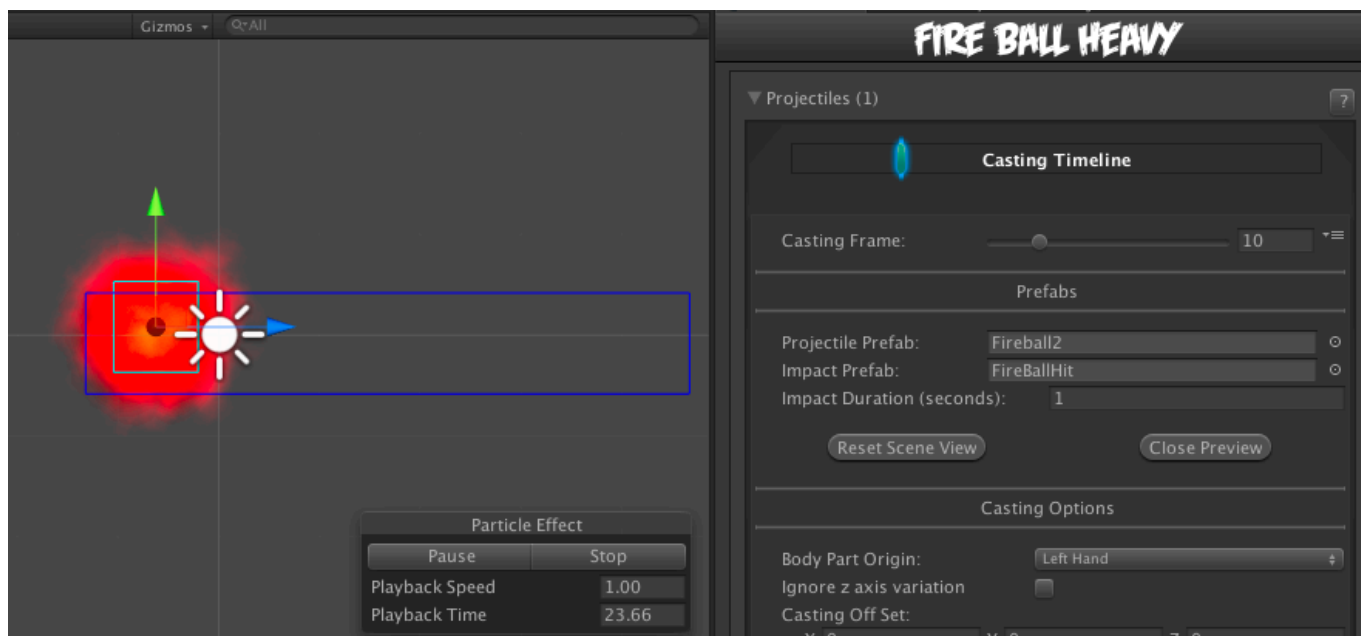


Projectiles

Casts a projectile with the following options

Casting Frame: When during the move the should the projectile be created.



Prefabs

Projectile Prefab: Drag a game prefab to represent the projectile.

Impact Prefab: A game prefab that is created upon projectile impact.

Impact Duration (seconds): Time until the Impact Prefab is destroyed.

Preview Click this button to preview the projectile. Cyan hitbox is the impact, Dark Blue hitbox is the blockable area. Note: your projectile may not appear if the particle effect is a child in the prefab. In this case, just select the child in the hierarchy and it will appear.

Casting Options

Body Part Origin: Where on the character the projectile is created.

Ignore z axis variation: If enabled, projectile is always cast with the same z position as the characters. Enable this for more accurate alignment and consistent hitbox behavior, especially if your camera tends to rotate a lot or has wide perspective.

Projectiles (1)

Casting Timeline

Casting Frame:

10

Prefabs

Projectile Prefab:

Fireball2

Impact Prefab:

FireBallHit2

Impact Duration (seconds):

0.6

Preview

Casting Options

Body Part Origin:

Left Hand

Spawn Position:

Attack HitBox

Ignore z axis variation

☒

Casting Off Set:

X 2

Y 1.43

Z 0

Duration (Seconds):

5

Mirror on Right Side

☒

Apply Gravity

☐

Speed:

15

Direction (Angle):

0

Apply Gravity

☒

Force Applied:

X 25

Y 5

Weight:

100

Limit Multicasting

☒

- Only This Projectile

☒

- On Screen Limit:

1

Hit Area

Shape:

Circle

Radius:

0.5

Off Set:

X 0

Y 0

Blockable Area

Unblockable

☐

Shape:

Rectangle

Rectangle:

X -1

Y -1

W 6

H 2

Hit Conditions

Hit projectiles

☒

Hit ground opponent

☒

Hit air opponent

☒

Hit down opponent

☒

Hit Type:

Mid

Collision Options

Total Hits:

1

Reset X Forces

☒

Reset Y Forces

☒

Ground Bounce

☒

Push Force:

X 20

Y 0

Apply Different Air Force

☒

Applied Force (Air)

X 10

Y 10

Apply Different Block Force

☒

Applied Force (Block)

X 0

Y 0

Obey Directional Hit

☒

Display Hit Effects On Strike

☒

Hit Strength:

Heavy

Armor Breaker

☒

Fix Rotation

☒

Override Hit Effects

☒

Hit Effects

Particle Effect:

None (Game Object)

Effect Duration:

0

Sound Effect:

None (Audio Clip)

Freezing Time:

0

Animation Speed (%):

0.1

Mirror on Right Side

☐

Shake Character On Hit

☒

- Shake Density:

0.8

Shake Camera On Hit

☒

- Shake Density:

0.8

Destroy When Off Camera

☒

Camera Bounds Offset:

2

Damage Options

Damage Type:

Points

Damage on Hit:

80

Damage on Block:

25

Damage Scaling

☒

Hit Doesn't Kill

☒

Hit Stun Options

Reset Hit Stun

☒

Hit Stun on Hit:

45

Hit Stun on Block:

10

Move Override

On Strike:

None (Move Info)

On Block:

None (Move Info)

On Parry:

None (Move Info)

Force Grounded

☒

New Projectile

Casting Offset: Offset the casting position relative to the bodypart origin.

Speed: The speed of the projectile.

Direction (Angle): Select the angle the projectile will be fired at. 0 is forward.

Duration (Seconds): How long the projectile will last on screen before self destruct.

Hit Area

Shape: Select between *circle* and *rectangle*.

(circle) **Hit Radius\Offset:** The radius and offset of the projectile's hurtbox. For a more consistent hit effect, by default radius is set to 0 (hit only detected when the "dot" crosses one of the opponent's hit boxes).

(rectangle) **Rectangle:** The position and size of the rectangle hurt box relative to the origin of the projectile prefab. If Follow Projectile Bounds is set below, you can use W and H to adjust the width and height relative to the projectile.

(rectangle) **Follow Projectile Bounds (X/Y):** If enabled, the rectangle will follow the render bounds of the projectile prefab.

Blockable Area

Unblockable: If enabled, this projectile is unblockable and this sections options are disabled.

Shape: Select between *circle* and *rectangle*.

(circle) **Hit Radius\Offset:** The radius and offset of the projectile's blockable area.

(rectangle) **Rectangle:** The position and size of the rectangle blockable area relative to the origin of the projectile prefab. If Follow Projectile Bounds is set below, you can use W and H to adjust the width and height relative to the projectile.

(rectangle) **Follow Projectile Bounds (X/Y):** If enabled, the rectangle will follow the render bounds of the projectile prefab.

Hit Conditions

Hit Projectiles: Should this projectile collide with other projectiles?

Hit ground opponent: If toggled on, this projectile can strike opponents that are on the ground (but not knockdown)

Hit air opponent: If toggled on, this projectile can strike air opponents.

Hit down opponent: If toggled on, this projectile can strike knocked down opponents. Make sure you have [knockdown](#) hitboxes toggled on.

Hit Type: Determine the hit conditions for this move to be blockable or trigger a different animation.

- High Low: Can be blocked high (standing) or low (crouching).
- Low: Can only be blocked low.
- Overhead: Can only be blocked high. Its recommended to always use this for air moves.
- Launcher: Can be blocked high or low. If hit, sets the opposing character direct into juggle animation.
- High Knockdown: Instantly send the opponent into Get Hit High Knockdown animation. This hit can only be blocked high.
- Mid Knockdown: Instantly send the opponent into Get Hit High Knockdown animation. This hit can be blocked either high or low.
- Knock Back: Instantly send the opponent into Get Hit Knock Back animation. Must apply vertical force for it to work.
- Sweep: Instantly send the opponent into Get Hit Sweep animation. This hit can only be blocked low.

Collision Options

Total Hits: If this projectile can hit multiple times, set them here.

Collision Options

Total Hits: 1

Push Force:
X 20 Y 0

Hit Strength: Heavy

Override Hit Effects ☐

► Damage Options

▼ Hit Stun Options

Reset Hit Stun ☒

Hit Stun on Hit: 35

Hit Stun on Block: 10

New Projectile

Space Between Hits: If the projectile has more than 1 one hit, how much interval between each hit should it hit again. Example: Selecting High will put more time between hits.

Push Force: When the projectile hits, how far should it push the opponent.

Hit Strength: Set what kind of hit this is based on your [hit effect](#) options.

Override Hit Effect: If enabled, it will replace the default hit effect by the one selected in the panel below this option.

- **Particle Effect:** The particle effect that is fired when the character gets hit.
- **Effect Duration:** Time before effect prefab is destroyed.
- **Sound Effect:** Sound effect that will play during hit.
- **Freezing Time:** The game will freeze for a very brief moment (in seconds) when a hit connects.
- **Animation Speed (%):** On hit, the animation speed of both characters will be set for this value.
- **Shake Character On Hit:** During the freezing time, the character can shake very slightly giving a more convincing hit effect.
- **Shake Camera On Hit:** To make the hit more impacting, you can also set the camera to shake slightly during freezing time.
- **Shake Density:** How much shake will be applied to the camera and character during freezing time.

Damage Options

Damage Type: Set if the damage is in points or percentage.

Damage on Hit: Damage caused on hit.

Damage on Block: Damage caused on block.

Damage Scaling: Whether or not damage is scaled down based on [Combo Options](#).

Hit Stun Options

Reset Hit Stun: If you are using [hit stun deterioration](#), enable this to reset the stun and its deterioration value.

Hit Stun on Hit: The amount of frames the opponent will remain stunned for after a hit.

Hit Stun on Block: The amount of frames the opponent will remain stunned for after a block.

Move Override

On Strike: If the projectile hits the opponent, override whatever the caster is doing with this move.

On Block: If the projectile hits the opponent while they are blocking, override whatever the caster is doing with this move.

On Parry: If the opponent parry this projectile, override whatever the caster is doing with this move.

Code example:

```
void OnHit(HitBox strokeHitBox, MoveInfo move, CharacterInfo hitter){  
    foreach(Projectile projectile in move.projectiles){  
        Debug.Log("Damage:" + projectile.damageOnHit);  
    }  
}
```

[< Back to Move Editor](#)

From:

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

Permanent link:

<http://www.ufe3d.com/doku.php/move:projectiles?rev=1431668753>

Last update: **2015/05/15 01:45**

