# 概述：

有参考外网 也有一些我自己的理解

一些用不到的或是我还没能理解的就没放在教程里

有什么地方写的不对可以提出来 我会修正

-------------------------------------------------------------------------------------------------

完整框架为：

<mamecheat version="1">

<cheat desc="作弊项描述文本"> <!-- 文档注释 -->

<comment>作弊菜单可见注释文本</comment>

<parameter min="最小值" max="最大值" step="加减间隔">

<item value="值">选项描述文本</item>

</parameter>

<script state="运行方式">

<action condition="条件">命令</action>

</script>

</cheat>

</mamecheat>

若只是简单的锁定数值，这样写就够了：

<mamecheat version="1">

<cheat desc="作弊项描述文本">

<script state="运行方式">

<action condition="条件">命令</action>

</script>

</cheat>

</mamecheat>

<commment>和<parameter>、<item>以及<!-- -->都是可选标签，其余为必须标签

-------------------------------------------------------------------------------------------------

# <mamecheat> 表示此xml文件内置命令及参数为mame专有，

version=”1”为默认参数也只能这样写，非必须，但还是有必要加上

# <cheat> 里的desc后内容为作弊项描述文本，显示在作弊菜单里，如

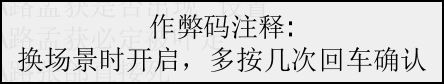
<cheat desc=”通用 - 全关卡”></cheat>，在作弊菜单里就会显示为



# <comment> 标签中的内容为作弊菜单可见注释文本，如

<comment>换场景时开启，多按几次回车确认</comment>，

当把光标移动到写了这一条的作弊项时，下方就会出现



# <parameter> 有两种写法：

① 增量参数 (常用于选关或选场景)

<parameter min=”” max=”” step=”” />

标签里的参数min为最小值，max为最大值，step为增加间隔，

如写1，每增加一次就是增加1，写2，每次就增加或减少2，

完整写法参考下面，大同小异，只是运行方式不同，

若是run，则切换时立刻变化，若是change，则切换后按回车才生效

② 选项参数

<parameter>

<item value=””></item>

</parameter>

<item>标签里value的值可以是十进制也可以是十六进制，

若写十六进制，必须以0x开头，如0x0000，

标签中间的内容为选项显示文本，若使用<parameter>标签，

想要其生效的指令赋值必须=param，如

maincpu.pb@地址=数值 要写成 maincpu.pb@地址=param

param代表<item>标签里value的值

且<item>标签可以写无数个，会按顺序排列，可按依次切换，如

<cheat desc="通用 - 时间 设置">

<parameter>

<item value="0x64">无限</item>

<item value="0x00">置0</item>

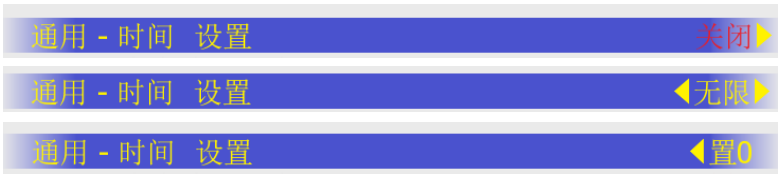
</parameter>

<script state="run">

<action>maincpu.pb@200201BE=param</action>

</script>

</cheat>



这样的写法即为持续锁定时间，开启后会一直往存储时间的地址里写值，

而如果我只想让操作时间的语句执行一次，就这样写，

<cheat desc="通用 - 时间 设置">

<parameter>

<item value="0x64">无限</item>

<item value="0x00">置0</item>

</parameter>

<script state="change">

<action>maincpu.pb@200201BE=param</action>

</script>

</cheat>

意为每当在选项上按回车，就会写入对应的值，若不按则不会执行

# <script> 标签唯一的存在意义就是其参数的运行方式，

有run | change | on | off，其中

run为持续执行，

change为选项列表单次执行，

on为单次执行，

off为关闭作弊项时执行的操作

详情看后文实例

# <action>标签为执行的命令，参数有condition，即条件判断，

比如满血量100，而我想让血量小于等于50的时候自动把血加满，可以这样写，

十六进制：

<script state=”run”>

<action condition=”maincpu.pb@血量地址 le 32”>maincpu.pb@血量地址=64</action>

</script>

十进制：

<script state=”run”>

<action condition=”maincpu.pb@血量地址 le #50”>maincpu.pb@血量地址=#100</action>

</script>

十进制和十六进制都可行，只不过默认是识别为十六进制，

十进制要在数值前加#才能被识别为十进制，不过读取后还是会转换成十六进制

此处的 le 可以是

== 等于 | != 不等于 | gt 大于 | lt 小于 | ge 大于等于 | le 小于等于

也可以同时写多个条件，如

<action condition=”条件1 AND 条件2 OR 条件3”></action>

条件可以用()包起来，如

(maincpu....) AND (maincpu....) OR (maincpu....)

后面的指令可用 , (英文逗号) 隔开多写，以实现一条判断同时执行多个命令，如

<action condition=”....”>maincpu.... , maincpu.... , maincpu....</action>

# <!-- --> 标签为文档注释，写在其内的文本不会影响作弊文件及作弊系统，

只是写给自己或是别人看的注释文字，

需要注意的是注释标签内不要再包含 < 或 > 之类的符号，会导致作弊系统报错崩溃

# temp0 ~ temp9 为10个缓存变量，相当于寄存器，默认值为0，

亦可给temp缓存变量赋值，独立于每个作弊项，

也就是说在锁定时间里使用了temp0且赋值1，不会对锁定血量里的temp0有任何影响，

当作弊项关闭后不会自动置0，只有在作弊菜单内重新载入后会自动置0

再附几个实例：

1 持续锁定数值

<cheat desc="通用 - 无限物品">

<script state="run">

<action>maincpu.pb@20060120=09</action> <!-- 1P -->

<action>maincpu.pb@200603D0=09</action> <!-- 2P -->

<action>maincpu.pb@20060680=09</action> <!-- 3P -->

<action>maincpu.pb@20060930=09</action> <!-- 4P -->

</script>

</cheat>

当往右切换到开启则持续锁定数值

2 单次选项菜单生效

<cheat desc="1P A面物品1道具选择">

<parameter>

<item value="0xA0">冰箭术</item>

<item value="0xA1">钻石尘</item>

<item value="0xA2">阎王的吐息</item>

</parameter>

<script state="change">

<action>maincpu.pb@2006011F=param</action>

</cheat>

当切换到对应的选项并按下回车则单次写入数值

3 判断选项执行对应命令

<cheat desc="1P A面物品1道具选择">

<parameter>

<item value="1">冰箭术</item>

<item value="2">钻石尘</item>

<item value="3">阎王的吐息</item>

</parameter>

<script state="change">

<action condition=”param==1”>maincpu.pb@2006011F=A0</action>

<action condition=”param==2”>maincpu.pb@2006011F=A1</action>

<action condition=”param==3”>maincpu.pb@2006011F=A2</action>

</cheat>

方便举栗，直接拿上面的代码改了下，能理解就行，这样写可以判断执行一堆指令

4 单次生效

<cheat desc="1P 获得满级经验">

<script state="on">

<action>maincpu.pd@2005F768=20FFFFFF</action> <!-- 1P -->

</script>

</cheat>

未使用<parameter>标签但又想回车单次生效的写法

5 关卡选择

<cheat desc="关卡选择">

<parameter min="1" max="F" step="1"/>

<script state="change">

<action>maincpu.pb@200201BA=param</action>

</script>

</cheat>

当切换到对应的选项并按下回车则单次写入数值，且最小选项为1，最大选项为F，

每次增加或减少值为1

6 ROM汇编地址修改及还原1

<cheat desc="通用 - 对敌强制破招">

<script state="run">

<action>maincpu.rw@1000D114=0000</action>

</script>

<script state="off">

<action>maincpu.rw@1000D114=D105</action>

</script>

</cheat>

当开启时修改ROM地址的值，关闭时还原ROM地址，off常用于还原值

7 ROM汇编地址修改及还原2

<cheat desc="通用 - 对敌强制破招">

<script state="on">

<action>temp0 = maincpu.rw@1000D114</action>

</script>

<script state="run">

<action>maincpu.rw@1000D114=0000</action>

</script>

<script state="off">

<action>maincpu.rw@1000D114=temp0</action>

</script>

</cheat>

当开启时会同时执行on的指令，因为on的指令是开启时触发，而回车和run一样都是开启，存储地址的值到缓存变量temp0，并修改ROM地址的值，

当关闭时从temp0读取并还原ROM地址的值

8 相对值引用

maincpu.pb@2005F69E+=1 等效于 maincpu.pb@2005F69E=maincpu.pb@2005F69E+1

+= 可以是 -= | \*= | /= | %= 等等

9 如神剑伏魔录的指令写法

<cheat desc="1P 角色无敌">

<script state="run">

<action>prot.pb@18014C46=01</action>

</script>

<script state="off">

<action>prot.pb@18014C46=00</action>

</script>

</cheat>

因为神剑伏魔录的地址是存在prot里的，所以不论指令是操作RAM还是ROM都要用prot来引用地址，当然了，大部分游戏都是maincpu

10 引用帧值

<action condition=”frame % 5==0”>maincpu.pb@血量地址=#100</action>

即每5帧加满血，也可用于后面指令，如

<action>maincpu.pb@血量地址=frame%5</action>

但这样作用和意义就完全不同了

注意事项：

1 不论是条件还是值引用都必须要加maincpu.pb@，并非一定是maincpu.pb，只是举例，

因为maincpu代表地址，后面@的只是偏移地址

2 更多的比较符号和运算符号可以在调试器里键入 help exporession 查看

3 若是写了off的script指令，

在作弊菜单内重置或重新载入或快捷键关闭作弊都会自动执行off里的指令