

UnlightVBE-QS 脚本開發說明書

2023/7/21

Ver. 1.70

目錄

1. 前言	1
2. 腳本運作流程介紹	2
3. 執行階段	3
I. 運作簡介與時間軸圖	3
II. 執行階段型態分類	6
III. 執行階段使用面向	6
IV. 執行階段詳細說明	7
普通型	7
事件型	9
特殊型	11
V. 啟動管道與發動順序	12
VI. 執行週期與待命執行	13
4. 執行指令集	14
I. 基本指令	14
i. 一般類型	15
啟動類	15
系統類	16
移動距離類	17
骰數控制類	18
人物類	19
卡牌類	22
ii. 附加狀態類型	25

人物實際狀態類	25
異常狀態類	26
II. 使用格式轉換器	28
i. 簡介	28
ii. 指令函數	29
指令函數宣告一覽表	30
5. 腳本引數變數	33
6. 腳本體系類型	41
7. 腳本及卡片人物資訊文件格式	43
I. 卡片人物資訊 (UnlightVBE Card Information)	43
II. 技能腳本文件	49
8. 主動技能之 AI 計算判斷設計	58
9. 腳本之錯誤處理	63
10. 附錄 1-人物角色之圖片資源概要	66
11. 附錄 2-人物角色之小人物立繪影子座標取得操作	68

1. 前言

UnlightVBE 經歷了 2 個正式版本的發布 (Version α、Version ζ)，雖然其遊戲系統完整性日益成熟，不過在人物角色的資訊的自由編輯及可攜性方面，仍然是相當的有限。在追求人物角色技能程式碼外部化，達到角色資訊完全獨立的期望之下，本版本 (Type QS) 也就此誕生。

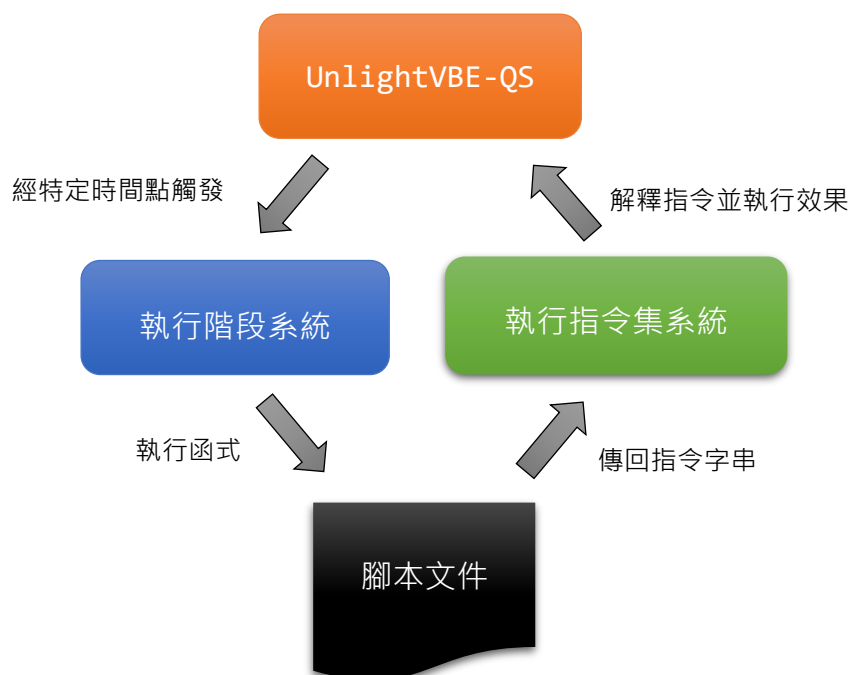
在當初開始作 UnlightVBE 時，其實最先是抱持著一種興致使然，一切都是原始於當時的一個小巧簡單構造的程式，而這股純粹的興致讓我不知不覺一邊在學習 VB 之下，另一邊在慢慢修改著程式，不知什麼時候它竟變得如此巨大，到最後還真的搞出一個可以玩的遊戲程式出來，就連現在的我回想當時之種種回憶也是很難以想像。

本說明書旨在說明本遊戲系統對於腳本之運作，以及說明腳本之架構與指令撰寫，本人也希望透過本遊戲平台及開發說明書，能夠讓更多二次元之人物角色有登上戰鬥舞台的機會。

UnlightVBE (QS) 製作者 AndyCiu.

2. 腳本運作流程介紹

本腳本文件採用 VBScript 語言撰寫，搭配 VB6 的 Microsoft Script Control 1.0 套件來讀取及執行腳本相關程式碼，透過在遊戲中特定之時間點觸發腳本文件，由其所回傳之指令字串來詮釋技能之效果。



UnlightVBE-QS 主要是利用 2 個子系統來實作：執行階段系統&執行指令集系統，在遊戲戰鬥過程中，依照時間順序及事件劃分成若干個「執行階段」，當階段觸發時，執行階段系統負責統計戰鬥資訊及先後順序，隨後執行該角色之腳本文件，並接收不等的「執行指令」字串，再透過執行指令集系統來依照個別指令去執行特定的效果。每個角色，乃至於每個必殺技/狀態都為一獨立之文件個體，可實現理論上若干多個角色之實裝而不受其遊戲主程式之影響與綁定限制。

3. 執行階段

I. 運作簡介與時間軸圖

本遊戲對於腳本之執行採用時間軸型態的編輯模式，對每個關鍵時間點設置執行階段，來達到「在什麼時間做什麼事」之目的。

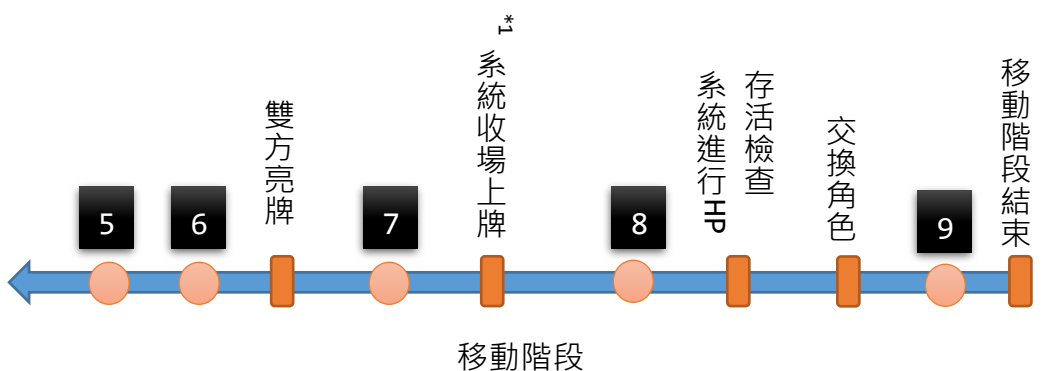
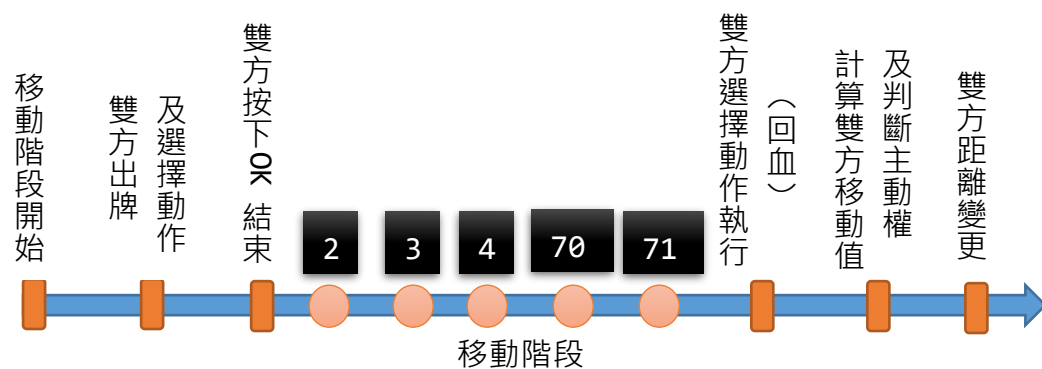
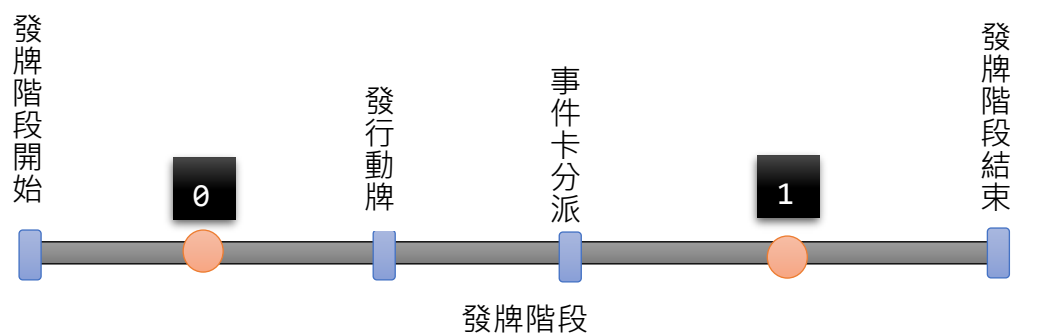
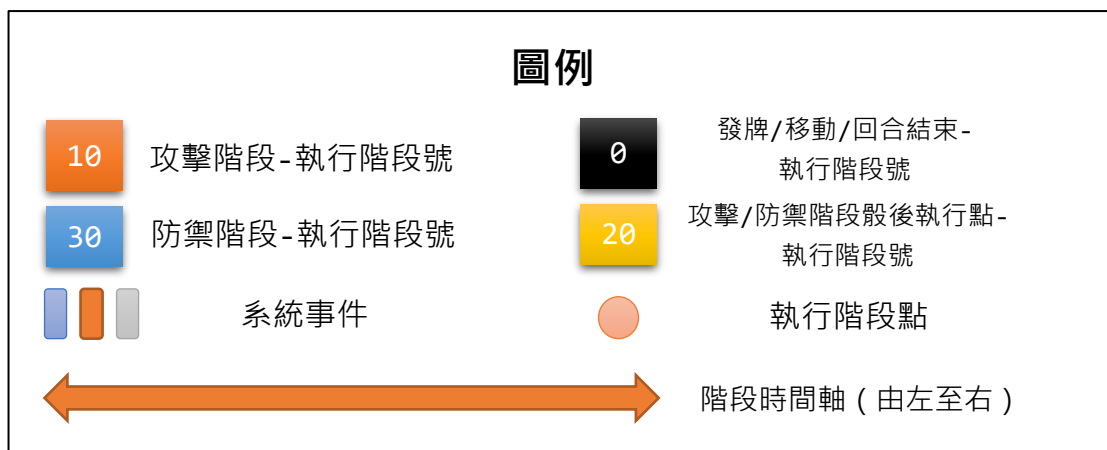
在戰鬥的過程中，有既定之戰鬥流程與規則，對本遊戲來說，能夠在適當的時間中打出最完美的牌，讓角色使用各種的必殺技，發揮出最大的戰鬥能力，是玩家所追求之首要。

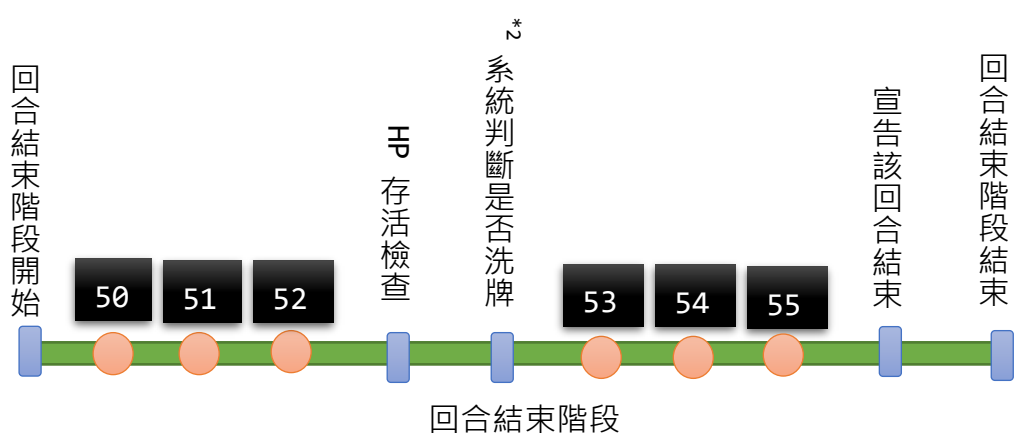
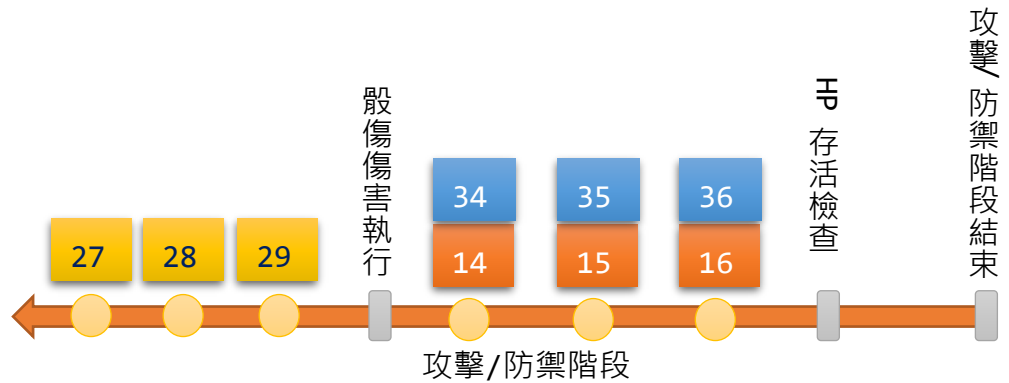
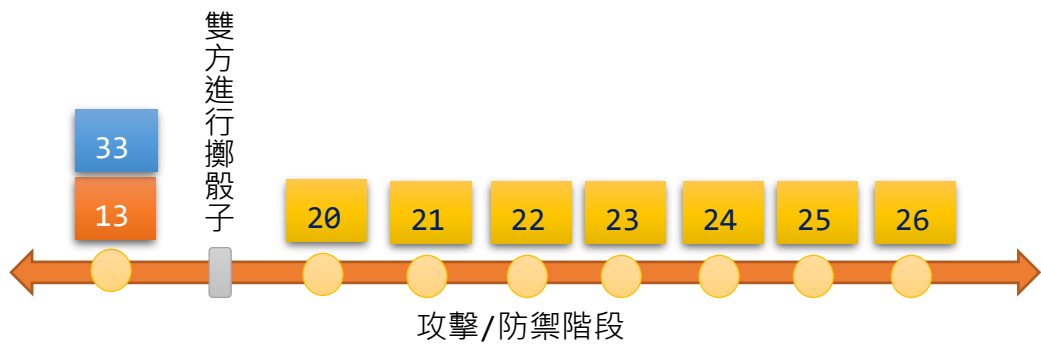
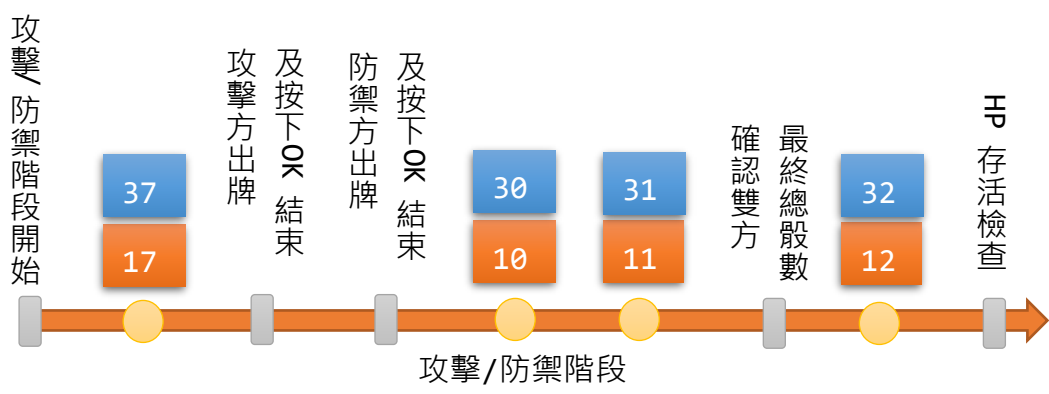
作為一位人物角色之設計者，在設計技能能力及數值之時，對於卡牌相對應的條件，對其他人物角色之對應關係尤其重要，這也牽涉到戰鬥過程與時間先後順序之流程應對。在甚麼樣的情況下某些特定之技能能夠發揮最大的效益，又在什麼情況下這些技能有可能遭受到箝制，甚至是無法顯現出原本應有的功能，這些事項想必設計者都會去再三思考吧。

為充分展現戰鬥流程之各種事件與應對關係，本系統採用時間軸制之事件列舉，並把設計者可用之時間點作編號，每一個時間點以一個「執行階段」作為稱呼。為方便設計者掌握各個執行階段點之時間先後順序，以下就以時間軸之概念敘述系統事件、執行階段點之時間關係：

(註¹：系統收場上牌時會將卡牌類型分類，公用牌類型>墓地牌堆，事件卡類型>棄牌堆)

(註²：若下一回合待發牌數量(公用牌)多於目前牌堆卡牌數量，則進行洗牌；洗牌時會將所有墓地牌重新洗回至既有之牌堆下方。)





II. 執行階段型態分類

在執行階段系統中，執行階段分為以下 3 種類型：

A. 普通型 - 在時間軸圖上之執行階段點：

發牌階段：0,1

移動階段：2~9,70,71

攻擊階段：10~17

防禦階段：30~37

攻擊/防禦階段最後執行點：20~29

回合結束階段：50~55

B. 事件型 - 在遊戲中某個時間點由事件所觸發。

C. 特殊型 - 在遊戲中屬於特殊類型之執行階段點。

若是您一開始不知該如何下手的話也沒關係，一般以主動技能來說，首先需要從出牌條件是否成立開始寫起，這時會利用到特殊型之相關出牌事件來做啟動之驗證，再來就是發動技能的執行效果的部分了，這部分就是依照技能來自由運用其事件，像是您可以利用時間軸上各個普通型執行階段事件來發動技能動畫，動畫執行中您可以利用 (61) 來執行技能效果，或者您也可以運用事件型執行階段事件來對戰鬥中某些特定的時間點做出相符合的動作等等。

III. 執行階段使用面向

在普通型執行階段中，通常都會依照玩家雙方以及各個不同的技能體系來逐一進行呼叫，讓有利用到此執行階段之各項技能腳本都能進行參與，不過像事件型與特殊型的執行階段則會因為其事件之需要而限制其使用面向範圍，像是該事件只能由我方參與，敵方不能參與之類的一些限制。以下表格為針對事件型、特殊型執行階段事件的使用面向範圍名詞解釋說明：

使用面向範圍

雙方	該事件玩家雙方都可參與
單方	該事件只有發動此事件之一方才可參與
全域	該事件可參與方的全部技能體系都可參與
個體	該事件可參與方只有發動此事件之特定技能體系才可參與

舉例來說，「雙方 / 全域」代表玩家雙方所有技能體系都能夠利用此執行階段事件，而「單方 / 個體」則只有發動此執行階段事件之特定一方的特定技能體系才能夠使用此事件。

IV. 執行階段詳細說明

普通型

0/1	主要對被動技-發牌階段效果實施	發牌
2/3/4	移動動作執行前之主動技效果發動實施	移動
70/71	移動動作執行前之異常狀態效果實施	移動
5/6/7	移動階段主要之效果實施	移動
8	針對於 HP 存活檢查前之效果實施	移動
9	針對於交換角色後之效果實施	移動
10	針對具優先權之技能效果實施 (EX: 主動技-因果之輪)	攻擊
11	攻擊階段主要之效果實施	攻擊
12	針對於雙方總骰數決定後之效果實施	攻擊
13	主要針對需要進行多次擲骰之技能效果實施	攻擊
14/15/16	在骰傷傷害執行後之技能效果實施 (EX: 主動技-因果之刻)	攻擊
17	主要對被動技-攻擊階段效果實施	攻擊
30	針對具優先權之技能效果實施	防禦

31	防禦階段主要之效果實施	防禦
32	針對於雙方總骰數決定後之效果實施	防禦
33	主要針對需要進行多次擲骰之技能效果實施	防禦
34/35/36	在骰傷傷害執行後之技能效果實施	防禦
37	主要對被動技-防禦階段效果實施	防禦
20	擲骰後效果實施-一般類型	骰後執行點
21	擲骰後效果實施-追加/減輕骰數類型	骰後執行點
22	擲骰後效果實施-其他骰數有關類型	骰後執行點
23	擲骰後效果實施-攻擊/防禦成功類型	骰後執行點
24	擲骰後效果實施-骰數乘數加成類型 (EX:異常狀態-恐怖・異常狀態-狂戰士)	骰後執行點
25	擲骰後效果實施-骰數有關類型-防禦 (EX:主動技-茨林)	骰後執行點
26	擲骰後效果實施-卡牌抽牌/回牌類型 (EX:主動技-反擊的狼煙)	骰後執行點
27	擲骰後效果實施-骰數特殊類型 (EX:異常狀態-不死)	骰後執行點
28	擲骰後效果實施-卡牌丟牌類型 (EX:主動技-雷擊)	骰後執行點
29	擲骰後效果實施-骰傷傷害類型 (EX:主動技-兩個身體・主動技-不屈之心)	骰後執行點
50/51/52	針對於 HP 存活檢查前之效果實施	回合結束
53/54/55	回合結束階段之主要效果實施 / 結束執行	回合結束

事件型

41	角色交換時	雙方 / 全域
人物角色交換時觸發事件。		
46	人物角色受到傷害時	雙方 / 全域
人物角色受到傷害時觸發事件，包含毆傷 / 直傷 / 直接死亡。		
47	人物角色移動時	雙方 / 全域
人物角色移動時觸發事件。		
48	人物角色回復血量時	雙方 / 全域
人物角色回復血量時觸發事件。		
49	人物角色復活時	雙方 / 全域
人物角色復活時觸發事件。		
61	動畫發生時途中之效果實行	單方 / 個體
技能動畫發動時觸發之被動事件。		
62	技能效果進行多次擲骰時	單方 / 個體
技能效果進行多次擲骰時觸發之被動事件，每次擲骰後須在本執行階段手動進行結果紀錄。		
72	異常狀態新增時	單方 / 個體
異常狀態新增時觸發之被動事件， 異常狀態專用 。		
73	異常狀態消滅時	單方 / 個體
異常狀態消滅時觸發之被動事件， 異常狀態專用 。		
74	人物實際狀態新增時	單方 / 個體
人物實際狀態新增時觸發之被動事件， 人物實際狀態專用 。		
75	人物實際狀態消滅時	單方 / 個體
人物實際狀態消滅時觸發之被動事件， 人物實際狀態專用 。		
76	人物角色附加狀態增加時	雙方 / 全域
人物角色附加狀態增加時觸發事件，於執行階段 72/74 後執行。		
77	人物角色附加狀態解除時	雙方 / 全域
人物角色附加狀態解除時觸發事件，於附加狀態消滅後執行。		
101	對手卡牌奪取效果執行時	雙方 / 全域
對手卡牌奪取效果執行時觸發事件。		
102	抽牌效果執行抽取前	雙方 / 全域
抽牌效果執行抽取前觸發事件。		

103	抽牌效果執行每次抽牌時	單方 / 全域
抽牌效果執行每次抽牌時觸發事件（公用牌類限定）。		
104	丟牌效果執行時	雙方 / 全域
丟牌效果執行時觸發事件。		
105	自身卡牌送與效果執行時	雙方 / 全域
自身卡牌送與效果執行時觸發事件。		
106	墓地牌/棄牌回牌效果執行時	雙方 / 全域
墓地牌/棄牌回牌效果執行時觸發事件。		
107	對某一方擁有之卡牌進行控制時	雙方 / 全域
對某一方擁有之卡牌進行控制時觸發事件。		

特殊型

42 攻擊階段-出牌時

單方 / 全域

玩家出牌時觸發事件（攻擊階段），主要為主動技能進行發動之驗證。

註：未啟動之技能可使用此事件，但待命角色之主動技能無論是否已啟動皆除外。

43 防禦階段-出牌時

單方 / 全域

玩家出牌時觸發事件（防禦階段），主要為主動技能進行發動之驗證。

註：未啟動之技能可使用此事件，但待命角色之主動技能無論是否已啟動皆除外。

44 移動階段-出牌時

雙方 / 全域

玩家出牌時觸發事件（移動階段），主要為主動技能進行發動之驗證。

註：未啟動之技能可使用此事件，但待命角色之主動技能無論是否已啟動皆除外。

45 雙方總骰數更新時

雙方 / 全域

雙方玩家總骰數更新時觸發事件，主要為一般效果骰數加成時之統一管道。

92 攻擊階段-階段開始時

單方 / 全域

雙方玩家出牌前系統事件（攻擊階段），主要為技能發動之驗證（例如：無卡牌要件者）。

註¹：未啟動之技能可使用此事件。

註²：本事件於普通型執行階段 17 後執行。

93 防禦階段-階段開始時

單方 / 全域

雙方玩家出牌前系統事件（防禦階段），主要為技能發動之驗證（例如：無卡牌要件者）。

註¹：未啟動之技能可使用此事件。

註²：本事件於普通型執行階段 37 後執行。

94 移動階段-階段開始時

雙方 / 全域

雙方玩家出牌前系統事件（移動階段），主要為技能發動之驗證（例如：無卡牌要件者）。

註¹：未啟動之技能可使用此事件。

註²：本事件於普通型執行階段 1 後執行。

99 智能判斷型 AI 計算卡牌之排列組合時

單方 / 個體

智能判斷型 AI 計算卡牌之排列組合時觸發事件，針對 AI 於徵求主動技能之意見時所使用。

V. 啟動管道與發動順序

不論是哪一型態的執行階段點，都需要先對系統進行「啟動」確認之後，方可進行使用。「啟動」分為 2 個部分，第 1 個部分是該技能腳本於執行階段點的登記，可參照第 7 章（腳本及卡片人物資訊文件格式-技能腳本文件-初始架構標頭資訊說明）。第 2 個部分則是在遊戲過程中，該技能是否處在發動中狀態。以下表格說明各執行階段型態中系統如何驗證其為發動中狀態：

第 2 部分啟動管道確認

執行階段型態	說明/備註
普通型	主動/被動技能體系： 已發送 AtkingTurnOnOff 指令為開啟者 (唯有在啟動後才能在其後的各個執行階段點進行使用)
	異常 / 人物實際技能體系： 已附著在人物角色者
事件型	僅具有觸發對應之 發動中 技能體系才具使用資格
特殊型	詳見 (執行階段詳細說明-特殊型) 之各執行階段說明

各技能體系之發送啟動指令時機

體系	說明
主動技能	於特殊型執行階段 42/43/44/92/93/94 發送
被動技能	除主動技能之管道外，可於任何 普通型 執行階段發送
異常狀態	不須主動發送 (狀態存續即在發動中狀態)
人物實際狀態	

在系統判斷是否啟動確認的部分，於執行階段系統會先做一次檢查，而後執行指令集系統會針對主/被動技能，在每次各執行指令解釋前都會再行驗證是否處在發動中狀態，若無者則不予解釋執行除 (**AtkingLineLight/AtkingTurnOnOff**) 之外的其他指令。

發動順序部分，普通型執行階段在啟動之後，因雙方都具有權力來發動各執行階段點，系統在不同時間下會照不同的規則來作為執行，以下表格為各階段之發動順序：

普通型各執行階段點之發動順序

執行階段號	發動順序
0~4/70/71	使用者方先發動
5~9	有主動權之一方則 <u>先</u> 發動
10~17	攻擊方先發動
30~37	防禦方後發動
20~29	攻擊方先發動
50~55	有主動權之一方則 <u>後</u> 發動

至於事件型與特殊型則一律以發動事件的那一方為主，而後再依照其使用面向範圍再決定是否允許敵方參與發動。

VI. 執行週期與待命執行

通常技能以一個回合當作開始與結束之執行週期，但由於原則上技能之開關與否都交由技能體系自行控制，故理論上能實現所謂跨執行週期（跨回合數）之寫法。

但要注意的一點是，在效果執行前務必要釐清自身角色於場上之狀態定位，亦及該角色是否為場上角色或為待命中角色。在被动技能 / 異常狀態 / 人物實際狀態中由於其特性之緣故，故都採取先行判斷角色場上狀態之邏輯寫法，而主動技能在一般情況下因不允許在待命角色中執行之關係，故邏輯判斷上可單純化無須介意，但若設計者開啟了主動技能可在待命角色中執行之功能，則應當在每個使用之執行階段注意其自身角色之場上狀態。

4. 執行指令集

I. 基本指令

本系統是以腳本回傳之「指令字串」作為效果之執行依據。指令字串是由數個指令串連而成，透過系統指定之指令格式，來達到效果詮釋之目的。

在指令字串撰寫中，指令名稱及指令引數上由「#」號分隔，引數與引數中間由「，」號分隔；如需傳回 2 個以上之指令者，指令與指令中間則以「=」號做分隔。

例如：於某階段中使用「AtkingLineLight」作狀態燈控制開啟，加上「AtkingTurnOnOff」作啟動碼控制開啟時，最後所構成之指令字串如下所示：

AtkingLineLight#1=AtkingTurnOnOff#1=

因應每個執行指令其適用範圍之不同，之後於每項指令右方都有其特定標記，說明如下：

(若無技能體系特定標記者，則屬全體適用)

特定標記圖例

主動技能	指定之技能體系適用
異常狀態	指定之技能體系 排除 適用
普通型	指定之執行階段型態或特定範圍適用
13/33	指定之執行階段型態或特定範圍 排除 適用
61	該指令所觸發對應之執行階段事件 (標記*者需參考備註)

以下為各執行指令之詳細說明：

i. 一般類型

啟動類				
AtkingLineLight	#n1	主動技能	被動技能	45/99
人物必殺技狀態燈控制(自身) (註：本指令僅限 <u>場上角色</u> 使用。)				
n1-(1.開啟/2.關閉)				
AtkingTurnOnOff	#n1	主動技能	被動技能	45/99
人物必殺技啟動碼控制(自身)				
n1-(1.開啟/2.關閉)				
AtkingLineLightAnother	#n1,n2,n3	主動技能	被動技能	普通型 事件型
人物必殺技狀態燈控制(其他) (註：本指令僅限 <u>場上角色</u> 、技能 <u>發動中</u> 之狀態才能使用。)				
n1-(1.主動技能/2.被動技能)				
n2-(第 x 個技能)				
n3-(1.開啟/2.關閉)				
AtkingTurnOnOffAnother	#n1,n2,n3	主動技能	被動技能	普通型 事件型
人物必殺技啟動碼控制(其他) (註：本指令僅限技能 <u>發動中</u> 之狀態才能使用。)				
n1-(1.主動技能/2.被動技能)				
n2-(第 x 個技能)				
n3-(1.開啟/2.關閉)				

系統類

AtkingStartPlay #n1,[n2] 61 普通型 異常狀態 13/33

技能動畫圖像執行 (靜態)

(註：若無第 2 張圖片者可不填寫或填「0」)

n1-(第 1 張圖片路徑)

[n2]-(第 2 張圖片路徑)

AtkingStartPlayAnimate #n1,[n2,n3] 61 普通型 異常狀態 13/33

技能動畫圖像執行 (逐格動畫)

(註¹：可擺放 16~24 張圖片，檔案以 1.png/2.png/...編號逐一命名)

(註²：圖片資料夾路徑字串結尾須加反斜線(\))

(註³：可選擇關鍵影格 1~2 張作為靜態技能動畫圖像執行使用，適用於在設定中關閉逐格動畫支援者；若無填寫時則系統預設取第 4 張與第 12 張影格作為使用)

n1-(圖片資料夾路徑)

[n2]-(關鍵影格第 1 張圖片編號)

[n3]-(關鍵影格第 2 張圖片編號)

BattleTurnControl #n1,n2 普通型 事件型

系統回合數控制

n1-(1.加/2.減)

n2-(數值)

BattleSendMessage #n1 99

發布訊息效果執行

n1- (訊息內容)

BattleStartDice #0 62 13/33/20/21/22/23/24/25/26/27/28/29

擲骰效果執行

(註：如需進行多次擲骰，則需重複執行多次本指令。另外在每次擲骰之後，會觸發執行階段-62 事件，技能需在本執行階段進行擲骰之結果紀錄，系統不予保留。)

AtkingInformationRecord #n1 主動技能 被動技能 99

技能自行註記之備註資訊記載

(註：本備註資訊會更新在引數 AtkingckVSF 中。)

n1-(技能自身註記之備註字串)

EventPlayerAllActionOff #n1

1/17/37

禁止玩家進行所有操作控制

(註¹: 本指令於玩家各行動階段之第 1 個執行階段進行效果, 例如欲在該玩家攻擊階段實施者, 則在執行階段 17 下達本指令, 依此類推。)

(註²: 本指令執行效果僅以當下行動階段為限。)

n1-(1.自身/2.對方)

EventActiveAIScore

#n1,n2,[n3...]

主動技能

99

智能判斷型 AI 個別技能判斷評分

(註¹: 自引數 n3 開始為選擇性填寫性質, 目的為讓技能去挑選具貢獻之卡牌, 並回報其編號。)

(註²: 本指令無法直接適用於格式轉換器之同名函數, 若想利用格式轉換器者, 請參照「II. 使用格式轉換器-新增/變更之指令函數」之說明)

(註³: 有關於智能判斷型 AI 請參閱「主動技能之 AI 計算判斷設計」)

n1-(1.推薦/2.無意見/3.不推薦)

n2-(1.一般型評分/2.骰數變化量優先型評分)

[n3...]- (個別卡牌編號 (VBEPAGECARDNUMVS 順序))

移動距離類

BattleMoveControl

#n1

47

普通型

事件型

47

角色相對距離控制

n1-(1.近距離/2.中距離/3.遠距離)

EventMoveActionOff

#0

47

原應執行之距離變更無效化

PersonMoveControl

#n1,n2,n3

2/3/4/70

人物移動階段總移動量控制

(註: 一旦進行「指定」數值之後, 即無視各指令原讓其增加或減少之變化量)

n1-(1.自身/2.對方)

n2-(1.增加/2.減少/3.指定)

n3-(數值)

PersonMoveActionChange

#n1,n2

2/3/4/70

人物角色移動階段行動控制

(註: 在 1v1 戰鬥模式下更換角色指令為無效果。)

n1-(1.自身/2.對方)

n2-(1:向遠/2:回復/3:向近/4:更換角色/5:不選擇)

PersonAttackFirstControl #n1

2/3/4/70/71

人物角色優先攻擊控制

(註¹：一旦指定後即無視移動總量對於先後攻之判定。)

(註²：若指定自身先攻即是讓對方後攻，反之亦然。)

n1-(1.自身/2.對方)

骰數控制類

EventTotalDiceChange #n1,n2,n3

45

攻擊/防禦階段系統骰數變化量控制

(註¹：本指令為攻擊/防禦階段「出牌時」使用，適用一般對骰量加成效果)

(註²：一旦指定數值之後，即無視在本指令下各增加或減少之控制)

n1-(1.自身/2.對方)

n2-(1.加/2.減/3.乘/4.除(尾數捨去)/5.除(尾數進位)/6.指定)

n3-(數值)

EventPersonAbilityDiceChange #n1,n2,n3

45

攻擊/防禦階段角色白值能力對骰數變化量控制

(註¹：本指令對骰量加成效果是在 EventTotalDiceChange 之前計算)

(註²：一旦指定數值之後，即無視在本指令下各增加或減少之控制，亦無視角色原本之基礎白值設定)

n1-(1.自身/2.對方)

n2-(1.加/2.減/3.指定)

n3-(數值)

PersonTotalDiceControl #n1,n2,n3

10/11/30/31

攻擊/防禦階段擲骰前系統總骰數直接控制

(註：本指令為攻擊/防禦階段於雙方按下 OK 後，擲骰之前使用，適用於對總骰量具影響之效果)

n1-(1.自身/2.對方)

n2-(1.加/2.減/3.乘/4.除(尾數捨去)/5.除(尾數進位)/6.指定)

n3-(數值)

AtkingTrueDiceControl #n1,n2

20/21/22/23/24/25/26/27/28/29

擲骰後正面骰數控制

(註：本指令是在擲骰後，針對雙方之正面骰數差數作控制，例如我方正面骰 5，對手正面骰 2，則在我方為攻擊階段時，系統正面骰數為 3，在我方為防禦階段時，系統正面骰數為 -3。)

n1-(1.加成/2.減少/3.指定)

n2-(數值)

人物類

PersonBloodControl #n1,n2,n3,n4

46/48*

普通型

事件型

46/48

人物角色血量控制

(註：執行本指令會觸發執行階段-46[直接傷害/直接死亡]或 48[回復]事件)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(1.直接傷害/2.回復/3.直接死亡)

n4-(數值)

EventBloodActionOff #0

46

原應執行之傷害無效化

EventBloodActionChange #n1,n2

46

原應執行之傷害效果變更

(註 1：若為立即死亡者，只能以指定方式來改變所受之傷害。)

(註 2：改變後之傷害執行視為直傷且無法再觸發執行階段-46 事件。)

n1-(1.加成/2.減少/3.指定)

n2-(數值)

EventBloodReflection #n1,n2,n3

46/48

傷害反射效果執行

(註 1：反射之傷害執行視為直傷且無法再觸發執行階段-46 事件。)

(註 2：本指令不影響原效果之執行。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(指定數值)

EventHPLActionOff #0

48

原應執行之回復無效化

EventHPLActionChange #n1,n2

48

原應執行之回復效果變更

n1-(1.加成/2.減少/3.指定)

n2-(數值)

回復反射效果執行

(註¹：反射之回復執行無法再觸發執行階段-48 事件。)

(註²：本指令不影響原回復之執行。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(指定數值)

PersonAtkingOff #n1,n2,n3

異常狀態

人物實際狀態

普通型

事件型

禁止人物主動技能發動效果控制(整體)

(註：在本指令標記開啟後，除已啟動之主動技外均無法在任何執行階段執行，直到再次使用本指令標記關閉後方可啟動。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(1.開啟/2.關閉)

PersonPassiveOff #n1,n2,n3

異常狀態

人物實際狀態

普通型

事件型

禁止人物被動技能發動效果控制(整體)

(註：在本指令標記開啟後，除已啟動之被動技外均無法在任何執行階段執行，直到再次使用本指令標記關閉後方可啟動。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(1.開啟/2.關閉)

PersonAtkingOffSelect #n1,n2,n3,n4

異常狀態

人物實際狀態

普通型

事件型

禁止人物主動技能發動效果控制(選擇)

(註¹：在本指令標記開啟後，除非該技能仍在啟動中，否則均無法在任何執行階段執行，直到再次使用本指令標記關閉後方可啟動。)

(註²：使用 PersonAtkingOff 指令時會複寫本指令之設定。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(1.開啟/2.關閉)

n4-(第 x 個主動技能)

PersonPassiveOffSelect #n1,n2,n3,n4

異常狀態

人物實際狀態

普通型

事件型

禁止人物被動技能發動效果控制(選擇)

(註 1：在本指令標記開啟後，除非該技能仍在啟動中，否則均無法在任何執行階段執行，直到再次使用本指令標記關閉後方可啟動。)

(註 2：使用 PersonPassiveOff 指令時會複寫本指令之設定。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(1.開啟/2.關閉)

n4-(第 x 個被動技能)

PersonAtkingInvalid

#n1,n2

普通型

事件型

人物必殺技無效化

(註：本指令只將技能強制關閉並解除對殷量造成之變化量，並無法回朔在發動本指令之前該技能對戰況之影響。)

n1-(1.自身/2.對方)

n2-(1.主動技/2.被動技)

PersonResurrect

#n1,n2

49

普通型

人物角色復活效果執行

(註 1：若發動指令時當下該指定之角色 HP 大於 0 者則無效。)

(註 2：利用本指令復活後之角色 HP 為 1，但其卡牌最大數量上限並不會隨之減少。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

EventPersonResurrectActionOff #0

49

原應執行之人物角色復活效果無效化

PersonChangeBattleImage

#n1,n2

主動技能

被動技能

普通型

事件型

人物角色更換戰鬥擲骰立繪

(註：本指令僅一次性覆寫目標人物之立繪資料，並非為狀態之儲存。)

n1-(1.自身/2.對方)

n2-(欲套用之大人物形象圖片路徑)

卡牌類

AtkingSeizeEnemyCards

#n1,n2

101

普通型

事件型

101

對手卡牌奪取效果執行

n1-(1.手牌類/2.出牌類)

n2-(牌編號(VBEPAGECardNumVS 順序))

EventAtkingSeizeEnemyCardsActionOff #0

101

原應執行之對手卡牌奪取效果無效化

AtkingDrawCards

#n1,n2,n3

102/103*

普通型

事件型

102/103

抽牌效果執行

(註¹: 抽取事件卡類時會直接推進回合數。)

(註²: 執行本指令時觸發執行階段之順序如下: 102[抽取前] > 103[每次抽牌時]。)

(註³: 執行階段 103 事件為抽取公用牌類限定。)

(註⁴: 抽取公用牌類時若牌庫不足以支應抽牌張數時會先執行洗牌再進行抽牌; 若仍舊不足時以能抽取之最大數量為限。)

n1-(1.自身/2.對方)

n2-(1.公用牌類/2.事件卡類)

n3-(抽牌張數)

EventAtkingDrawCardsActionOff #0

102/103

原應執行之抽牌效果無效化

(註: 若在執行階段 103[每次抽牌時]執行本指令, 則當前所預計抽取之卡牌亦將取消抽取。)

EventAtkingDrawCardsAddOnce #0

103

抽牌效果執行時原應抽取之卡牌數量增加 1 次

(註: 抽取公用牌類時若牌庫不足以支應抽牌張數時不會再進行洗牌前置動作。)

EventAtkingDrawCardsContinue #0

103

抽牌效果執行時原應抽取之卡牌略過

(註: 執行本指令時當前所預計抽取之卡牌將取消抽取。)

BattleDeckShuffle

#0

普通型

事件型

102/103/106

強制洗牌執行

AtkingDestroyCards	#n1,n2	104	普通型	事件型	104
手牌丟牌效果執行					
n1-(1.自身/2.對方)					
n2-(牌編號(VBEPageCardNumVS 順序))					
EventAtkingDestroyCardsActionOff	#0	104			
原應執行之手牌丟牌效果無效化					
AtkingGiveCards	#n1	105	普通型	事件型	105
自身手牌送與效果執行					
n1-(牌編號(VBEPageCardNumVS 順序))					
EventAtkingGiveCardsActionOff	#0	105			
原應執行之手牌送與效果無效化					
AtkingGetUsedCards	#n1	106	普通型	事件型	106
墓地牌/棄牌回牌效果執行					
n1-(牌編號(VBEPageCardNumVS 順序))					
EventAtkingGetUsedCardsActionOff	#0	106			
原應執行之墓地牌/棄牌回牌效果無效化					
AtkingOneSelfCardControl	#n1,n2,n3	107	2/3/4/70/10/11/12/17/30/31/32/37		
對某一方擁有之卡牌進行控制					
n1-(1.自身/2.對方)					
n2-(1.手牌出牌/2.出牌回牌/3.轉牌)					
n3-(牌編號(VBEPageCardNumVS 順序))					
EventAtkingOneSelfCardControlActionOff	#0	107			
原應執行之卡牌控制效果無效化					
PersonMaxCardsNumControl	#n1,n2,n3	普通型		事件型	
對人物角色之卡牌最大數量上限控制					
n1-(1.自身/2.對方)					
n2-(1.增加/2.減少)					
n3-(數值)					

插入事件卡效果執行

(註¹：事件卡標題為系統指定之字串，詳細請參照下方表格內容。)

(註²：插入之回合數若小於等於目前回合數者則無效。)

(註³：現有之事件卡牌堆若其數量不足以到達所指定插入之回合數者，則插入至目前牌堆之最後。)

n1-(1.自身/2.對方)

n2-(第 x 張(戰鬥回合數順序))

n3-(事件卡標題)

單面事件卡							
劍 1	劍 2	劍 3	劍 4	劍 5	劍 6	劍 7	劍 8
槍 1	槍 2	槍 3	槍 4	槍 5	槍 6	槍 7	槍 8
防 1	防 2	防 3	防 4	防 5	防 7		
特 1	特 2	特 3	特 4	特 5			
移 1	移 2	移 3	移 4	移 5			
機會 1	機會 2	機會 3	機會 4	機會 5			
詛咒術 1	詛咒術 2	詛咒術 3	詛咒術 5				
HP 回復 1	HP 回復 2	HP 回復 3	聖水				
雙面事件卡							
劍 3/槍 1	劍 4/槍 2	劍 5/槍 3	劍 3/移 1	劍 4/移 1	劍 5/移 1	劍 3/防 1	劍 5/槍 5
防 3/移 1	防 4/移 1	防 5/移 1	特 1/防 1	特 2/防 2	特 3/防 3		
槍 3/劍 1	槍 4/劍 2	槍 5/劍 3	槍 3/移 1	槍 4/移 1	槍 5/移 1	槍 3/防 1	
移 1/特 1	移 2/特 2	移 3/特 3					

註*：使用聖水事件卡視同人物角色移除異常狀態(全部)，會觸發執行階段-73 事件，並視為聖水事件卡解除。

ii. 附加狀態類型

人物實際狀態類

PersonAddActualStatus #n1,n2,n3,n4

74/76*

主動技能

被動技能

普通型

事件型

74/75

人物角色新增人物實際狀態

(註：執行本指令時觸發執行階段之順序如下：74[人物實際狀態新增時] > 76[人物角色附加狀態增加時] 。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(人物實際狀態之技能唯一識別碼)

n4-(效果回合數)

EventAddActualStatusData #n1,n2,n3,n4,n5,n6,n7,n8

人物實際狀態

普通型

事件型

人物實際狀態新增資料

(註：本指令為人物實際狀態於被新增時之擴充資料填入，原則上在執行階段-74中使用，但該指令亦可在人物實際狀態週期內重複執行，每次執行指令均會立即變更目標人物之立繪資料。)

n1-(欲套用之大人物形象圖片路徑)

n2-(欲套用之小人物形象圖片路徑)

n3-(欲套用之小人物形象影子圖片路徑)

n4-(欲套用之顯示列人物頭像圖片路徑)

n5-(小人物形象影子座標 Left(使用者))

n6-(小人物形象影子座標 Top(使用者))

n7-(小人物形象影子座標 Left(電腦))

n8-(小人物形象影子座標 Top(電腦))

ActualStatusEnd

#0

75/77*

人物實際狀態

普通型

事件型

74/75

人物實際狀態宣告結束

(註¹：本指令執行後即宣告狀態終結，並非減少回合數。)

(註²：使用本指令觸發之執行階段-75事件視為主動解除。)

(註³：執行本指令時觸發執行階段之順序如下：75[人物實際狀態消滅時] > 77[人物角色附加狀態解除時]；惟當人物實際狀態消滅後才會接續觸發執行階段 77 事件。)

PersonRemoveActualStatus #n1,n2

75/77*

主動技能

被動技能

普通型

事件型

74/75

人物角色移除人物實際狀態

(註 1：使用本指令觸發之執行階段-75 事件視為被動解除。)

(註 2：執行本指令時觸發執行階段之順序如下：75[人物實際狀態消滅時] > 77[人物角色附加狀態解除時]；惟當人物實際狀態消滅後才會接續觸發執行階段 77 事件。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

EventRemoveActualStatusActionOff #0

人物實際狀態

75

原應執行之人物實際狀態消除無效化

(註：若為主動解除時則無法觸發。)

異常狀態類

PersonAddBuff #n1,n2,n3,n4,n5

72/76*

普通型

事件型

異常狀態

72/73

人物角色新增異常狀態

(註 1：若異常狀態已存在者則直接覆寫其效果設定。)

(註 2：執行本指令時觸發執行階段之順序如下：72[異常狀態新增時] > 76[人物角色附加狀態增加時]。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(異常狀態之技能唯一識別碼)

n4-(效果變化量)

n5-(效果回合數)

BuffTurnEnd

#0

73/77*

異常狀態

普通型

事件型

72/73

異常狀態宣告當回合結束

(註 1：本指令執行後會將狀態之回合數減少 1 回合，並非直接宣告結束。)

(註 2：若使用本指令後回合數歸 0 者，會觸發執行階段-73 事件，並視為主動解除。)

(註 3：若觸發執行階段 73 事件時，執行本指令觸發執行階段之順序如下：73[異常狀態消滅時] > 77[人物角色附加狀態解除時]；惟當異常狀態消滅後才會接續觸發執行階段 77 事件。)

PersonRemoveBuffAll #n1,n2

73/77*

普通型

事件型

異常狀態

72/73

人物角色移除異常狀態(全部)

(註¹: 使用本指令會觸發執行階段-73 事件, 並視為被動解除。)

(註²: 執行本指令時觸發執行階段之順序如下: 73[異常狀態消滅時] > 77[人物角色附加狀態解除時]; 惟當異常狀態消滅後才會接續觸發執行階段 77 事件。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

PersonRemoveBuffSelect #n1,n2,n3

73/77*

普通型

事件型

異常狀態

72/73

人物角色移除異常狀態(指定)

(註¹: 使用本指令會觸發執行階段-73 事件, 並視為被動解除。)

(註²: 執行本指令時觸發執行階段之順序如下: 73[異常狀態消滅時] > 77[人物角色附加狀態解除時]; 惟當異常狀態消滅後才會接續觸發執行階段 77 事件。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(異常狀態之技能唯一識別碼)

PersonBuffTurnChange #n1,n2,n3,n4,n5

73/77*

普通型

事件型

異常狀態

72/73

人物角色異常狀態變更回合數

(註¹: 使用本指令若減少或指定回合數至 0 者, 會觸發執行階段-73 事件, 並視為主動解除。)

(註²: 若觸發執行階段 73 事件時, 執行本指令觸發執行階段之順序如下: 73[異常狀態消滅時] > 77[人物角色附加狀態解除時]; 惟當異常狀態消滅後才會接續觸發執行階段 77 事件。)

n1-(1.自身/2.對方)

n2-(第 x 位(VBEAtkingVSF 順序))

n3-(異常狀態之技能唯一識別碼)

n4-(1.增加/2.減少/3.指定)

n5-(數值)

EventRemoveBuffActionOff #0

異常狀態

73

原應執行之異常狀態消除無效化

(註: 若為主動解除時則無法觸發。)

II. 使用格式轉換器

i. 簡介

自 Version 1655 開始，您可以利用於各執行腳本中皆適用之「執行指令格式轉換器」參考物件（命名空間為 `VBECCommand`），來讓指令字串之組合編排更加便利。

需注意的是，由格式轉換器所呼叫之各項函數，只作為編製組合其指令字串，而非效果之執行，效果執行之實作仍依照「2. 腳本運作流程介紹」所示，傳回指令字串後由系統內部來解釋指令並執行效果。

例如：於某階段中使用「`AtkingLineLight`」作狀態燈控制開啟，加上「`AtkingTurnOnOff`」作啟動碼控制開啟時，最後所構成之指令字串如下：

```
AtkingLineLight#1=AtkingTurnOnOff#1=
```

以往為了能組合出如上之字串，程式寫法多半如下所示：

```
atking = atking & "AtkingLineLight#1" & "="  
atking = atking & "AtkingTurnOnOff#1" & "="
```

使用格式轉換器之程式寫法上可以更加簡潔且一目了然：

```
VBECCommand.AtkingLineLight 1  
VBECCommand.AtkingTurnOnOff 1
```

你也可以使用專屬於格式轉換器之指令函數：

```
VBECCommand.AtkingTurnOnOffWithLineLight 1
```

為了能讓格式轉換器所編製組合之指令字串進行回傳，於腳本函數執行之最後須呼叫「`VBECCommand.Output()`」方法來輸出指令字串：

（以主動技能為例）

```
atking = VBECCommand.Output()
```


ii. 指令函數

於「I. 基本指令」所列之各項指令中，絕大部分皆可直接適用在格式轉換器之同名函數當中（除外之部分請參照下方之說明）。

新增/變更之指令函數

AtkingTurnOnOffWithLineLight (n1 As Integer)

人物必殺技狀態燈控制+啟動碼控制(自身)

(註：本指令函數為複合指令，實際編製為「AtkingLineLight」+「AtkingTurnOnOff」)

n1-(1.開啟/2.關閉)

EventActiveAIScore (n1 As Integer, n2 As Integer)

智能判斷型 AI 個別技能判斷評分(指令函數)

(註¹：有關個別推薦牌編號部分請參照指令函數「EventActiveAIScoreAddCardNum」)

(註²：執行本函數時會將「EventActiveAIScoreAddCardNum」個別卡牌回報之資料予以清除重設)

n1-(1.推薦/2.無意見/3.不推薦)

n2-(1.一般型評分/2.骰數變化量優先型評分)

EventActiveAIScoreAddCardNum (n1 As Integer)

智能判斷型 AI 個別卡牌回報(指令函數)

(註¹：該函數對於智能判斷型 AI 為選擇性回傳性質，目的為讓技能去挑選具貢獻之卡牌，並回報其編號。)

(註²：使用時須先執行「EventActiveAIScore」指令函數後方可接續進行呼叫)

(註³：如有多張卡牌需回報時可重複執行本指令函數)

n1-卡牌編號(VBEPAGECardNumVS 順序)

舉例來說，基本指令「AtkingLineLight」具有其參數 n1，相對應格式轉換器之同名函數則為「VBCommand.AtkingLineLight」，需注意的是其參數 n1 為 Integer 型態，一般來說若該參數為數字特徵者則指令函數就會是 Integer 型態；若該參數為字串特徵者則指令函數就會是 String 型態。詳細之指令函數宣告可參照下方表格：

指令函數宣告一覽表

```
Public Sub AtkingLineLight(ByVal n1 As Integer)
Public Sub AtkingTurnOnOff(ByVal n1 As Integer)
Public Sub AtkingTurnOnOffWithLineLight(ByVal n1 As Integer)
Public Sub AtkingLineLightAnother(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub AtkingTurnOnOffAnother(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub AtkingStartPlay(ByVal n1 As String, Optional ByVal n2 As String)
Public Sub AtkingStartPlayAnimate(ByVal n1 As String, Optional ByVal n2 As Integer, Optional ByVal n3 As Integer)
Public Sub BattleTurnControl(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub BattleSendMessage(ByVal n1 As String)
Public Sub BattleStartDice()
Public Sub AtkingInformationRecord(ByVal n1 As String)
Public Sub EventPlayerAllActionOff(ByVal n1 As Integer)
Public Sub EventActiveAIScore(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub EventActiveAIScoreAddCardNum(ByVal n1 As Integer)
Public Sub BattleMoveControl(ByVal n1 As Integer)
Public Sub EventMoveActionOff()
Public Sub PersonMoveControl(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub PersonMoveActionChange(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub PersonAttackFirstControl(ByVal n1 As Integer)
Public Sub EventTotalDiceChange(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
```

```

Public Sub EventPersonAbilityDiceChange(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub PersonTotalDiceControl(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub AtkingTrueDiceControl(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub PersonBloodControl(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer, ByVal n4 As Integer)
Public Sub EventBloodActionOff()
Public Sub EventBloodActionChange(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub EventBloodReflection(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub EventHPLActionOff()
Public Sub EventHPLActionChange(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub EventHPLReflection(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub PersonAtkingOff(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub PersonPassiveOff(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub PersonAtkingOffSelect(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer, ByVal n4 As Integer)
Public Sub PersonPassiveOffSelect(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer, ByVal n4 As Integer)
Public Sub PersonAtkingInvalid(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub PersonResurrect(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub EventPersonResurrectActionOff()
Public Sub PersonChangeBattleImage(ByVal n1 As Integer, ByVal n2 As String)
Public Sub AtkingSeizeEnemyCards(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub EventAtkingSeizeEnemyCardsActionOff()
Public Sub AtkingDrawCards(ByVal n1 As Integer, ByVal n2 As Integer, ByVal n3 As Integer)
Public Sub EventAtkingDrawCardsActionOff()
Public Sub EventAtkingDrawCardsAddOnce()
Public Sub EventAtkingDrawCardsContinue()

```

```

Public Sub BattleDeckShuffle()
Public Sub AtkingDestroyCards(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub EventAtkingDestroyCardsActionOff()
Public Sub AtkingGiveCards(ByVal n1 As Integer)
Public Sub EventAtkingGiveCardsActionOff()
Public Sub AtkingGetUsedCards(ByVal n1 As Integer)
Public Sub AtkingOneSelfCardControl(ByVal n1 As Integer, ByVal n2 As
Integer, ByVal n3 As Integer)
Public Sub EventAtkingOneSelfCardControlActionOff()
Public Sub PersonMaxCardsNumControl(ByVal n1 As Integer, ByVal n2 As
Integer, ByVal n3 As Integer)
Public Sub BattleInsertEventCard(ByVal n1 As Integer, ByVal n2 As
Integer, ByVal n3 As String)
Public Sub PersonAddActualStatus(ByVal n1 As Integer, ByVal n2 As
Integer, ByVal n3 As String, ByVal n4 As Integer)
Public Sub EventAddActualStatusData(ByVal n1 As String, ByVal n2 As
String, ByVal n3 As String, ByVal n4 As String, ByVal n5 As Integer,
ByVal n6 As Integer, ByVal n7 As Integer, ByVal n8 As Integer)
Public Sub ActualStatusEnd()
Public Sub PersonRemoveActualStatus(ByVal n1 As Integer, ByVal n2 As
Integer)
Public Sub EventRemoveActualStatusActionOff()
Public Sub PersonAddBuff(ByVal n1 As Integer, ByVal n2 As Integer, ByVal
n3 As String, ByVal n4 As Integer, ByVal n5 As Integer)
Public Sub BuffTurnEnd()
Public Sub PersonRemoveBuffAll(ByVal n1 As Integer, ByVal n2 As Integer)
Public Sub PersonRemoveBuffSelect(ByVal n1 As Integer, ByVal n2 As
Integer, ByVal n3 As String)
Public Sub PersonBuffTurnChange(ByVal n1 As Integer, ByVal n2 As Integer,
ByVal n3 As String, ByVal n4 As Integer, ByVal n5 As Integer)
Public Sub EventRemoveBuffActionOff()

```

5. 腳本引數變數

為了能讓腳本進行遊戲場上之情勢做出判斷，遊戲系統提供了各式之場上資訊讓各技能體系腳本做為效果判斷之依據，以下開始為各引數變數之詳細說明：

特定標記圖例

主動技能	指定之技能體系適用
異常狀態	指定之技能體系 排除 適用

(若無適用技能體系特定標記者，則屬全體適用)

腳本引數變數

VBEAtkingVSF (1 to 2, 1 to 3, 1 to 2)

雙方血量資訊一覽

(1. 自身/2. 對方,
1~3. 場上順序排列-場上 1 位>場下待機 2 位,
1. 當前血量/2. 當前血量最大值)

VBEAtkingVSS (0 to 20)

戰鬥場上資訊一覽

0. 目前角色之場上位置(VBEAtkingVSF 順序)
1. 出牌數-自身
2. 手牌數-自身
3. 出牌數-對手
4. 手牌數-對手
5. 系統公骰擲骰後正骰數量(攻擊正面骰減去防禦正面骰)
6. 當前距離(1. 近距離/2. 中距離/3. 遠距離)
7. 自身目前總骰數(系統公骰)
8. 對手目前總骰數(系統公骰)
9. 當前回合數
10. 程式執行根目錄路徑
11. 當前公牌數量
12. 目前階段(1-DRAW/2-MOVE/3-ATK/4-DEF)
13. 目前何方攻擊(1-自身/2-對方)

- 14. 擲骰後自身正骰數量(系統公骰)
- 15. 擲骰後對手正骰數量(系統公骰)
- 16. 本回合何方先攻(0-未決定/1-自身/2-對方)
- 17. 我方戰鬥總人數
- 18. 敵方戰鬥總人數
- 19. 我方目前卡牌最大數量上限
- 20. 敵方目前卡牌最大數量上限

AtkingPagetotVS (1 To 2, 1 To 5)

出牌種類及數值統計資料

- (1. 自身/2. 對方,
- 1. 劍/2. 防/3. 移/4. 特/5. 槍)

VBEPersonVS

(1 to 2, 1 to 3, 1 to 3, 1 to 8, 1 to 11)

異常狀態

人物角色總資訊一覽

- (1. 自身/2. 對方,
- 1~3. 第 x 位人物(VBEAtkingVSF 順序),
- 1. 基本資料系/2. 座標系/3. 技能系,
- 1~8. 各類別(參照下方說明),
- 1~11. 各項目(參照下方說明))

基本資料系

- 1. 名稱類(1. 中文名稱/2. 英文名稱/3. 稱號)
- 2. 牌等級類(1. 等級類別/2. 等級數)
- 3. 人物能力值類(1. HP/2. ATK/3. DEF/4. 人物事件卡顏色格數字串/5. 能力資訊是否為新樣式[1. 是/0. 否])
- 4. ID 類(1. VBEID/2. ULID/3. AI 版本號)
- 5. 人物圖片路徑類(1. 小人物圖路徑/2. 小人物影子圖路徑/3. 大人物圖路徑/4. 小人物頭像圖路徑/5. 人物卡片圖路徑)

座標系

- 1. 小人物立繪類(1~4. (無資料)/5. 影子 Left/6. 影子 Top)
- 2. 大人物立繪類(1. Height/2. Width/3. Top/4. 移動最大 Left/5. 是否直接移動到底[1. 是/0. 否])
- 3. 技能字體大小類(1~4. 第 n 個個別技能說明字體大小/5. 是否套用預設值 13[1. 是/0. 否])

技能系

- 1~4. 第 1~4 種主動技能詳細類(1. 名稱/2. 階段/3. 距離/4. 卡片/5. 效果/6. 卡片字型大小(選)/7. 效果字型大小(選)/8. 階段代碼/9. 距離代碼/10. 卡片代碼/11. 技能唯一識別碼)
- 5~8. 第 1~4 種被動技能詳細類(1. 名稱/2. 效果/3. 技能唯一識別碼)

VBEPageCardNumVS (1 To ?, 1 To 7)

異常狀態

戰鬥系統場上卡牌資訊一覽

(註：對於場上之卡牌張數為不確定數，可由「UBound(VBEPageCardNumVS, 1)」得知。)

(第 x 編號，

1. 正面類型/2. 正面數值/3. 反面類型/4. 反面數值/

5. (0)系統-(1)自身-(2)對方/

6. (1)手牌-(2)出牌-(3)墓地牌/棄牌-(4)牌堆-(5)未使用/

7. (1)公用牌/(2)事件卡)

(註¹：卡牌類型如下：1. 劍/2. 防/3. 移/4. 特/5. 槍/6. 機會/7. 詛咒術/8. 回復/9. 聖水/0. 其他)

(註²：系統將雙方卡牌收回成墓地牌/棄牌時並未清除第 5 項之資訊，待經洗牌後使其重設為 0(系統))

(註³：事件卡於系統收回時將標記為[棄牌]而非[墓地牌])

(註⁴：於執行階段 99 所給與之資訊中，卡牌一律視為[自身]/[手牌]/[公用牌]之設定)

AtkingckVSS (1 to 8, 1 to 3)

主動技能

被動技能

主動技能/被動技能資訊一覽(一般)

(1~4 人物自身主動技能項目; 5~8 人物自身被動技能項目，

1. 技能啟動標記/2. 這回合間啟動次數/3. 這場戰鬥間啟動次數)

AtkingckVSF (1 to 8, 1 to 1)

主動技能

被動技能

主動技能/被動技能資訊一覽(字串)

(1~4 人物自身主動技能項目; 5~8 人物自身被動技能項目，

1. 技能備註資訊)

VBEPersonBuffVSF (1 to 2, 1 to 3, 1 To ?, 1 To 2)

異常狀態

異常狀態資訊一覽(一般)

(註：對於場上人物之異常狀態數量為不確定數，可由「UBound(VBEPersonBuffVSF, 3)」得知。)

(1. 自身/2. 對方，

第 x 位(VBEAtkingVSF 順序)，

第 x 個異常狀態，

1. 狀態數值/2. 狀態統計數(剩餘回合/累計))

VBEPersonBuffVSS (1 to 2, 1 to 3, 1 To ?)

異常狀態

異常狀態資訊一覽(技能唯一識別碼)

(註：對於場上人物之異常狀態數量為不確定數，可由「UBound(VBEPersonBuffVSS, 3)」得知。)

(1. 自身/2. 對方，
第 x 位(VBEAtkingVSF 順序)，
第 x 個異常狀態 - 技能唯一識別碼)

VBEActualStatusVS (1 to 2, 1 to 3, 1 to 2)

主動技能

被動技能

人物實際狀態資訊一覽

(1. 自身/2. 對方，
第 x 位(VBEAtkingVSF 順序)，
1. 技能唯一識別碼/2. 人物實際狀態名稱)

BuffPersonType

異常狀態

該角色目前場上狀態

(1) 場上角色
(2) 待機角色

ActualStatusPersonType

人物實際狀態

該角色目前場上狀態

(1) 場上角色
(2) 待機角色

PassivePersonType

被動技能

該角色目前場上狀態

(1) 場上角色
(2) 待機角色

VBEVSBuffNum (1 to 2)

異常狀態

異常狀態自身資訊一覽

1. 異常狀態自身變化量數
2. 異常狀態自身回合數

VBEVSStageNum (?)

執行階段多用途紀錄資訊

(註：對於多用途紀錄資訊變數為不確定數，且僅部分之執行階段才會提供資訊，詳見以下之列表：)

執行階段-2/3/4/70/71(普通型-移動前)

- (1)自身目前總計之移動數值
- (2)對方目前總計之移動數值
- (3)自身目前移動階段選擇項目(0:無/1:向遠/2:回復/3:向近/4:更換角色)
- (4)對方目前移動階段選擇項目(0:無/1:向遠/2:回復/3:向近/4:更換角色)

執行階段-41(人物交換)

- (1)執行效果方(1.自身/2.對方)
- (2)交換前人物編號(VBEAtkingVSF 順序)
- (3)交換後人物編號(VBEAtkingVSF 順序)

執行階段-42/43/44(特殊型-各階段出牌時)

- (1)觸發事件方(1.自身/2.對方)

執行階段-46(傷害)

- (1)受到傷害方(1.自身/2.對方)
- (2)受到傷害方人物編號(VBEAtkingVSF 順序)
- (3)受到傷害之形式(1.毆傷/2.直傷/3.立即死亡)
- (4)受到傷害之數值
- (5)觸發事件方(0.系統/1.自身/2.對方)
- (6)觸發狀態類別(0.系統/1.主動技能/2.被動技能/3.異常狀態/4.人物實際狀態)

執行階段-47(移動)

- (1)變更前距離
- (2)變更後距離
- (3)觸發事件方(0.系統/1.自身/2.對方)

執行階段-48(回復)

- (1)回復方(1.自身/2.對方)
- (2)回復方人物編號(VBEAtkingVSF 順序)
- (3)回復之數值
- (4)觸發事件方(0.系統/1.自身/2.對方)
- (5)觸發狀態類別(0.系統/1.主動技能/2.被動技能/3.異常狀態/4.人物實際狀態)

執行階段-49(角色復活)

- (1)角色復活方(1.自身/2.對方)
- (2)角色復活方人物編號(VBEAtkingVSF 順序)
- (3)觸發事件方(1.自身/2.對方)

執行階段-62(技能效果進行多次擲骰時)

- (1)自身總骰數
- (2)對手總骰數
- (3)擲骰後自身正骰數量
- (4)擲骰後對手正骰數量
- (5)擲骰後正骰數量(攻擊正面骰減去防禦正面骰)

執行階段-73(異常狀態消滅時)

- (1)是否為被動消除(0.否/1.是/2.聖水事件卡)

執行階段-74(人物實際狀態-新增)

- (1)原引發之技能所要求之效果回合數

執行階段-75(人物實際狀態消滅時)

- (1)是否為被動消除(0.否/1.是)

執行階段-76/77(人物角色附加狀態增加/解除時)

- (1)觸發事件方(1.自身/2.對方)
- (2)觸發狀態類別(1.異常狀態/2.人物實際狀態)
- (3)該觸發狀態之技能唯一識別碼(Type:String)(若為異常狀態全體解除者，則為空白)

執行階段-101(對手卡牌奪取效果執行時)

- (1)觸發事件方(1.自身/2.對方)
- (2)受影響之卡牌編號(VBEPAGECardNumVS 順序)
- (3)受影響之卡牌正面類型(VBEPAGECardNumVS 卡片類型)
- (4)受影響之卡牌正面數值
- (5)受影響之卡牌反面類型(VBEPAGECardNumVS 卡片類型)
- (6)受影響之卡牌反面數值
- (7)受影響之卡牌場上位置類別(1.手牌/2.出牌)

執行階段-102(抽牌效果執行時)

- (1)觸發事件方(1.自身/2.對方)

執行階段-103(抽牌效果執行每次抽牌時)

- (1)預計抽牌之卡牌編號(VBEPAGECARDNUMVS 順序)
- (2)預計抽牌之卡牌正面類型(VBEPAGECARDNUMVS 卡片類型)
- (3)預計抽牌之卡牌正面數值
- (4)預計抽牌之卡牌反面類型(VBEPAGECARDNUMVS 卡片類型)
- (5)預計抽牌之卡牌反面數值

執行階段-104(手牌丟牌效果執行時)

- (1)觸發事件方(1.自身/2.對方)
- (2)目標方(1.自身/2.對方)
- (3)受影響之卡牌編號(VBEPAGECARDNUMVS 順序)
- (4)受影響之卡牌正面類型(VBEPAGECARDNUMVS 卡片類型)
- (5)受影響之卡牌正面數值
- (6)受影響之卡牌反面類型(VBEPAGECARDNUMVS 卡片類型)
- (7)受影響之卡牌反面數值

執行階段-105(自身卡牌送與效果執行時)

- (1)觸發事件方(1.自身/2.對方)
- (2)受影響之卡牌編號(VBEPAGECARDNUMVS 順序)
- (3)受影響之卡牌正面類型(VBEPAGECARDNUMVS 卡片類型)
- (4)受影響之卡牌正面數值
- (5)受影響之卡牌反面類型(VBEPAGECARDNUMVS 卡片類型)
- (6)受影響之卡牌反面數值

執行階段-106(墓地牌/棄牌回牌效果執行時)

- (1)觸發事件方(1.自身/2.對方)
- (2)受影響之卡牌編號(VBEPAGECARDNUMVS 順序)
- (3)受影響之卡牌正面類型(VBEPAGECARDNUMVS 卡片類型)
- (4)受影響之卡牌正面數值
- (5)受影響之卡牌反面類型(VBEPAGECARDNUMVS 卡片類型)
- (6)受影響之卡牌反面數值
- (7)受影響之卡牌場上位置類別(1.墓地牌/2.棄牌)

執行階段-107(對某一方擁有之卡牌進行控制時)

- (1)觸發事件方(1.自身/2.對方)
- (2)目標方(1.自身/2.對方)
- (3)受影響之卡牌編號(VBEPAGECARDNUMVS 順序)
- (4)受影響之卡牌正面類型(VBEPAGECARDNUMVS 卡片類型)
- (5)受影響之卡牌正面數值
- (6)受影響之卡牌反面類型(VBEPAGECARDNUMVS 卡片類型)
- (7)受影響之卡牌反面數值
- (8)受影響之卡牌預定控制效果(1.手牌出牌/2.出牌回牌/3.轉牌)

6. 腳本體系類型

在遊戲系統中，會影響人物角色之技能體系主要有以下 4 種：

I. 主動技能：人物角色所擁有，也是最基礎之技能體系，需要透過指定之出牌條件來發動，每個人物角色最多擁有 4 個主動技能。

II. 被動技能：人物角色所擁有，相對於主動技能，被動技能擁有在不特定之時間點下發動，且無須設定出牌條件，每個人物角色最多擁有 4 個被動技能。

III. 異常狀態：獨立之技能體系，會附著在人物角色身上，需要透過其他技能體系來引導發動；每位角色對異常狀態最多可附加每種類各 1 個，UI 最多可顯示前 18 種附加異常狀態。

IV. 人物實際狀態：獨立之技能體系，相對於異常狀態之附著特性，人物實際狀態為在特定時間內對人物角色造成其實體改變，需要透過其他技能體系來引導發動，每位角色最多擁有 1 個人物實際狀態。

為了能讓各位更了解這些技能體系之歸屬，以下舉例為引用 Unlight 官方

人物角色-復活角色艾茵 N1 (ULID:2111) 之各技能歸屬：

艾茵(N1)

主動技能	一顆心、七個願望、十三隻眼
被動技能	兩個身體
人物實際狀態	貓化狀態

在遊戲設計中，人物實際狀態為最具有權力及影響力之技能體系，其次是異常狀態、被動技能，最後是主動技能。在每個執行階段中，人物角色會依照技能體系權力大小之順序來執行效果：「**人物實際狀態>異常狀態>被動技能>主動技能**」。

另外在攻擊/防禦階段之**骰數變化量**計算之執行階段-45事件中，各個技能體系所分別給予之總骰數變化量計算則依照影響力之順序來**先後**計算加成：「**主動技能>被動技能>異常狀態>人物實際狀態**」，另因執行階段-45是系統計算雙方之總骰數變化量的唯一事件，如果在雙方都有讓對方具總骰數變化量的影響時，人物角色計算加成之順序如下：「**自身之主動技能>自身之被動技能>自身之異常狀態>自身之人物實際狀態>對手之主動技能>對手之被動技能>對手之異常狀態>對手之人物實際狀態**」。

以下表格範例為引用 Unlight 官方人物角色-史塔夏 LV5 (ULID:335, VBEID:N00505) 之實際計算狀況，假設史塔夏已變身殺戮狀態，對手全員活著，在攻擊階段於近距離出一張劍 6 行動卡的情況下，自身之總骰數計算如下：

史塔夏(LV5)自身總骰數變化量加成

攻擊白值	行動卡	主動技能	被動技能	異常狀態	人物實際狀態	總骰數
4 (LV5)	+6 (劍 6)	+12 (愚者之手)			*2 (殺戮狀態)	44

7. 腳本及卡片人物資訊文件格式

I. 卡片人物資訊 (UnlightVBE Card Information)

卡片人物資訊文件 (*.uleci) 為記載人物角色之各式基本資訊之純文字文件，儲存在 **character** 資料夾中以人物角色名稱作為個別人物角色之資料夾下。以下表格為引用 UnlightVBE 中自創人物角色-蕾 (VBEID:S001) 之卡片人物資訊文件基礎架構：

卡片人物資訊文件架構

架構程式碼	說明
Q4VX435S MenuName=蕾 MenuList=LV5=R1=R2=R3=R4=R5 EndFirst	初始架構，紀錄選單及文件內所包含的等級
StartPerson=LV5=蕾 cardjpg= Ria\RiaLV5.jpg personhp=9 personatk=5 persondef=10 EndPerson	主資訊架構，紀錄角色某一等級的基本資訊
StartTalk=蕾=LV=R EndTalk	對話架構，紀錄角色的人物對話資訊

在一份文件中必須擁有初始架構及至少 1 項等級之主資訊架構，對話架構則為選擇性填寫。

以下表格為架構中每項設定之詳細說明：

(部分內容因說明之需求純屬舉例)

卡片人物資訊文件架構設定項說明 (初始架構)

標題	內容	說明
Q4VX435S		VBE-QS 1.0 版本文件檢查碼 (G15d466E : VBE-1.0 版本文件檢查碼 XzPF5922 : VBE-2.0 版本文件檢查碼)
MenuName	蕾	人物名稱
MenuList	LV5=R1=R2=R3=R4=R5	此文件所納入的角色等級。 (等級與等級中間以「=」相隔)
EndFirst		結束初始架構文字

卡片人物資訊文件架構設定項說明 (主資訊架構)

[若該選項有紅框者，則為選擇性填寫。]

標題	內容	說明
StartPerson	LV5=蕾	開始架構文字 LV5：本段落資訊所指的角色等級 蕾：人物名稱
cardjpg	Ria\RiaLV5.jpg	卡面圖片 (註 ³)
personhp	9	血量 (白值)
personatk	5	攻擊力 (白值)
persondef	10	防禦力 (白值)
cardInfisNewType	0	角色能力資訊是否為新樣式 [0-否/1-是] (新舊樣式會影響數字之位置顯示)
personname	蕾	角色名稱
personengname	Ria	角色英文名稱
personpname	Aria	角色稱號
personlevel1	LV	等級分類 (例：LV、R)
personlevel2	5	等級數字
cardid	S00105	VBE 角色 ID (註 ¹)
persontg	357000	角色卡色格 (註 ²)
personbig	Ria\Riaperson1.png	大人物立繪 (註 ³)
personmini	Ria\Riamini1.png	小人物立繪 (註 ³)
personf	Ria\Riaf1.png	顯示列臉圖 (註 ³)
personsmalldown	Ria\Riaminidown1.png	小人物立繪影子 (註 ³)

卡片人物資訊文件架構設定項說明（主資訊架構）

[若該選項有紅框者，則為選擇性填寫。]

標題	內容	說明
atkingfontck	0	角色技能欄字體大小是否為預設值 [1-預設值大小(12)/0-自訂] (自訂時須加上以下 4 列)
atkingfont1	10	第 1 個主動技能字體大小 (7~)
atkingfont2	10	第 2 個主動技能字體大小 (7~)
atkingfont3	10	第 3 個主動技能字體大小 (7~)
atkingfont4	10	第 4 個主動技能字體大小 (7~)
atkingjpgleftallzero	0	是否由系統自動控制大人物立繪移動位置 (註 4) [0-自動/1-對齊邊框]
smalldownleftus	0	使用者方小人物立繪影子座標 (Left)
smalldowntopus	-30	使用者方小人物立繪影子座標 (Top)
smalldownleftcom	40	電腦方小人物立繪影子座標 (Left)
smalldowntopcom	-30	電腦方小人物立繪影子座標 (Top)
atkingname(1)	PNAKS00101=atking1.ulevsa	第 1~4 項人物技能腳本連結 (註 5)
...
atkingname(5)	PNPES00101=passive1.ulevsp	第 5~8 項人物技能腳本連結 (註 5)
...

卡片人物資訊文件架構設定項說明（主資訊架構）

[若該選項有紅框者，則為選擇性填寫。]

標題	內容	說明
atkingcfont(1)	8	第 1~4 項主動技能發動條件 字體大小 (像是有「無合計 1 以上、無合計 1 以上、無合計 1 以上」超過 2 個條件者適用，建議大小為 8)
atkingdfont(1)	8	第 1~4 項主動技能發動條件 字體大小 (效果文字太長以致系統無法顯示全部者適用，建議大小為 8)
EndPerson		結束架構文字

卡片人物資訊文件架構設定項說明（對話架構-自身）

標題	內容	說明
StartTalk	蕾=LV=R	開始架構文字 蕾：人物名稱 LV、R：該對話架構所適用之自身角色等級類別 (若含 2 個以上等級類別，中間以「=」相隔)
TalkA1	為什麼要那麼傻呢，笨蛋。=凱爾	第 N 個相對角色類型對話(至多 20 個) 凱爾-本對話所指定之人物
TalkA2	時間是會流逝的，不管在哪裡。=阿貝爾&利恩=LV5	*註 1：對話若有 2 個以上指定人物者，人物與人物中間以「&」相隔（如 TalkA2 所示）。 *註 2：對話可限制其適用之自身角色等級，附加在人物區塊之後；若指定 2 個以上者，等級與等級中間以「&」相隔（如 TalkA2/TalkB2 所示）。
...	...	
TalkB1	吶，聽我唱首歌吧！	第 N 個無相對角色類型對話(至多 10 個)
TalkB2	也就是說，死亡並不是終點嗎.....=LV4&LV5	*註：對話可限制其適用之自身角色等級；若指定 2 個以上者，等級與等級中間以「&」相隔（如 TalkA2/TalkB2 所示）。

卡片人物資訊文件架構設定項說明 (對話架構-自身)

標題	內容	說明
...	...	
TalkC1	<p>你是我的光。來，跟我一起擁抱一切吧。</p> <p>=N01811</p>	<p>第 N 個特別指定角色類型對話 (至多 10 個)</p> <p>N01811 - 本對話所指定之人物 VBEID</p> <p>*註 1：對話若有 2 個以上指定人物 VBEID 者，ID 與 ID 中間以「&」相隔。</p> <p>*註 2：對話可限制其適用之自身角色等級，附加在 ID 區塊之後；若指定 2 個以上者，等級與等級中間以「&」相隔 (如 TalkA2/TalkB2 所示) 。</p>
EndTalk		結束架構文字

卡片人物資訊文件架構設定項說明 (對話架構-對手方)

標題	內容	說明
StartTalkCom	蕾=LV=R	<p>開始架構文字</p> <p>蕾：人物名稱</p> <p>LV、R：該對話架構所適用之角色等級類別</p> <p>(若含 2 個以上等級類別，中間以「=」相隔)</p>
...	...	(中間皆與「對話架構-自身」相同)
EndTalk		結束架構文字

於類型對話中，會依照「特別指定角色類型>相對角色類型>無相對角色類型」之優先順序來進行篩選；若同類型中有複數個符合條件之對話，則會以隨機方式進行指定。

(註¹)：VBE 角色 ID 為人物角色之唯一編號字串，本人制定時共為 6 碼，開發者亦可自行使用自行編製之編號字串，但需要具唯一性。

以下表格為 VBE 角色 ID 之編號說明：

VBE 角色 ID 編號說明

項目		說明
主分類	N	官方人物角色 (不限官方分類)
	S	自創人物角色
人物主編號	001	編號自 001~999
分項目號	01	編號自 01~99 通常會對應人物角色等級數字

(註²)：角色卡色格字串共 6 碼，對應人物角色之 6 格事件卡格：

人物角色卡色格

代號	對應顏色	
0	(無)	
1	紅	
2	綠	
3	藍	
4	紫	
5	白	
6	黑	
7	黃	

當遇到人物角色非 6 格事件卡格時，先將擁有之事件卡格進行編號，剩下之長度則補 0。

(註³)：所有圖片都要放在「gif\<人物角色名>」之資料夾內，且所有之圖片路徑需加入母資料夾名稱 (人物角色名)。

(註⁴)：部分人物角色立繪在自動控制時會有顯示上的不健全，主要原因為該立繪圖片主體有削邊之情形發生。

(註⁵)：技能腳本連結共有 2 個分項目，第 1 個為該技能之技能唯一識別碼，第 2 個為該技能腳本之檔名 (含副檔名)，第 1~4 項人物技能為主動技能，第 5~8 項人物技能則為被動技能。

II. 技能腳本文件

對於 4 個技能體系腳本文件架構而言，其實大部分都是相同的，只是在 **Function** 的名稱，以及引數變數上面會有所差異。

以下表格為技能體系腳本文件之分類：

腳本文件副檔名分類

技能體系	文件副檔名
主動技能 (Active)	*.ulevsa
被動技能 (Passive)	*.ulevsp
異常狀態 (Buff)	*.ulevsbf
人物實際狀態 (Character Actual Status)	*.ulevsc

腳本文件存放路徑分類

技能體系	文件存放路徑
主動技能 (Active)	\Character\<人物角色名>
被動技能 (Passive)	\Character\<人物角色名>
異常狀態 (Buff)	\Buff\<異常狀態名>
人物實際狀態 (Character Actual Status)	\CharacterActualStatus\<人物實際狀態名>

注意以上之路徑都是以主程式所在資料夾作為開始之路徑。

以下表格為引用 UnlightVBE 中自創人物角色-蕾 (VBEID:S001) 之主動技能腳本文件基礎架構：

技能腳本文件架構

架構程式碼範例	說明
<pre> Function main(a) Select Case a Case 1 main = "PNAKS00101" Case 2 main = "42#45#11" Case 3 main = "輪旋曲-琉璃色的微風" Case 4 main = "攻擊#1" Case 5 main = "近距離#001" Case 6 main = "劍合計 4 以上#14a" Case 7 main = "ATK+6・對手這回合有效防禦值-3" End Select End Function </pre>	<p>初始架構・紀錄技能之基本資訊</p>
<pre> Function atking(a, VBEPersonVS(), VBEPageCardNumVS(), AtkingPagetotVS(), VBEPersonBuffVSF(), VBEPersonBuffVSS(), AtkingckVSS(), AtkingckVSF(), VBEAtkingVSF(), VBEAtkingVSS(), VBEActualStatusVS(), VBEVSStageNum()) Select Case a Case 42 If VBEAtkingVSS(6) = 1 Then If AtkingPagetotVS(1, 1) >= 4 Then If AtkingckVSS(1, 1) = 0 Then VBECommand.AtkingTurnOnOffWithLineLight 1 End If Else If AtkingckVSS(1, 1) = 1 Then VBECommand.AtkingTurnOnOffWithLineLight 2 End If End If End If Case 45 VBECommand.EventTotalDiceChange 1, 1, 6 VBECommand.EventTotalDiceChange 2, 2, 3 Case 11 VBECommand.AtkingStartPlay "\gif\Ria\atking1.png" VBECommand.AtkingTurnOnOffWithLineLight 2 End Select atking = VBECommand.Output() End Function </pre>	<p>主架構・紀錄技能之效果撰寫</p>

注意不管是哪種技能技系・在一份文件中必須要擁有以上之 2 個架構。

在不同之技能體系中，初始架構之 Function 名稱都為 main，不過在主架構 Function 之名稱會有所差異：

主架構 Function 名稱分類

技能體系	Function 名稱
主動技能 (Active)	atking
被動技能 (Passive)	passive
異常狀態 (Buff)	buff
人物實際狀態 (Character Actual Status)	ActualStatus

初始架構之意義為一個技能之標頭資訊，以下就 4 個技能體系之不同的標頭資訊做說明（若該選項有紅框者，則為選擇性填寫）：

初始架構標頭資訊說明（主動技能）

（引用：自創人物角色-蓄 LV5（VBEID:S00105）主動技能-輪旋曲-琉璃色的微風）

編號	內容	說明
1	PNAKS00101	技能唯一識別碼（註 ¹ ）
2	42#45#11	欲使用之執行階段號（註 ² ）
3	輪旋曲-琉璃色的微風	主動技能名稱
4	攻擊#1	主動技能發動階段 1-攻擊階段 2-防禦階段 3-移動階段
5	近距離#001	主動技能發動距離 0-該技能是否適用於遠距離(0-否/1-是) 0-該技能是否適用於中距離(0-否/1-是) 1-該技能是否適用於近距離(0-否/1-是)
6	劍合計 4 以上#14a	主動技能發動條件（註 ³ ） 1-條件發動牌類型 (1-劍/2-盾/3-移/4-特/5-槍/6-其他) 4-條件發動牌數值 (0~9/a->10/b->?) a-條件發動數值之採取方式 (a-合計以上/b-指定等於/c-合計以下)
7	ATK+6・對手這回合有效防禦值-3	主動技能發動效果
8	(無)	主動技能於角色待命狀態時仍可執行之啟動旗標 (ON-開啟/非 ON 者則關閉)

初始架構標頭資訊說明 (被動技能)

(引用:官方人物角色-莉莉 LV1 (ULID:40001、VBEID:N02701) 被動技能-小公主)

編號	內容	說明
1	PNPEN02701	技能唯一識別碼 (註 ¹)
2	1#22#55	欲使用之執行階段號 (註 ²)
3	小公主	被動技能名稱
4	20%機率受到的傷害為 0。只會在 HP5 以上時發動。	被動技能發動效果

初始架構標頭資訊說明 (異常狀態)

(引用:增益系異常狀態-ATK+)

編號	內容	說明
1	BUFFN00101	技能唯一識別碼 (註 ¹)
2	45#53	欲使用之執行階段號 (註 ²)
3	ATK+	異常狀態名稱
4	\Buff\atkup\atkupnew.gif	異常狀態圖片路徑 (註 ⁴)

初始架構標頭資訊說明 (人物實際狀態)

(引用:官方人物角色-史塔夏 LV5 (ULID:335、VBEID:N00505) 人物實際狀態-殺戮模式)

編號	內容	說明
1	UCASN00101	技能唯一識別碼 (註 ¹)
2	74#45#54	欲使用之執行階段號 (註 ²)
3	殺戮狀態	人物實際狀態名稱

(註 ¹) : 技能唯一識別碼為每個技能之唯一編號字串，本人制定時共為 10 碼，開發者亦可自行使用自行編製之編號字串，但需要具唯一性，否則在遊戲執行時會造成錯誤。以下表格技能唯一識別碼之編號說明：

技能唯一識別碼編號說明

項目		說明
主分類	PNAK	主動技能代號
	PNPE	被動技能代號
	BUFF	異常狀態代號
	UCAS	人物實際狀態代號
技能從屬編號	N001	主動 / 被動技能：欲使用該技能之人物角色編號 異常狀態 / 人物實際狀態：個別效果體系編號 (001~999)
分項目號	01	編號自 01~99

(註²)：對於執行階段點的使用採登記制，如該技能需要使用到特定之執行階段點，需要在本欄設定上註記執行階段號，如有 2 個以上者，中間以「#」號間隔。

(註³)：若該技能有 2 個發動條件以上者，代碼與代碼中間以「&」相隔，至多 5 個。EX：發動條件為「槍 1、槍 2、槍 3」時，則代碼為「51b&52b&53b」。

(註⁴)：異常狀態之圖片路徑是以主程式所在資料夾作為開始之參照。

主架構包含程式之引數變數，登記之執行階段號以及效果程式碼之部分，而在主動技能方面則還包括 AI 判斷設計之內容。本章節主要討論主架構之大綱部分，有關於其他之各部分會在其他之章節再做說明，以下表格開始為主架構大綱部分及效果程式碼之說明：

(因版面狹小，部分程式碼會採取換行之寫法)

主架構效果程式碼說明 (主動技能)

(引用: 自創人物角色-蕾 LV5 (VBEID:S00105) 主動技能-輪旋曲-琉璃色的微風[PNAKS00101])

架構程式碼

```
Function atking(a, VBEPersonVS(), VBEPageCardNumVS(), AtkingPagetotVS(),  
VBEPersonBuffVVSF(), VBEPersonBuffVSS(), AtkingckVSS(), AtkingckVVSF(),  
VBEAtkingVVSF(), VBEAtkingVSS(), VBEActualStatusVS(), VBEVSSStageNum())  
    Select Case a  
        Case 42      '攻擊階段(出牌時)  
            If VBEAtkingVSS(6) = 1 Then      '當現在為近距離時  
                If AtkingPagetotVS(1, 1) >= 4 Then  
                    '如果出的劍卡總數值超過 4 時  
                        If AtkingckVSS(1, 1) = 0 Then      '當技能未啟動時  
                            VBECOMMAND.AtkingTurnOnOffWithLineLight 1  
                        End If  
                    Else  
                        If AtkingckVSS(1, 1) = 1 Then  
                            '當出牌條件未符合且技能已啟動時  
                                VBECOMMAND.AtkingTurnOnOffWithLineLight 2  
                            End If  
                        End If  
                    End If  
                End If  
            Case 45      '總骰數變化量影響  
                VBECOMMAND.EventTotalDiceChange 1, 1, 6  
                VBECOMMAND.EventTotalDiceChange 2, 2, 3  
            Case 11      '發動技能動畫  
                VBECOMMAND.AtkingStartPlay "\gif\Ria\atking1.png"  
                VBECOMMAND.AtkingTurnOnOffWithLineLight 2  
            End Select  
            atking = VBECOMMAND.Output()      '格式轉換器輸出指令字串  
End Function
```

主架構效果程式碼說明（被動技能）

（引用：官方人物角色-莉莉 LV1（ULID:40001、VBEID:N02701）被動技能-小公主[PNPEN02701]）

架構程式碼

```
Function passive(a, VBEPersonVS(), VBEPageCardNumVS(), AtkingPagetotVS(),  
VBEPersonBuffVSF(), VBEPersonBuffVSS(), AtkingckVSS(), AtkingckVSF(),  
VBEAtkingVSF(), VBEAtkingVSS(), VBEActualStatusVS(), PassivePersonType,  
VBEVSStageNum())  
    Select Case a  
        Case 0      '發牌階段  
            If VBEAtkingVSF(1, 1, 1) >= 5 And PassivePersonType = 1 Then  
                '如果自身血量大於5且為場上角色時  
                VBECommand.AtkingTurnOnOffWithLineLight 1  
            End If  
        Case 22      '攻擊/防禦階段最後執行點-其他骰數有關類型  
            If PassivePersonType = 1 And VBEAtkingVSS(12) = 4 Then  
                '如果現在為場上角色且為防禦階段時  
                Dim m  
                Randomize Timer  
                m = Int(Rnd() * 100) + 1      '機率取樣  
                If (AtkingckVSS(1, 1) = 1 And m >= 53 And m <= 92) Or _  
                    (AtkingckVSS(1, 1) = 0 And m Mod 5 = 0) Then  
                    '40%/20%之機率  
                    VBECommand.AtkingTrueDiceControl 3, 0  
                    VBECommand.BattleSendMessage "因為小公主的效果，傷害被無效了。"  
                End If  
            End If  
        Case 55      '回合結束階段  
            If PassivePersonType = 1 Then '如果現在為場上角色時  
                VBECommand.AtkingTurnOnOffWithLineLight 2  
            End If  
    End Select  
    passive = VBECommand.Output()      '格式轉換器輸出指令字串  
End Function
```

主架構效果程式碼說明 (異常狀態)

(引用:增益系異常狀態-ATK+[*BUFFN00101*])

架構程式碼

```
Function buff(a, atkingpagetotVS(), VBEAtkingVSF(), VBEAtkingVSS(),  
VBEVSBuffNum(), BuffPersonType, VBEVSSStageNum())  
    Select Case a  
        Case 45 '總骰數變化量影響  
            If BuffPersonType = 1 Then '如果附身之人物角色在場上的話  
                If VBEAtkingVSS(12) = 3 Then '如果現在為攻擊階段的話  
                    VBECommand.EventPersonAbilityDiceChange 1, 1, VBEVSBuffNum(1)  
                End If  
            End If  
        Case 53 '回合結束階段  
            If BuffPersonType = 1 Then '如果附身之人物角色在場上的話  
                VBECommand.BuffTurnEnd  
            End If  
        End Select  
        buff = VBECommand.Output() '格式轉換器輸出指令字串  
    End Function
```

主架構效果程式碼說明 (人物實際狀態)

(引用:官方人物角色-史塔夏 LV5 (ULID:335 、 VBEID:N00505) 人物實際狀態-殺戮模式[UCASN00101])

架構程式碼

```
Function ActualStatus(a, VBEPersonVS(), VBEPageCardNumVS(), AtkingPagetotVS(),  
VBEPersonBuffVSF(), VBEPersonBuffVSS(), VBEAtkingVSF(), VBEAtkingVSS(),  
ActualStatusPersonType, VBEVSSStageNum())  
    Select Case a  
        Case 74 '人物實際狀態-新增事件  
            '新增殺戮狀態立繪資料  
            VBECOMMAND.EventAddActualStatusData _  
                "\CharacterActualStatus\KillerMode\personb.png", _  
                "\CharacterActualStatus\KillerMode\personmini.png", _  
                "\CharacterActualStatus\KillerMode\personminidown.png", _  
                "\CharacterActualStatus\KillerMode\personf.png", _  
                -90, -60, 90, -60  
        Case 45 '總骰數變化量影響  
            If ActualStatusPersonType = 1 Then '如果現在為場上角色時  
                If VBEAtkingVSS(12) = 3 Then  
                    VBECOMMAND.EventTotalDiceChange 1, 3, 2  
                End If  
            End If  
        Case 54 '回合結束階段  
            If ActualStatusPersonType = 1 Then '如果現在為場上角色時  
                VBECOMMAND.ActualStatusEnd  
            End If  
    End Select  
    ActualStatus = VBECOMMAND.Output() '格式轉換器輸出指令字串  
End Function
```

8. 主動技能之 AI 計算判斷設計

自 Version 7 開始實裝之 AI 系統-智能判斷型人工智慧 AI-「夢」，以透過手牌之排列組合計算作為判斷出牌之依據，本章為說明 AI 系統之個別技能計算部分。

在 Type QS 中，AI 在計算卡牌的排列組合期望值之時，為了能更加貼切每個人物角色之需求，會徵求人物角色之各主動技能之意見，而各技能則依據各自之判斷，回傳其判斷結果給 AI 系統。

在實際運作之中，與技能之間之意見傳遞會透過執行階段系統-99 號來執行，與效果程式碼不同的是，在執行階段-99 之腳本引數有關卡牌變數部分，會直接套用當前該排列組合之計算資訊，而非實際在場上之情況，主動技能僅須直接就當前之資訊給予判斷即可。

以下為在執行階段-99 之中，技能所要回傳之指定指令內容：

執行階段-99 指定指令			
EventActiveAIScore	#n1,n2,[n3...]	主動技能	99
智能判斷型 AI 個別技能判斷評分			
(註 ¹ ：自引數 n3 開始為選擇性填寫性質，目的為讓技能去挑選具貢獻之卡牌，並回報其編號。)			
(註 ² ：本指令無法直接適用於格式轉換器之同名函數，若想利用格式轉換器者，請參照「4.執行指令集- II. 使用格式轉換器-新增/變更之指令函數」之說明)			
n1-(1.推薦/2.無意見/3.不推薦)			
n2-(1.一般型評分/2.骰數變化量優先型評分)			
[n3...]- (個別卡牌編號(VBEPAGECARDNUMVS 順序))			

對於一個卡牌排列組合，如果有任何主動技能「推薦」的話，將讓該組合的期望值獲得提高，而 AI 系統最終將會挑選出最高期望值的排列組合，當作出牌的依據。反之若有任何主動技能回報「不推薦」的話，將使該組合期望值評比降低，進而淘汰這些不適合的排列組合。

若主動技能並無納入 AI 計算判斷設計，或者是該組合被回報為「無意見」的話，AI 系統將不會對這些排列組合作任何期望值上的技能加成，但技能仍可以回報具有貢獻的卡牌編號，這依然會有因 AI 系統的篩選而被選中的可能性存在。

在判斷評分類型上分成一般型評分與骰數變化量優先型評分，主要不同在於期望值的加權方式，一般型評分只會做期望值固定加成，對於以骰量加成為主要效果之技能來說，一般型評分的加成結果是相同的，這時就可選擇後者之評分方式，因骰數變化量優先型評分著重於總骰數變化量之加成取向，AI 系統會參考該技能於執行階段-45 之骰量加成數據，來當作期望值加成的依據。

從另一個角度來說，個別技能判斷評分講求的是個人在出牌方面的經驗，這並沒有所謂的標準寫法，而一個好的判斷評分會讓玩家在對戰上面會有更佳的使用者體驗。以下為了能讓各位更加了解個別技能判斷評分的運作，本人將示範設計 Unlight 官方人物角色-雪莉 LV5 (ULID:105 , VBEID:N00105) 之主動技能「自殺傾向」作為這次的範例。

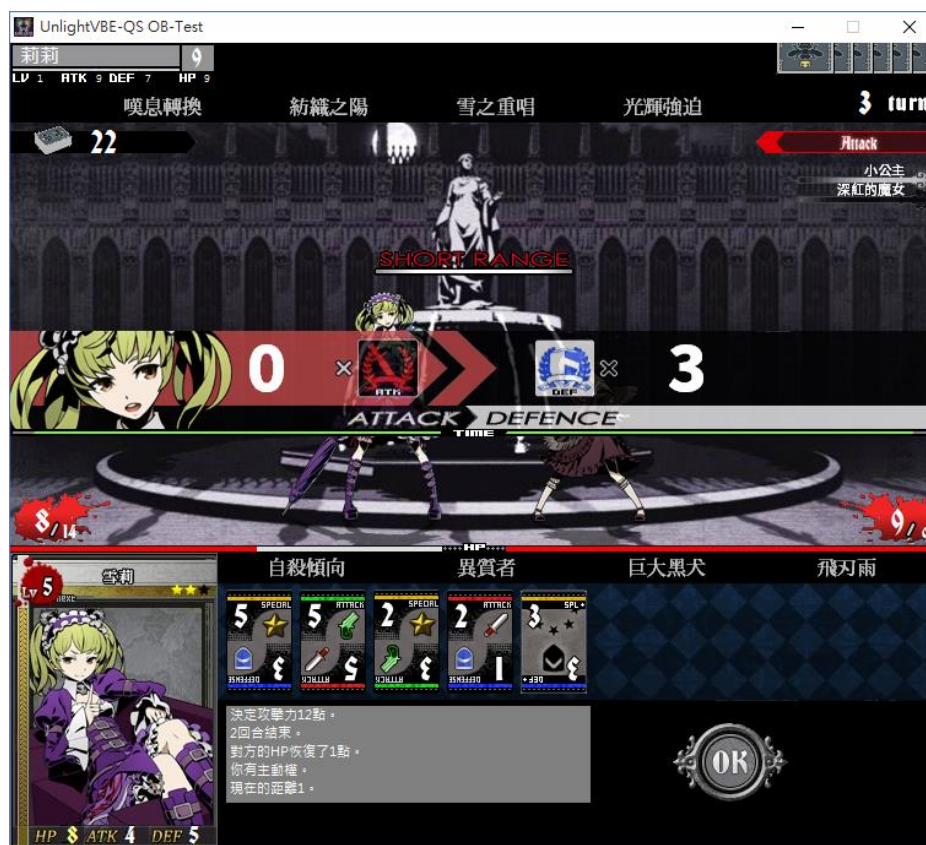
我們先來看一下該技能的基本資料：

雪莉 LV5-主動技能「自殺傾向」

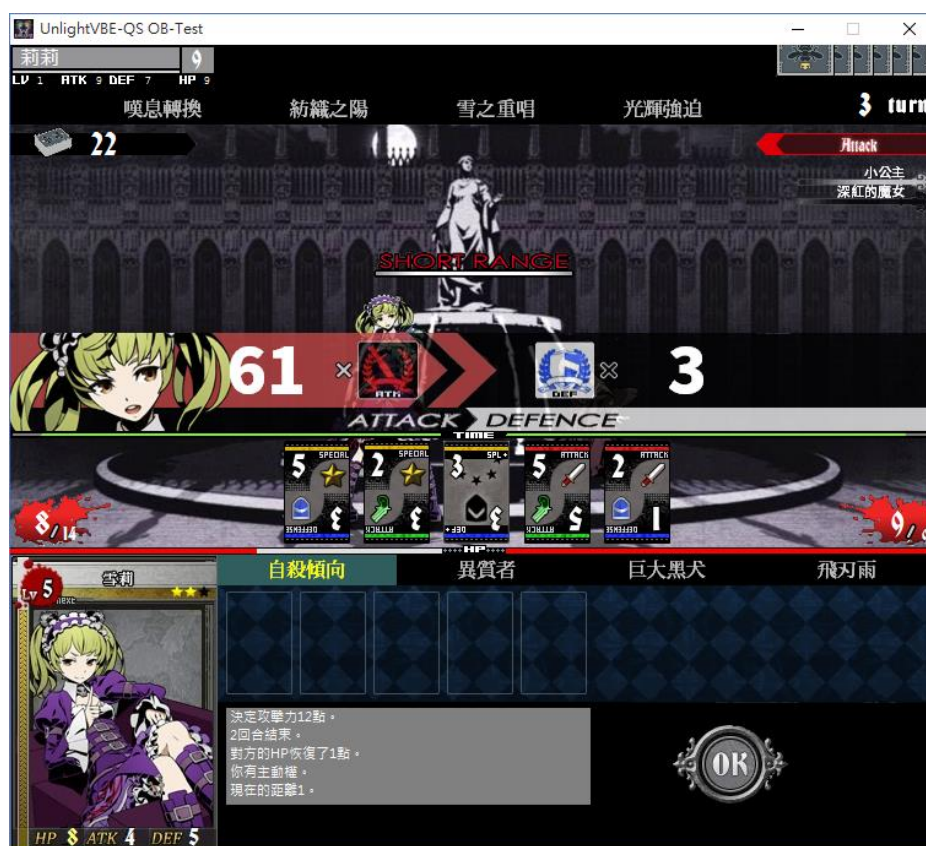
項目	內容
技能唯一識別碼	PNAKN00101
名稱	自殺傾向
階段	攻擊
距離	全距離
發動條件	特合計 1 以上
發動效果	HP-特，ATK+特 x5

自殺傾向是一個能夠創造高攻擊力的一項技能，雖然要以自身割血作為代價，但是當能夠一次累積相當數量的特卡之時，其對對手之破壞力就不在話下了。

接下來我們加入一些戰鬥的條件假設如圖(8-1)所示，雪莉在第 3 回合近距離下先攻，在剩餘 8 滴血的情況下，要對圖片中之手牌作出牌的判斷選擇。如果以技能最單純的目的，也就是創造高攻擊力去想的話，最理想的情況就是把所有的特卡都出出去，如圖(8-2)所示，這種判斷方法是否可行呢？



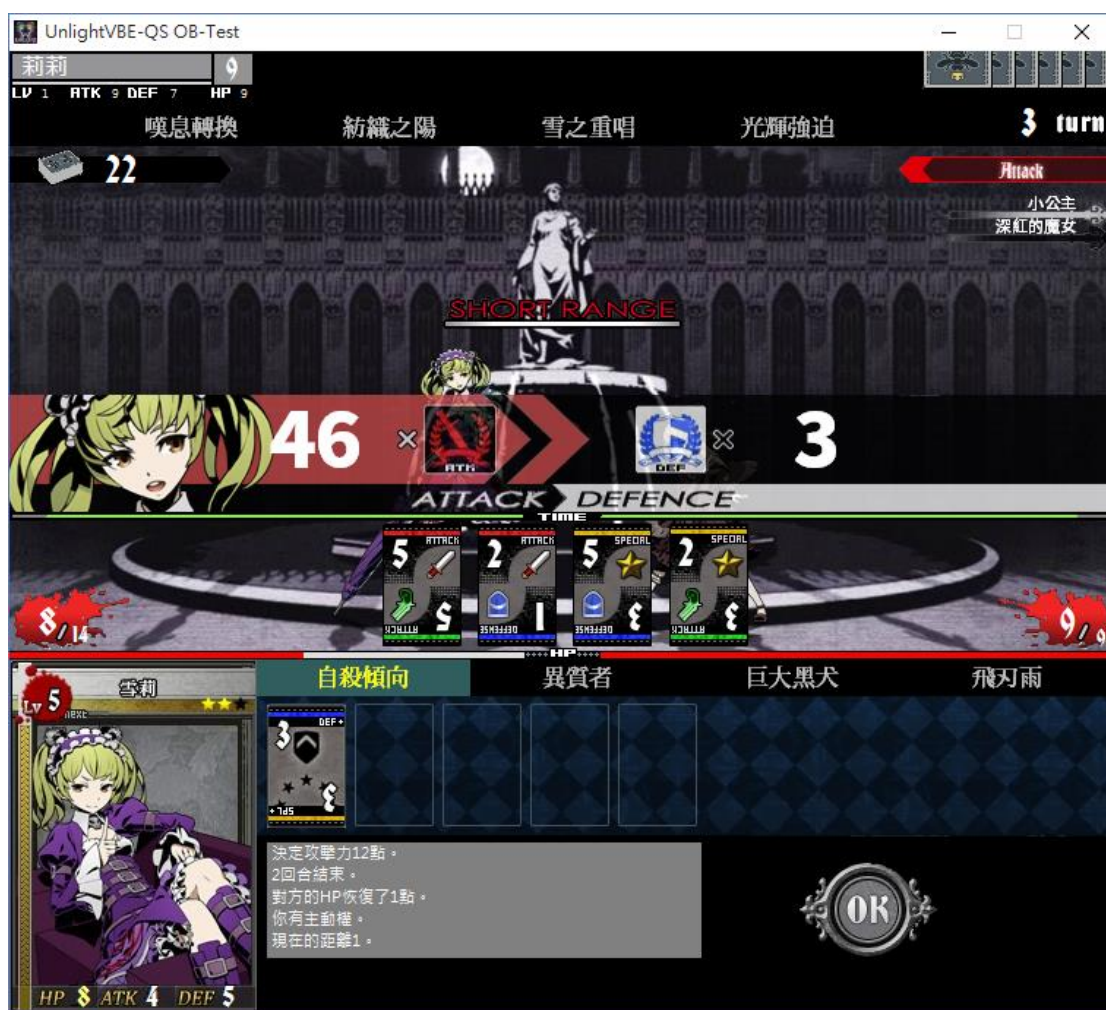
圖(8-1)



圖(8-2)

答案我想各位都很清楚，這種出牌可謂真正的「自殺」，根本毫不可行，但 Unlight 官方曾經有一段時間的 PVP-AI 真的是這種判斷方式。

要避免這樣的「自殺」攻擊，我們需要在針對特卡的數量來下更多判斷，當像圖(8-2)這種已經會危害到人物角色生命的這種出牌排列組合，我們應當對 AI 系統作出「不推薦」的一個判斷評分，反之在適當的特卡數量之出牌排列組合下，我們就能夠向 AI 系統進行「推薦」了。圖(8-3)是本人覺得最適當的出牌排列組合。



圖(8-3)

綜合以上的思考，最後我們要將這種判斷實行在腳本之中，以下為原效果程式碼加上 AI 判斷設計程式碼之主架構說明：

主架構-效果&AI 判斷設計程式碼說明 (主動技能)

(引用:官方人物角色-雪莉 LV5 (ULID:105 · VBEID:N00105) 主動技能-自殺傾向)

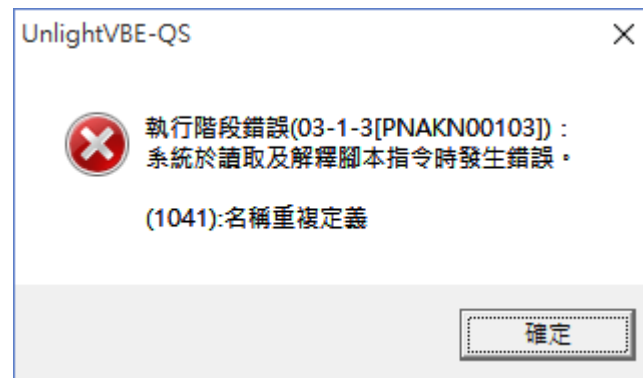
架構程式碼

```
Function atking(a, VBEPersonVS(), VBEPageCardNumVS(), AtkingPagetotVS(),  
VBEPersonBuffVSF(), VBEPersonBuffVSS(), AtkingckVSS(), AtkingckVSF(),  
VBEAtkingVSF(), VBEAtkingVSS(), VBEActualStatusVS(), VBEVSStageNum())  
    Select Case a  
        Case 42      ' 攻擊階段 ( 出牌時 )  
            If AtkingPagetotVS(1, 4) >= 1 Then      ' 如果出的特卡總數值超過 1 時  
                If AtkingckVSS(1, 1) = 0 Then      ' 當技能未啟動時  
                    VBECommand.AtkingTurnOnOffWithLineLight 1  
                End If  
            Else  
                If AtkingckVSS(1, 1) = 1 Then      ' 當出牌條件未符合且技能已啟動時  
                    VBECommand.AtkingTurnOnOffWithLineLight 2  
                End If  
            End If  
        Case 45      ' 總骰數變化量影響  
            VBECommand.EventTotalDiceChange 1, 1, AtkingPagetotVS(1, 4) * 5  
            ' 增加出牌之特總數*5 之總骰量  
        Case 11      ' 發動技能動畫  
            VBECommand.AtkingStartPlay "\gif\Sheri\atking1.png"  
            VBECommand.AtkingTurnOnOffWithLineLight 2  
        Case 61      ' 技能動畫中效果實施  
            VBECommand.PersonBloodControl 1, 1, 1, AtkingPagetotVS(1, 4)  
        Case 99      ' AI 判斷設計  
            If VBEAtkingVSS(12) = 3 And VBEAtkingVSS(13) = 1 Then  
                ' 如果現在為攻擊階段/我方攻擊時  
                If AtkingPagetotVS(1, 4) >= 1 And _  
                    (VBEAtkingVSF(1, 1, 1) - AtkingPagetotVS(1, 4)) > 0 Then  
                    ' 如果該排列組合之特數量超過 1，且自身之目前血量扣除特數量仍大於 0 時  
                    VBECommand.EventActiveAIScore 1, 2  
                    ' 判斷為「推薦」之排列組合、評分加權為骰數變化量優先型評分  
                    For i = 1 To UBound(VBEPageCardNumVS, 1)  
                        If VBEPageCardNumVS(i, 1) = 4 Then  
                            ' 在排列組合的牌中，如果其中類型為特卡時  
                            VBECommand.EventActiveAIScoreAddCardNum i  
                            ' 標記為具貢獻之卡牌  
                        End If  
                    Next  
                ElseIf (VBEAtkingVSF(1, 1, 1) - AtkingPagetotVS(1, 4)) <= 0 Then  
                    ' 但如果該排列組合會造成自己「自殺」時  
                    VBECommand.EventActiveAIScore 3, 2  
                    ' 判斷為「不推薦」之排列組合  
                End If  
            End If  
        End Select  
        atking = VBECommand.Output()      ' 格式轉換器輸出指令字串  
    End Function
```

9. 腳本之錯誤處理

當腳本愈寫愈大之際，難免的裡面一定會有所謂的臭蟲 (Bug) 出現，雖然本系統不能像一般程式設計軟體一樣，具對每一行之逐步偵錯之功能，但我們還是有提供一些錯誤訊息給各位做參考。

當遊戲系統發生錯誤時，在執行階段系統以及執行指令集系統有提供錯誤訊息之顯示，如圖(9-1)。前面 2 行為遊戲系統之錯誤顯示訊息，而最後一行則為 VB6 自身所顯示之錯誤訊息。



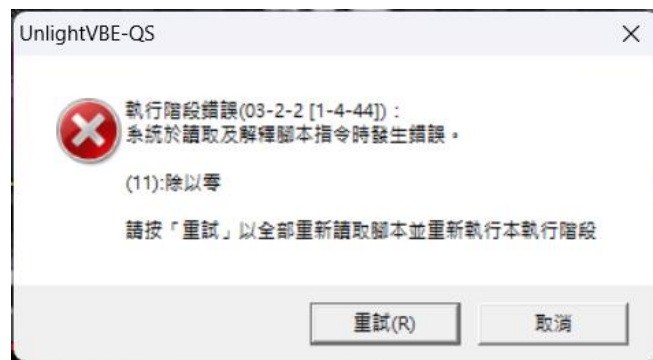
圖(9-1)

以下表格為系統之錯誤編號說明：

遊戲系統錯誤編號表

類別	主編號	次編號	詳細說明
執行階段系統 (03)	03-1-1	-	異常狀態類腳本搜尋錯誤
	03-1-2	-	Microsoft Script Control 物件創立錯誤
	03-1-3	[<技能唯一識別碼>]	主動技能/被動技能類 腳本讀入錯誤(AddCode)
	03-1-3	[<腳本檔案路徑>]	人物實際狀態/異常狀態類 腳本讀入錯誤(AddCode)
	03-1-4	[<1.使用者方/2.電腦方> -<人物技能編號>]	主動技能/被動技能類 初始資料讀入錯誤(初始架構)
執行階段系統 (03)	03-1-5	[<腳本檔案路徑>]	異常狀態類 初始資料讀入錯誤(初始架構)
	03-1-6	-	人物實際狀態類腳本搜尋錯誤
	03-1-7	[<腳本檔案路徑>]	人物實際狀態類 初始資料讀入錯誤(初始架構)
	03-2-1	[<1.使用者方/2.電腦方> -<執行階段號> -<技能唯一識別碼>]	執行階段系統 執行技能驗證時錯誤
	03-2-2	[1 -<人物技能編號> -<執行階段號>]	主動技能類 執行腳本時錯誤(主架構)
	03-2-2	[2 -<人物技能編號> -<執行階段號>]	被動技能類 執行腳本時錯誤(主架構)
	03-2-2	[3 -<技能唯一識別碼> -<執行階段號>]	異常狀態類 執行腳本時錯誤(主架構)
	03-2-2	[4 -<技能唯一識別碼> -<執行階段號>]	人物實際狀態類 執行腳本時錯誤(主架構)
執行指令集系統 (04)	04	-<執行階段號> -<執行指令名稱> -<執行指令集分階段號>	執行指令集系統 指令執行錯誤
	04	-Run-CommadNotFound [執行指令名稱]-0 -<執行指令集分階段號>	執行指令集系統 未找到該指令之錯誤

在腳本中的大部分的語法錯誤都應會在(03-1-3)號錯誤，也就是腳本物件在 **AddCode** 的時候就會被發現，有些錯誤則會遊戲執行中的時候才會發生；若該錯誤是在「執行腳本時錯誤(主架構)」(03-2-2)範圍內的話，您可以藉由啟動測試模式(**Admin**)來開啟重新載入全部腳本檔案的功能(如圖(9-2))；但須注意的是，此動作亦將重設所有在腳本內部設定之自定義變數，於測試時還請留意。



圖(9-2)

10. 附錄 1-人物角色之圖片資源概要

本章節將列舉一位人物角色在遊戲系統中所需要之圖片資源，來讓各

位做為製作角色時之參考：

人物角色圖片資源表

(引用:自創人物角色-蕾 (VBEID:5001))

項目	圖片範例	格式
角色卡片		PNG/JPG [須為(168*240)px] (註：因系統已自動標上人物白值，如圖片上含有白值數字者請將白值數字部分去除。)
大人物立繪		PNG
小人物立繪		PNG
小人物立繪影子		PNG (註：影子須加上半透明效果約 50%)

人物角色圖片資源表

(引用: 自創人物角色-蕾 (VBEID: S001))

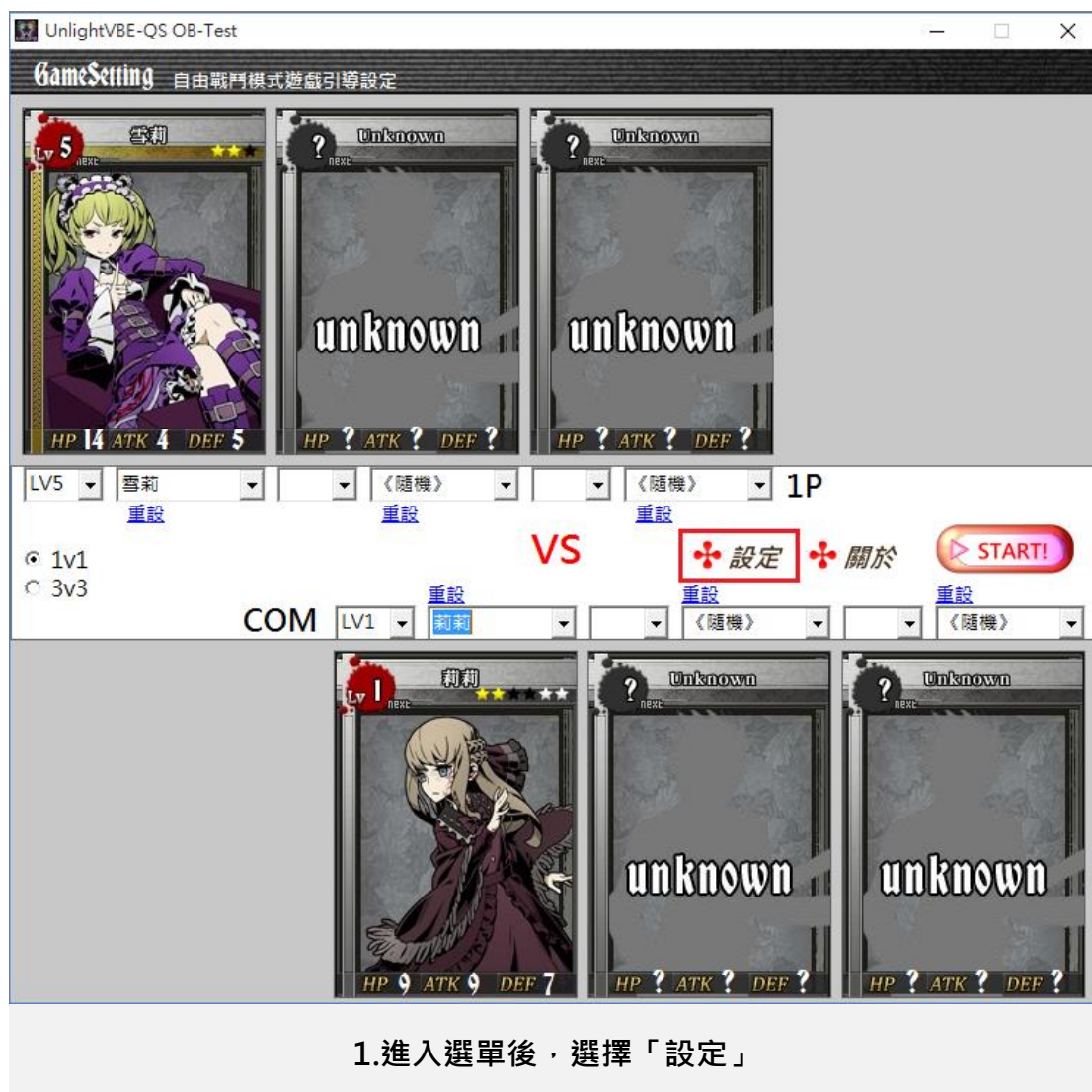
項目	圖片範例	格式
小人物頭像		<p>PNG</p> <p>[須為(248*80)px]</p> <p>(註: 於人物頭像外為全透明效果)</p>
各技能圖		<p>PNG</p> <p>(註¹: 靜態與逐格動畫皆適用之)</p> <p>(註²: 高度 match 為 660px)</p>

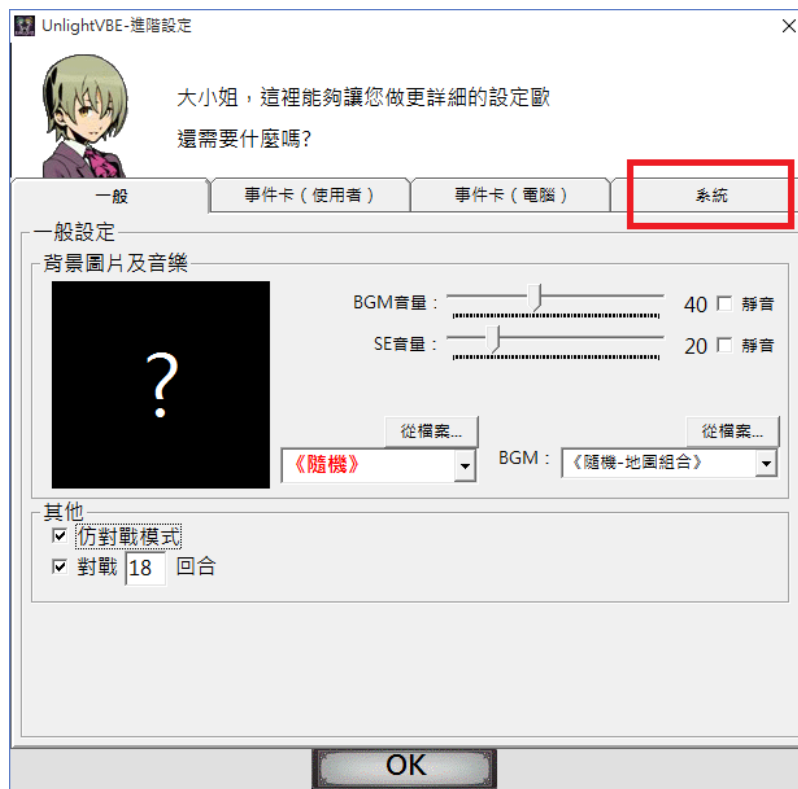
11. 附錄 2-人物角色之小人物立繪影子座標取得操作

目前對於小人物立繪之影子座標指定只能以手動方式輸入，本章節將教學如何取得影子座標之數據。

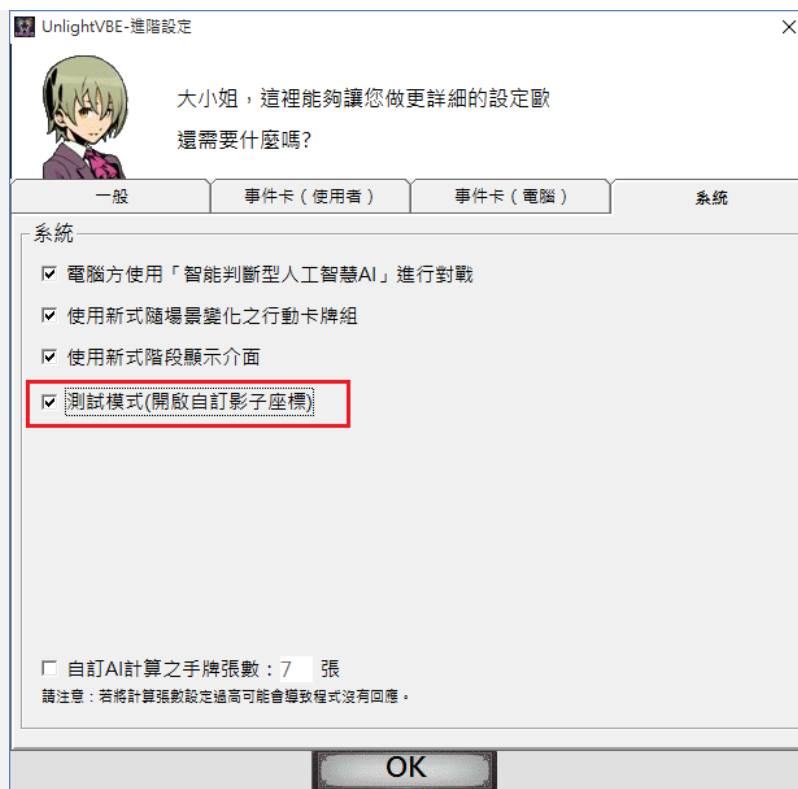
(實際之操作介面可能會根據您遊玩之版本不同而有所差異)

小人物立繪影子座標取得說明





2.選擇「系統」標籤



3.選擇「測試模式(開啟自訂影子座標)」



4.在進入戰鬥後，選擇右下方之「影子設定」按鈕



5.可操作 4 向按鈕使影子移動，當與小人物立繪相吻合時即可抄下座標。

L：影子朝左邊移動 10 單位 R：影子朝右邊移動 10 單位
 Tup：影子朝上方移動 10 單位 Tdn：影子朝下方移動 10 單位
 1P：使用者方 COM：電腦方