# Скрипт Smart merge для vMIX 4к и выше

## 1 Описание

Скрипт создан для формирования изображения плавного перехода между суперсорс источниками (***ME линейки***) при использовнии перехода типа **MERGE**.

Пример работы скрипта представлен в в папке ../Example/Example.mp4

Основной **принцип** работы: динамическое формирование **STINGER** перехода для каждого сочетания **Preview/Program**, эмулирующего эффект перехода типа **MERGE**

Источник для **STINGER’а** – вход типа **COLOUR** с **прозрачным фоном**, в котором собраны все (заранее определенное количество) источники для суперсорс. Каждый источник представлен отдельным входом типа **MIX** для которого применена маска, которая, в свою очередь тоже является входом типа **MIX**. И по факту происходит **MERGE** для каждого источнка по отдельности с последующим объединением этих изображений в один **STINGER**. Для каждого сочетания **Preview/Program** формируется свой **Preview/Program** внутри кажого такого **MIX** и в момент перехода происходит **MERGE** в каждом **MIX** а в переход в основной программе производится как **STINGER** в изображении которого отображеные все **MERGE** источников. (см. Рис.1.1)

Главным приемущества данного скрипта:

* при его внедрении в проект не вносится конструктивных изменений в изначальную структуру и логику проекта
* сохраняются все талли и в программе находятся INPUTы, состоящие из реальных источников,

все дополнительные шоты и INPUTы, используемые скриптом участвуют ТОЛЬКО в момент перехода и не вносят дополнительной существенной нагрузки на ЦП.

* все «тяжелые» манипуляции с API происходят только в случае, если все условия для выполнения скрипта выполены. Т.е. сначала происходит полный анализ текущей ситуации и лишь после всех манипуляций с данными формируется результрующее решение по переназначениям и другим взаимодейтсвиям с API.

Работа скрипта глобально состоит из следующих шагов

1. Формирования Program и Preview для каждого источника в индливидуальных входах типа MIX для изображения и маски
2. Одновременный запуск MERGE внутри всех MIXов
3. В случае успешного MERGE в основной программе примеяется STINGER, при ошибке или невозможности произветси MEGRE применяется FADE

Скрипт имеет возможность полной кастомизации:

* настраиваемая длительность перехода из GUI vMIX
* настраиваемое количество источников (ограничения только количеством уже задействованных входов типа MIX в проекте (всего до 16 в последней версии ))
* настраиваемое количество ME линеек.

Полное описание и исходный код представлен в приложении.

ME Y

PREVIEW

SRC1

SRC2

SRC3

ME X

PROGRAM

SRC1

SRC2

SRC3

>>>STINGER >>>

SRC1

VIDEO preview

SRC1

MASK preview

SRC2

VIDEO preview

SRC2

MASK preview

SRC3

VIDEO preview

SRC3

MASK preview

SRC1

VIDEO program

SRC1

MASK program

SRC2

VIDEO program

SRC2

MASK program

SRC3

VIDEO program

SRC3

MASK program

SOURCE 3 MIX

SRC3

MIX

preview

SRC3

MIX

preview

>MERGE>>

SOURCE 2 MIX

SRC2

MIX

preview

SRC2

MIX

preview

>MERGE>>

SOURCE 1 MIX

SRC1

MIX

preview

SRC1

MIX

preview

>MERGE>>

`

STINGER SOURCE INPUT

Рис. 1.1 Схема динамического формирования перехода STRINGER

## 2 Логика работы скрипта

При переходе из одного ME происходит трансформация положения и содержимого для каждого входящего источника SOURCE.

К примеру, мы переходим из ME X в ME Y

ME X

SOURCE 1

(INPUT 12)

SOURCE 2

(INPUT 32)

SOURCE 3

(INPUT 42)

↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓

ME Y

SOURCE 3

(INPUT 45)

SOURCE 1

(INPUT 12)

~~SOURCE 2~~

SOURCE 4

(INPUT 32)

~~SOURCE 3~~

SOURCE 2

(INPUT 15)

***Примечание: в примере выше в ME Y не совсем корректно изначально выбрано распределение SOURCE по номерам (зачеркнутый вариант). Желаетельное выбирать распределение (расположение) SOURCE по номерам так, чтобы при трансформации внутри одного номера SORCE вероятность перекрытия других источников было минимальным***

Итак, для данного примера имеем:

Максимальное количесвто SOURCE в конфигурации – 4

При переходе от ME X в ME Y

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | ME X |  | ME Y | Состояние |
| SOURCE 1 | INPUT 12 | → | INPUT 12 | ≡ |
| SOURCE 2 | INPUT 32 | → | INPUT 15 | → SOURCE 4 |
| SOURCE 3 | INPUT 42 | → | INPUT 45 | ≠ |
| SOURCE 4 | X | → | INPUT 32 | FROM SOURCE 2 ← |

Таблица 2.1

Работу скрипта в данном примере можно разделить на два этапа

1. Формирование Program внутри MIXов для SOURCE для того, чтобы получить изображение ME X
2. Подготовка Preview внутри MIXов для SOURCE для того, чтобы MERGE прошел максимально плавно в ME Y

Для каждого SOURCE каждом ME нам будут необходимы минимум(в рамках примера):

1. INPUT с позицией SOURCE для каждой ME (далее POS INPUT)
2. Маска с ОТКРЫТЫМ (прозрачность) положением для SOURCE которые присутствуют в ME ( далее для краткости MASK\_ON)
3. Маска с закрытым (непрозрачным) положением для SOURCE который отсутсвтует в ME ( далее для краткости MASK\_OFF)

Т.к. возможна (как в нашем примере) смена источника в SOURCE с одним номером в разных ME необходимо предусмотреть следующие ситуации:

1. Источник меняется внутри SOURCE с одним номером
2. Источник меняется внутри SOURCE идентичен источнику SOURCE с другим номером в другом ME.

Это влечет за собой необходимость добавить дополнительные INPUTы для этих сиутаций.

Распишем какие дополнительные INPUTы нам понадобятся для нашего примера (обозначение **a→b** следует понимать POS INPUT/MASK\_ON/MASK\_OFF для ситуации перехода SOURCE **a** в **SOURCE** b)

Т.к. в PROGRAM находится ME Y, поэтому составим таблицу вначале для него (данная конфигурация считает, что в PREVIEW находится ME, в которой у всех SOURCE источники те же, что и у текущей ME)

**ME Y**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SRC №** | **Y input** | **POS INPUT** | **MASK\_ON** | **MASK\_OFF** |
| 1 | 12 | 1→1 | 1→1 | none |
| 2 | 15 | 2→2 | 2→2 | none |
| 3 | 45 | 3→3 | 3→3 | none |
| 4 | 32 | 4→4 | 4→4 | none |

Таблица 2.2

Затем для ME X, который находится в PREVIEW, но с учетом того, какой ME находится в PROGRAM

**ME X**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SRC №** | **X input** |  | **Y input** | **Состояние** | **POS INPUT** | **MASK\_ON** | **MASK\_OFF** | **SIMPLE CHANGE** |
| 1 | 12 | → | 12 | ≡ | 1→1 | 1→1 | none | none |
| 2 | 32 | → | 15 | TO → SOURCE 4 | 2→4 | 2→4 | none | none |
| 3 | 42 | → | 45 | ≠ | 3→3 | 3→3 | none | √ |
| 4 | X | → | 42 | FROM SOURCE 2 ← | 4→4 | none | 4 ← 2 | none |

Таблица 2.3

Из этих таблиц видно, что по сути, для каждого SOURCE сущетвует четыре сценария:

1. **≡** *- источники для SOURCE одинаковы*
2. ≠ *- источники отличаются и ни один из них не соответствуют источникам для* ***SOURCE*** *c другими номерами*
3. TO → SOURCE B *– источник в* ***SOURCE A*** *соответсвует источнику в* ***SOURCE B****, происходит «трансформация»* ***SOURCE A*** *в* ***SOURCE B.***
4. FROM SOURCE A ← - *источник в* ***SOURCE B*** *соответсвует источнику в* ***SOURCE A****, происходит «трансформация»* ***SOURCE A*** *в* ***SOURCE B***

Даллее остается подставить нужные POS INPUT/MASK в соответтвующие PREVIEW/PROGRAM в соответсвующих MIXах и произвести одновременный MERGE в этих миксах. На этом скрипт заканчивает свою работу.

***Примечание: Если для сценария 3 или 4 нет INPUTа – выполнится сценарий 2.***

***Примечание: Скрипт перед началом своей работы собирает информацию об имеющихся POS INPUT/MASK\_ON/MASK\_OFF для каждой ME в виде таблицы вида. Для удобства рекомендуется заполнить предварительно таблицу необходимыми пунктами, чтобы минимизировать вероятность ошибки.***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **ME Y IN PREVIEW** | | | | |
| **ME X IN PROGRAM** |  | SOURCE 1 | SOURCE 2 | … | SOURCE n |
| SOURCE 1 | **1→1** | **1→2** | **1→…** | **1→n** |
| SOURCE 2 | **2→1** | **2→2** | **2→…** | **2→n** |
| ... | **…→1** | **…→2** | **…→…** | **…→n** |
| SOURCE n | **n→1** | **n→2** | **n→…** | **n→n** |

***Таблица 2.4***

## 2 Подготовка проекта для внедрения скрипта

Здесь и далее так => ( \*) будут обозначаться параметры, которые можно кастомизировать в настройках скрипта

Определимся с терминами

Схема формирования ME/SUPERSOURCE (далее для краткости просто ME)

INPUT 4

INPUT **5**

INPUT 1

INPUT M

INPUT **2**

INPUT 3

SOURCE 1

SOURCE 2

SOURCE 3

SOURCE N

ME / SUPERSOURCE INPUT

INPUT VIRTUAL MATRIX

...

Вход (INPUT) может быть преставлен непосредственно источником (Camera/Video/List….) или **COLOUR**, в котором целевой источник находится на определенном слое в **Multiview / Layers**

**SOURCE**, источник – в каждой **ME** виртуальный «вход» на который можеть быть «назначен» любой вход **INPUT** внутри проекта.

**INPUT VIRTUAL MARIX** – виртуальная матрица коммутации входных **INPUT** и выходных **SOURCE** для **ME**

«Мощность», т.е. количесвто выходных «каналов» **SOURCE** определяется самым «требовательным» **ME** источником.

Т.к. физически в vMIX нет никаких матричных коммутаторов и т.д. фунционал данной схемы заключен в самом INPUT для каждого **ME**.

### 2.1 Подготовка ME источника

Пример ME источника

SOURCE 1

SOURCE 2

SOURCE 3

У каждого источника в каждом **ME** есть свое положение (далее **POSITION**). Трансформация из одного **ME** в другой – есть трансформация положения каждого **SOURCE** в рамках своей позиции (**POSITION**). Поэтому даже если в конктретной ME не присутсвуют другие SOURCE, которые есть в других ME (скажем в одной ME – 2 источника, в другой 3, в третьей 4), в ней все равно дожна быть информация об остальных (по факту они становятся мниными/скрытыми).

В итоге: под **каждый** SOURCE необходимо **зарезервировать** слой в **Multiview / Layers.**

**СПИСОК СЛОЕВ (представлено для значений по умолчанию)**

1. DO NOT USE (fullFrameLayer) – (\*) не используйте этот слой т.к. он настроен по умолчанию как возможный слой для полнорастрового источника
2. Слой свободен. Можно исользовать для фона и проч...
3. Слой для источника номер 4 (\*) (если источник не задействован – **оставить слой пустым**)
4. Слой для источника номер 3 (\*) (если источник не задействован – **оставить слой пустым**)
5. Слой для источника номер 2 (\*)
6. Слой для источника номер 1 (\*)
7. Слой свободен. Можно использовать для альтернативных наложений (overlays)
8. Слой свободен. Можно использовать для альтернативных наложений (overlays)
9. Слой свободен. Можно использовать для альтернативных наложений (overlays)
10. DO NOT USE fullFramePOSlayer – (\*) слой на который скрипт перемещает целевой слой источника N если происхоит переход на полный растр из ME

***Позиции источников желательно окончательно утвердить перед тем, как приступать к подготовке дополнительных необходимых для работы скрипта шотов***

### 2.2 Использование виртуальной PTZ

Елси предполагается использование VIRTUAL PTZ на источнике, то необходимо:

Создать два виртуальных входа для этого источника

1. Виртруальный вход с активированной VIRTUAL PTZ и диапазоном зума 5x. Переименовать этот виртуальный вход

***“названиевхода\_POS”***,

где “\_POS” – ***POSpostFix*** – (\*) обязательный постфикс для VIRTUAL PTZ. Именно этот вход необходимо будет использовать в конфигурациях **ME**

1. Виртруальный вход для эмуляции Переименовать этот виртуальный вход

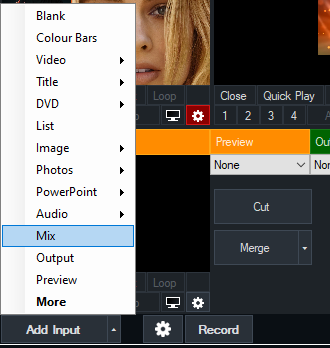
***“названиевхода\_FAKE\_PTZ”***,

где “***\_FAKE\_PTZ*** ” – ***fakePTZpostfix*** – (\*) обязательный постфикс для входа эмулятора VIRTUAL PTZ. **Это должен быть просто Virtual Input**

**В дальнейшем назначать на SOURCE для ME нужно будет именно витруальный «\*\_POS» вход, что позволит делать плавные merge на не подверженный VIRTUAL PTZ вход и позволит кадрировать source не меняя настройки его Cropping/Position**

### 2.3 Создание MIXов, необходимых для работы скрипта

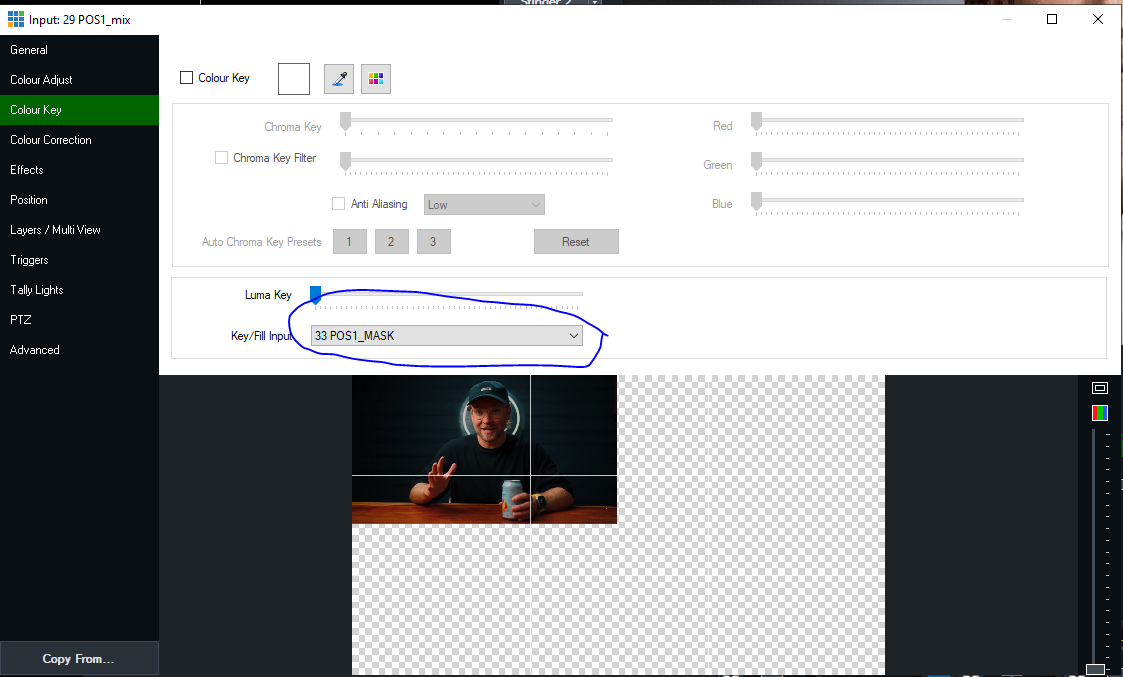
Необходимо создать 2\*N INPUTов типа MIX. N – определяется максимальным количеством SOURCE, используемых в ME линейках. Названия MIXов выберите любые на Ваше усмотрение (POS\*\_mix, POS\*\_MASK – (\*) – названия \*ных MIXов по умолчанию в скрипте)



N – MIXов основные для вывода каждого SOURCE

N – MIXов для масок.

Для каждого основного MIXа необходимо выбрать в качесве маски MIX-маску



*в дальнейшем номера и названия этих MIXов необходимо будет внести в раздел настроек скрипта.*

### 2.4 Создание суммирующего входа ME STINGER SOURCE и настройка перехода типа STINGER

#### ME STINGER SOURCE

Необходимо создать **INPUT** типа **COLOUR** с таким же как у ME INPUTов фоном, название можно выбрать произвольным («ME\_STINGER\_SOURCE» – (\*) – название по умолчанию в скрипте)

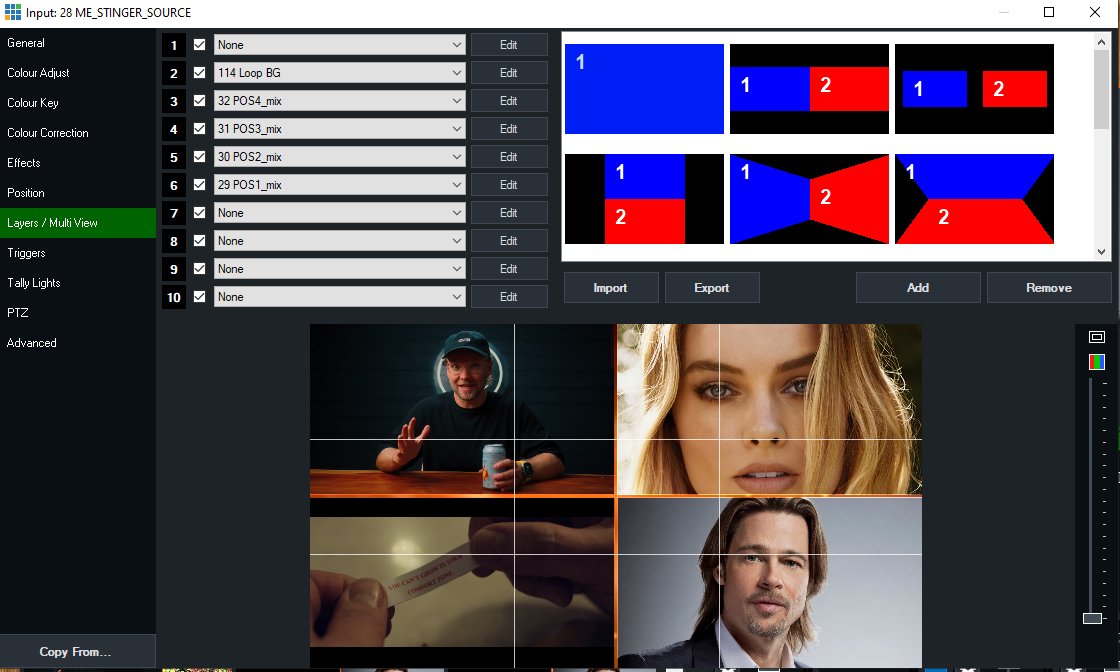


Рис. 2.4.1

Назначение слоев

1. DO NOT USE **fullFrameLayer** – (\*) не используйте этот слой т.к. он настроен по умолчанию как возможный слой для полнорастрового источника
2. Слой свободен. Можно исользовать для фона и проч...
3. POSITION MIX 4 (\*)
4. POSITION MIX 3 (\*)
5. POSITION MIX 2 (\*)
6. POSITION MIX 1 (\*)
7. Слой свободен. Можно использовать для альтернативных наложений (overlays)
8. Слой свободен. Можно использовать для альтернативных наложений (overlays)
9. Слой свободен. Можно использовать для альтернативных наложений (overlays)
10. DO NOT USE fullFramePOSlayer – (\*) слой на который скрипт перемещает целевой слой источника N если происхоит переход на полный растр из ME

Примечание. Т.к. по умолчанию vMix при использовании входа как источника STINGER перехода безусловно (т.е. при любых установненных настройках Auto Play/Restart/Play after transition) перезапускает и останавливает после окончания все источники, которые можно перезапустить, рекомендуется при необходимости «живого» фона добавить его через дополнительный MIX

ВАЖНО! ЭТОТ INPUT не должен иметь прозрачности!

#### Настройка STINGER

К сожалению, легального метода изменения этих настроек по API нет, поэтому приходится руками....

Заходим в настройки overlay

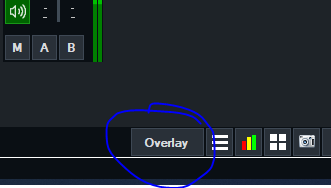


Рис. 2.4.2

Выбираем свободный STRINGER

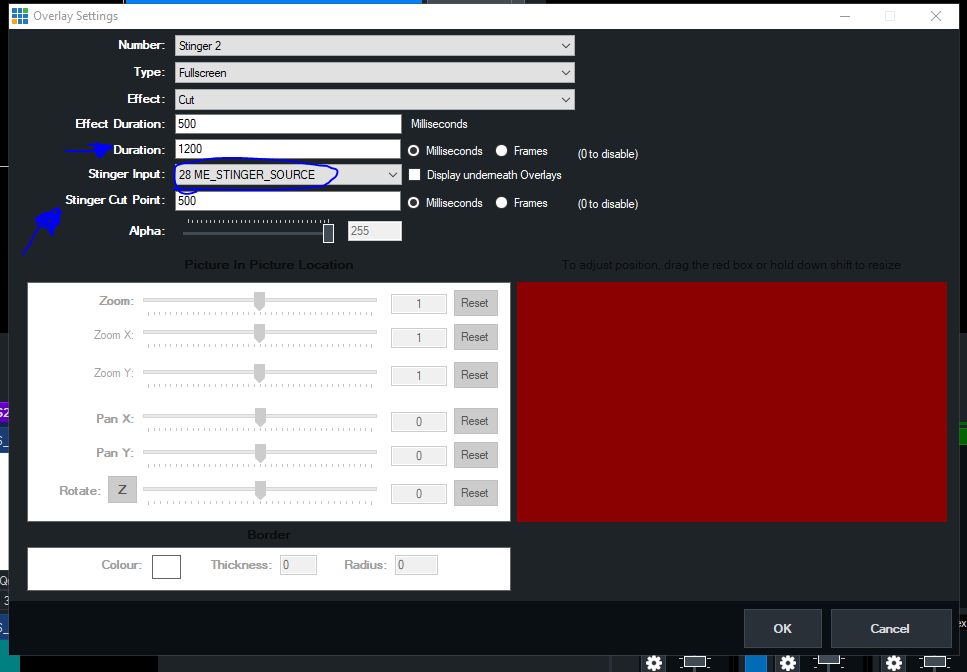


Рис 2.4.3

Указываем Stinger input – наш ME STINGER SOURCE

Duration должна быть больше, чем планируемая длительность MERGE

Cut Point – половина от длительности MERGE

## 3 Создание дополнительных INPUTов, необходимых для работы скрипта

### 3.1 Правила присвоения имени для дополнительных INPUTов

Т.к. для многих ситуаций, представленных в таблице 2.4, конфигурация INPUTов может быть идентична, поэтому есть **строгие правила** присвоения имен для дополнительных **POS INPUT/MASK\_ON/MASK\_OFF** INPUTов

Шаблон названия:

PREFIX\_AxBxC….xZ\_AAvBBvCC…vZZ\_ONorOFF\_M

PREFIX – префикс, общий для всех дополнительных INPUTов, позволит понять скрипту, что это именно интересующий его INPUT

AxAxBxC….xZ – A, B, C, …. Z – A – номер SOURCE для ME, через «x» указываются случаи **A→B, C, … Z** (см. Таблицу 2.4)

AAvBBvCC…vZZ – AA, BB, CC – номера ME для которых применим данный INPUT. Перечисление происходит через «v» (аналог «или»)

ONorOFF – флаг, обозначающий «октрывающая» (ON) или «закрывающая» OFF это маска. **Только для масок**

M – флаг, обозначающий, что этот INPUT – маска. **Только для масок**

### 3.2 Создание POS INPUT

POS INPUT представляет из себя **COLOUR** c **прозрачным** фоном, в котором необходимо сформировать слои в Mulitview следующим образом

1. POSmainLayer – (\*) – Основной слой, его позиционирование полностью соответсвует положению в ME
2. POSchangingLayer – (\*) – слой подмены, должен находиться за пределами видимости. Из этого положения происходит простая замена источника для SOURCE
3. POSmainNewLayer– (\*) – дубликат основного слоя, обязательный слой, в него происходит merge при простой замене
4. POSX1Layer – (\*) – слой виртуального положения SOURCE 1
5. POSX2Layer – (\*) – слой виртуального положения SOURCE 2
6. POSX3Layer – (\*) – слой виртуального положения SOURCE 3
7. POSX4Layer– (\*) – слой виртуального положения SOURCE 4

Для ситуации A→A см. Рис. 3.2.1

Для ситуации A→B см. Рис. 3.2.1

Рис 3.2.1 A→A

ЗОНА ВИДИМОСТИ

TO B

Положения SOURCE B

POSBLayer

changingLayer

POSX2Layer

POSX3Layer …..

MAIN

newMAIN

Рис 3.2.2 A→B

ЗОНА ВИДИМОСТИ

POSBLayer

changingLayer

POSX2Layer

POSX3Layer ….

MAIN

newMAIN

***Примечание: По сути необходимо настроить лишь POSITION/CROP для каждого слоя. Содержимое слоя задается скриптом***

### 3.3 Создание MASK ON/OFF

MASK ON/OFF представляет из себя **COLOUR** c **прозрачным** фоном. В качестве слоев могут выступать любые INPUTы (COLOUR с целиковым заполнением). А также могут быть растровые заранее подготовленные изображения.

vMIX счтиает прозрачность по красному каналу, поэтому прозрачными считаются все цвета у которых R = 0 G – любой B – любой

К примеру

|  |  |
| --- | --- |
| Непрозрачный R=255 | Прозрачный R = 0 |
| R=255 G = 0 B = 0 | R=0 G = 255 B = 0 |
| R=255 G = 255 B = 0 | R=0 G = 0 B = 255 |
| R=255 G = 0 B = 255 | R=0 G = 255 B = 255 |
| R=255 G = 100 B = 0 | R=0 G = 100 B = 0 |
| R=255 G = 0 B = 100 | R=0 G = 0 B = 100 |
| R=255 G = 50 B = 50 | R=0 G = 50 B = 50 |

Таблица 3.3.1

***Примечание: Для наглядности удобно использовать маски разных цветов, чтобы контролировать перемещения масок.***

Рис. 3.3.1 Пример MASK ON маски

WHITE LEFT

SOURCE 1

BLACK LEFT

BLACK BOTTOM

BLACK TOP

ЗОНА ВИДИМОСТИ

Рис. 3.3.2 Пример MASK OFF маски

WHITE LEFT

SOURCE 1

BLACK LEFT

BLACK BOTTOM

BLACK TOP

ЗОНА ВИДИМОСТИ

***Примечание: Желательно основной «закрывающий» слой поместить самым последним (верхним). В примере на рис. 3.3.1 это маска BLACK BOTTOM. Слоев может быть больше, в зависимотси от сложности переходов. WHITE LEFT маска применяется при создании масок переходов A=>B***

### 3.4 FullFrame MASK ON

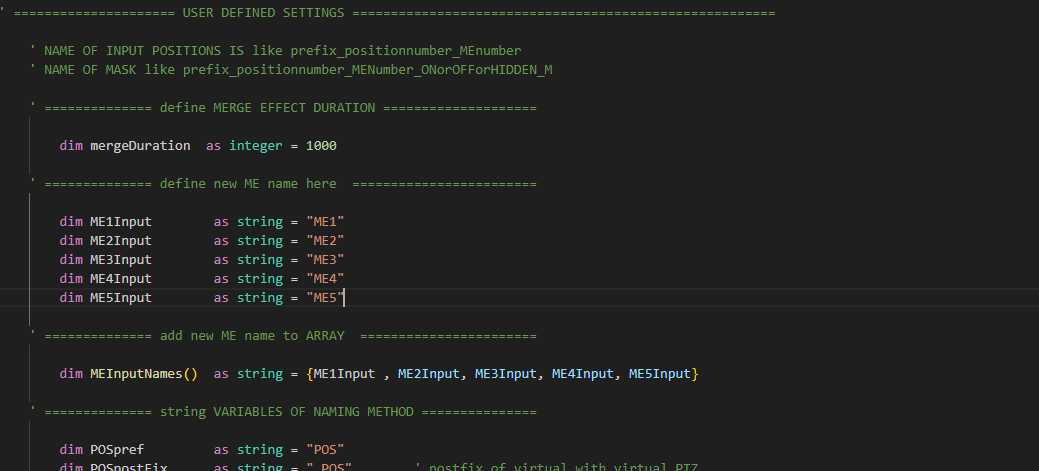
Создайте полностью «открытую» маску и дайте ей имя **FullFrame\_M** (\*) или любое другое, потом внеся изменения в скрипт. Это маска необходима для перехода в полный растр из ME или наоборот.

ПРИМЕЧАНИЕ: ЭТО ОБЯЗАТЕЛЬНЫЙ ПУНКТ

## 4 Настройка скрипта

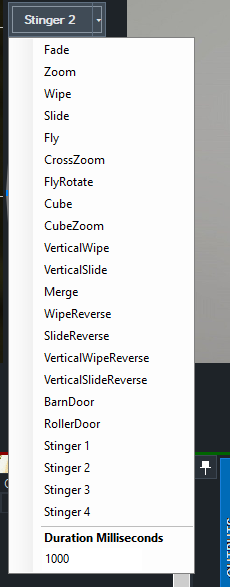
Все настройки вносятся в любом удобном для вас текстовом редакторе (в VS Code это делать очень удобно)

Настройки скрипта заключены в области USER DEFINED SETTINGS



### 4.1 Длительность MERGE и номер STINGER

Вносим номер STINGER, если нужно

        dim stingerNumber   as integer = 2

Длительность определяется значением Duration Milliseconds в поле настройки быстрых переходов в GUI vmix.

Если STINGER, выбранный вами в качестве целевого для скрипта среди переходов быстрого доступа. Его можно выбрать среди представленных вариантов. По умолчанию длительность будет равна 1000 мс.

Если длительность Duration (см. Рис. 2.4.3) окажется меньше, чем длительность по умолчанию или указанная вами в Duratiom Milliseconds, будет выбрана та, что указана в Overlay Settings (см. Глава 2.4)

### 4.2 Добавление ME

Для добавления нового ME INPUTа в скрипт необходимо создать новую переменную MEXInput

        dim ME1Input        as string = " YOUR\_ME\_NAME\_1"

        dim ME2Input        as string = " YOUR\_ME\_NAME\_2"

        dim MEXInput        as string = "YOUR\_ME\_NAME\_X"

Где X – следующий после последнего среди имеющихся (именно этот номер используется в наименованиях POS INPUT/MASK ON/OFF)

После этого необходимо добавить эту переменную в после запятой в скобки (в массив) MEInputNames()

        dim MEInputNames()  as string = {ME1Input , ME2Input, MEXInput}

### 4.3 Постфиксы и префиксы имен INPUTов

        dim POSpref         as string = "POS"

        dim POSpostFix      as string = "\_POS"

        dim fakePTZpostfix  as string = "\_FAKE\_PTZ"

        dim MEStingerInput  as string  = "ME\_STINGER\_SOURCE"

### 4.4 Настройка номеров MIXов и их имен

Задаем, либо вносим изменения, в названия MIXов.

! ОБЯЗАТЕЛЬНО после добавления переменных внести их в соответствующие массивы!

***!!!! Добавление или уменьшение количества SOURCEs влечет изменения во всех последующих пунктах !!!!!!***

'=============== POS MIXES SETTINGS==========================

dim POS1\_Input as string = "POS1\_mix"

dim POS2\_Input as string = "POS2\_mix"

dim POS3\_Input as string = "POS3\_mix"

dim POS4\_Input as string = "POS4\_mix"

dim POSMixesNames() as string = { POS1\_Input , POS2\_Input , POS3\_Input, POS4\_Input}

' =========================================================

dim POS1\_MASK\_Input as string = "POS1\_MASK"

dim POS2\_MASK\_Input as string = "POS2\_MASK"

dim POS3\_MASK\_Input as string = "POS3\_MASK"

dim POS4\_MASK\_Input as string = "POS4\_MASK"

dim POSMasksNames() as string = { POS1\_MASK\_Input , POS2\_MASK\_Input , POS3\_MASK\_Input, POS4\_MASK\_Input}

### 4.5 Номера слоев в ME

В п.п. 2.1 вы настраивали слои в ME INPUTах если выбрали какие-то иные слои внесите изменения в следующие переменные и внесите в массив

        dim position1Layer  as integer = 6

        dim position2Layer  as integer = 5

        dim position3Layer  as integer = 4

        dim position4Layer  as integer = 3

        dim positionLayers() as integer = { position1Layer , position2Layer , position3Layer , position4Layer }

### 4.6 Номера слоев POS INPUT

В п.п. 3.2 вы настраивали слои в POS INPUTах если выбрали какие-то иные слои внесите изменения в следующие переменные и внесите в массив

            dim POSmainLayer        as integer = 1

            dim POSchangingLayer    as integer = 2  '

            dim POSmainNewLayer     as integer = 3

        ' =======================================================

            dim POSX1Layer          as integer = 5  '   Layer to change for Px=P1

            dim POSX2Layer          as integer = 6  '   Layer to change for Px=P2

            dim POSX3Layer          as integer = 7  '   Layer to change for Px=P3

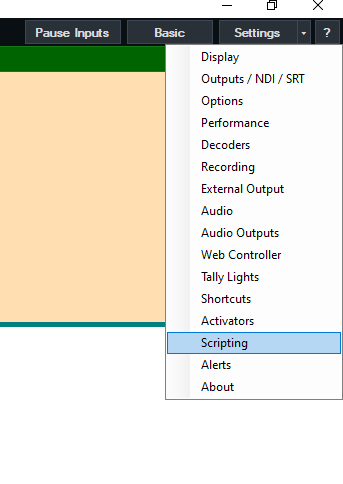
            dim POSX4Layer          as integer = 8  '   Layer to change for Px=P4

        ' =======================================================

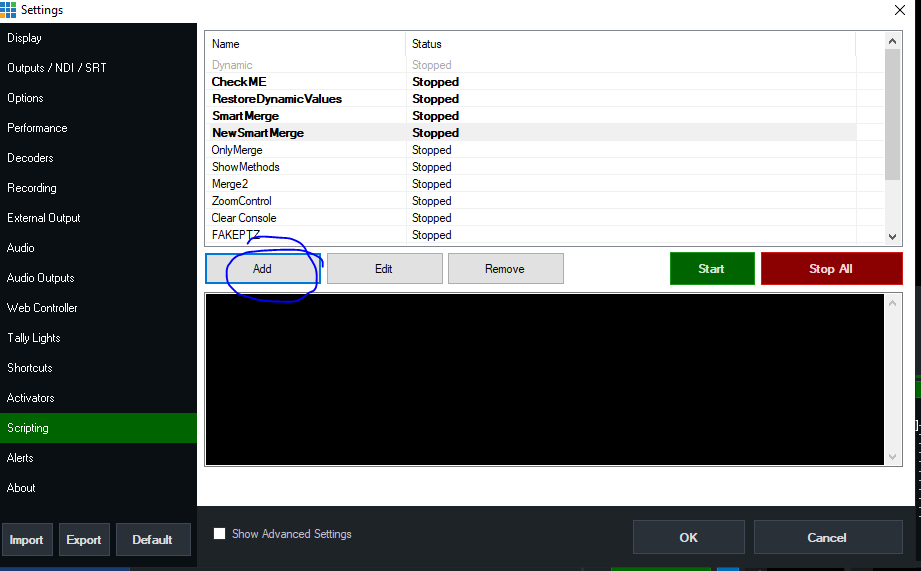
            dim POSXLaysers() as integer = { POSX1Layer, POSX2Layer, POSX3Layer, POSX4Layer}

## 5 Запуск скрипта

Сохраните внесенные изменения как новый файл. Скопируйте в буфер обмена весь текст. Откройте settings=>scripting



Добавьте новый или обновите существующий скрипт

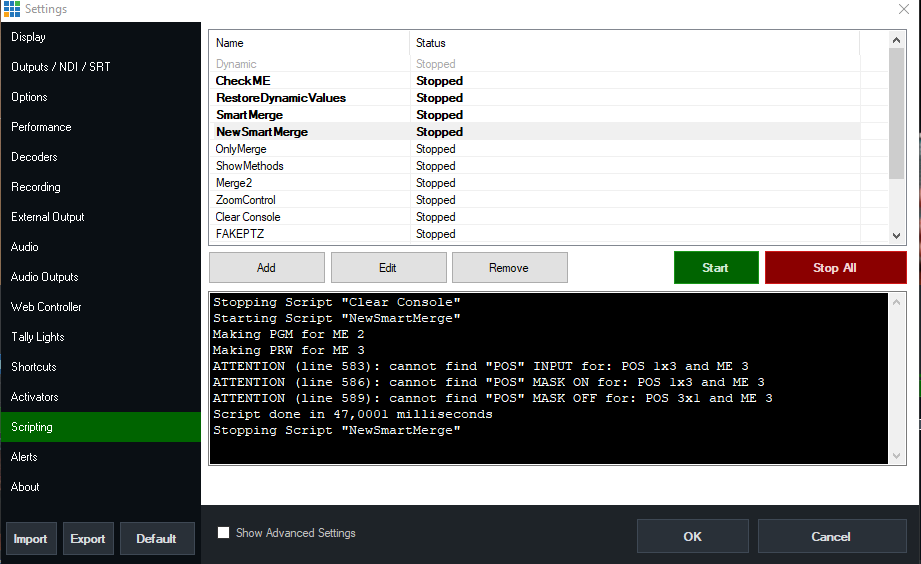


Назначьте горячую клавишу на выполнение скрипта. Если все ностроено все будет работать

## 6 Отладка и сообщения в консоли

В случае ошибок будет работать простой Fade

Также если скрипту не хватает каких-то дополнительных INPUTов – он об этом скажет. Для чтения сообщений в консоли перейдите во вкладку settings=>scripting



К примеру в данной ситуации скрипт сказал, что для полноценного выполнения сценария трансформации ему не хватает POS INPUT 1x3 а также MASK ON 1x3 и MASK OFF 3x1 для ME3

## X Приложение

### Сводная таблица слоев

Таблица X.1 Сводная таблица слоев для INPUTов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер слоя** | **REGULAR INPUT** | **ME STINGER** | **ME INPUT** | **POS\_X\_Y** |
| 1 | **fullFrameLayer** | **-** | **DO NOT USE**  **(fullFrameLayer)** | **main** |
| 2 | **-** | **-** | **-** | **xChange** |
| 3 | **DO NOT USE**  **(POS LAYERS)** | **POSITION 4 MIX** | **POSITION 4 SOURCE** | **new main** |
| 4 | **DO NOT USE**  **(POS LAYERS)** | **POSITION 3 MIX** | **POSITION 3 SOURCE** |  |
| 5 | **DO NOT USE**  **(POS LAYERS)** | **POSITION 2 MIX** | **POSITION 2 SOURCE** | **POSITION 1 XCHANGE** |
| 6 | **DO NOT USE**  **(POS LAYERS)** | **POSITION 1 MIX** | **POSITION 2 SOURCE** | **POSITION 2 XCHANGE** |
| 7 | **-** | **-** | **-** | **POSITION 3 XCHANGE** |
| 8 | **-** | **-** | **-** | **POSITION 4 XCHANGE** |
| 9 | **-** | **-** | **-** |  |
| 10 | **-** | **DO NOT USE**  **(fullFramePOSlayer)** | **-** |  |

### **Исходный код**

находится ../src/smartMerge\_script.vb

### Демо проект

Залит в бандл

находится ../proj/smartmegre\_demo.vmix

### Обратная связь

Все пожелания писать на semenov\_nuck@mail.ru