

Moves (Move Sets)

Set your animations, stances and moves here, as well as how hard (or easy) special moves from this character are to execute.



Execution Timing: How fast, in seconds, the keys listed in button sequence and execution have to be pressed in order to complete a special move.

Possible Air moves: How many moves can this character perform while jumping.

Blending Duration: How smooth the transition between basic moves are. Be careful as high numbers can cut down your animation and create small visual bugs as it trades response time for smoothness. The recommended (and default) value is 0.1.

Animation Type:

- Legacy: Tells UFE to read the animations from this character as *Legacy*.
- Mecanim 3D: Tells UFE to read the animations from this character as *Humanoid*. [Click here to watch Mecanim tutorial](#)
- Mecanim 2D: Tells UFE the character is 2D (uses [SpriteRender](#))

Avatar: If Mecanim was selected, drag the character's [avatar](#) here.

Animation Control: Tell the engine which animation control it should use.

- UFE Engine: The animation flow is fully controlled by UFE's own Update cycles. Due to limitations on the Unity engine, use this system to ensure there are no desynchronization when playing network games.

- **Unity Engine:** The animation flow is controlled by Unity's own update cycles. UFE will attempt to match the hitbox positions to control the moves AFTER Unity updates the character's mesh position. Selecting this option may cause minor inconsistencies. If your game requires precise frame data, use *UFE Engine* instead.

Normalize Animation Frames: Stabilizes 2D animation frames by fixing frame offsets while manually controlling the animator component (UFE Engine)

Use Animation Maps: Toggle this option if you are using the auto-maps and animation mapper for online play ([rollback netcode](#)).

Combat Stances

Each character can have one or several move sets, all classified by a combat stance. Add a new Combat Stance by clicking the **New Combat Stance** button at the bottom of this panel's section.

Combat Stance: A character can have several different fight styles, each with their own unique animations and special moves. In Street Fighter 4 for example, we have Gen, a character that can swap between 2 different fight styles. Both fighting styles are completely different, from the way they move to the special moves you can execute. Combat Stances can also be used as "power up" or anything you like. Combat styles can be applied by simply changing the combat stance in the [move options](#).

Cinematic Intro: (Optional, **PRO** and **SOURCE** only) a special move designed with just an intro animation and cinematics. UFE will check if you have this and play it at the beginning of each game. Check the example provided at `.\UFE\Characters\Mike\Moves\Intro.asset`

Round Won: Plays this move when the round is won.

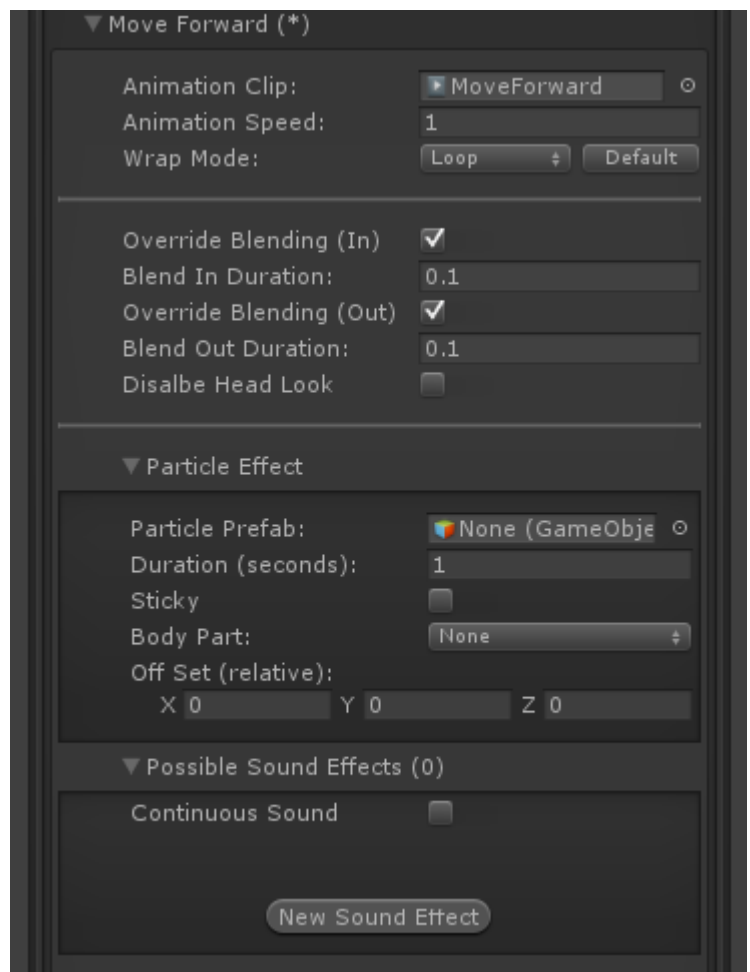
Time Out: Plays this move when a timeout happens.

Cinematic Outro: (Optional, **PRO** and **SOURCE** only) a special move designed with just an outro animation and cinematics. UFE will check if you have this and play it at the end of each game if this character wins. Check the example provided at `.\UFE\Characters\Mike\Moves\Outro.asset`

Basic Moves

Set the basic animations and sounds for your character, as well as any particular effect you want to add during these events. Basic Moves are categorised into Standard Animations, Blocking Animations and Hit Animations.

Enabled Moves Disable certain basic moves in case your game doesn't require it. You may disable Move, Jump, Crouch, Block or Parry. Disabling these options completely ignores any requests, including animation and input (from both player and AI).



Basic Information

Animation Clip: The animation clip related to this move. Animation files are normally located under the file structure of an *FBX* file, or just as a single *.anim* file.

Animation Speed: Tune the animation speed to match the desired effect during the game. Some moves like Jumping, Falling and Landing are automatically calculated by UFE based on the jump height.

Override Blending (In): Overrides the previous out blending or default blending duration with the value below.

Blend In Duration: If *override blending (in)* is toggled on, this is the new blending in value for this animation.

Override Blending (Out): Overrides the blend in set for the next animation or default blending duration with the value below.

Blend Out Duration: If *override blending (out)* is toggled on, this is the new blending out value for this animation.

Disable Head Look: If [Head Look](#) is enabled, it will be disabled during this move.

Particle Effects:

- **Particle Prefab:** A prefab with particle effects that is fired when this animation plays.
- **Duration (seconds):** How long this effect will last on the screen.
- **Sticky:** If toggled on, the particle will follow the body part as it emits.
- **Body Part:** The body part relative to the character (defined in the hitbox setup) where this particle will appear.
- **Off Set (relative):** The position (relative to the body part) this particle will appear when this move plays.

Possible Sound Effect: A list of possible sounds that plays at random when this move is executed. If only one sound is added it will only play that sound. UFE works with multi-channels, so don't worry about overlapping.

Continuous Sound: Should the selected sound effect keep playing while this move is in motion?

▼ Basic Moves
▶ Enabled Moves
Standard Animations
▶ Idle (*)
▶ Move Forward (*)
▶ Move Back (*)
▶ Crouching (*)
Jump Animations
▶ Take Off
▶ Jump Straight (*)
▶ Jump Back
▶ Jump Forward
Fall Animations
▶ Fall Straight (*)
▶ Fall Back
▶ Fall Forward
▶ Landing
Block Animations
▶ Standing Pose
▶ Standing High Hit
▶ Standing Low Hit
▶ Crouching Pose
▶ Crouching Hit
▶ Air Pose
▶ Air Hit
Parry Animations
▶ Standing High Hit
▶ Standing Low Hit
▶ Crouching Hit
▶ Air Hit
Hit Reactions
▶ Standing High Hit (*)
▶ Standing Low Hit
▶ Crouching Hit (*)
▶ Air Juggle [Launcher]
▶ Knock Back [Knockdown]
▶ Standing High Hit [Knockdown]
▶ Standing Mid Hit [Knockdown]
▶ Sweep [Knockdown]
▶ Crumple [Knockdown]
Stage Reactions
▶ Ground Bounce [Launcher]
▶ Standing Wall Bounce
▶ Standing Wall Bounce [Knockdown]
▶ Air Wall Bounce [Knockdown]
Fall Down Reactions
▶ Default (*)
▶ From Air Juggle [Knockdown]
▶ From Ground Bounce [Knockdown]
▶ Air Recovery
Stand Up Animations
▶ Default (*)
▶ From Air Juggle
▶ From Knock Back
▶ From Standing High Hit
▶ From Standing Mid Hit
▶ From Sweep
▶ From Crumple
▶ From Standing Wall Bounce
▶ From Air Wall Bounce
▶ From Ground Bounce
* Required

Specifics

Standard Animations: Used when character is not moving (Idle), moving back and forward, and crouching (if your game uses it). When choosing the Idle you can also set its resting animations:

- Resting Clips (1-5): When character isn't moving, you can have certain animations play at random.
- Resting Interval (seconds): Every interval cycle the engine will attempt to play one of the resting clips from above. If the randomly selected slot has no clip, it won't play anything. Example: If you want a clip to run every 6 seconds with 100% consistency, fill all the fields with that animation.

Jump Animations: These animations trigger when the character *initiates* a jump. If you don't have the respective Fall Animation setup, the engine will attempt to use these animation as single jump cycles.

- If Auto Speed is toggled the engine will attempt to synchronize the height of the jump with the animation length so both can end at the correct time to link with its next animation. Untoggle to adjust how your animation should play out.
- Take Off animation speeds are used in conjunction with the Jump Delay variable (located on the [physics option](#))

Fall Animations: These animations trigger when the character is *falling* from a jump.

- If Auto Speed is toggled the engine will attempt to synchronize the height of the fall with the animation length. Untoggle to adjust how your animation should play out.
- Landing animation speeds are used in conjunction with the Jump Delay variable (located on the [physics option](#))

Block Animations: Triggered when the character [blocks](#) (or attempting to block) an attack.

- Standing High Hit is triggered when a an attack hits the *upper part* of a character while standing. Standing Low Hits is triggered on low hits.
- Blocking Poses triggers when the character is attempting to block as its *under* the blockable area, but *not* blocking (only happens when using hold back as the [block type](#)).
- If Auto Speed is toggled the engine will use the [block stun](#) information to synchronize it with the animation length. Untoggle to run your own speed and [WrapMode](#) values.

Parry Animations: Triggers when the character successfully [parries](#) an attack.

- Standing High Hit is triggered when a an attack hits the *upper part* of a character while standing. Standing Low Hits is triggered on low hits.
- If Auto Speed is toggled the engine will use the *parry stun* information from [Block Options -> Parry](#) to synchronize it with the animation length. Untoggle to run your own speed and [WrapMode](#) values.

Hit Reactions: Triggers when the character gets hit.

- Standing High Hit is triggered when a an attack hits the *upper part* of a character while standing. Standing Low Hits is triggered on low hits.
- For *regular* standing and crouching attacks you can set different clips based on the Move's [Hit Type](#).

- Launcher animations are triggered when the character is launched *up*.
- Knockdown animations knocks the character down (which then triggers respective Stand Up animations).
- If Auto Speed is toggled the engine will use the [hit stun](#) applied to synchronize it with the animation length. Untoggle to run your own speed and [WrapMode](#) values.

Stage Reactions: Triggers when the character getting hit *interacts* with the stage (ground and wall bounce).

- You must have the respective bounce options under [Bounce Options](#).
- Unlike Air Hit, Ground Bounce does not operate using the auto-speed algorithm. Use [WrapMode](#) on *Clamp Forever* so the character stays in a single air pose.
- To adjust a stage reaction of your move go to [Active Frame](#) → Stage Reactions.
- Untoggle Grounded (Standing) under [Active Frame](#) → Stage Reactions → Wall Bounce Options to use Standing Wall Bounce (knockdown).

Fall Down Reactions: Triggered when the character hits the floor after being knocked down.

- The Default animation is triggered whenever the engine can't find the related knockdown animation.
- In certain animations you can set a Fall Clip and a Down Clip. Use this sequence when you want the Down Clip to be a loop (like breathing).
- If you don't want to have 2 clips it's important that the [WrapMode](#) is set to *Clamp Forever* under Fall Clip, so the character stays down (as the last frame should have the character laying down).
- Air Recovery is triggered if the character is recovered from Stun in mid-air. Make sure you have the proper option enabled under [Combo Options](#) -> [Air Recovery Type](#).
- Use *Auto Speed* to get the animation to follow the height path or have it end in a single pose with [WrapMode](#) set to *Clamp Forever*.

Stand Up Animations: Triggered after the knockdown timer runs out (Adjustable under [Global](#) -> [Knock Down Options](#)).

- The Default animation is triggered whenever the engine can't find the related stand up animation.
- All *Knockdown* animations can have its own follow up *stand up* animation.
Example: You may have a sweep animation that have your character facing down instead of facing up. You can set that animation to end at the down state and have a specific stand up for the sweep animation as a continuation of that clip.

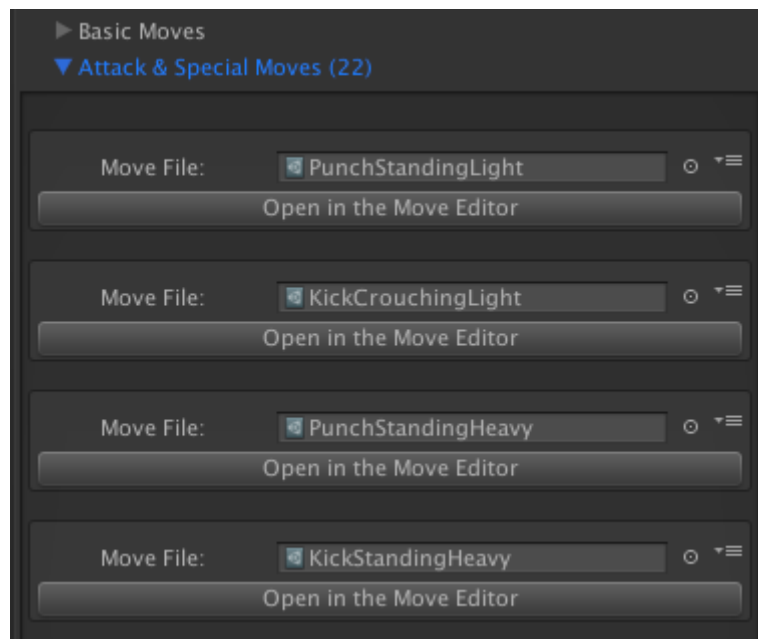
Notes:

- You can return the default speed and [wrapMode](#) of any Basic Move by clicking on the *Default* button next to [WrapMode](#).
- Because of how fast a [hit stun](#) can be, *it's recommended* that you set the override blending (in) values from *Hit*, *Block* and *Parry* animations to 0.
- None of the *Parry* moves are required if you don't have [parry enabled](#) in your game.
- None of the *Block* moves are required if you don't have [block enabled](#) in your game.
- *Bounce* and *Falling From Bounce* animations are not required if you don't have bounce enabled in your game.
- Some animations allow you to hide the hitboxes. Can be useful if you want certain animations to leave the character invincible during a certain state. A good example is having *Air Recovery*

invincible to emulate the Street Fighter style of air recovery (along with the option **Cant Move** under [Air Recovery Type](#)).

Attack & Special Moves

After creating a [new move](#), add the move to the character by clicking **New Move** than dragging the file to the following field.



Notes:

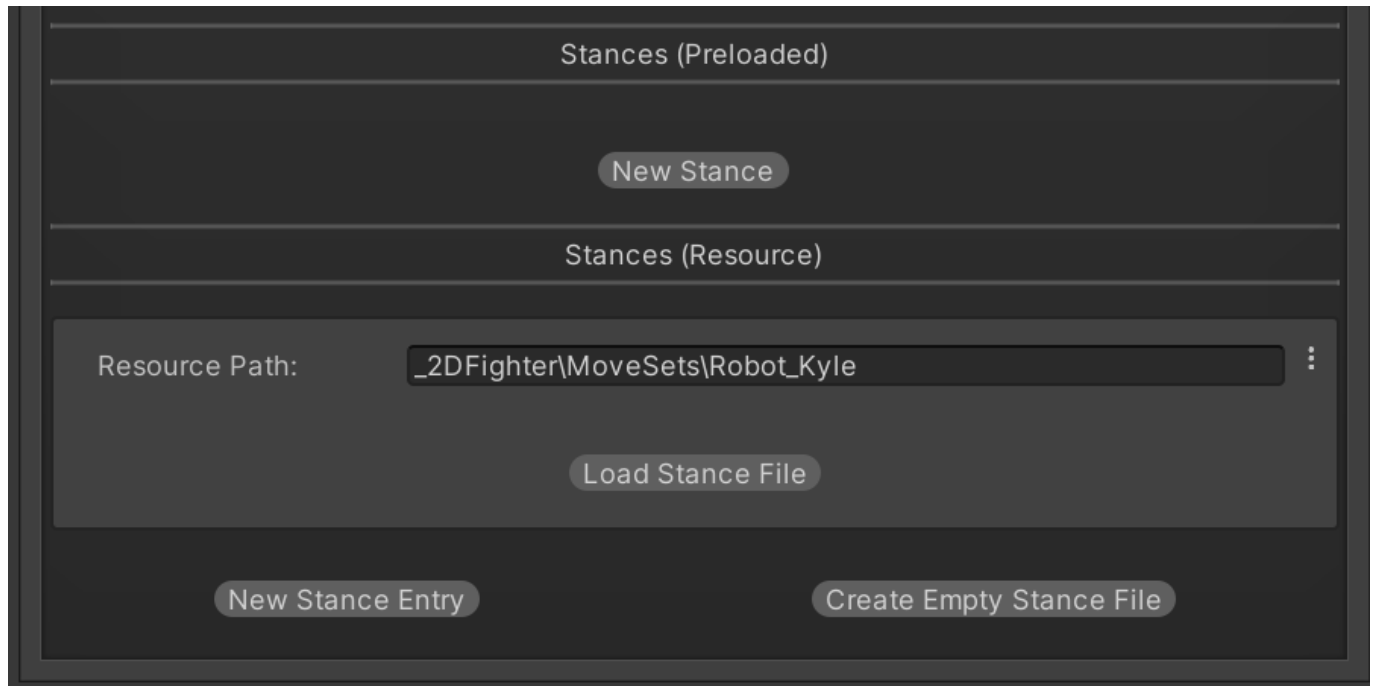
- Moving a move file up/down with the panel options has no effect on priority of moves. It only affects the list order displayed in this panel.
- You can find several move samples under `. \UFE\Characters\Mike\Moves`

Code access:

```
UFE.config.player1Character.moves
```

```
UFE.config.player2Character.moves
```

Stances



Stances (Preloaded):

- The stance file will always be loaded into memory.
- Loading times will likely increase when using this.

Stances (Resource):

- The stance file will be loaded from a Resources folder.
- Loading times will likely decrease when using this.

[< Back to Character Editor](#)

From:

<http://www.ufe3d.com/> - **Universal Fighting Engine**

Permanent link:

<http://www.ufe3d.com/doku.php/character:movesets>

Last update: **2023/07/29 12:51**

