEY=X*

Description: This sets the player's camera rotation around the y-axis (vertical) by a fixed angle.

Range: X = -360 to 360

Example: `:state=1:plroffsetangley=45`

3.8.2.42 *PLROFFSETANGLEZ=X*

Description: This sets the player's camera rotation around the z-axis (line-of-sight) by a fixed angle.

Range: X = -360 to 360

Example: `:state=1:plroffsetanglez=15`

3.8.2.43 *PLRPICKON=X*

Description: This sets whether the player can pick up using the right mouse click or not.

Range: X = 0 – pick-up is disabled,

X = 1 – pick-up is enabled (default).

Example: `:state=1:plrpickon=0`

3.8.2.44 *PLRPICKRANGE=X*

Description: This defines the range at which a player can pick the entity up from.

Range: X = Any value.

Example: `:state=1:plrpickrange=200`

3.8.2.45 *PLRPOINTATOBJECT=X*

Description: This sets the player to look directly at the entity. X defines the offset in the vertical axis in units.

Range: X = Any number.

Example: `:state=1:plrpointatobject=25`

3.8.2.46 *PLRROTATEX=X*

Description: This sets the angle of the player's camera around the x-axis (horizontal) by X degrees.

Range: X = -360° to 360°.

Example: `:state=1:plrrotatex=5`

3.8.2.47 *PLRROTATEY=X*

Description: This sets the angle of the player's camera around the y-axis (vertical) by X degrees. 0° is defined as North. To adjust the rotational angle from the current set angle, use the internal variable $CAY together with the addvar command.

Range: X = -360° to 360°.

Example: `:state=1:plrrotatey=5`

3.8.2.48 *PLRSETHEALTH=X*

Description: This sets the player's health to X. If set above the maximum health the health will automatically reset to the maximum.

Range: X = 0 to maximum health value.

Example: `:state=1:plrsethealth=1`

3.8.2.49 *PLRSETIMMUNE=X*

Description: This sets whether the player is immune to damage or not.

Range: X = 0 – player can be damaged,

X = 1 – player is immune from damage.

Example: `:state=1:plrsetimmune=1`

3.8.2.50 *PLRSOUND=X*

Description: This makes the player make a sound where X defines the sound-set number or a sound file path.

Range: X = 0 to 6, or a file path.

Example: `:state=1:plrsound=1`

3.8.2.51 *PLRSPEEDMOD=X*

Description: This modifies the speed of the player's walk or run by a percentage.

Range: X = Any value.

Example: `:state=1:plrspeedmod=200`

3.8.2.52 *PLRSTRENGTH=X*

Description: This multiplies the player's maximum health by a percentage.

Range: X = Any value.

Example: `:state=1:plrstrength=200`

3.8.2.53 *PLRSUBHEALTH=X*

Description: This subtracts X amount of health from the player. Negative values can be used to increase the player's health. The FPSC engine will prevent the player's health exceeding the maximum.

Range: X = Any number up to maximum health value.

Example: `:state=1:plrsubhealth=1`

3.8.2.54 *PLRWOBBLE=X*

Description: This sets the amount of 'wobble' applied to the camera as the player moves around.

Range: X = Any number.

Example: `:state=1:plrwobble=25`

3.8.2.55 *PRESETGUNANIMATION=X*

Description: This plays a preset gun animation for the current gun held. No animation is played if the animation does not exist.

Range: X = *cock. endreload, fixjam, handbutton, handdead, handpush, handtake, idle, jammed, move, run, select, startreload, swaptoalt*

Example: `:state=1:presetgunanimation=cock`

3.8.2.56 *PREVENTPLRSELECTINGGUN=X*

Description: This sets whether the player can select a weapon or not.

Range: X = 0 – weapon selection is enabled,

X = 1 – weapon selection is disabled.

Example: `:state=1:preventplrselectinggun=1`

3.8.2.57 *REMOVECURRENTWEAPON*

Description: This removes the player's current weapon and also removes it from the weapon slot.

Example: `:state=1:removecurrentweapon`

3.8.2.58 *REMOVEPLRWEAPON=X*

Description: This removes the gun from gun slot X or the gun named X and frees the gun slot.

Range: X = Any number up to the maximum gun slot number,

X = the file path to the gun from the generic guns folder.

Examples: `:state=1:removeplrweapon=5`

`:state=1:removeplrweapon=colt45 ww2\colt45`

3.8.2.59 *RESETPLRWEAPONS*

Description: This removes all the player's weapons.

Example: `:state=1:`**`resetplrweapons`**

3.8.2.60 *SETARM=X*

Description: This sets the armour level (if body armour is being used).

Range: X = Any value.

Example: `:state=1:`**`setarm`**`=2000`

3.8.2.61 *SETMAXWEAPONS=X*

Description: This sets the maximum number of weapons a player can carry. Once the player has collected X weapons they can no longer pick any more up.

Range: X = Any value.

Example: `:state=1:`**`setmaxweapons`**`=3`

3.8.2.62 *SHOWPLRWEAPON*

Description: This puts the player's weapon in view.

Example: `:state=1:`**`showplrweapon`**

3.8.2.63 *STARTCINEMATICHANDS=X Y*

Description: This plays the play's hand animation X and loops it Y times.

Range: X = 1 – start cinematic hands.

Y = >1 – number of times to loop cinematic hands.

Example: `:state=1:`**`startcinematichands`**`=1 3`

3.8.2.64 *SWAPPLRWEAPON=X or SWAPPLRWEAPON=FilePath*

Description: This sets the weapon, defined by the weapon slot X or by the FilePath, to the active weapon for the player.

Range: X = Any occupied weapon slot for the player.

FilePath = path to weapon – the player must have the weapon in their inventory.

Examples: `:state=1:`**`swapplrweapon`**`=6`

`:state=1:`**`swapplrweapon`**`=scifi\autoslug`

3.8.2.65 *SWAPTOALT*

Description: This changes the fire mode of the currently held weapon to its alternative fire mode.

Example: `:state=1:`**`swaptoalt`**

3.8.2.66 *UNASSOCIATEPLAYER*

Description: This dissociates the player with the entity.

Example: `:state=1:`**`unassociateplayer`**

3.8.2.67 *WEAPONTOSLOT=X*

Description: This defines the next weapon slot to use when the player picks up a weapon.

Range: X = Any value up to the maximum weapon slot number (default is 9).
Set X to 0 to automatically use the next free slot.

Example: `:state=1:weapontoslot=6`

3.9 Multi-Player

These actions are applicable specifically to the multi-player aspects of FPSC.

3.9.1 ACTIONS

3.9.1.1 *HOSTGAME*

Description: This triggers a multi-player game to be hosted.

Example: `:state=1:`**`hostgame`**

3.9.1.2 *JOINGAME*

Description: This triggers a multi-player game to be joined.

Example: `:state=1:`**`joingame`**

3.9.1.3 *REPEATGAME*

Description: This triggers a multi-player game to repeat the same level.

Example: `:state=1:`**`repeatgame`**

3.10 Camera

These conditions and actions are applicable specifically to the player's field of view (FoV).

3.10.1 CONDITIONS

3.10.1.1 *INVIEW=X*

Description: Is true if the entity is on screen and X is 1 or the entity is not on screen and X is 0.

Range: X = 0 – True if entity is not on screen,

X = 1 – True if entity is on screen.

Example: `:inview=1:state=1`

3.10.1.2 *PICKOBJECT=X*

Description: Is true if the entity is dead-centre screen and X is 1 or the entity is not dead-centre screen and X is 0.

Range: X = 0 – True if the entity is not in the centre of view.

X = 1 – True if the entity is in the centre of view.

Example: `:pickobject=1:playertake`

3.10.2 ACTIONS

3.10.2.1 *CAMFOV=X*

Description: This sets the camera's FoV to a default value plus X.

Range: X = 0 to 180

Example: `:state=1:camfov=10`

3.10.2.2 *CAMFOVINC=X*

Description: This increases the camera's FOV by X.

Range: X = 0 to 180

Example: `:state=1:camfovinc=1`

3.10.2.3 *CAMROTATIONON=X*

Description: This defines whether the camera rotates when the player moves the mouse.

Range: X = 0 – camera rotation is fixed to that of the entity,

X = 1 – camera rotation is linked to the mouse movement.

Example: `:state=1:camrotationon=0`

3.10.2.4 *CAMSHAKE=X*

Description: This causes the camera to shake. X defines the magnitude of the shake. The shake decays over time.

Range: X = 0 to 20

Example: `:state=1:camshake=10`

3.10.2.5 *ENTITYCAM*

Description: This moves the camera view to that of the entity calling the command. The camera uses the entity's rotation as it's angle.

Example: `:state=1:entitycam`

3.10.2.6 *LASTCAM*

Description: This moves the camera view to that of the previous camera association (player or entity).

Example: `:state=1:lastcam`

3.10.2.7 *PLAYERCAM*

Description: This restores the camera view to that of the player.

Example: `:state=1:playercam`

3.10.2.8 *PLRCAMOFFSETON=X*

Description: This enables the player's camera to be offset.

Range: X = 0 – Player's camera is not offset,

X = 1 – Player's camera can be offset.

Example: `:state=1:plrcamoffseton=1`

3.10.2.9 *PLRCAMOFFSETX=X*

Description: This offsets the player's camera position in the x-axis (horizontal).

Range: X = Any value

Example: `:state=1:plrcamoffsetx=25`

3.10.2.10 *PLRCAMOFFSETY=X*

Description: This offsets the player's camera position in the y-axis (vertical).

Range: X = Any value

Example: `:state=1:plrcamoffsety=25`

3.10.2.11 *PLRCAMOFFSETZ=X*

Description: This offsets the player's camera position in the z-axis (forward/backwards).

Range: X = Any value

Example: `:state=1:plrcamoffsetz=25`

3.10.2.12 *SETCAMOFFSETX=X*

Description: This offsets the camera position in the x-axis (horizontal).

Range: X = Any value

Example: `:state=1:`**`setcamoffsetx`**`=25`

3.10.2.13 *SETCAMOFFSETY=X*

Description: This offsets the camera position in the y-axis (vertical).

Range: X = Any value

Example: `:state=1:`**`setcamoffsety`**`=25`

3.10.2.14 *SETCAMOFFSETZ=X*

Description: This offsets the camera position in the z-axis (forwards/backwards).

Range: X = Any value

Example: `:state=1:`**`setcamoffsetz`**`=25`

3.10.2.15 *SETCAMROTX=X*

Description: This rotates the camera around the x-axis (pitch).

Range: X = -360 to 360

Example: `:state=1:`**`setcamrotx`**`=45`

3.10.2.16 *SETCAMROTY=X*

Description: This rotates the camera around the y-axis (yaw).

Range: X = -360 to 360

Example: `:state=1:`**`setcamroty`**`=45`

3.10.2.17 *SETCAMROTZ=X*

Description: This rotates the camera around the z-axis (roll).

Range: X = -360 to 360

Example: `:state=1:`**`setcamrotz`**`=45`

3.11 Lights & Lighting

These actions are applicable to light entities, general level lighting as well as the player flash light.

3.11.1 ACTIONS

3.11.1.1 *AMBIENCE=X*

Description: This sets the amount of ambient light for the level.

Range: X = 0 to 100.

Example: `:state=1:`**`ambience`**`=100`

3.11.1.2 *AMBIENCEBLUE=X*

Description: This sets the amount of blue ambient light for the level.

Range: X = 0 to 255

Example: `:state=1:`**`ambienceblue`**`=100`

3.11.1.3 *AMBIENCEGREEN=X*

Description: This sets the amount of green ambient light for the level.

Range: X = 0 to 255

Example: `:state=1:`**`ambiencegreen`**`=100`

3.11.1.4 *AMBIENCERED=X*

Description: This sets the amount of red ambient light for the level.

Range: X = 0 to 255

Example: `:state=1:`**`ambiencered`**`=100`

3.11.1.5 *BLOOMACTIVE=X*

Description: This sets the bloom active variable in FPSC to X and is part of the light rays system within FPSC.

Range: X = Any value.

Example: `:state=1:`**`bloomactive`**`=1`

3.11.1.6 *FLASHLIGHT=X*

Description: This sets the player's flash light ON or OFF.

Range: X = 0 – Flash light OFF (default)

X = 1 – Flash light ON

Example: `:state=1:`**`flashlight`**`=1`

3.11.1.7 *FLASHLIGHTBLUE=X*

Description: This sets the amount of blue light the flash light emits.

Range: X = 0 to 255

Example: `:state=1:`**`flashlightblue`**`=128`

3.11.1.8 *FLASHLIGHTGREEN=X*

Description: This sets the amount of green light the flash light emits.

Range: X = 0 to 255

Example: `:state=1:`**`flashlightgreen`**`=128`

3.11.1.9 *FLASHLIGHTRANGE=X*

Description: This sets the radius of the flash light around the player.

Range: X = Any value.

Example: `:state=1:`**`flashlightrange`**`=500`

3.11.1.10 *FLASHLIGHTRED=X*

Description: This sets the amount of red light the flash light emits.

Range: X = 0 to 255

Example: `:state=1:`**`flashlightred`**`=128`

3.11.1.11 *LIGHTBLUE=X*

Description: This sets the amount of blue light the light entity emits.

Range: X = 0 to 255

Example: `:state=1:`**`lightblue`**`=128`

3.11.1.12 *LIGHTGREEN=X*

Description: This sets the amount of green light the light entity emits.

Range: X = 0 to 255

Example: `:state=1:`**`lightgreen`**`=128`

3.11.1.13 *LIGHTINTENSITY=X*

Description: This sets the 'brightness' of the light entity to X% and maintains the RGB ratio.

Range: X = 0 to 100

Example: `:state=1:`**`lightintensity`**`=50`

3.11.1.14 *LIGHTOFF*

Description: This switches the light OFF.

Example: `:state=1:`**`lightoff`**

3.11.1.15 *LIGHTON*

Description: This switches the light ON.

Example: `:state=1:`**`lighton`**

3.11.1.16 *LIGHTRANGE=X*

Description: This sets how far (in units) the light emitted from the light entity can reach.

Range: X = 0 to any value.

Example: `:state=1:lightrange=1000`

3.11.1.17 *LIGHTRED=X*

Description: This sets the amount of red light the light entity emits.

Range: X = 0 to 255

Example: `:state=1:lightred=128`

3.11.1.18 *LRAYACTIVE=X*

Description: This sets the light ray samples to X.

Range: X = Any value.

Example: `:state=1:lrayactive=1`

3.11.1.19 *LRDEBUDDEACTIVE=X*

Description: This deactivates the light ray debugging if X is set to 1.

Range: X = 0 or 1.

Example: `:state=1:lrdebugdeactive=1`

3.11.1.20 *LRAYSET=X*

Description: This sets the rotvar variable in FPSC to X.

Range: X = any value.

Example: `:state=1:lrayset=1`

3.12 HUDs

These conditions and actions are applicable specifically to Heads-Up Displays (HUDs). The conditions and actions added as part of GUI x9 are included within their own chapter.

3.12.1 CONDITIONS

3.12.1.1 *HUDEDITDONE=X*

Description: Is true when an 'editable' HUD is used.

Range: Name of HUD

Example: `:hudeditdone=PlayerName:resumegame`

3.12.1.2 *HUDHAVENAME*

Description: Is true when no player name has been entered.

Example: `:hudhavename:pausegame`

3.12.1.3 *HUDSELECTED=X*

Description: Is true if The mouse has clicked on the HUD.

Range: X = HUD name.

Example: `:hudselected=ShotEnemy:state=1`

3.12.1.4 *HUDSELECTIONMADE=X*

Description: Is true when the user has clicked the HUD button X.

Range: X = HUD name.

Example: `:hudselectionmade=BigGun:state=50`

3.12.2 ACTIONS

3.12.2.1 *ADDRAWTEXT=Text*

Description: This adds text to the current text being displayed using *fpgcrawtext*.

Range: Text = Any text string.

Example: `:state=1:addrawtext=...complete.`

3.12.2.2 *ADDRAWVAR=VarName OR ADDRAWVAR=X*

Description: This adds the value contained within VarName or the value X to the current text being displayed using *fpgcrawtext*.

Range: VarName = Any local or global variable.

X = Any Value

Examples: `:state=1:addrawvar=ItemUnitCost`

`:state=1:addrawvar=5.8`

3.12.2.3 *CHANGEHUDALPHA=HUDName Alpha*

Description: This sets the alpha channel of the HUD identified to the value given.

Range: HUDName = Any valid HUD.

Alpha = 0 to 255.

Example: `:state=1:changehudalpha=MainTitle 50`

3.12.2.4 *COMPASSOFF*

Description: This hides the compass from the screen.

Example: `:state=1:compassoff`

3.12.2.5 *COMPASSON*

Description: This displays the compass on screen.

Example: `:state=1:compasson`

3.12.2.6 *COMPASSSPIN*

Description: This forces the compass background to rotate, rather than the compass needle.

Example: `:state=1:compassspin`

3.12.2.7 *COMPASSX=X*

Description: This defines the position of the compass on screen where X is a percentage of the screen width.

Range: X = 0 to 100

Example: `:state=1:compassx=25`

3.12.2.8 *COMPASSY=X*

Description: This defines the position of the compass on screen where X is a percentage of the screen height.

Range: X = 0 to 100

Example: `:state=1:compassy=75`

3.12.2.9 *CROSSHAIR=X*

Description: This defines whether a crosshair (if available) is shown in the screen or not.

Range: X = 0 – Removes crosshair from screen,

X = 1 – Displays crosshair on screen, if available (default).

Example: `:state=1:crosshairt=0`

3.12.2.10 *FPGCRAWTEXT=X*

Description: This displays raw text to the screen as defined by the other *fpgcrawtext* actions. This function can also be used to display the value of a variable to screen. Local or Global variables shall be displayed by placing '%' in front of the variable name.

Range: X = Text to display, or variable (internal, local or global).

Example: `:state=1:`**`fpgcrawtext`**`=Hello world!`

3.12.2.11 *FPGCRAWTEXTB=X*

Description: This defines the level of blue in the font colour.

Range: X = 0 to 255

Example: `:state=1:`**`fpgcrawtextb`**`=127`

3.12.2.12 *FPGCRAWTEXTFONT=X*

Description: This defines the font used when displaying text. The font defined must be available on the computer or the default font will be used.

Range: X = Font name.

Example: `:state=1:`**`fpgcrawtextfont`**`=verdana`

3.12.2.13 *FPGCRAWTEXTG=X*

Description: This defines the level of green in the font colour.

Range: X = 0 to 255

Example: `:state=1:`**`fpgcrawtextg`**`=127`

3.12.2.14 *FPGCRAWTEXTOFF*

Description: This removes the text from the screen.

Example: `:state=1:`**`fpgcrawtextoff`**

3.12.2.15 *FPGCRAWTEXTR=X*

Description: This defines the level of red in the font colour.

Range: X = 0 to 255

Example: `:state=1:`**`fpgcrawtextr`**`=127`

3.12.2.16 *FPGCRAWTEXTSIZE=X*

Description: This defines the size of the text to display.

Range: X = 0 to any value.

Example: `:state=1:`**`fpgcrawtextsize`**`=18`

3.12.2.17 *FPGCRAWTEXTX=X*

Description: This defines the position of the text on screen where X is a percentage of the screen width. The text is centred around the position.

Range: X = 0 to 100

Example: `:state=1:`**`fpgcrawtextx`**`=50`

3.12.2.18 *FPGCRAWTEXTY=X*

Description: This defines the position of the text on screen where X is a percentage of the screen height. The text is centred around the position.

Range: X = 0 to 100

Example: `:state=1:fpgcrawtexty=50`

3.12.2.19 *HUDANIM=X*

Description: This defines the file path to the animated HUD image.

Range: X = File name (*.bmp, *.dds, *.jpg, *.png, *.tga)

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudanim=gamecore\huds\TestTubes\PressEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.20 *HUDBLUE=X*

Description: This defines the level of blue in the colour of the HUD item.

Range: X = 0 to 255

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudanim=gamecore\huds\TestTubes\PressEnterToUse.png,hudred=0,hudgreen=0,hudblue=255,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.21 *HUDFADEOUT=X*

Description: This starts the fade out sequence for a HUD image. The sequence shows the HUD and then increases the alpha phade of the HUD over time. Repeatedly calling this action, therefore, re-starts the sequence from the start. This does not work if *hudtext* is used.

Range: X = HUD name.

Example: `:state=1:hudfadeout=CollectHealth`

3.12.2.22 *HUDFONT=X*

Description: This defines the font used when displaying *hudtext*. If the font does not exist on the computer, the default font is used.

Range: X = Font name.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudsize=22,hudfont=verdana,hudtext=Press Enter To Use,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.23 *HUDGREEN=X*

Description: This defines the level of green in the colour of the HUD item.

Range: X = 0 to 255

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudanim=gamecore\huds\TestTubes\PressEnterToUse.png,hudred=0,hudgreen=0,hudblue=255,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.24 *HUDHIDE=X*

Description: This, when used during the definition of a HUD, sets the HUD to be hidden or not depending on the value of X.

Range: X = 0 – Show HUD,

X = 1 – Hide HUD.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudanim=gamecore\huds\TestTubes\PressEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.25 *HUDIMAGE=X or HUDIMAGEFINE=X*

Description: This defines the file path to the image to use as the HUD.

Range: X = File path (*.bmp, *.dds, *.jpg, *.png, *.tga).

Examples:
```
:state=0:hudreset,hudx=50,hudy=90,hudimage=gamecore\huds\TestTubes\PressEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

:
```
state=0:hudreset,hudx=50,hudy=90,hudimagefine=gamecore\huds\TestTubes\PressEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.26 *HUDLAYER =X Y*

Description: This defines which layer the HUD appears in. X defines the HUD name and Y defines the layer number. The 11th HUD layer is reserved for the mouse pointer.

Range: X = HUD name.

Y = 1 to 10.

Example: `:state=1:hudlayer=YouAreDead 2`

3.12.2.27 *HUDMAKE=X*

Description: This creates the HUD entity a should be the last command called when creating a HUD. X defines the 'make' of HUD that is being created.

Range: X = display – a static image such as text or your current weapon,

X = pointer – a mouse pointer,

X = button – for HUDs acting as buttons (i.e. exit/save/continue game),

X = status - for HUDs displaying lives, health, frags, time, etc.,

X = internal – HUDs used by internal game actions such as the zoom or loading screen,

X = numeric – for HUDs displaying numbers,

X = anim – for animated HUDs,

X = edit – for editing game set up parameters,

X = list – for displaying a list, such as saved games,

X = ipaddress – for displaying the ip address,

X = winnersname – for displaying the winner's name during a multi-player game,

X = textprompt - for enabling the player to type text.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudimage=gamecore\huds\TestTubes\PressEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.28 *HUDNAME=X*

Description: This defines the name of the HUD.

Range: X = HUD name.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudimage=gamecore\huds\TestTubes\PressEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.29 *HUDRED=X*

Description: This defines the level of red in the colour of the HUD item.

Range: X = 0 to 255

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudimage=gamecore\huds\TestTubes\PressEnterToUse.png,hudred=0,hudgreen=0,hudblue=255,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.30 *HUDRESET*

Description: This resets the HUD variables in preparation for a new HUD to be created. This action should be used before any other action used to define a new HUD.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudimage=gamecore\huds\TestTubes\PressEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.31 *HUDSHOW=X*

Description: This displays the HUD named X on screen. If the HUD type is an edit box, the font size is set to 30.

Range: X = HUD name.

Example: `:state=1:hudshow=Choice1`

3.12.2.32 *HUDSIZE=X*

Description: This sets the font size for the HUD if the HUD is displaying text.

Range: X = 0 to any value.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudsize=22,hudfont=verdana,hudtext=
Press Enter To Use,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.33 *HUDSIZEX=X*

Description: This defines the selectable width size of the HUD in pixels.

Range: X = 0 to any integer.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudsizex=16,hudsizey=16,hudimage=ga
mecore\huds\TestTubes\numeric8.png,hudname=UserNumber,huduservar=Test
Variable,hudtype=6,hudmake=numeric
```

3.12.2.34 *HUDSIZEY=X*

Description: This defines the selectable height size of the HUD in pixels.

Range: X = 0 to any integer.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudsizex=16,hudsizey=16,hudimage=ga
mecore\huds\TestTubes\numeric8.png,hudname=UserNumber,huduservar=Test
Variable,hudtype=6,hudmake=numeric
```

3.12.2.35 *HUDSIZEZ=X*

Description: This HUD dimension does not do anything, it is simply included because it is contained within the source code.

3.12.2.36 *HUDTEXT=X*

Description: This defines the text to display if no image is defined for the HUD.

Range: X = Text to display.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudsize=22,hudfont=verdana,hudtext=
Press Enter To Use,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.37 *HUDTYPE=X*

Description: This defines the type of HUD based on the 'make' of the HUD defined.

Range: X = 0 – default

	X=	0	1	2	3	4	5	6	11
HUD Make	**display**	Always display	-	-	-	-	-	-	-
	status	-	Lives	Health	Weapon	Frags	Time	-	Blip
	numeric	-	Lives	Health	Ammo	-	-	User variable	Blip
	button	-	New	Load	Save	Continue	Exit	-	-
	internal	-	Load	Eye HUD	Fader	Zoom	-	-	-
	anim	-	Lives	Health	Ammo	-	-	-	Blip
	edit	-	-	-	-	-	-	-	-
	list	-	-	-	-	-	-	-	-
	ipaddress	-	-	-	-	-	-	-	-
	winnersname	-	-	-	-	-	-	-	-
	textprompt	-	-	-	-	-	-	-	-

NOTE: The HUD makes "edit", "list", "ipaddress", "winnersname" and "textprompt" do not require HUD type to be defined.

Example:

```
:state=0:hudreset,hudx=50,hudy=90,hudsizex=16,hudsizey=16,hudimage=gamecore\huds\TestTubes\numeric8.png,hudname=User Number,huduservar=TestVariable,hudtype=6,hudmake=numeric
```

3.12.2.38 *HUDUNSHOW=X*

Description: This hides the HUD named X.

Range: X = HUD name.

Example: `:state=1:`**`hudunshow`**`=GameOver`

3.12.2.39 *HUDUSERVAR=VarName*

Description: This uses a declared global variable as a numerical HUD.

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudsizex=16,hudsizey=16,hudimage=ga
mecore\huds\TestTubes\numeric8.png,hudname=UserNumber,huduservar=Test
Variable,hudtype=6,hudmake=numeric
```

3.12.2.40 *HUDX=X*

Description: This sets the position of the new HUD as a percentage of the screen's width.

Range: X = 0 to 100

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudimage=gamecore\huds\TestTubes\Pr
essEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.41 *HUDY=X*

Description: This sets the position of the new HUD as a percentage of the screen's height.

Range: X = 0 to 100

Example:
```
:state=0:hudreset,hudx=50,hudy=90,hudimage=gamecore\huds\TestTubes\Pr
essEnterToUse.png,hudname=usedoorprompt,hudhide=1,hudmake=display
```

3.12.2.42 *HUDZ=X*

Description: This FPI command does not perform any action. It is included only for reference because it is within the FPSC source code.

3.12.2.43 *NEEDLESPIN*

Description: This forces the compass needle to rotate, rather than the compass background.

Example: `:state=1:`**`needlespin`**

3.12.2.44 *RADARON=X*

Description: This defines whether the dark radar feature is displayed on screen. The radar displays 'blip' images based on the Dark AI Teams set up within the game. Blip1.png corresponds to friendly characters and non-Dark AI characters show up as blip2.png.

Range: X = 0 – Radar is not shown,

X = 1 – Radar is shown on screen.

Example: `:state=1:`**`radaron`**`=1`

3.12.2.45 *RADARRANGE=X*

Description: This sets the range at which characters showup on the radar in segments.

Range: X = Any value

Example: `:state=0:radarrange=10`

3.12.2.46 *RADARX=X*

Description: This sets the position of the radar HUD as a percentage of the screen's width.

Range: X = 0 to 100

Example: `:state=0:radarx=50`

3.12.2.47 *RADARY=X*

Description: This sets the position of the radar HUD as a percentage of the screen's height.

Range: X = 0 to 100

Example: `:state=0:radary=50`

3.12.2.48 *ROTATEBLIP=X*

Description: This defines whether the radar blips rotate with the character's direction.

Range: X = 0 – The radar blips do not rotate,

X = 1 – The radar blips rotate.

Example: `:state=0:rotateblip=1`

3.12.2.49 *SCALEHUDX=HUDName Percentage*

Description: This resizes the HUD identified in the x-axis (horizontal) to the Percentage given.

Range: HUDName = Any valid HUD.

Percentage = 0 to any value.

Example: `:state=1:scalehudx=MainTitle 50`

3.12.2.50 *SCALEHUDY=HUDName Percentage*

Description: This resizes the HUD identified in the y-axis (vertical) to the Percentage given.

Range: HUDName = Any valid HUD.

Percentage = 0 to any value.

Example: `:state=1:scalehudy=MainTitle 50`

3.12.2.51 *SETAIRX=X*

Description: This sets the position of the air HUD as a percentage of the screen's width.

Range: X = 0 to 100

Example: `:state=0:`**`setairx`**`=50`

3.12.2.52 *SETAIRY=X*

Description: This sets the position of the air HUD as a percentage of the screen's height.

Range: X = 0 to 100

Example: `:state=0:`**`setairy`**`=50`

3.12.2.53 *SETARMX=X*

Description: This sets the position of the armour HUD as a percentage of the screen's width.

Range: X = 0 to 100

Example: `:state=0:`**`setarmx`**`=50`

3.12.2.54 *SETARMY=X*

Description: This sets the position of the armour HUD as a percentage of the screen's height.

Range: X = 0 to 100

Example: `:state=0:`**`setarmy`**`=50`

3.12.2.55 *SETOBJECTIVEMODE=X*

Description: This defines when the objective direction HUD is displayed.

Range: X = 0 – Objective direction marker is hidden from the screen,

X = 1 – Objective direction marker is always shown on screen,

X = 2 – Objective direction marker only shows when the player is on the same floor as the objective.

Example: `:state=0:`**`setobjectivemode`**`=2`

3.12.2.56 *SETOBJECTIVEX=X*

Description: This sets the position of the objective direction HUD as a percentage of the screen's width.

Range: X = 0 to 100

Example: `:state=0:`**`setobjectivex`**`=50`

3.12.2.57 *SETOBJECTIVEY=X*

Description: This sets the position of the objective direction HUD as a percentage of the screen's height.

Range: X = 0 to 100

Example: `:state=0:`**`setobjectivey`**`=50`

3.13 GUI-x9

These conditions and actions are applicable specifically to the GUI x9 system.

Note: Internal and custom variables will not work in place of values in GUI commands (i.e. usinf %MyVar or $PH will not work).

3.13.1 CONDITIONS

3.13.1.1 *CHOICEVALUEEQUAL=HUDName X*

Description: Is true if the choice slider HUDName equals Y.

Range: X = Choice value.

Example: :**choicevalueequal**=Difficulty Easy:state=1

3.13.1.2 *CURSORIMAGE=ImageName X*

Description: Is true if the mouse image is *ImageName* and X is 1 or the mouse image is not *ImageName* and Y is 0.

Range: X = 0 – True if image name is ImageName,

X = 1 – True if image name is not ImageName.

Example: :**cursorimage**=Target 1:state=1

3.13.1.3 *HUDMOUSEDOWN=HUDName X*

Description: Is true if the left mouse button is held down over HUDName and X is 1 or the left mouse button is not held down over HUDName and X is 0.

Range: X = 0 – True if the left mouse button is not held down over HUDName,

X = 1 - True if the left mouse button is held down over HUDName.

Example: :**hudmousedown**=Answer1 1:state=1

3.13.1.4 *HUDMOUSEOVER=HUDName X*

Description: Is true if the mouse is over HUDName and X is 1 or the mouse is not over HUDName and X is 0.

Range: X = 0 – True if the mouse is not over HUDName,

X = 1 - True if the mouse is over HUDName.

Example: :**hudmouseover**=Answer1 1:state=1

3.13.1.5 *HUDMOUSEUP=HUDName X*

Description: Is true if the left mouse button is clicked over HUDName and X is 1 or the left mouse button is not clicked over HUDName and X is 0.

Range: X = 0 – True if the left mouse button is not clicked over HUDName,

X = 1 - True if the left mouse button is clicked over HUDName.

Example: :**hudmouseup**=Answer1 1:state=1

3.13.1.6 *LOADGAMEVISIBLE=X*

Description: Is true if the load game screen is visible and X is 1 or the load screen is not visible and X is 0.

Range: X = 0 – True if load game is not visible,

X =1 – True if load game is visible.

Example: `:loadgamevisible=1:state=1`

3.13.1.7 *SAVEGAMEVISIBLE=X*

Description: Is true if the save game screen is visible and X is 1 or the save screen is not visible and X is 0.

Range: X = 0 – True if save game is not visible,

X =1 – True if save game is visible.

Example: `:savegamevisible=1:state=1`

3.13.1.8 *SLIDERVALUEEQUAL=HUDName X*

Description: Is true if the slider named HUDName equals X.

Range: X = 0 to any value.

Example: `:slidervalueequal=EnemyCount 1:state=1`

3.13.1.9 *SWGREATER=StopWatchName X*

Description: Is true if the stop watch called StopWatchName is greater than X milliseconds.

Range: X = 0 to any value.

Example: `:swgreater=Lap01 5000:state=1`

3.13.1.10 *SWLESS=StopWatchName X*

Description: Is true if the stop watch called StopWatchName is less than X milliseconds.

Range: X = 0 to any value.

Example: `:swless=Lap01 5000:state=1`

3.13.1.11 *SWRUNNING=StopWatchName X*

Description: Is true if the stop-watch called StopWatchName is running and X is 1 or if the stop-watch is not running and X is 0.

Range: X = 0 – True if stop-watch is not running,

X = 1 – True if stop-watch is running.

Example: `:swrunning=Lap1 1:state=1`

3.13.2ACTIONS

3.13.2.1 *ADDCHOICEVALUE=X Y*

Description: This adds value Y to the choice slider X. The choice is added to the right of all previous choices.

Range: X = Choice slider name,

Y = 0 to any value.

Example: `:state=1:addchoicevalue=GameDifficulty Easy`

3.13.2.2 *EYEHUDTIME=X*

Description: This defines the duration the 'hurt' HUD is visible on screen for X milliseconds.

Range: X = 0 to any value.

Example: `:state=1:eyehudtime=500`

3.13.2.3 *FADERSPEED=X*

Description: This defines the time taken for the end of level fader to complete in X milliseconds.

Range: X = 0 to any value.

Example: `:state=1:faderspeed=3000`

3.13.2.4 *HIDEALL*

Description: This hides all GUI related HUDs.

Example: `:state=1:hideall`

3.13.2.5 *HIDECHECKBOX=X*

Description: This hides the check-box named X.

Range: X = Check-box name.

Example: `:state=1:hidecheckbox=Invincible`

3.13.2.6 *HIDECHOICE=X*

Description: This hides the choice slider named X.

Range: X = Choice slider name.

Example: `:state=1:hidechoice=Difficulty`

3.13.2.7 *HIDECURSOR*

Description: This hides the mouse cursor.

Example: `:state=1:hidecursor`

3.13.2.8 *HIDEHUD=X*

Description: This hides the GUI s9 HUD named X.

Range: X = HUD name.

Example: `:state=1:`**`hidehud`**`=Options`

3.13.2.9 *HIDESLIDER=X*

Description: This hides the slider HUD named X.

Range: X = Slider HUD name.

Example: `:state=1:`**`hideslider`**`=Volume`

3.13.2.10 *HUDCLICKABLE=HUDName X*

Description: This makes the HUD called HUDName clickable like a button if X is 1 or makes the HUD non-clickable if X is 0.

Range: X = 0 – HUD is not clickable,

X = 1 – HUD is clickable.

Example: `:state=1:`**`hideclickable`**`=ON_OFF`

3.13.2.11 *LOADIMAGE=X Y*

Description: This loads an image, defined by Y, and names it X.

Range: X = HUD image name,

Y = HUD image file path (*.bmp, *.dds, *.jpg, *.png, *.tga)

Example: `:state=1:`**`loadimage`**`=Volume gamecore\huds\gamevolume.tga`

3.13.2.12 *MAKECHECKBOX=A B C X Y W H*

Description: This defines the parameters to make a check-box HUD.

Range: A = Check box HUD name,

B = Image name for "Unchecked" status,

C = Image name for "Checked" status,

X = 0 to 100 - Position of check box HUD as percentage of screen width,

Y = 0 to 100 – Position of check box HUD as percentage of screen height,

W = 0 to 100 - Width of the check box HUD as a percentage of the screen width (this is optional – leave blank if not needed),

H = 0 to 100 – Height of the check box HUD as a percentage of the screen height (this is optional – leave blank if not needed).

Example: `:state=1:`**`makecheckbox`**`=Mute MuteOFF MuteON 50 50`

3.13.2.13 *MAKECHOICE=A B C D E F X Y W H*

Description: This defines the parameters to make a choice slider HUD.

Range: A = Choice slider HUD name,

B = Image name for the background of the choice slider,

C = Image name for the fill of the choice slider (if no fill wanted set as "*none*"),

D = Image name for the choice slider itself,

E = Choice 1 value,

F = Choice 2 value,

X = 0 to 100 - Position of choice slider as a percentage of screen width,

Y = 0 to 100 – Position of choice slider as a percentage of screen height,

W = 0 to 100 - Width of the choice slider as a percentage of the screen width (this is optional – leave blank if not needed),

H = 0 to 100 – Height of the choice slider as a percentage of the screen height (this is optional – leave blank if not needed).

Example1:
```
:state=1:makechoiceslider=Difficulty DiffBack none
    DiffSlide Easy Hard 50 50
```

3.13.2.14 *MAKEHUD=A B C X Y W H*

Description: This defines the parameters to make a HUD.

Range: A = HUD name,

B = Image name for standard HUD,

C = Image name for HUD if the mouse is over it (setting this to "*none*" makes the HUD invisible when the mouse is over it),

X = 0 to 100 - Position of HUD as a percentage of screen width,

Y = 0 to 100 – Position of HUD as a percentage of screen height,

W = 0 to 100 - Width of the HUD as a percentage of the screen width (this is optional – leave blank if not needed),

H = 0 to 100 – Height of the HUD as a percentage of the screen height (this is optional – leave blank if not needed).

Example1:
```
:state=1:makehud=Options OptionIM none 50 50
```

3.13.2.15 *MAKESLIDER=A B C D X Y W H*

Description: This defines the parameters to make a slider HUD.

Range: A = Slider HUD name,

B = Image name for background of slider HUD,

C = Image name for slider HUD fill (if not wanted set as "*none*"),

D = Image name for slider itself,

X = 0 to 100 - Position of slider HUD as a percentage of screen width,

Y = 0 to 100 – Position of slider HUD as a percentage of screen height,

W = 0 to 100 - Width of the slider HUD as a percentage of the screen width (this is optional – leave blank if not needed),

H = 0 to 100 – Height of the slider HUD as a percentage of the screen height (this is optional – leave blank if not needed).

Example1: `:state=1:`**`makeslider`**`=Volume VolBack VolFill VolSlider 50 50`

3.13.2.16 *MAKESW=X Y*

Description: This creates a stop watch

Range: X = Stop watch name,

Y = 0 – do not start stop watch,

Y = 1 – start stop watch automatically.

Example: `:state=1:`**`makesw`**`=GameTime 1`

3.13.2.17 *READSETUPLINE=X Y*

Description: This reads the specified line from the setup.ini and stores it within the named setup variable.

Range: X = Setup variable name,

Y = Setup.ini line.

Example: `:state=1:`**`readsetupline`**`=JumpHeight PlayerJumpHeight`

3.13.2.18 *REPLACEIMAGE=X Y*

Description: This replaces the image currently designated to an image name with a new image. All HUDs using the original image are replaced with the new one.

Range: X = GUI image name,

Y = File path to new image (*.bmp, *.dds, *.jpg, *.png, *.tga).

Example: `:state=1:`**`replaceimage`**`=SetUpScreen gamecore\huds\setup2.tga`

3.13.2.19 *RESETGUI*

Description: This removes all GUI HUD data.

Example: `:state=1:`**`resetgui`**

3.13.2.20 *SETCHECKBOXALPHA=X Y*

Description: This sets the transparency of the check box.

Range: X = Check box name,

Y = Transparency value.

Example: `:state=1:setcheckboxalpha=VolumeMute 50`

3.13.2.21 *SETCHECKBOXCHECKED=X Y*

Description: This sets the check box's status.

Range: X = Check box name,

Y = 0 – check box is 'unchecked',

Y = 1 – check box is 'checked'.

Example: `:state=1:setcheckboxchecked=UnlimitedAmmo 1`

3.13.2.22 *SETCHECKBOXCIMAGE=X Y*

Description: This sets which image name to use as the check box's 'checked' image.

Range: X = Check box name,

Y = GUI image name.

Example: `:state=1:setcheckboxcimage=InfiniteLives ILChecked`

3.13.2.23 *SETCHECKBOXCOLOUR=X R G B or SETCHECKBOXCOLOR=X R G B*

Description: This defines the colour of the check box.

Range: X = Check box name,

R = 0 to 255 – The red component of the colour,

G = 0 to 255 – The green component of the colour,

B = 0 to 255 – The blue component of the colour.

Example: `:state=1:setcheckboxcolour=WeakEnemies 255 148 0`

3.13.2.24 *SETCHECKBOXH=X Y*

Description: This sets the check box height in pixels.

Range: X = Check box name,

Y = 0 to any value. – check box height.

Example: `:state=1:setcheckboxh=ShowHints 200`

3.13.2.25 *SETCHECKBOXNIMAGE=X*

Description: This sets which image name to use as the check box's 'unchecked' image.

Range: X = Check box name,

Y = GUI image name.

Example: `:state=1:setcheckboxnimage=InfiniteLives ILUnchecked`

3.13.2.26 *SETCHECKBOXW=X Y*

Description: This sets the check box width in pixels.

Range: X = Check box name,

Y = 0 to any value. – check box width.

Example: `:state=1:setcheckboxw=ShowHints 200`

3.13.2.27 *SETCHECKBOXX =X*

Description: This sets the check box position across the screen in pixels.

Range: X = Check box name,

Y = 0 to any value. – check box position along the width of the screen.

Example: `:state=1:setcheckboxx=ShowHints 1000`

3.13.2.28 *SETCHECKBOXY=X Y*

Description: This sets the check box position up and down the screen in pixels.

Range: X = Check box name,

Y = 0 to any value. – check box position up and down the screen.

Example: `:state=1:setcheckboxy=ShowHints 1000`

3.13.2.29 *SETCHOICEALPHA=X Y*

Description: This sets the transparency of the choice slider.

Range: X = Choice slider name,

Y = Transparency value.

Example: `:state=1:setchoicealpha=Volume 50`

3.13.2.30 *SETCHOICECOLOUR=X R G B or SETCHOICECOLOR=X R G B*

Description: This defines the colour of the choice slider.

Range: X = Choice slider name,

R = 0 to 255 – The red component of the colour,

G = 0 to 255 – The green component of the colour,

B = 0 to 255 – The blue component of the colour.

Example: `:state=1:setchoicecolour=Health 255 148 0`

3.13.2.31 *SETCHOICEH=X Y*

Description: This sets the choice slider height in pixels.

Range: X = Choice slider name,

Y = 0 to any value. – choice slider height.

Example: `:state=1:setchoiceh=FXVol 200`

3.13.2.32 *SETCHOICEVALUE=X Y*

Description: This sets the choice slider to a specific value.

Range: X = Choice slider name,

Y = Choice slider value.

Example: `:state=1:setchoicevalue=DifficultyLevel Easy`

3.13.2.33 *SETCHOICEW=X*

Description: This sets the choice slider width in pixels.

Range: X = Choice slider name,

Y = 0 to any value. – choice slider width.

Example: `:state=1:setchoicew=FXVol 1000`

3.13.2.34 *SETCHOICEX=X Y*

Description: This sets the choice slider position across the screen in pixels.

Range: X = Choice slider name,

Y = 0 to any value. – choice slider position across the screen.

Example: `:state=1:setchoicex=FXVol 300`

3.13.2.35 *SETCHOICEY=X Y*

Description: This sets the choice slider position up and down the screen in pixels.

Range: X = Choice slider name,

Y = 0 to any value. – choice slider position up and down the screen.

Example: `:state=1:setchoicey=FXVol 200`

3.13.2.36 *SETCURSOR=X*

Description: This sets the mouse cursor image.

Range: X = Mouse cursor image.

Example: `:state=1:setcursor=cursor4`

3.13.2.37 *SETHUDALPHA=X Y*

Description: This sets the transparency or alpha value of the HUD.

Range: X = HUD name,

Y = 0 to 255 – transparency value.

Example: `:state=1:sethudalpha=Confirm 50`

3.13.2.38 *SETHUDCOLOUR=X R G B or SETHUDCOLOR=X R G B*

Description: This defines the colour of the HUD.

Range: X = HUD name,

R = 0 to 255 – The red component of the colour,

G = 0 to 255 – The green component of the colour,

B = 0 to 255 – The blue component of the colour.

Example: `:state=1:`**`sethudcolour`**`=Button1 255 148 0`

3.13.2.39 *SETHUDH=X Y*

Description: This sets the HUD height in pixels.

Range: X = HUD name,

Y = 0 to any value. – HUD height.

Example: `:state=1:`**`sethudh`**`=Cancel 50`

3.13.2.40 *SETHUDNIMAGE=X Y*

Description: This sets which image name to use as the HUD's normal image.

Range: X = HUD name,

Y = GUI image name.

Example: `:state=1:`**`sethudnimage`**`=ConfirmButton ConfirmNotOver`

3.13.2.41 *SETHUDNUMERIC=X Y*

Description: This defines the HUD as a numeric type.

Range: X = HUD name,

Y = 0 – HUD not numeric,

Y = 1 – HUD is numeric.

Example: `:state=1:`**`sethudnumeric`**`=Cash 1`

3.13.2.42 *SETHUDOIMAGE=X*

Description: This sets which image name to use as the HUD's image when the mouse is over it.

Range: X = HUD name,

Y = GUI image name.

Example: `:state=1:`**`sethudoimage`**`=ConfirmButton ConfirmIsOver`

3.13.2.43 *SETHUDVALUE=X Y*

Description: This sets the variable to use for the numeric HUD.

Range: X = HUD name,

Y = "lives" – displays the number of lives the player has,

Y = "health" – displays the health of the player,

Y = "ammo" – displays the ammo available for the current weapon,

Y = "frags" – displays the number of grenades the player has,

Y = "time" – displays the time remaining,

Y = Variable name.

Example: `:state=1:sethudvalue=Cash %PlayerCash`

3.13.2.44 *SETHUDW=X Y*

Description: This sets the HUD width in pixels.

Range: X = HUD name,

Y = 0 to any integer – HUD width.

Example: `:state=1:sethudw=Cancel 100`

3.13.2.45 *SETHUDX=X Y*

Description: This sets the HUD position across the screen in pixels.

Range: X = HUD name,

Y = 0 to any integer – HUD position across the screen.

Example: `:state=1:sethudx=Cancel 300`

3.13.2.46 *SETHUDXPOS=X Y*

Description: This sets the HUD position across the screen as a percentage of the screen width.

Range: X = HUD name,

Y = 0 to 100 – HUD position across the screen.

Example: `:state=1:sethudxpos=Cancel 50`

3.13.2.47 *SETHUDY=X Y*

Description: This sets the HUD position up and down the screen in pixels.

Range: X = HUD name,

Y = 0 to any integer – HUD position up and down the screen.

Example: `:state=1:sethudy=Cancel 240`

3.13.2.48 *SETSLIDERALPHA=X Y*

Description: This sets the transparency of the slider.

Range: X = Slider name,

Y = 0 to 100 – transparency value.

Example: `:state=1:setslideralpha=MusicVol 50`

3.13.2.49 *SETSLIDERCOLOUR=X R G B or SETSLIDERCOLOR=X R G B*

Description: This defines the colour of the slider.

Range: X = Slider name,

R = 0 to 255 – The red component of the colour,

G = 0 to 255 – The green component of the colour,

B = 0 to 255 – The blue component of the colour.

Example: `:state=1:setslidercolour=Lighting 255 148 0`

3.13.2.50 *SETSLIDERH=X Y*

Description: This sets the slider height in pixels.

Range: X = Slider name,

Y = 0 to any integer – Slider height.

Example: `:state=1:setsliderh=NumPlayers 50`

3.13.2.51 *SETSLIDERVALUE=X Y*

Description: This sets the value for the slider.

Range: X = Slider name,

Y = 0 to ??? – Slider value.

Example: `:state=1:setslidervalue=NumPlayers 4`

3.13.2.52 *SETSLIDERW=X Y*

Description: This sets the slider width in pixels.

Range: X = Slider name,

Y = 0 to any integer – Slider width.

Example: `:state=1:setsliderw=NumPlayers 400`

3.13.2.53 *SETSLIDERX=X Y*

Description: This sets the slider position across the screen in pixels.

Range: X = Slider name,

Y = 0 to any integer – Slider position across the screen.

Example: `:state=1:setsliderx=NumPlayers 512`

3.13.2.54 *SETSLIDERY=X Y*

Description: This sets the slider position up and down the screen in pixels.

Range: X = Slider name,

Y = 0 to any integer – Slider position up and down the screen.

Example: `:state=1:setslidery=NumPlayers 512`

3.13.2.55 *SHOWCHECKBOX=X*

Description: This makes the named check box visible.

Range: X = Check box name.

Example: `:state=1:showcheckbox=Mute`

3.13.2.56 *SHOWCHOICE=X*

Description: This makes the named choice slider visible.

Range: X = Choice slider name.

Example: `:state=1:showchoice=Difficulty`

3.13.2.57 *SHOWCURSOR*

Description: This makes the mouse cursor visible.

Example: `:state=1:showcursor`

3.13.2.58 *SHOWHUD=X*

Description: This makes the named HUD visible.

Range: X = HUD name.

Example: `:state=1:showhud=SetupBkGd`

3.13.2.59 *SHOWSLIDER=X*

Description: This makes the named slider visible.

Range: X = Slider name.

Example: `:state=1:showslider=FXVolume`

3.13.2.60 *STARTSW=X Y*

Description: This starts the named stop watch.

Range: X = Stop watch name,

Y = 1 or "y" – This resets the stop watch and keeps it running.

Y = anything else – This starts the stop watch.

Example: `:state=1:startsw=GameTime y`

3.13.2.61 *STOPSW=X*

Description: This stops the named stop watch.

Range: X = Stop watch name.

Example: `:state=1:`**`stopsw`**`=GameTime`

3.13.2.62 *UPDATEGUI*

Description: This forces GUI x9 to update the status of all the GUI x9 HUDs.

Example: `:state=1:`**`updategui`**

3.13.2.63 *USEGUI=X*

Description: This sets FPSC up so that the GUI x9 conditions and actions can be used.

Range: X = 0 – GUI x9 is not used.

X = 1 – GUI x9 is used.

Example: `:state=1:`**`usegui`**`=1`

3.14 Level Navigation & Setup

These conditions and actions are applicable specifically to global level properties and moving between levels.

3.14.1 CONDITIONS

3.14.1.1 *LEVELEQUAL=X*

Description: Is true when the current level is equal to X.

Range: X = 0 to highest level.

Example: `:levelequal=4:state=1`

3.14.1.2 *LEVELNOTEQUAL=X*

Description: Is true when the current level is not equal to X.

Range: X = 0 to highest level.

Example: `:levelnotequal=4:state=1`

3.14.2 ACTIONS

3.14.2.1 *FOG=X*

Description: This switches the fog ON/OFF and simultaneously sets the distance of the fog.

Range: X = 0 – switches fog OFF,

X = 1 - switches fog ON and sets the fog between 0 and 1000 units,

1 < X < 1000 – switches fog ON and sets the fog between X and 1000 units,

X > 4000 – switches fog ON and sets the fog between X and 4000.

Example: `:state=1:fog=1200`

3.14.2.2 *FOGBLUE=X*

Description: This sets the blue component of the fog colour.

Range: X = 0 to 255

Example: `:state=1:fogblue=127`

3.14.2.3 *FOGEND=X*

Description: This defines the end of the fog, or the point where the fog is not transparent.

Range: X = Any value.

Example: `:state=1:fogend=4000`

3.14.2.4 *FOGGREEN=X*

Description: This sets the green component of the fog colour.

Range: X = 0 to 255

Example: `:state=1:foggreen=127`

3.14.2.5 *FOGRED=X*

Description: This sets the red component of the fog colour.

Range: X = 0 to 255

Example: `:state=1:fogred=127`

3.14.2.6 *FOGSTART=X*

Description: This defines the start of the fog, or the distance from the player where the fog doesn't exist.

Range: X = Any value.

Example: `:state=1:fogstart=200`

3.14.2.7 *HOLSTER=X*

Description: This sets whether the player can holster the weapon using the holster key.

Range: X = 0 – player cannot holster or swap weapons,

X = 1 – player can holster or swap weapons (default).

Example: `:state=1:holster=0`

3.14.2.8 *NEXTLEVEL=X*

Description: This defines which level will be the next when the win zone is triggered. In order to make the action work, it may be necessary to use the "*suspend*" action after the "nextlevel=" action.

Range: X = 0 to highest level.

Example: `:state=1:nextlevel=4,suspend`

3.14.2.9 *SETPOSTEFFECT=X*

Description: This sets which post-processing shader to use. The effect can be switched at any point in the game.

Range: X = any valid *.fx post-processing shader file.

Example: `:state=1:setposteffect=gasmask`

3.14.2.10 *SKY=FilePath*

Description: This sets the location of the sky box to use for the level.

Range: Filepath = *.x files only.

Example: `:state=1:sky=gamecore\myskys\level2sky.x`

3.14.2.11 *SKYSCROLL=FilePath*

Description: This sets the sky scroll texture for the level.

Range: Filepath = *.tga files only

Example: `:state=1:skyscroll= gamecore\myskys\level2skytexture.tga`

3.14.2.12 *WIN*

Description: This sets all player objectives as complete and ends the level. Use this action if you want to end the level without actually entering the Win Zone.

Example: `:state=1:win`

3.15 Water

These actions are applicable specifically to the water entity and the drowning mechanic of the player.

3.15.1 CONDITIONS

3.15.1.1 *AIREQUAL=X*

Description: This is true if the air level of the player is equal to X.

Range: X = Any value.

Example: `:airequal=40:state=2`

3.15.1.2 *AIRGREATER=X*

Description: This is true if the air level of the player is greater than X.

Range: X = Any value.

Example: `:airgreater=40:state=2`

3.15.1.3 *AIRLESSER=X*

Description: This is true if the air level of the player is less than X.

Range: X = Any value.

Example: `:airlesser=40:state=2`

3.15.1.4 *WATEREQUAL=X*

Description: This is true if the water level is equal to X units high.

Range: X = 0 – 20,000.

Example: `:waterequal=40:state=2`

3.15.1.5 *WATERGREATER=X*

Description: This is true if the water level is greater than X units high.

Range: X = 0 – 20,000.

Example: `:watergreater=40:state=2`

3.15.1.6 *WATERISON=X*

Description: This is true if the water is active and X is 1 or not active and X is 0.

Range: X = 0 – true if water is not active,

X = 1 – true if water is active.

Example: `:waterison=1:state=2`

3.15.1.7 *WATERLESSER=X*

Description: This is true if the water level is less than X units high.

Range: X = 0 – 20,000.

Example: `:waterlesser=40:state=2`

3.15.2ACTIONS

3.15.2.1 *ADDAIR=X*

Description: This adds X to the current air level of the player.

Range: X = Any value.

Example: `:state=1:addair=20`

3.15.2.2 *AIRON=X*

Description: This defines whether the player can drown when underwater or not.

Range: X = 0 – player cannot drown,

X = 1 – player can drown, but carries an oxygen supply that slowly decreases when underwater. Supply is not replenished when out of the water.

X = 2 – player can drown and does not carry an oxygen supply. Instead, the player can hold their breath for a duration before drowning. Lungs are re-filled when above the water level.

Example: `:state=1:airon=2`

3.15.2.3 *CREATEBULLETSPLASHSOUND=X*

Description: This sets whether a sound plays when bullets hit the water.

Range: X = 0 – splash sounds are OFF,

X = 1 – splash sounds are ON.

Example: `:state=1:createbulletsplashsound=1`

3.15.2.4 *GLOBALNOAIR=X*

Description: This sets the entire level to be un-breathable if X is set to one. For example, if the level was set on the surface of the Moon.

Range: X = 0 – Breathable air is ON,

X = 1 – Breathable air is OFF.

Example: `:state=1:globalnoair=1`

3.15.2.5 *INSTANTDROWN=X*

Description: This defines whether running out of air causes instant death.

Range: X = 0 – player does not instantly die when drowning,

X = 1 – player instantly dies when air level is reduced to zero.

Example: `:state=1:instantdrown=1`

3.15.2.6 *SETAIR=X*

Description: This sets the current air level of the player to X.

Range: X = Any value.

Example: `:state=1:setair=200`

3.15.2.7 *SETAIRMAX=X*

Description: This sets the maximum air level that can be carried by the player.

Range: X = Any value.

Example: `:state=1:setairmax=2000`

3.15.2.8 *SETAIRTIME=Milliseconds*

Description: This sets the time in milliseconds before the air level of the player is reduced.

Range: X = Any value.

Example: `:state=1:setairtime=1000`

3.15.2.9 *SETDROWNTIME=Milliseconds*

Description: This sets the time in milliseconds before drowning damage is applied to the player.

Range: X = Any value.

Example: `:state=1:setairtime=1000`

3.15.2.10 *SETNOAIRDAMAGE=X*

Description: This sets the damage applied to the player when there is no air left.

Range: X = Any value.

Example: `:state=1:setnoairdamage=200`

3.15.2.11 *WATER=X*

Description: This switches the water entity ON or OFF.

Range: X = 0 – Switches water OFF,

X = 1 – Switches water ON.

Example: `:state=1:water=1`

3.15.2.12 *WATERBLUE=X*

Description: This sets the blue component of the water's colour.

Range: X = 0 to 255

Example: `:state=1:waterblue=127`

3.15.2.13 *WATERCURRENT=X*

Description: This sets the water flow direction, clockwise from north. Setting X to zero makes the water flow north, 180 is south, etc.

Range: X = -360 to 360

Example: `:state=1:watercurrent=90`

3.15.2.14 *WATERFLOW=X*

Description: This sets the force of the water flow, moving the player and dynamic entities when within it.

Range: X = Any value.

Example: `:state=1:waterflow=1000`

3.15.2.15 *WATERFOGBLUE=X*

Description: This sets the blue component of the water when the player is underneath the water's surface.

Range: X = 0 to 255

Example: `:state=1:waterfogblue=160`

3.15.2.16 *WATERFOGDIST=X*

Description: This sets the distance of the fog underneath the water's surface.

Range: X = 0 to 4000

Example: `:state=1:waterfogdist=1000`

3.15.2.17 *WATERFOGGREEN=X*

Description: This sets the green component of the water when the player is underneath the water's surface.

Range: X = 0 to 255

Example: `:state=1:waterfoggreen=160`

3.15.2.18 *WATERFOGRED=X*

Description: This sets the red component of the water when the player is underneath the water's surface.

Range: X = 0 to 255

Example: `:state=1:waterfogred=160`

3.15.2.19 *WATERGREEN=X*

Description: This sets the green component of the water's colour.

Range: X = 0 to 255

Example: `:state=1:watergreen=127`

3.15.2.20 *WATERHEIGHT=X*

Description: This sets the height of the water within the level in units. The height of water is references to layer 1 of the level. For example, if you want the water to by 50 units high (half way up a segment wall) on layer 6 set the water height to 550.

Range: X = 0 to 2,000

Example: `:state=1:waterheight=3000`

3.15.2.21 *WATERRED=X*

Description: This sets the red component of the water's colour.

Range: X = 0 to 255

Example: `:state=1:waterred=127`

3.15.2.22 *WATERSPEED=X*

Description: This sets the speed and height of the ripples on the water's surface.

Range: X = 0 to 100

Example: `:state=1:waterspeed=20`

3.15.2.23 *WATERHEIGHTOFZONE=X*

Description: This sets the height of the water with reference to height of the entity calling this script command.

Range: X = 0 to 20000

Example: `:state=1:waterheightofzone=20`

3.16 Media (Sounds, Videos, Web, etc.)

These conditions and actions are applicable specifically to various media sources used within the game including video textures for entities.

3.16.1 CONDITIONS

3.16.1.1 *SOUNDFINISHED=X*

Description: Is true if the entity is not playing a sound and X is 1 or the entity is playing a sound and X is 0.

Range: X = 0 – True if the entity is playing a sound,

X = 1 – True if the entity is not playing a sound.

Example: `:soundfinished=1:state=1`

3.16.2 ACTIONS

3.16.2.1 *DELETEVIDEOTEXTURE=TextureID*

Description: This removes the video texture from memory.

Range: TextureID = 0 - 10

Example: `:state=1:deletevideotexture=1`

3.16.2.2 *LOOPSOUND=FilePath*

Description: This plays a sound at the entities location continuously.

Range: FilePath = *.ogg and *.wav files only.

Example: `:state=1:loopsound=audiobank\atmos\acidhum.wav`

3.16.2.3 *MAKEVIDEOTEXTURE=FilePath TextureID*

Description: This creates a video texture that can be called back an entity.

Range: FilePath = *.avi files only???

TextureID = 0 - 10

Example:
```
:state=1:makevideotexture=texturebank\videotextures\generic\tvscreen.avi 0
```

3.16.2.4 *MUSIC=FilePath*

Description: This plays a sound that can be heard wherever the player is, i.e. game music. The file is played continuously.

Range: FilePath = *.ogg and *.wav files only.

Example: `:state=1:music=audiobank\music\generic\main.wav`

3.16.2.5 *MUSICOVERRIDE=FilePath*

Description: This replaces the current background sound/music with a new file and plays the new file continuously.

Range: FilePath = *.ogg and *.wav files only.

Example: `:state=1:musicoverride=audiobank\music\generic\main.wav`

3.16.2.6 *MUSICVOLUME=X*

Description: This sets the volume level of the background sound/music. The volume defined by X is converted by FPSC using the following formula:

$$Volume = 80 + \frac{X}{5}$$

The internal volume can be set between 0 and 100 so to find the value of X to set the desired volume use the following formula:

$$X = (5 \times Volume) - 400$$

Range: X = -400 to 100

Example: `:state=1:musicvolume=-350`

3.16.2.7 *PAUSEVIDEOTEXTURE=TextureID*

Description: This pauses the video texture.

Range: TextureID = 0 - 10

Example: `:state=1:pausevideotexture=2`

3.16.2.8 *PLAYFULLVIDEO=X*

Description: This defines whether the player can interrupt any video played by using the "video=" command. The video will play to completition, i.e. clicking the mouse or spacebar will not stop the video.

Range: X = 0 – Video can be interrupted.

X = 1 – Video cannot be interrupted.

Example: `:state=1:playfullvideo=1`

3.16.2.9 *RESUMEVIDEOTEXTURE=TextureID*

Description: This continues the video texture playing from when it was paused.

Range: TextureID = 0 - 10

Example: `:state=1:resumevideotexture=3`

3.16.2.10 *SETVIDEOTEXTURESPEED=TextureID SpeedValue*

Description: This sets the speed at which the video texture plays as a percentage of its nominal speed.

Range: TextureID = 0 – 10

SpeedValue = 0 – 100.

Example: `:state=1:setvideotexturespeed=4 75`

3.16.2.11 *SETVIDEOTEXTUREVOLUME=TextureID Volume*

Description: This sets the volume of any sound the video texture plays as a percentage of its nominal speed.

Range: TextureID = 0 – 10

Volume = 85 – 100. (no sound is played below 85)

Example: `:state=1:setvideotexturevolume=5 90`

3.16.2.12 *SOUND=FilePath*

Description: This plays a sound at the entity's location once.

Range: FilePath = *.ogg and *.wav files only.

Example: `:state=1:sound=audiobank\voices\hello.wav`

3.16.2.13 *SOUNDSCALE=X*

Description: This scales the sound for the entity.

Range: X = 0 to 100

Example: `:state=1:soundscale=50`

3.16.2.14 *STOPSOUND=X*

Description: This stops all sounds created by the entity.

Range: X = 0 – Stops sound set 0,

X = 1 – Stops sound set 1.

Example: `:state=1:stopsound=0`

3.16.2.15 *STOPVIDEOTEXTURE=TextureID*

Description: This stops the video texture from playing.

Range: TextureID = 0 - 10

Example: `:state=1:stopvideotexture=6`

3.16.2.16 *USEVIDEOTEXTURE=TextureID*

Description: This sets the entity to use the video texture set as the TextureID. The command *makevideotexture* must have been used before this is called.

Range: TextureID = 0 - 10

Example: `:state=1:usevideotexture=1`

3.16.2.17 *VIDEO=FilePath*

Description: This interrupts the game and plays a video once. If the mouse is clicked, the spacebar is pressed or the escape keys is pressed the video will end.

Range: FilePath = *.avi files only.

Example: `:state=1:video=videobank\gamevideos\intro.avi`

3.16.2.18 *WEBLINK=WebAddress*

Description: This opens the website address using Microsoft Internet Explorer.

Range: WebAddress = Active website address starting with "http:".

Example: `:state=1:weblink=http://www.google.co.uk`

3.17 Game Controls & Inputs

These conditions and actions are applicable specifically to user inputs and controls within the game.

3.17.1 CONDITIONS

3.17.1.1 *ESCAPEKEYPRESSED=X*

Description: Is true when the Escape key is pressed and X is 1 or the Escape key is not pressed and X is 0.

Range: X = 0 – True if escape key is not pressed,

X = 1 – True if escape key is pressed.

Example: `:escapekeypressed=1:quitgame`

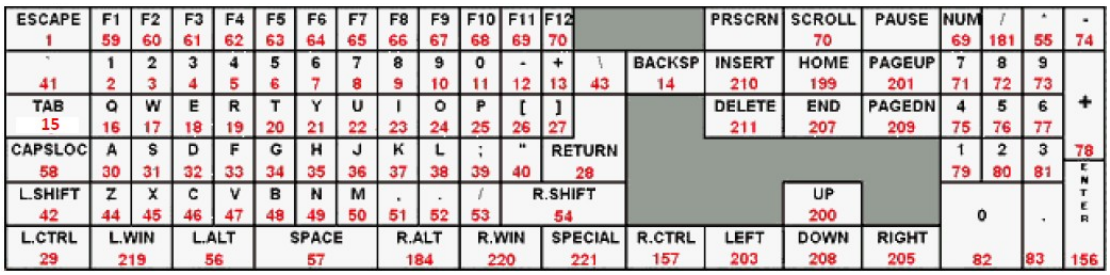
3.17.1.2 *KEYPRESSED=X Y*

Description: Is true if the key corresponding to the keycode X is pressed and Y is 1 or the key is not pressed and Y is 0.

Range: X = Keycode, see figure below.

Y = 0 – True if key is not pressed,

Y = 1 – True if key is pressed.



Example: `::keypressed=46 1:state=1`

3.17.1.3 *MOUSECLICK=X*

Description: This returns true if the corresponding mouse button has been pressed

Range: X = 1 – Left button is pressed,

X = 2 – Right button is pressed,

X = 3 – Both left and right buttons are pressed.

Example: `:mouseclick=1:state=1`

3.17.1.4 *MOUSESTATE=X*

Description: Is true if a specific combination of mouse buttons are pressed an X equals the sum of the mouse button codes, or no mouse button is clicked on X is zero.

Range: X = Mouse state see table below (primary codes shown in blue):

Mouse State Code (X)	Left Button	Right Button	Middle Button	Forth Button
0	-	-	-	-
1	O	-	-	-
2	-	O	-	-
3	O	O	-	-
4	-	-	O	-
5	O	-	O	-
6	-	O	O	-
7	O	O	O	-
8	-	-	-	O
9	O	-	-	O
10	-	O	-	O
11	O	O	-	O
12	-	-	O	O
13	O	-	O	O
14	-	O	O	O
15	O	O	O	O

Example: `:mousestate=1:state=1`

3.17.1.5 *PLRUSINGACTION=X*

Description: Is true if the player is pressing the action button and X is 1 or the player is not pressing the action button and X is 0. The default action button is enter.

Range: X = 0 – True if player is not pressing action button,

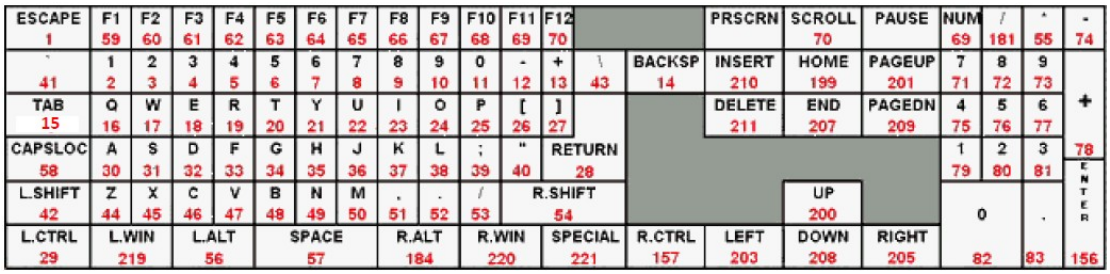
X = 1 – True if player is pressing action button.

Example: `:plrusingaction=1:state=1`

3.17.1.6 *SCANCODEKEYPRESSED=X*

Description: Is true if the key corresponding to the key code X is pressed.

Range: X = Keycode see figure below.



Example: `:scancodekeypressed=46:state=1`

3.17.2ACTIONS

3.17.2.1 *ARROWKEYS=X*

Description: This defines whether the arrow keys are used for player movement.

Range: X = 0 – Arrow keys are not used for player movement.

X = 1 – Arrow keys control the player's movement.

Example: `:state=1:arrowkeys=0`

3.17.2.2 *BACKDROP=FilePath*

Description: This sets the static backdrop image for out-of-game moments.

Range: FilePath = *.bmp, *.dds, *.jpg, *.png, *.tga.

Example: `:state=1:backdrop=backdrops\newgame.dds`

3.17.2.3 *BACKDROPVIDEO=X*

Description: This sets a video played in the background for out-of-game moments.

Range: FilePath = *.avi files only.

Example: `:state=1:backdropvideo=backdrops\newgame.avi`

3.17.2.4 *CONTINUEGAME*

Description: This action is used to move onto the next screen.

Example: `:state=1:continuegame`

3.17.2.5 *CROUCHKEY=X*

Description: This defines whether the crouch key is used.

Range: X = 0 – Crouch key is not used.

X = 1 – Crouch key is used.

Example: `:state=1:crouchkey=0`

3.17.2.6 *JUMPKEY=X*

Description: This defines whether the jump key is used.

Range: X = 0 – Jump key is not used.

X = 1 – Jump key is used.

Example: `:state=1:jumpkey=0`

3.17.2.7 *LOADGAME*

Description: This opens a dialog box to select a previously saved game. If GUI x9 is active, the GUI load game dialog box is opened.

Example: `:state=1:loadgame`

3.17.2.8 *NEWGAME*

Description: This starts the game from level 1.

Example: `:state=1:`**`newgame`**

3.17.2.9 *PAUSEGAME*

Description: This pauses the game and displays the in-game menu.

Example: `:state=1:`**`pausegame`**

3.17.2.10 *PEEKKEY=X*

Description: This defines whether the peek keys are used.

Range: X = 0 – Peek keys are not used.

X = 1 – Peek keys are used.

Example: `:state=1:`**`peekkey`**`=0`

3.17.2.11 *QUICKLOADGAME*

Description: This loads the game saved in the first saved game slot without showing the load game dialog box.

Example: `:state=1:`**`quickloadgame`**

3.17.2.12 *QUICKSAVEGAME*

Description: This saves the game to the first saved game slot without showing the save game dialog box.

Example: `:state=1:`**`quicksavegame`**

3.17.2.13 *QUITGAME*

Description: This stops the current game.

Example: `:state=1:`**`quitegame`**

3.17.2.14 *RESET*

Description: This removes all resources from the game including all entities, sounds, scripts, etc. This can only be used in final compiled game and cannot be used in the test game.

Example: `:state=1:`**`reset`**

3.17.2.15 *RESUMEGAME*

Description: This resumes the game and closes the in-game menu.

Example: `:state=1:`**`resumegame`**

3.17.2.16 *RUNKEY=X*

Description: This defines whether the run key is used.

Range: X = 0 – Run key is not used.

X = 1 – Run key is used.

Example: `:state=1:`**`runkey`**`=0`

3.17.2.17 *SAVEGAME*

Description: This opens a dialog box to select which slot to use to save the game. If GUI x9 is active, the GUI save game dialog box is opened.

Example: `:state=1:savegame`

3.17.2.18 *SETLISTKEY=X Y*

Description: This defines what key codes are used for each of the player functions.

Range: X = 1 – Move forward,

X = 2 – Move backwards,

X = 3 – Strafe left,

X = 4 – Strafe right,

X = 5 – Jump,

X = 6 – Crouch,

X = 7 – Use/Action,

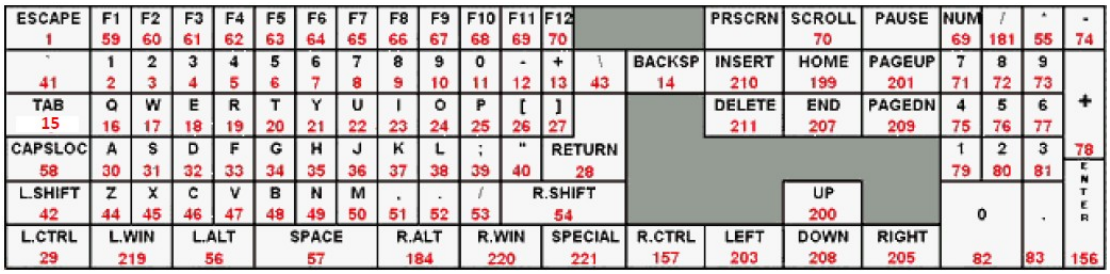
X = 8 – Reload,

X = 9 – Peek left,

X = 10 – Peek right,

X = 11 – Run

Y = Keycode see figure below.



Example: `:state=1:setlistkey=7 156`

3.17.2.19 *WALKKEY=X*

Description: This defines whether the walk key is used. What is the walk key?

Range: X = 0 – Walk key is not used.

X = 1 – Walk key is used.

Example: `:state=1:walkkey=0`

3.18 SETUP.INI

These conditions and actions are applicable specifically to the game's setup.ini file.

3.18.1 CONDITIONS

3.18.1.1 *ANTIALIASVAR=X*

Description: Is true if the anti-alias variable is equal to X.

Range: X = 0 to any value.

Example: :**antialiasvar**=1:state=1

3.18.1.2 *ASPECTRATIOVAR=X*

Description: Is true if the aspect ratio variable is equal to X.

Range: X = 0 to any value.

Example: :**aspectratiovar**=1:state=1

3.18.1.3 *AUTORESVAR=X*

Description: Is true if the auto resolution variable is equal to X.

Range: X = 0 to any value.

Example: :**autoresvar**=1:state=1

3.18.1.4 *DEPTHVAR=X*

Description: Is true if the depth variable is equal to X.

Range: X = 0 to any value.

Example: :**depthvar**=1:state=1

3.18.1.5 *HEIGHTVAR=X*

Description: Is true if the height variable is equal to X.

Range: X = 0 to any value.

Example: :**heightvar**=1:state=1

3.18.1.6 *MOUSEINVERTVAR=X*

Description: Is true if the mouse vertical axis inversion variable is equal to X.

Range: X = 0 to any value.

Example: :**mouseinvertvar**=1:state=1

3.18.1.7 *MOUSESENSITIVITYVAR=X*

Description: Is true if the mouse sensitivity variable is equal to X.

Range: X = 0 to any value.

Example: :**mousesensitivityvar**=1:state=1

3.18.1.8 *PARTICLESVAR=X*

Description: Is true if the particles variable is equal to X.

Range: X = 0 to any value.

Example: :**particlevar**=1:state=1

3.18.1.9 *POSTPROCESSINGVAR=X*

Description: Is true if the post processing variable is equal to X.

Range: X = 0 to any value.

Example: :**postprocessingvar**=1:state=1

3.18.1.10 *SHADERVAR=X*

Description: Is true if the shader variable is equal to X.

Range: X = 0 to any value.

Example: :**shadervar**=1:state=1

3.18.1.11 *SHADOWSVAR=X*

Description: Is true if the shadow variable is equal to X.

Range: X = 0 to any value.

Example: :**shadowvar**=1:state=1

3.18.1.12 *TEXQUALITYVAR=X*

Description: Is true if the texture quality variable is equal to X.

Range: X = 0 to any value.

Example: :**texqualityvar**=1:state=1

3.18.1.13 *WIDTHVAR=X*

Description: Is true if the width variable is equal to X.

Range: X = 0 to any value.

Example: :**widthvar**=1:state=1

3.18.2ACTIONS

3.18.2.1 *PASSTOSETUP=VarName X*

Description: This writes the value X to the VarName line within the setup.ini.

Range: VarName = "useeffects", "dynamicshadows", "dividetexturesize", "mousesensitivity", "aspectratio", "postprocessing", "width", "height", "depth", "antialias", "invmouse", "particleused", "autores".

X = any value.

Example: `:state=1:passtosetup=invmouse 1`

3.18.2.2 *SAVESETUP*

Description: This writes all setup variables to the game set.ini.

Example: `:state=1:savesetup`

3.18.2.3 *SETUP_ANTIALIAS=X*

Description: This sets the anti alias variable to X.

Range: X = 0 to any value.

Example: `:state=1:setup_antialias=1`

3.18.2.4 *SETUP_ASPECTRATIO=X*

Description: This sets the aspect ratio variable to X.

Range: X = 0 to any value.

Example: `:state=1:setup_aspectratio=1`

3.18.2.5 *SETUP_AUTORES =X*

Description: This sets the auto resolution variable to X.

Range: X = 0 to any value.

Example: `:state=1:setup_autores=1`

3.18.2.6 *SETUP_DEPTH=X*

Description: This sets the depth variable to X.

Range: X = 0 to any value.

Example: `:state=1:setup_depth=1`

3.18.2.7 *SETUP_DIVIDETEXTURESIZE =X*

Description: This sets the divide texture size variable to X.

Range: X = 0 to any value.

Example: `:state=1:setup_dividetexturesize=1`

3.18.2.8 *SETUP_DYNAMICSHADOWS =X*

Description: This sets the dynamic shadows variable to X.

Range: X = 0 to any value.

Example: `:state=1:`**`setup_dynamicshadows`**`=1`

3.18.2.9 *SETUP_HEIGHT =X*

Description: This sets the height variable to X.

Range: X = 0 to any value.

Example: `:state=1:`**`setup_height`**`=1`

3.18.2.10 *SETUP_MOUSEINVERT=X*

Description: This sets the invert mouse variable to X.

Range: X = 0 to any value.

Example: `:state=1:`**`setup_mouseinvert`**`=1`

3.18.2.11 *SETUP_MOUSESENSITIVITY=X*

Description: This sets the mouse sensitivity variable to X.

Range: X = 0 to any value.

Example: `:state=1:`**`setup_mousesensitivity`**`=1`

3.18.2.12 *SETUP_PARTICLES=X*

Description: This sets the particles variable to X.

Range: X = 0 to any value.

Example: `:state=1:`**`setup_particles`**`=1`

3.18.2.13 *SETUP_POSTPROCESSING=X*

Description: This sets the post processing variable to X.

Range: X = 0 to any value.

Example: `:state=1:`**`setup_postprocessing`**`=1`

3.18.2.14 *SETUP_USEEFFECTS =X*

Description: This sets the useeffects variable to X.

Range: X = 0 to any value.

Example: `:state=1:`**`setup_useeffects`**`=1`

3.18.2.15 *SETUP_WIDTH=X*

Description: This sets the width variable to X.

Range: X = 0 to any value.

Example: `:state=1:`**`setup_width`**`=1`

3.19 Debugging

These actions are applicable specifically to presenting information to the screen that help with debugging scripts.

3.19.1 Actions

3.19.1.1 *DEBUGCURSOR=X Y*

Description: This sets the cursor position to X and Y on the screen.

Range: X = 0 to the screen width,

Y = 0 to the screen height.

Example: `:state=1:debugcursor=512 384`

3.19.1.2 *DEBUGTEXT=Text*

Description: This displays the text to screen.

Range: Text = Any valid text string. Avoid using colon and comma characters.

Example: `:state=1:debugtext=Enemy trying to aim here.`

3.19.1.3 *DEBUGVAR=VarName*

Description: This displays the variable value on the screen.

Range: VarName = Any valid variable name.

Example: `:state=1:debugvar=ItemsCollected`

3.19.1.4 *WIREFRAME=X*

Description: This sets display to wireframe mode if X is 1, or switches OFF wireframe mode if X is 0.

Range: X = 0 – turn OFF wire frame mode,

X = 1 – turn ON wire frame mode.

Example: `:state=1:wireframe=1`

4 Appendices

4.1 Script Command Value Reference

The following is a short explanation of the ranges useable with the script commands, ranges specified with [] are optional.

1/0 = Either a 1 or 0 value.

ActionType = 1 – Fires current weapon,
2 – Zoomed current weapon,
3 – Reloads current weapon,
4 – Crouches player,
5 – Jumps player,
6 – Peeks player left,
7 – Peeks player right,
8 – Forces use key,
9 – Forces left mouse click,
10 – Forces right mouse click,
11 – Jams current weapon.

AirType = 1 – Aqualung (air depleted underwater, not replenished),
2 – Held breath (air depleted underwater, replenished above water).

AmmoClips = Refers to the number of ammo carried by the character.

Angle = -360 to 360 degrees.

AnimationNumber = Refers to the animation list of the entity.

AnyValue = Can be any value.

BloodDirection = 1 – Last damaged position.
2 – Head position.
3 – Centre of model, blood falls vertically down.

BlueColourValue = 0 - 255

BodyPartNumber = 1 – Head.
2 – Body.
3 – Left Arm.
4 – Right Arm.
5 – Left Leg.
6 – Right Leg.

CoverNumber = 0 – nearest cover.
1 – cover based on enemy positions.

DAIActionValue = 0 – idle.
1 – follow player.
2 – in cover.

3 – stop moving.

4 – stop following player.

5 – move to sound.

6 – move away from sound.

7 – attack.

8 – move to point.

9 – move to cover based on enemy positions.

10 – move to nearest cover regardless of enemy positions.

DamageType = "collision" – damaged by falling/thrown object,

"falling" – damaged by falling from a height,

"flak" – damaged by flak,

"indirect" – damaged by indirect event, like an explosion,

"melee" – damaged by melee attack.

DeathType = 1 – Fall left,

2 – Fall right,

3 – Fall forwards,

4 – Fall backwards,

5 – Fall randomly.

DecalMode = Refers to the decal mode of the entity.

EntityName = Refers to the name of an entity.

FilePath = Refers to the path to a specific file.

FontName = Refers to the name of a font.

FontSize = Refers to the size of a font.

FPIValue = 0 – Appear script.

1 – Main script.

2 – Destroy script.

3 – Shoot script.

FrameNumber = Refers to a specific frame within an animation.

GreenColourValue = 0 – 255

GunFolder = Folder within which gun exists (i.e. scifi)

GunName = Name of weapon (i.e. autoslug)

GunPath = File path to weapon (i.e. scifi\autoslug)

HUDMakeName = Refers to the HUD make name.

HUDName = The name of the HUD.

HUDNumberType = Refers to the type of number to link to the HUD.

HUDTypeNumber = Refers to the HUD sub-type.

ImageName = Name of the image loaded into FPSC using GUI-X9.

Integer = Any whole number.

KeyCode = Refers to the value corresponding to a key on the keyboard.

LevelNumber = Refers to the number of the level within the game.

LimbID = 1 = Head,
2 = Left clavicle,
3 = Right clavicle,
4 = Left foot,
5 = Right foot,
6 = Left thigh,
7 = Right thigh,
8 = Left hand,
9 = Right hand,
10 = Left fore arm,
11 = Right fore arm,
12 = Left upper arm,
13 = Right upper arm,
14 = Spine base,
15 = Spine middle,
16 = Spine top,
17 = neck

Milliseconds = 1000 milliseconds = 1 second.

MouseState = 1 – Left mouse button,
2 – Right mouse button,
3 – Middle mouse button.

ObjectiveType = 1 – Always show objective,
2 – Only show objective when on same floor.

Percentage = 0 – 100.

Pixels = Refers to the number of pixels.

PointValue = 0 – 50.

RedColourValue = 0 - 255

SetupVariable = Refers to the name of the setup.ini variable.

ShaderName = Name of valid *.fx file.

ShaderVariable = Refers to the shader variables – these are not supported.

ShakeValue = 0 – 20.

SoundSetNumber = Refers to the sound file number within the sound set.

SWName = Refers to the name of a stop watch.

TeamValue = 1 – 20.

Text = Refers to any text.

Units = 100 units equals 1 segment.

VariableName = Refers to the name of a variable created by the user.

VariableNumber = 0 – 99.

VolumeValue = -400 to 100

WeaponPath = Path to weapon within the files folder.

WeaponSlot = Number equal to the weapon slot of the player.

4.2 Conditions List

ACTIVATED=Integer

AIACTION=DAIActionValue

AIATCOVER=1/0

AIATPOINT=PointValue 1/0

AICALLED=1/0

AICALLEDBYPLR=1/0

AICANSHOOT=1/0

AIHASTARGET=1/0

AIHEARDSOUND=1/0

AIREQUAL=AnyValue

AIRGREATER=AnyValue

AIRLESSER=AnyValue

AITARGETDISTFURTHER=Units

AITARGETDISTWITHIN=Units

AITEAM=TeamValue

ALPHAFADEEQUAL=Percentage

ALWAYS

ANIMATIONOVER=AnimantionNumber

ANTIALIASVAR=AnyValue

ANYFURTHER=Units

ANYKEYWITHINZONE=1/0

ANYWITHIN=Units

ANYWITHINZONE=1/0

ASPECTRATIOVAR=AnyValue

AUTORESVAR=AnyValue

CANTAKE

CHOICEVALUEEQUAL=HUDName AnyValue

CURRENTWEAPON=WeaponSlot

CURRENTWEAPONJAMMED

CURSORIMAGE=ImageName 1/0

DAMAGEBY=DamageType

DAMAGEBY=WeaponPath

DAMAGEBY=WeaponSlot

DAMAGETIMEGREATER=Milliseconds

DEPTHVAR=AnyValue

DUCKING=1/0

ENTITYFLOOREQUAL=AnyValue

ENTITYFLOORHIGHER=AnyValue
ENTITYFLOORLOWER=AnyValue
ENTITYISHIGHER=Units
ENTITYISLOWER=Units
ENTITYWITHINZONE
ESCAPEKEYPRESSED=1/0
ETIMERGREATER=Milliseconds
FIREMODE=1/0
FLASHING=1/0
FRAMEATEND=AnimationNumber
FRAMEATSTART=AnimationNumber
FRAMEBEYOND=AnimationNumber FrameNumber
FRAMEWITHIN=AnimationNumber FrameNumber
HASWEAPON=1/0
HEADANGLEGREATER=Angle
HEADANGLELESS=Angle
HEALTH=AnyValue
HEALTHGREATER=AnyValue
HEALTHLESS=AnyValue
HEIGHTVAR=AnyValue
HUDEDITDONE=HUDName
HUDHAVENAME
HUDMOUSEDOWN=HUDName 1/0
HUDMOUSEOVER=HUDName 1/0
HUDMOUSEUP=HUDName 1/0
HUDSELECTED=HUDName
HUDSELECTIONMADE=HUDName
IDLE=1/0
IFMARKER=1/0
IFPLRTRAIL=1/0
IFWEAPON=1/0
INVIEW=1/0
ISDARKAI=1/0
ISENTITYIMMUNE
ISENTITYNOTIMMUNE
KEYPRESSED=KeyCode 1/0
LEVELEQUAL=LevelNumber
LEVELNOTEQUAL=LevelNumber

LOADGAMEVISIBLE=1/0
LOSETARGET=Integer
MOUSECLICK=MouseState
MOUSESENSITIVOTYVAR=AnyValue
MOUSEINVERTVAR=AnyValue
MOUSESTATE=MouseState
MOVINGBACKWARDS=1/0
MOVINGFORWARDS=1/0
NEARACTIVATABLE=Integer
NEVER
NEWWEAPONCANBESEEN
NOISEHEARD
NORAYCASTUP=Units1 Units2
NOTONRADAR
ONRADAR=1/0
PARTICLESVAR=AnyValue
PICKOBJECT=1/0
PLAYERASSOCIATED
PLRALIVE=1/0
PLRBLOCKING=1/0
PLRCANBESEEN
PLRCANNOTBESEEN
PLRDISTFURTHER=Units
PLRDISTWITHIN=Units
PLRELEVFURTHER=Angle
PLRELEVWITHIN=Angle
PLRFACING=Angle
PLRHASKEY=1/0
PLRHEALTHGREATER=AnyValue
PLRHEALTHLESS=AnyValue
PLRHIGHER=Units
PLRINGUNSIGHT
PLRISCROUCHING
PLRISIMMUNE
PLRISJUMPING
PLRISONGROUND=1/0
PLRISRUNNING
PLRISUSINGRELOAD

PLRISZOOMED
PLRLASTFIRED-WeaponPath
PLRLASTFIRED=WeaponSlot
PLRNOTCROUCHED
PLRNOTFACING=Angle
PLRNOTJUMPING
PLRNOTRUNNING
PLRNOTUSINGRELOAD
PLRNOTZOOMED
PLRPREVENTEDSELECTINGGUN
PLRUNDERWATER=1/0
PLRUSINGACTION=1/0
PLRWEAPONIDLE
PLRWEAPONNOTIDLE
PLRWEAPONSEQUAL=WeaponSlot
PLRWEAPONSGREATER=WeaponSlot
PLRWEAPONSLESSER=WeaponSlot
PLRWITHINZONE=1/0
POSTPROCESSINGVAR=AnyValue
QUANTITY=AmmoClips
RADAREQUAL=Units 1/0
RADARGREATER=Units 1/0
RANDOM=AnyValue
RATEOFFIRE
RAYCAST=Units1 Units2
RAYCASTBACK=Units1 Units2
RAYCASTUP=Units1 Units2
REACHTARGET
RUNNINGFORWARDS=1/0
SAMEFLOORASPLR
SAVEGAMEVISIBLE=1/0
SCANCODEKEYPRESSED=KeyCode
SHADERVAR=AnyValue
SHADERVARIABLE=ShaderVariable
SHADERVARIABLEGREATER=ShaderVariable
SHADERVARIABLELESS=ShaderVariable
SHADOWSVAR=AnyValue
SHOTDAMAGE=AnyValue

SHOTDAMAGETYPE=BodyPartNumber
SLIDERVALUEEQUAL=HUDName AnyValue
SOUNDFINISHED=1/0
SPAWNSGREATER=AnyValue 1/0
SPAWNSLEFT=AnyValue 1/0
SPEED=AnyValue
STATE=Integer
STRAFINGLEFT=1/0
STRAFINGRIGHT=1/0
SVAREQUAL=SetupVariable AnyValue
SWGREATER=SWName Milliseconds
SWLESS=SWName Milliseconds
SWRUNNING=SWName 1/0
TEXQUALITYVAR=AnyValue
TIMERGREATER=Milliseconds
UNDERWATER=1/0
VAREQUAL=AnyValue
VAREQUAL=VariableName AnyValue
VARGREATER=AnyValue
VARGREATER=VariableName AnyValue
VARLESS=AnyValue
VARLESS=VariableName AnyValue
VARNOTEQUAL=AnyValue
VARNOTEQUAL=VariableName AnyValue
VELOCITYGREATER=AnyValue
WAYPOINTSTATE=WayPointValue
WEAPONINSLOT=WeaponSlot
WIDTHVAR=AnyValue

4.3 Actions List

ACTIVATE=Integer

ACTIVATEALLINZONE=Integer

ACTIVATEIFUSED=Integer

ACTIVATEIFUSEDNEAR=Integer

ACTIVATETARGET=Integer

ACTIVEALLINZONE=Integer

ADDAIR=AnyValue

ADDAITEAM=TeamNumber

ADDCHOICEVALUE=HUDName AnyValue

ADDHEALTH=AnyValue

ADDRAWTEXT=Text

ADDRAWVAR=AnyValue

ADDRAWVAR=VarName

ADDVAR=VariableName AnyValue

ADVFRAME=Percentage

AIACTION=DAIActionNumber

AIADDALLY=TeamNumber TeamNumber(Ally1) [TeamNumber(Ally2 – 19)]

AIADDENEMY=TeamNumber TeamNumber(Enemy1) [TeamNumber(Enemy2 – 19)]

AIADDNEUTRAL=TeamNumber TeamNumber(Neutral1) [TeamNumber(Neutral2 – 19)]

AIADDPOINT=PointNumber

AIATTACKAWARENESS=1/0

AIAUTOFACTIONOFF=1/0

AICALLTEAM=TeamNumber

AICLEARTARGET

AIFOLLOWPLR=1/0

AIGOTTOPOINT=PointNumber

AILOOKAROUND=Angle1 Angle2

AIMOVEAWAYFROMSOUND

AIMOVERANDOM

AIMOVETOCOVER=CoverNumber

AIMOVETOSOUND

AIMOVETOTARGET

AIPLRCALLTEAM=Units

AIREMOVE

AIRESPONDTOCALL

AIRESPONDTOPLRCALL

AIRON=AirType
AIROTATETOSOUND
AIROTATETOTARGET
AIROTATEY=Angle
AISETEYELEVEL=Units
AISETMELEEDAMAGE=AnyValue
AISETSPEED=AnyValue
AISETTARGET
AISETVIEWRANGE=Units
AISTOP
AIUSEFULLAIM=1/0
AIUSEMELEE=1/0
ALLOWMOVE
ALLYFOLLOW
ALTTEXTURE=1/0
ALWAYSACTIVE=1/0
AMBIENCE=Percentage
AMBIENCEBLUE=BlueColourValue
AMBIENCEGREEN=GreenColourValue
AMBIENCERED=RedColourValue
ANIMATE=AnimationNumber
ANIMATIONNORMAL
ANIMATIONREVERSE
ARMDEC=AnyValue
ARMINC=AnyValue
ARMON=1/0
ARROWKEYS=1/0
ASSOCIATEPLAYER
BACKDROP=FilePath
BACKDROPVIDEO=FilePath
BLOODSPLASH=Units
BLOODSPURT=BloodDirection
BLOOMACTIVE=?
CAMFOV=?
CAMFOVINC=?
CAMROTATIONON=1/0
CAMSHAKE=ShakeValue
CARRYALL

CHANGEHUDALPHA=HUDName Percentage
CHOOSESTRAFE
COLLECTTARGET
COLOFF
COLON
COMPASSOFF
COMPASSON
COMPASSSPIN
COMPASSX=Percentage
COMPASSY=Percentage
CONTINUEGAME
COS=VarName Degrees
CROSSHAIR=1/0
CROUCHKEY=1/0
CULLMOD=AnyValue
CULLMODE=1/0
CULLRANGE=Units
DEBUGCURSOR=Percentage Percentage
DEBUGTEXT=Text
DEBUGVAR=VarName
DECALPHAFADE=Percentage
DECFRAME=AnimationNumber
DECSHADERVARIABLE=ShaderVariable
DECVAR=AnyValue
DESTROY
DIMLOCALVAR=VariableName
DIMVAR=VariableName
DISABLEFIREBUTTON=1/0
DIVVAR=VariableName AnyValue
DROPMARKER
ENTITYACCMULT=Percentage
ENTITYCAM
ENTITYDAMAGEMULT=Percentage
ENTITYSETIMMUNE=1/0
ENTROTATEX=Angle
ENTROTATEY=Angle
ENTROTATEZ=Angle
EMITFLASH

EMITFORCE=AnyValue
ETIMERSTART
EXPLODE
EYEHUDTIME=Milliseconds
FADERSPEED=Milliseconds
FLOATRATE=AnyValue
FLOORLOGIC
FOG=AnyValue
FOGBLUE=BlueColourValue
FOGEND=AnyValue
FOGGREEN=GreenColourValue
FOGRED=RedColourValue
FOGSTART=AnyValue
FOLLOWPLR
FORCEBACK
FORCEBOUNCE=1/0
FORCEDAMAGEON=1/0
FORCEFORE
FORCELEFT
FORCERIGHT
FPGCRAWTEXT=Text
FPGCRAWTEXTB=BlueColourValue
FPGCRAWTEXTFONT=FontName
FPGCRAWTEXTG=GreenColourValue
FPGCRAWTEXTOFF
FPGCRAWTEXTR=RedColourValue
FPGCRAWTEXTSIZE=FontSize
FPGCRAWTEXTX=Percentage
FPGCRAWTEXTY=Percentage
FREEZE
GETCINEMATICHANDS=GunName GunFolder
GIVEPLRWEAPON=GunPath
GLOBALNOAIR=1/0
GLOBALVAR=VariableNumber
HEADSHOT=Units
HEADSHOTDAMAGE=AnyValue
HIDE
HIDEALL

HIDECHECKBOX=HUDName
HIDECHOICE=HUDName
HIDECURSOR
HIDEHUD=HUDName
HIDELIMB=LimbID
HIDEPLRWEAPON
HIDESHADOW=1/0
HIDESLIDER=HUDName
HOLSTER
HOSTGAME
HUDANIM=FilePath
HUDBLUE=BlueColourValue
HUDCLICKABLE=HUDName 1/0
HUDFADEOUT=HUDName
HUDFONT=FontName
HUDGREEN=GreenColourValue
HUDHIDE=HUDName
HUDIMAGE=FilePath
HUDIMAGEFINE=FilePath
HUDLAYER=Integer
HUDMAKE=HUDMakeName
HUDNAME=HUDName
HUDRED=RedColourValue
HUDRESET
HUDSHOW=HUDName
HUDSIZE=FontSize
HUDSIZEX=Pixels
HUDSIZEY=Pixels
HUDSIZEZ=Pixels?
HUDTEXT=Text
HUDTYPE=HUDTypeNumber
HUDUNSHOW=HUDName
HUDUSERVAR=VariableName
HUDX=Percentage
HUDY=Percentage
HUDZ=Percentage?
INCALPHAFADE=Percentage
INCFRAME=AnimationNumber

INCSHADERVARIABLE=ShaderVariable

INCSTATE

INCVAR=AnyValue

INSTANTDROWN=1/0

ISALTAMMO

JOINGAME

JUMPKEY=1/0

LASTCAM

LIGHTBLUE=BlueColourValue

LIGHTGREEN=GreenColourValue

LIGHTINTENSITY=Percentage

LIGHTOFF

LIGHTON

LIGHTRANGE=Units

LIGHTRED=RedColourValue

LOADGAME

LOADIMAGE=ImageName FilePath

LOCALVAR=VariableNumber

LOGICBURST

LOOKATPLR

LOOKATTARGET

LOOPSOUND=FilePath

LRAYACTIVE=?

LRAYSET=?

LRDEBUGDEACTIVE=?

MAKECHECKBOX=HUDName ImageName(Unchecked) ImageName(Checked) Percentage(Width) Percentage(Height) [Pixels(Width)] [Pixels(Height)]

MAKECHOICE=HUDName ImageName(Back) ImageName(Fill) ImageName(Slider) AnyValue1 AnyValue2 Percentage(Width) Percentage(Height) [Pixels(Width)] [Pixels(Height)]

MAKEHUD=HUDName ImageName ImageName(Over) Percentage(Width) Percentage(Height) [Pixels(Width)] [Pixels(Height)]

MAKESLIDER=HUDName ImageName(Back) ImageName(Fill) ImageName(Slider) Percentage(Width) Percentage(Height) [Pixels(Width)] [Pixels(Height)]

MAKESVAR=VariableName SetupName AnyValue

MAKESW=SWName 1/0

MODVAR=VariableName AnyValue

MOVEBACK

MOVEFORE=Percentage

MOVEPLRX=AnyValue
MOVEPLRY=AnyValue
MOVEPLRZ=AnyValue
MOVETOTARGET=1/0
MOVEUP=Percentage
MULVAR=VariableName AnyValue
MUSIC=FilePath
MUSICOVERRIDE=FilePath
MUSICVOLUME=VolumeValue
NEEDLESPIN
NEWGAME
NEWJUMPHEIGHT=Units
NEXTLEVEL=LevelNumber
NEXTMARKER
NOBULLETCOL
NONE
NOGRAVITY
NOROTATE=1/0
PASSTOSETUP=SetupName AnyValue
PAUSEGAME
PEEKKEY=1/0
PIVOTRANDOM=Angle
PLAYERCAM
PLAYERDROP
PLAYERDROPCURRENT
PLAYERTAKE
PLAYFULLVIDEO=1/0
PLAYGUNANIMATION=FrameNumber FrameNumber
PLRACTION=ActionType
PLRADDHEALTH=AnyValue
PLRCAMOFFSETON=1/0
PLRCAMOFFSETX=Units
PLRCAMOFFSETY=Units
PLRCAMOFFSETZ=Units
PLRDAMAGEMULT=AnyValue
PLRDEATH=DeathType
PLRDEATHBOUNCE=AnyValue
PLRDEATHSPEED=AnyValue

PLRDISABLE=Milliseconds
PLRFORCEMOVE=AnyValue
PLRFREEZE=Milliseconds
PLRMOVEDOWN=Units
PLRMOVEEAST=Units
PLRMOVEIFUSED=EntityName
PLRMOVENORTH=Units
PLRMOVESOUTH=Units
PLRMOVETO=EntityName
PLRMOVEUP=Units
PLRMOVEWEST=Units
PLRPICKON=1/0
PLRPICKRANGE=Units
PLRPOINTATOBJECT=Units
PLRROTATEX=Degrees
PLRROTATEY=Degrees
PLRSETHEALTH=AnyValue
PLRSETIMMUNE=1/0
PLRSOUND=SoundSetNumber
PLRSPEEDMOD=Percentage
PLRSTRENGTH=Percentage
PLRSUBHEALTH=AnyValue
PLRWOBBLE=Degrees
PRESETGUNANIMATION=GunAnimation
PREVENTPLRSELECTINGGUN=1/0
QUICKLOADGAME
QUICKSAVEGAME
QUITGAME
RADARON=1/0
RADARRANGE=Units
RADARX=Percentage
RADARY=Percentage
RAGDOLL
RANDOMIZE
READSETUPLINE=VariableName SetupName
RELOADWEAPON
REMOVECURRENTWEAPON
REMOVEPLRWEAPON=GunPath

REMOVEPLRWEAPON=SlotNumber
REPEATGAME
REPLACEIMAGE=ImageName FilePath
RESET
RESETGLOBALSONRELOAD=1/0
RESETGUI
RESETHEAD=Angle
RESETMARKERS
RESETPLRWEAPONS
RESUMEGAME
ROTATEBLIP=1/0
ROTATEHEAD=Angle
ROTATEHEADRANDOM=Angle
ROTATEIY=Angle
ROTATETOPLR
ROTATETOTARGET
ROTATEY=Angle
RUNDECAL=DecalMode
RUNFORE=Percentage
RUNFPI=FilePath
RUNFPIDEFAULT=FPIValue
RUNKEY=1/0
SAVEGAME
SAVESETUP
SAVESVARS
SCALE=AnyValue AnyValue
SCALEHUDX=HUDName Percentage
SCALEHUDY=HUDName Percentage
SCALELIMB=LimbID Percentage
SELECTSHADERVARIABLE=ShaderVariable
SETAIACTIVE=1/0
SETAIR=AnyValue
SETAIRMAX=AnyValue
SETAIRTIME=Milliseconds
SETAIRX=Percentage
SETAIRY=Percentage
SETALPHAFADE=Percentage
SETANIMATESPEED=AnyValue

SETARM=AnyValue

SETARMX=Percentage

SETARMY=Percentage

SETCAMOFFSETX=Units

SETCAMOFFSETY=Units

SETCAMOFFSETZ=Units

SETCAMROTX=Degrees

SETCAMROTY=Degrees

SETCAMROTZ=Degrees

SETCHECKBOXALPHA=HUDName Percentage

SETCHECKBOXCHECKED=HUDName 1/0

SETCHECKBOXCIMAGE=HUDName ImaheName

SETCHECKBOXCOLOR=HUDName RedColourValue GreenColourValue BlueColourValue

SETCHECKBOXCOLOUR=HUDName RedColourValue GreenColourValue BlueColourValue

SETCHECKBOXNIMAGE=HUDName ImageName

SETCHECKBOXH=HUDName Pixels

SETCHECKBOXW=HUDName Pixels

SETCHECKBOXX=HUDName Pixels

SETCHECKBOXY=HUDName Pixels

SETCHOICEALPHA=HUDName Percentage

SETCHOICECOLOR=HUDName RedColourValue GreenColourValue BlueColourValue

SETCHOICECOLOUR=HUDName RedColourValue GreenColourValue BlueColourValue

SETCHOICEH=HUDName Pixels

SETCHOICEVALUE=HUDName AnyValue

SETCHOICEW=HUDName Pixels

SETCHOICEX=HUDName Pixels

SETCHOICEY=HUDName Pixels

SETCURSOR=ImageName

SETDROWNTIME=Milliseconds

SETENTITYSPEED=AnyValue

SETFLASHBLUE=BlueColourValue

SETFLASHGREEN=GreenColourValue

SETFLASHRANGE=Units

SETFLASHRED=RedColourValue

SETFORCEDAMAGE=AnyValue

SETFRAME=AnimationNumber

SETHEALTH=AnyValue

SETHUDALPHA=HUDName Percentage

SETHUDCOLOR=HUDName RedColourValue GreenColourValue BlueColourValue

SETHUDCOLOUR=HUDName RedColourValue GreenColourValue BlueColourValue

SETHUDNIMAGE=HUDName ImageName

SETHUDH=HUDName Pixels

SETHUDNUMERIC=HUDName 1/0

SETHUDOIMAGE=HUDName ImageName

SETHUDVALUE=HUDName HUDNumberType

SETHUDW=HUDName Pixels

SETHUDX=HUDName Pixels

SETHUDXPOS=HUDName Percentage

SETHUDY=HUDName Pixels

SETIFUSED=EntityName

SETISOBJECTIVE

SETMAXWEAPONS=WeaponSlot

SETNOAIRDAMAGE=AnyValue

SETOBJECTIVEMODE=ObjectiveMode

SETOBJECTIVEX=Percentage

SETOBJECTIVEY=Percentage

SETPOSTEFFECT=ShaderName

SETSHADERVARIABLE=ShaderVariable

SETSLIDERALPHA=HUDName Percentage

SETSLIDERCOLOR=HUDName RedColourValue GreenColourValue BlueColourValue

SETSLIDERCOLOUR=HUDName RedColourValue GreenColourValue BlueColourValue

SETSLIDERH=HUDName Pixels

SETSLIDERVALUE=HUDName AnyValue

SETSLIDERW=HUDName Pixels

SETSLIDERX=HUDName Pixels

SETSLIDERY=HUDName Pixels

SETTARGET

SETTARGETNAME=EntityName

SETUP_ANTIALIAS=AnyValue

SETUP_ASPECTRATIO=AnyValue

SETUP_AUTORES=AnyValue

SETUP_DEPTH=AnyValue

SETUP_DIVIDETEXTURESIZE=AnyValue
SETUP_DYNAMICSHADOWS=AnyValue
SETUP_HEIGHT=AnyValue
SETUP_MOUSEINVERT=AnyValue
SETUP_MOUSESENSITIVITY=AnyValue
SETUP_PARTICLES=AnyValue
SETUP_POSTPROCESSING=AnyValue
SETUP_USEEFFECTS=AnyValue
SETUP_WIDTH=AnyValue
SETUSEKEY=EntityName
SETVAR=AnyValue
SETVAR=VariableName AnyValue
SETVARLINE=VariableName SetupName
SETVARRND=Variable Name AnyValue
SETVARRND=AnyValue
SETVARTOGUI=VariableName HUDName
SETVARVALUE=VariableName AnyValue
SHAPEDECAL
SHOOTPLR
SHOW
SHOWCHECKBOX=HUDName
SHOWCHOICE=HUDName
SHOWCURSOR
SHOWHUD=HUDName
SHOWLIMB=LimbID
SHOWPLRWEAPON
SHOWSLIDER=HUDName
SIN=VarName Degrees
SKY=FilePath
SKYSCROLL=FilePath
SOUND=FilePath
SOUNDSCALE=Percentage
SPAWNOFF
SPAWNON
SPINRATE=AnyValue
STARTCINEMATICHANDS=1/0 Integer
STARTSW=SWName 1/0
STATE=Integer

STOPSOUND=1/0
STOPSW=SWName
STRAFE=Angle
SUBHEALTH=AnyValue
SUBVAR=VariableName AnyValue
SUSPEND
SWAPTOALT
TALK=FilePath
TALKORDERED=FilePath
TALKRANDOM=FilePath
TIMERSTART
TRIGGERFORCE=AnyValue
UNASSOCIATEPLAYER
UPDATEGUI
USEGUI=1/0
USEWEAPON
VIDEO=FilePath
WALKKEY=1/0
WATER=1/0
WATERBLUE=BlueColourValue
WATERCURRENT=Degrees
WATERFLOW=AnyValue
WATERFOGBLUE=BlueColourValue
WATERFOGDIST=Units
WATERFOGGREEN=GreenColourValue
WATERFOGRED=RedColourValue
WATERGREEN=GreenColourValue
WATERHEIGHT=Units
WATERHEIGHTOFZONE=Units
WATERRED=RedColourValue
WATERSPEED=AnyValue
WAYPOINTNEXT
WAYPOINTPREV
WAYPOINTRANDOM
WAYPOINTREVERSE
WAYPOINTSTART
WAYPOINTSTOP
WEAPONTOSLOT=WeaponSlot

WEBLINK=WebAddress

WIN

WRAPVAR=VariableName AnyValue

4.4 Animation List

Animation Value	Description	Animation Value	Description
0	Spawn	50	Weapon Spawn
1	Idle	51	Weapon Idle
2	Move Slow	52	Weapon Move Slow
3	Strafe Left	53	Weapon Strafe Left
4	Strafe Right	54	Weapon Strafe Right
5	Move Fast	55	Weapon Move Fast
6	Reload Weapon (or Toss)	56	Weapon Reload Weapon (or Toss)
8	Melee	57	Weapon NEW Climb
10	Climb	61	Weapon Impact Front
11	Impact Front	62	Weapon Bounce Front
12	Bounce Front	63	Weapon Get Up Front
13	Get Up Front	64	Weapon Impact Back
14	Impact Back	65	Weapon Bounce Back
15	Bounce Back	66	Weapon Get Up Back
16	Get up Back	67	Weapon Impact Left
17	Impact Left	68	Weapon Bounce Left
18	Bounce Left	70	Weapon Impact Right
20	Impact Right	71	Weapon Bounce Right
21	Bounce Right	81	Weapon Crouched Idle
31	Crouched Idle	82	Weapon Crouched Move Slow
32	Crouched Move Slow	83	Weapon Crouched Strafe Left
33	Crouched Strafe Left	84	Weapon Crouched Strafe Right
34	Crouched Strafe Right	85	Weapon Crouched Move Fast
35	Crouched Move Fast	86	Weapon Crouched Reload Weapon
36	Crouched Reload Weapon	90	Weapon Freeform Idle
40	Freeform Idle	91	Weapon Freeform Move
41	Freeform Move		

4.5 Internal Variables

Here is a list of the internal variables used by FPSC:

- $PH – Player's health,
- $PL – Player's lives,
- $RA – Player's total ammo in right-hand,
- $A - Player's total ammo,
- $RCA - Player's ammo in right-hand gun,
- $CA - Player's ammo in gun,
- $EH - Entity's health,
- $EA - Entity's ammo in gun (there is no total ammo for entities as they have unlimited ammo),
- $EPX - Entity's position in x-axis (read-only),
- $EPY - Entity's position in y-axis (read-only),
- $EPZ - Entity's position in z-axis (read-only),
- $EAX - Entity's angle around x-axis,
- $EAY - Entity's angle around y-axis,
- $EAZ - Entity's angle around z-axis,
- $CPX - Camera's position in x-axis,
- $CPY - Camera's position in y-axis,
- $CPZ - Camera's position in z-axis,
- $CAX - Camera's angle around x-axis,
- $CAY - Camera's angle around y-axis,
- $CAZ - Camera's angle around z-axis,
- $MMX – Mouse movement in x direction,
- $MMY – Mouse movement in y direction,
- $AIR – Amount of air, or air level,
- $ARM – Amount of armour, or armour level,
- $DIF – Distance from player in feet (read-only),
- $DIM – Distance from player in meters (read-only),
- $DIS – The distance between the entity and the player (read only),
- $FPS – Frame rate (read-only),
- $MAX – Number of weapon slots,
- $WAT – Water height.

NOTE: These variables can be altered by using the "setvar=" command unless stated as read-only. For example "setvar=$PH 500" sets the player's health to 500 units.