

Getting Started with Animation Combo System:

- Watch the YouTube videos for extra details and information:

<https://www.youtube.com/playlist?list=PLTRqRMix9RtjK9vCxeFFT8qqbtrNGbPx->

Manuel Steps:

- In your script Create a ComboSequencer Object
- Call the Initialise() method in Start()
- Call the Update() method in your Update Loop
- Assign combos and the Animator references in the inspector
- Hit play!

```
{
    public class ComboSystem : MonoBehaviour
    {
        public ComboSequencer Sequencer;

        protected virtual void Start()
        {
            Sequencer.Initialise();
        }

        protected virtual void Update()
        {
            Sequencer.Update();
        }
    }
}
```

Input Sequencer

Common Traits:

- The EnableTimeLimit checkbox, instructs the sequencer to clear its buffer every X seconds (resetting the key input to 0)
- If the InputSequence list is longer in length than the Attacks list, the Sequencing type will set itself automatically to “Full”
- If the Attacks list is longer than the InputSequence list in length, the last correct Key stroke will chain the remaining Attack Animations together in 1 long sequence

Sequencing Type	Description	Link Begin Function	Link End Function
Full Sequencer	Player must input the full Key Sequence before triggering the combo and its Animations	The time at which we link/transition to the next animation in the combo chain	Does not have an effect
Partial Sequencer	<ul style="list-style-type: none"> - First correct key stroke triggers the first Animation in the sequence - Onwards: every correct key stroke within the LinkBegin and LinkEnd time span of the animation triggers the next Animation in the chain - Keystrokes (correct or false) that aren't within that timespan are completely ignored 	The point at which we start listening for input strokes	The point at which we stop listening for input strokes
Partial Appending Sequencer	With every Keystroke (correct or false) a command is put into a buffer (no waiting times). The Scheduler will execute the next command in queue accordingly	No effect	The time at which we link/transition to the next animation in the combo chain

Events: you could subscribe to the Dispatcher's Events (Dispatcher.cs) to extend and create custom logic

list of the current Events:

```
/// <summary>
/// an event sent at a specific timing for functions to handle the logic/to extend the hit scanning
/// </summary>
public static event HitScanHandler HitScanning;

/// <summary>
/// an event triggered at a specified timing for functions that want to implement a custom logic
/// </summary>
public static event GenericEventHandler GenericEvent;

/// <summary>
/// an event triggered when an attack is triggered
/// </summary>
public static event AttackTriggeredHandler AttackTriggered;

/// <summary>
/// an event triggered when a combo has been successfully completed
/// </summary>
public static event ComboCompletedHandler ComboCompleted;
```

```
private void OnEnable()
{
    Dispatcher.HitScanning += OnHitScanning;
    Dispatcher.ComboCompleted += OnComboCompleted;
    Dispatcher.AttackTriggered += OnAttackTriggered;
    Dispatcher.GenericEvent += OnGenericEvent;
}

private void OnDisable()
{
    Dispatcher.HitScanning -= OnHitScanning;
    Dispatcher.ComboCompleted -= OnComboCompleted;
    Dispatcher.AttackTriggered -= OnAttackTriggered;
    Dispatcher.GenericEvent -= OnGenericEvent;
}
```