

Input Options

Choose the buttons or button sequences this move requires in order to be executed.

▼ Input

▼ Default Inputs

▼ Button Sequences (3)

Charge Move ☒

Charge Timing: 0.7

Allow Input Leniency ☒

Leniency Input Buffer: 4

Allow Negative Edge ☒

Force Axis Precision ☒

Button: Forward

Button: Down

Button: Forward

New Button Sequence

▼ Button Executions (1)

On Button Press ☒

On Button Release ☒

- Requires Press First ☒

Button: Button 1

New Button Execution

► Alternative Inputs

Charge Move: If toggled, the first button press of your *Button Sequence* must be held down by the *Charge Timing* value defined below.

(Charge Move) **Charge Timing:** How long, in seconds, input must be held down for.

Allow Input Leniency: If this move has a sequence, this allows the player to input extra buttons between its button sequence. For a more detailed explanation on what it is and how it works, check out this [link](#).

(Allow Input Leniency) **Leniency Input Buffer:** If *Allow Input Leniency* is toggled on, define how many inputs can be pressed during this move's input sequence.

Allow Negative Edge: When toggled, the engine will read for [button up](#) when allowing this move to be executed.

Button Sequences: The buttons required to trigger the first sequence of your special move. You can set how fast a sequence must be executed by changing the *execution timing* in the [character's move set](#). If your move is just a standard attack, leave this option empty.

Button Executions: After the sequence of buttons have been successfully executed, use this option to select which button(s) must be pressed in order to successfully trigger the move. If your move is just a standard attack, just assign its button here.

- **On Button Press:** Move is executed on press of the button(s).
- **On Button Release:** Move is executed on release of the button(s).
 - *Requires Press First:* If toggled, this move will only come out if the engine detects the button was previously pressed. Useful to have in focus attack like moves where input execution sometimes occur during buffers.

In Street Fighter, special moves can be executed either on Press or on Release. Enable both options to mimic that behaviour.

Normal attacks should only be executed on Press.

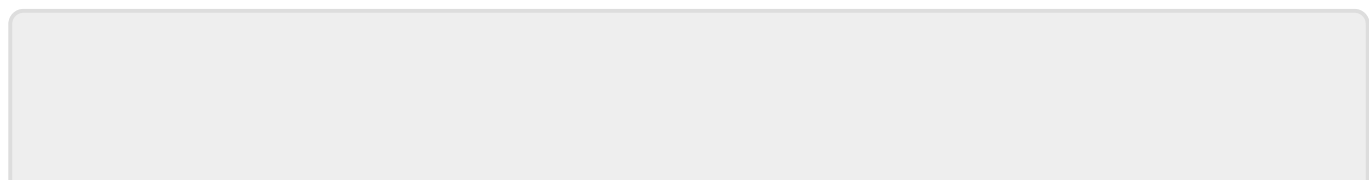
Boxer's TAP or MvC3's Zero's Hyper Zero Blaster should only be executed on Release.

NOTE: If both Press and Release are off, then the move cannot be executed. At least one of the must be enabled to allow the move to be executed.

Code example:

```
void OnMove(MoveInfo move, CharacterInfo player){
    // Creates a glow effect while holding the charge for a blaster
    if (move.moveName == "Charging Blaster"){
        _specialGlow = (GameObject)Instantiate(specialGlow);
        _specialGlow.transform = player.characterPrefab.transform;
    }else if (move.moveName == "Release Blaster"){
        Destroy(_specialGlow);
    }
}
```

[< Back to Move Editor](#)



From:

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

Permanent link:

<http://www.ufe3d.com/doku.php/move:input?rev=1484120333>

Last update: **2017/01/11 02:38**

