

# Ramza Dual Wield Usage Tutorial

Version 1.2

[Introduction:](#)

[Plugin Downloads:](#)

[Plugin Dependencies:](#)

[What It Does:](#)

[Plug And Play - What does it do?:](#)

[Advanced:](#)

[Plugin Parameter Descriptions:](#)

[Disable State ID:](#)

[Barehanded Attack:](#)

[Damage Modifiers:](#)

[Special Note:](#)

[Really Easy Mode:](#)

[Easy Mode:](#)

[Intermediate Mode:](#)

[Advanced Mode:](#)

[The Menu Extension Plugin:](#)

[What does it do?](#)

[Integration with Shield Block Plugin:](#)

[Plugin Parameters:](#)

[Split Attack:](#)

[Mainhand Slot Name:](#)

[Offhand Slot Name:](#)

[Show Block Chance:](#)

[Block Chance Text:](#)

[YEP\\_StatusMenuCore Block % Color:](#)

[YEP\\_StatusMenuCore Block Chance Name:](#)

[YEP\\_StatusMenuCore Mainhand Hit Chance Name:](#)

[YEP\\_StatusMenuCore Offhand Hit Chance Name](#)

[YEP\\_StatusMenuCore Mainhand Crit Chance Name](#)

[YEP\\_StatusMenuCore Offhand Crit Chance Name](#)

[YEP\\_ShopMenuCore Show Possession # on Split Pane](#)

[Additional Info About the Menu Extension](#)

[Closing Statement:](#)

## Introduction:

Thanks for checking out the dual wield plugin. This tutorial aims to make it as simple as possible to include some of the many features of the dual wield plugin in your own RPG Maker MV project.

## Plugin Downloads:

You can always find the latest downloads for all of my RMMV plugins from my [itch.io page](#). The dual wield menu extension plugin is also located on my itch.io page. The first section of the actual tutorial will assume you're only using the base dual wield plugin, for simplicity, but the more advanced sections will require you to use the menu extension plugin as well.

## Plugin Dependencies:

- You are required to be using [Yanfly's Battle Engine Core](#) plugin.
- Please note that if you are also using [Yanfly's Base Parameter Control](#) plugin, the dual wield plugin needs to load after it in the plugin load order in your project.
- If you also choose to use the menu extension plugin, it needs to load after any yanfly plugins that it modifies, (status menu core, equip core, party change, shop menu core) but these yanfly plugins are not required.
- Optionally, the menu extension plugin has several features that also enhance the shield block plugin (also located on my [itch.io](#) page).

## What It Does:

This plugin is very customizable, but it does several things on it's own without being configured by the user.

- Normally, an actor set as dual wield type in the database loses the ability to equip armor in their second armor slot (the shield by default) in order to equip a weapon in that slot instead. This plugin makes it so that a dual wield actor can choose to equip either a weapon or a shield in that slot. The **only** equipment type that can be equipped in this slot is equipment type ID 2, which in the default database is the shield. This value is **not configurable**.
- Dual wield type actors will repeat an attack action a second time if their offhand is armed. Note that having no weapon in the offhand is still considered armed, but that there is a plugin parameter to disable this if desired.
- When the dual wield actor makes multiple attacks, the weapon sprite and attack animation shown for the attack is based on the weapon being held in each hand. The

first hit will show the mainhand weapon, the second will show the offhand. Any subsequent attacks will continue to alternate in this fashion (if you have states that add even more attacks, as an example).

- The damage formulas for the default attack skill for dual wielding actors will be edited automatically when a.atk is used in the calculation to only use the atk value from the mainhand weapon, except on the second swing, where it will pull the offhand weapons atk value instead.
  - For example, if your damage formula is normally  $(a.atk * 4) - (b.def * 2)$  it will be automatically changed to  $(a.mhcalc() * 4) - (b.def * 2)$
- The mainhand attack of a dual wield actor will ignore the traits on the offhand weapon (unless tagged as an offhand skill, see below). The result of this is that exparam changes on the offhand weapon are not taken into account at all on normal physical skills, and the attack skills mainhand attack.
  - For example, if one were to wield a weapon with 30% accuracy in his mainhand, and a weapon with 90% accuracy in his offhand, his accuracy for skills and attacks would not be treated as if it were 120%.
  - The same is true for critical strike chance, and any elemental rates your weapons have. Wielding a weapon with ice element in your offhand will not give you a type advantage for your mainhand attacks, nor will it cause them to deal no damage to something that is immune to ice damage.
- The offhand hit in the normal attack skill, as well as any skills tagged with the <offhand attack> note tag will use the traits and atk parameter from only the offhand weapon.
- You may tag skills in certain ways in order to use specific values for your atk parameters in those skills. There is extra functionality if you are also making use of [Yanfly's Action Sequence](#) plugins to make skills that hit multiple times. An untagged skill will use the combined traits and attack power of both weapons for a dual-wielding actor which is how the default dual wielding engine handled it before.
  - <Mainhand Skill> The attack damage from this skill will be calculated by using only the mainhand weapon's traits and parameters. If you're using action sequences, and this attack hits multiple times, each hit will only show the mainhand weapon for a dual wielding actor.
  - <Dualwield Skill> Tagging a skill with this tag will make it deal split damage. Each swing for this skill will alternate between the mainhand and offhand attack sprite/animation, and the mainhand and offhand weapon traits. This will work with skills that hit multiple times by default, but is much more suited to using the action sequences from the above mentioned plugin.
  - <Offhand Skill> A skill tagged as an offhand skill will only be usable by a dual wield actor who is considered armed, and all hits from that skill will only use the traits and atk param of the weapon held in the offhand.

## Plug And Play - What does it do?:

Importing this plugin into your database without changing any of the default has the following immediately noticeable effects:

- Dual wield actors hit twice with their basic attack
- Dual wield actors can equip shields in their offhand slot, where before it was only weapons.
- Each hit from the default attack skill deals split damage, which is to say that the first swing uses the parameters of the mainhand weapon, and the second swing uses the parameters of the offhand.
- Default weapon modifiers are applied to the offhand. Wielding two identical weapons will see the offhand have 25% less atk value from the actor's base attack.
- Non dual wielding actors are completely unaffected by these changes.

### Advanced:

- You may tag weapons in the database as being <Mainhand> <Offhand> or <Twohanded>. Main and offhand weapons can only be equipped in those slots, while a twohanded weapon will cause the wielder to release any weapon or shield in his offhand.
- The default modifier for a twohanded weapon is +50%
- You can also tag shields with <Offhand> to allow them to be held while using a twohanded weapon.
- You can tag weapons with <TwoHanded Attack Modifier> or <Offhand Attack Modifier> to modify the base atk value that weapon grants when being two-handed, or wielded in the offhand. This is different from the damage modifier plugin parameter, as it only affects the parameters of the weapon itself, not the attack value of the actor.

## Plugin Parameter Descriptions:

Here I will list a slightly more in depth description for each plugin parameter in the dual wield plugin, what it does, and an example for how one might use it.

### Disable State ID:

- Any actor who has this state on them will be ignored by the dual wield plugin. They will not strike twice, will not be considered dual wield type (even if set on the actors page in the editor), and will be unable to equip two weapons.
- One use for this state is to be able to prevent using an offhand weapon with a specific weapon in the mainhand, or as some sort of disabling state that makes the actor's offhand unusable for some reason.

## Barehanded Attack:

- By default, the offhand is considered armed when a dual wield actor has no weapon in it, this means if he is using one weapon in his mainhand and no weapon in his offhand, he will still strike twice, but the second hit will be barehanded.
- Setting this to false will disable the extra barehanded attack, unless both hands are empty.

## Damage Modifiers:

- As mentioned briefly above, the damage formula for default attacks is automatically changed to grab only the value for the weapon being used for that attack or skill. These modifiers change the value of the atk parameter coming from the weapon before it is used in the damage calculation, allowing a developer to make the offhand weaker, or both attacks weaker when dual wielding. The twohanded multiplier also applies to non-dual wield actors who are using a weapon tagged with <twohanded>.
- The default values in these boxes make the offhand have 25% lower atk value, as an offhand penalty. These boxes can also use javascript evals as inputs, meaning you can make advanced systems where different actors, or classes, or even items, can have different penalties (or bonuses).
- The dual wield modifier is applied to both mainhand and offhand attacks made while dual wielding, the offhand penalty is only applied to the offhand attacks, and the twohanded modifier is only applied to attacks made using a two-handed weapon.

## Weapon Modifier Tags:

Added in version 2.56

- Individual weapons can be tagged with new note tags that modify their base attack value based on whether they are being two-handed, or off-handed.
- These tags are designed so that a javascript eval can be used to determine a variable change to the weapons base attack value. The variable value is added to the base attack value of the weapon at the end of the calculation. 'Item' can be used to reference the weapon itself, as well. Variables and switches can also be referenced with v[x] and s[x] respectively.

## Special Note:

As mentioned above, the skill note tags for dual wielding skills get the best use if they're being used in an action sequence. I have some additional action sequence notes that don't really fit in anywhere else, so I'll mention them here:

- The default attack skill simply replays the entire <Target Action> step of the action sequence. This was to maximize compatibility with the plug and play aspect of the plugin. If you have camera moves or sprite movements in your target action section of the sequence, you can prevent them from playing a second time by moving them to the <setup action> section, where they should've been to begin with :).
- A skill that simply has its "Repeat" option set to more than one will repeat the entire sequence, not just the target action section, but won't alternate weapon swings, or use the split damage formula.
- A skill tagged <Mainhand Skill> or <Offhand Skill> will use split damage values for all hits, and show the correct weapon.
- Skills tagged <Dualwield Skill> work very differently from the default attack skill. Instead of repeating the entire <Target Action> section of the action sequence, a new action sequence command is added to force specific sections of the sequence to repeat if you're dual wielding. DUALWIELD ACTION: *steps* repeats the next *steps* number of actions sequence commands (using the offhand weapon sprite/animation). You can still use two separate MOTION *attack* commands and ATTACK ANIMATION: *target* commands in the <Target Action> section, and it will still split damage in this way, but it will not automatically hit twice unless you add in the new command. This command was added to give you the ability to make a very neat looking sequence, that doesn't get all jumbled up by having unwanted commands repeat (such as camera moves, etc). Note that this command has no effect if the user of the skill isn't dual wielding.
- If you tag a skill as <Dual Wield Skill> but don't set any dual wield action repeat steps, the entire sequence will play only once. If it hits multiple times (multiple motion attack commands in the same sequence), each hit will alternate between the main and offhand weapon, and all hits will damage split between main and offhand damage values.
- If a dual wield actor uses a multi-hit skill that isn't tagged <Dual Wield Skill>, each swing will still alternate between the main and offhand weapons, but all hits will deal damage as if they were the mainhand weapon only.

## Really Easy Mode:

Import the plugin anywhere after the YEP\_BattleEngineCore plugin. You now have dual wielding.

## Easy Mode:

Alright, the above is kind of basic, so this is the most realistic use example for this plugin.

- Import the plugin into your project below your other battle plugins. Follow Yanfly's recommended [plugin load order](#).
- Modify the plugin parameter values for the attack modifiers. The default values cause the offhand to have a 75% penalty to the base atk parameter of the actor, meaning if an actor has 100 attack, and is dual wielding 10 attack weapons, his main hand will have 110 attack, and his offhand will have 85.
- Set any dual wielding actor to dual wield type, either by using a state, on his/her class, or on his actor page in the database.
- Tag any weapons you don't want to be held in the offhand as <Mainhand>, any offhand only weapons as <Offhand>, and if desired, any weapons that should prevent dual wielding entirely as <Twohanded>.
- Your weapons need to be equipment type 1, and shields need to be type 2. This is not



negotiable.

- Tag any shield that you want to still be equippable by someone who is also using a two handed weapon as <Offhand> (like a buckler or something).
- Tag any multi-hit skills that should only ever show the mainhand weapon as <Mainhand Skill>
- Tag skills that can only be used by dual wield actors who have an offhand weapon equipped, and should only be performed with the offhand weapon as <Offhand Skill>

Any multi-hit skills will automatically swap weapons after each hit unless tagged to use one weapon or the other specifically.

Any normal attacks performed by a dual wield actor who is using two weapons will strike twice automatically, with the first hit ignoring any traits on the offhand weapon, and the second strike ignoring traits from the mainhand weapon.

If the actor has a state that gives him Attack+, each additional attack will alternate between the main and offhand weapons in the same way.

## Intermediate Mode:

Using the action sequence plugins, you can tag a skill a <Dualwield Skill>. The only thing this tag does on it's own is cause split damage to happen, and if there are multiple hits in the skill, this will be seen by each hit alternating which weaponsprite is shown.

Unlike with the default attack, a dualwield skill doesn't automatically repeat the <Target Action> section of your action sequence, instead, you must use the new action sequence command DUALWIELD ACTION: *steps*. This will cause the sequence to perform the next *steps* commands, and then immediately repeat them again. The helpfile gives an example sequence:

```
* <target action>
* dualwield action: 6
* motion attack: user
* wait: 12
* attack animation: target
* wait: 4
* action effect: target
* wait for animation
* </target action>
```

In the given example, the sequence will repeat the next 6 actions after it a second time, which will cause it to swap weapons. In this example, the 6 steps are the entire <target action> sequence, but that doesn't have to be the case, as your skill sequence could include camera moves, or sprite moves that you do not want to be repeated.

It is also worth noting that the command doesn't have any effect if the user of the skill isn't dual wielding, meaning you can put this command in skills that non-dual wielding actors use, and not have to worry about the sequence being wrong, or hitting twice, when they aren't dual wielding.

# Advanced Mode:

## The Menu Extension Plugin:

The menu extension plugin is a lot less plug and play, it has many settings that should be looked at before using it.

### What does it do?

The menu extension plugs into several of the default menus, and modifies the way they look to include the option to show the individual atk value of each equipped weapon for dual wielding actors. By default, the status scene, equip scene, and shop scene are modified to include info about the offhand item.

Default equip scene:



Modified equip scene:



Default Status Menu Scene:



Modified Status Menu:



It also plugs into several yanfly engine menu plugins to include the offhand information on the following scenes:

### Status Menu:

General  
Parameters  
Elements  
States

Therese  
Lv 1

Hero  
HP 450/450  
MP 95/95  
TP 0

Parameters

Experience

Level	1	Current EXP	0
Max HP	450		
Max MP	95	To Next Level	
Attack	33	Defense	37
M.Attack	17	M.Defense	16
Agility	32	Luck	32
		Total EXP for Next Level	50

General  
Parameters  
Elements  
States

Therese  
Lv 1

Hero  
HP 450/450  
MP 95/95  
TP 0

Parameters

Experience

Max HP	450	Current EXP	0
Max MP	95		
Main	21	Off	24
Defense	37	M.Attack	17
M.Defense	16	Agility	32
Luck	32	Level	1
		Total EXP for Next Level	50

### Parameter Gauge scene:

General  
Parameters  
Elements  
States

Therese  
Lv 1

Hero  
HP 450/450  
MP 95/95  
TP 0

Parameter Graph

Attack	33
Defense	37
M.Attack	17
M.Defense	16
Agility	32
Luck	32

General  
Parameters  
Elements  
States

Therese  
Lv 1

Hero  
HP 450/450  
MP 95/95  
TP 0

Parameter Graph

Main Attack	21	Off Attack	24
Defense	37		
M.Attack	17		
M.Defense	16		
Agility	32		
Luck	32		

### Equip Scene:

Equip  
Optimize  
Clear  
Finish

Therese  
Lv 1

Hero  
HP 450/450  
MP 95/95  
TP 0

Weapon Sword  
Weapon Spear  
Head Hat  
Body Cloth  
Accessory ☐ <Empty>

Max HP 450 →  
Max MP 95 →  
Attack 33 →  
Defense 37 →  
M.Attack 17 →  
M.Defense 16 →  
Agility 32 →  
Luck 32 →

Equip  
Optimize  
Clear  
Finish

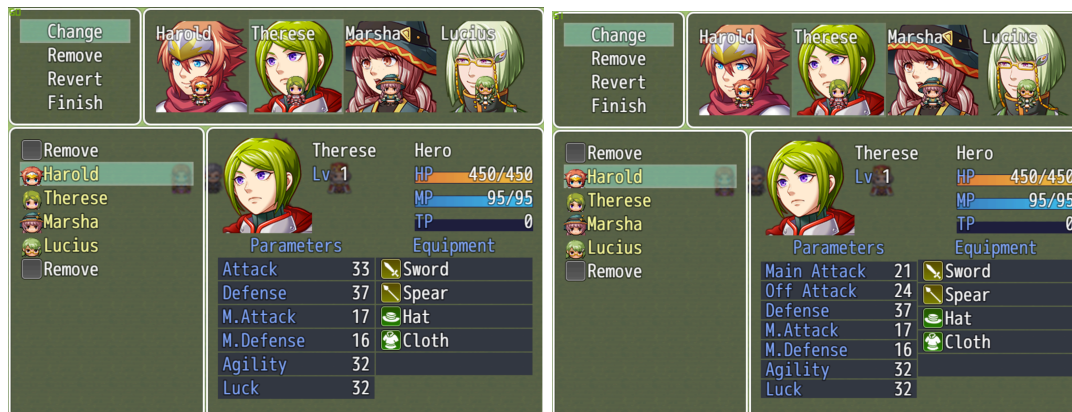
Therese  
Lv 1

Hero  
HP 450/450  
MP 95/95  
TP 0

Main Sword  
Off Spear  
Head Hat  
Body Cloth  
Accessory ☐ <Empty>

Max HP 450 →  
Max MP 95 →  
Main Attack 21 →  
Off Attack 24 →  
Defense 37 →  
M.Attack 17 →  
M.Defense 16 →  
Agility 32 →  
Luck 32 →

## Party Change Scene:



## Shop Scene:



## Integration with Shield Block Plugin:

The menu extension also integrates with my shield block plugin to show block chance in place of the offhand weapon values when a shield is equipped. This will apply to any actor with a shield equipped, not just dual wield type ones!



## Plugin Parameters:

Here I will describe each parameter in the menu extension.

### Split Attack:

#### **True/False**

- Show the individual atk values of the main and offhands in the modified menu scenes.
- If this is disabled, all scenes will look like the default examples above.

### Mainhand Slot Name:

- The text that will preface the atk params name for the mainhand slot. The default 'Main' will cause it to show 'Main Attack' in a place where it would show the mainhand attack's name. See above screenshots.

### Offhand Slot Name:

- As above, only affecting the offhand slot.

### Show Block Chance:

#### **Requires Ramza\_BlockChance**

- Shows the Block Chance exParam in place of the offhand attack slot if the actor is using a shield in his/her offhand.
- If you have split attacks turned off above, this parameter has no visual effect, as there is no section to show it in.

### Block Chance Text:

- The text that is shown on the menu scenes to indicate that the value is for block chance.

### YEP\_StatusMenuCore Block % Color:

- The color of the gauge that shows up specifically on the gauges scene for the YEP Status menu screen.

### YEP\_StatusMenuCore Block Chance Name:

- The long text which shows over the block gauge on the gauges scene, as well as on the parameter scene when you input 'blk' as the short code.

### YEP\_StatusMenuCore Mainhand Hit Chance Name:

- The long name that shows in the parameters scene for showing the mainhand hit chance.

### YEP\_StatusMenuCore Offhand Hit Chance Name

- Same as above except for the offhand hit chance.

### YEP\_StatusMenuCore Mainhand Crit Chance Name

- The long name that shows on the parameters scene for mainhand crit chance.

### YEP\_StatusMenuCore Offhand Crit Chance Name

- Same as above for the offhand crit chance.

### YEP\_ShopMenuCore Show Possession # on Split Pane

#### **True/False**

- Shows the # of a selected dualwield weapon on the bottom of the parameters pane, this is off by default because of how crowded that window normally is. This only affects the pane when it is showing the split params as in the screenshot for the shop scene above, although in that screenshot it isn't turned on.

## Additional Info About the Menu Extension

One other thing that can't accurately be portrayed by screenshots alone, is the fact that the offhand slot updates in real time when you change equips on a dual wield actor. If you swap a weapon for a shield, the param change section will indicate that your offhand attack will be going away entirely, rather than showing a loss of atk parameters. If you have the block chance plugin as well, it will show the increased block chance instead. Likewise, if you equip a twohanded weapon, the param change preview will show that your offhand weapon will be unequipped, rather than just a parameter change.

Additionally, equipment optimization has been changed to make sure it takes into account offhand only equipment, the modifiers from the base dual wield plugin will also play into whether or not it considers a specific weapon as an upgrade. If you have the block chance plugin, it will also consider block chance as a stat for weighting better items.

## Closing Statement:

Thanks again for using this plugin. Keep doing what you do, and make the best game you can. If you have any questions not covered in this document, feel free to ask on the [rpgmakerweb.com forums](http://rpgmakerweb.com/forums).