

CONTENTS

KCDK について	1
導入方法	2
本体の説明	3
本体の操作方法	4
コマンドの設定	4
スコアボードの説明	7
PTP (Pass To Plugin)	9
その他	9
トラブルシューティング	10
コマンドチェックリスト	11

※本説明ではバージョン表記を2.0として進めます。
使用するバージョンに合わせて適宜読み替えてください。

本説明で使用したシンタックスハイライト: Marshal Command Code

KCDK について

KCDK / Kojosen Command Development Kit / 攻城戦コマンド開発キットは
攻城戦(攻守に分かれて戦うゲーム)を簡単に作成する目的で作られたコマンドセットです。

- 1～4チームに対応しています。1～4チーム(青 緑 赤 黄)でできる全てのチームの組み合わせに対応しています。
- 多様な試合形式に対応しているため、可変性と拡張性が高いのが特徴です。

導入方法

1. 「KCDK 2.0.zip」を minecraft/saves/<world>/datapacks/ >に入れてください。
zipファイルの解凍はしなくてもかまいません。

※データパックは<データパック名/pack.mcmeta>というフォルダ構造でないと読み込めません。GitHubからダウンロードした場合<データパック名.zip/データパック名/pack.mcmeta>となっていますので注意してください。

tips: フォルダの構造さえ正しければzipファイルでも読み込むことができます。なので<データパック名.zip/pack.mcmeta>でも問題ありません。ただしフォルダ内に日本語名のファイルがある場合は、zip形式で読み込むとエラーになりますのでご注意ください。

2. セットアップを行うために以下のコマンド(1)を実行してください。セットアップはKCDKを使用するための前準備です。設定の初期化¹をスキップして最低限必要なセットアップだけを行いたい場合はコマンド(2)を実行してください。

セットアップの詳細については minecraft/data/kcdk/functions/ >の「setup.mcffunction」または「simplified_setup.mcffunction」をご覧ください。

```
1. function kcdk:setup (1) セットアップ (+初期化)
2. function kcdk:simplified_setup (2) 簡易セットアップ (初期化なし)
```

- **キットの入手:**
ストラクチャブロックのストラクチャ名に「kcdk:2.0」と入力するか以下のコマンドを実行することでキットを入手することができます。

繰り返しになりますが、本説明ではバージョン表記を2.0として進めます。
使用するバージョンに合わせて適宜読み替えてください。

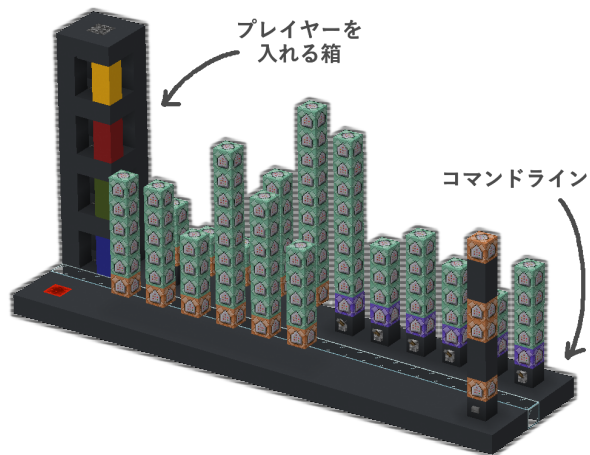
```
1. function kcdk:get
```

- **拡張機能の導入方法:**
拡張機能がzip形式の場合は解凍し minecraft/data/kcdk/functions/plugin/ >に拡張機能のフォルダを入れてください。使い方や設定方法、正確な導入方法は拡張機能フォルダ内の「_README.txt」ファイルをご覧ください。

KCDK を1つのワールドに複数導入する際「KCDK Manager」も導入すると KCDKバージョンの競合防止に役立ちます。KCDK がいくつ有効になっているか調べてくれるデータパックです。機能がそれだけなのと、KCDKを複数導入する機会がそもそも無いので、この機能が役立つことはあまり無いかもしれませんが、せっかく作ったので念のため。

¹ 設定の初期化:「難易度」「ゲームルール」「ワールドボーダー」「ワールドのスポン地点」等の設定をデフォルトの状態にします。

本体の説明



- ストラクチャブロックから KCDK の本体を読み込みましょう。
左の図が KCDK の本体です。この基板上でコマンドの設定と実行を行います。

- **プレイヤーケージ:**
図の左、直方体状の構造物はプレイヤーを待機させておくためのケージです。デフォルトでは各チームのプレイヤーを同じ色の箱の中に配置するように設定しています。(赤チームは赤色の箱に入れられる)プレイヤーの配置方法については『コマンドの設定』で説明します。

- **コマンドライン:**
ガラスブロックで区切られた長方形の土台のことを、コマンドラインと呼んでいます。奥のラインにはシステム関連のコマンドが配置されていて、手前のラインにはプレイヤー関連のコマンドが配置されています。

tips: 複雑なコマンドを組んだり規模の大きいゲームを作ると、コマンドの量も増えていきコマンドを置くスペースが無くなっていきます。そんな時はこのラインを増やしましょう。ラインは奥より手前に増やしたほうがキレイに見えます。



- **コマンドの後ろの看板:**
コマンドブロックの後ろに、上の図のような看板が設置されていると思います。これは中のコマンドがどのフェーズに実行されるかを示しています。フェーズはゲームの進行状況と同じ意味です。
表示が「phase:2」なら「フェーズが2の時に実行される」という意味です。「^1」の表示は「上のコマンドも同じフェーズに実行される」という意味です。フェーズの詳細については『スコアボードの説明』で説明します。
- **コマンドタワー:**
縦に積み上げられたコマンドブロックの連なりを、コマンドタワーと呼んでいます。コマンドタワー正面の根本に付いている看板で処理の概要(名前)を確認できます。

本体の操作方法

※『～』はコマンドタワーの名前を指しています。「～」は名前以外の看板を指しています。ゲームを開始する前にリポートコマンドブロックのレバーを下げてください。

ゲームの開始	『start/end game』のボタンを押します。
ゲームの終了	『start/end game』のボタンをもう一度押します。

チームの割り振り	『assign teams』のボタンを押します。
チームの入れ替え	『change teams』のボタンを押します。
チームを空にする	『empty teams』のボタンを押します。

ゲーム開始前のプレイヤーの配置:

『set players』を手動実行²します。ゲーム開始時のプレイヤーの配置は自動で実行されます。

キットの削除:

『remove kit』を実行して『click to remove』の看板を右クリックします。

※デフォルトでは全チームが無効に設定されています。チームの有効・無効の設定はスコアボードの説明で解説しています。

コマンドの設定

この章では以下の項目の設定方法を解説します。

1. 制限時間
2. 勝利条件
3. プレイヤーの配置
4. 死亡時の処理
5. アイテムの設定
6. エフェクトの設定
7. 役職の設定
8. 勝利時のsubtitle表示

※『～』はコマンドタワーの名前を指しています。「～」は名前以外の看板を指しています。フェーズについては次の章の『スコアボードの説明』で説明します。

² 手動実行: コマンドタワーに直接信号を与えて実行すること。コマンドタワーの真下にレッドストーンブロックを設置することを指す。

1. 制限時間の設定:

『set timer』の「timer_s」「timer_m」に秒分を入力してください。この設定を省略することで自動的に経過時間が表示されるようになります。

2. 勝利条件の設定:

『victory conditions』に勝利条件を記述してください。

条件には必ず `if score phase kcdk.control matches 2` を含めてください。

`scoreboard players set victory kcdk.control <数字>` で勝利を判定します。
1が青、2が緑、3が赤、4が黄色の勝利判定です。

```
1. execute if score phase kcdk.control matches 2 unless block ~ ~  
~ minecraft:beacon run scoreboard players set victory  
kcdk.control 1
```

解説:座標 ~ ~ ~ のブロックがビーコンでない場合、青チームを勝利させます。

3. プレイヤーの配置:

『set deploy』の `as @a[team=<team>] positioned <xyz座標>` に、配置する座標を入力してください。「<team> deploy in advance」がゲーム開始前の設定で、「<team> deploy」がゲーム開始時の設定です。間違えないように注意してください。

コマンド内で呼び出している「set_deploy.mcfuction」は tp と spawnpoint を併せた関数です。tp または spawnpoint だけを設定したい場合は `function ...` の部分を書き換えてください。

```
1. execute if score phase kcdk.control matches 0 as @a[team=blue]  
positioned 10 4 20 run function kcdk:player/set_deploy
```

解説:青チームを 10, 4, 20 に配置します。

4. 死亡時の処理:

『death』で死亡したプレイヤーをスペクテーターモード(観戦)に変更させます。

`scores={kcdk.death=<死亡回数>..}` で観戦までの死亡回数を調整できます。

スペクテーターモードへの変更を無効化する場合はコマンドブロック自体を破壊するか、コマンドの末尾に「!」など、構文に反する文字を差し込んでください。

```
1. execute if score phase kcdk.control matches 2 as  
@a[team=blue,scores={kcdk.death=5..}] run function  
kcdk:player/spectate
```

解説:5回死亡した青チームのプレイヤーを観戦させます。

5. アイテムの設定:

『set items』でゲーム開始時に配布されるアイテムを設定します。
ゲーム開始前に配布するアイテムは『set general settings』に設定してください。

```
1. give @a[scores={kcdk.role=0..}] minecraft:stone_sword
```

解説: kcdk.role 0 以上のプレイヤーに石の剣を与えます。※『set items』の処理。

```
2. execute if score phase kcdk.control matches 0 run give @a[scores={kcdk.role=0..}] minecraft:cooked_beef 64
```

解説: kcdk.role 0 以上のプレイヤーにステーキを与えます。※『set general settings』の処理。

6. エフェクトの設定:

『set effects』でゲーム開始前、ゲーム開始時およびゲーム終了時に付与または除去するエフェクトを設定します。

```
1. execute if score phase kcdk.control matches 3 run effect give @a[scores={kcdk.role=0..}] minecraft:resistance 120 4 true
```

解説: ゲーム終了時に kcdk.role 0 以上のプレイヤーへ耐性エフェクトを付与します。

7. 役職の設定:

『set role』で役職を設定します。役職は配布するアイテムをグループごとに分けたり、一部のプレイヤーに特殊効果を付与するときなどに使用します。

```
1. scoreboard players set @a[team=blue] kcdk.role 0
```

解説: 青チームの役職を 0 に設定します。

8. 勝利時の subtitle 表示:

『announce start/end』の「<team> subtitle」に任意のテキストを入力してください。

```
1. execute if score victory kcdk.control matches 1 run title @a subtitle [ "", {"text": "test"} ]
```

解説: 青チームが勝利したときに「test」と表示させます。

tips: デフォルトではゲーム中のゲームモードはアドベンチャーモードに設定されています。ゲームモードの設定コマンドは『set general settings』に配置されています。

スコアボードの説明

※『～』は『スコアボードオブジェクト名:変数名』を指しています。〈～〉はディレクトリを指しています。

『kcdk.control:phase』

ゲームの進行状況を管理しています。

phase: 0	ゲーム開始前
phase: 1	カウントダウン
phase: 2	ゲーム中
phase: 3	ゲーム終了

『kcdk.control:teams』

チームの有効・無効を管理しています。デフォルトで「0000」が設定されています。

teams: 1000	青チームが有効
teams: 0100	緑チームが有効
teams: 0010	赤チームが有効
teams: 0001	黄色チームが有効
teams: 1010	赤と青チームが有効

『kcdk.button』

triggerオブジェクトです。0番から29番までを一般的な使用のために予約しています。したがって、基本機能及び拡張機能が上記の番号を利用することはできないものとします。

『kcdk.config』

キット内拡張機能の設定情報が格納されています。仕様や設定方法は各拡張機能フォルダ内の「_README.txt」ファイルをご覧ください。

『kcdk.control:victory』

勝利の判定をしています。

victory: 1	青チームの勝利
victory: 2	緑チームの勝利
victory: 3	赤チームの勝利
victory: 4	黄色チームの勝利
victory: 5..	その他の章に記載

『kcdk.control:time / timer』

経過時間および制限時間を管理しています。変数末尾の「_t」「_s」「_m」はそれぞれtick、秒、分の意味です。

『kcdk.control:version』

Minecraftのバージョンを格納しています。特定できるバージョンはメジャーバージョンまでです。パッチバージョンの特定には対応していません。

『kcdk.role』

プレイヤーの役職を設定します。役職の設定方法は『コマンドの設定』の『役職の設定』をご覧ください。

『kcdk.death』

deathCountオブジェクトです。プレイヤーの死亡回数を管理しています。

『kcdk.players』

プレイヤーの人数を格納しています。

『kcdk.control: return』

function専用の汎用変数です。汎用変数とは、様々な用途で使うことのできる変数のことを指します。

関数は<data/kcdk/functions/score/return/>に配置してください。ファイルの煩雑化を避けるために、配置する場所を限定しています。なお、限定されるのはreturn変数に値を格納するための関数であり、それらと呼ば出す関数にフォルダの指定はありません。例えば、プレイヤーの人数をreturn変数に格納する関数はreturnフォルダに配置しますが、それと呼ば出す関数をreturnフォルダに配置する必要はありません。

前回return変数に格納された値は、必ず関数内で初期化または上書きされるようにしてください。これは古い値が処理に影響を与えてしまうのを防ぐためです。

return変数を用いる関数は呼び出しから値の読み取りまでを同じtick内で処理しなければいけません。tickを跨いでしまうと、意図しない関数が値に干渉できてしまうためです。例えば、プレイヤーの人数をreturn変数に格納する関数と、それと呼ば出す関数は同じtick内に処理しなければいけません。

『kcdk.control: temp』

汎用変数です。return変数より制限が緩く、フォルダの指定はありませんが、処理の主題がスコアボードである場合は<data/kcdk/functions/score/>に配置する必要があります。

return変数同様、前回変数に格納された値は、必ず関数内で初期化または上書きされるようにしてください。またtemp変数を用いる関数は呼び出しから値の読み取りまでを同じtick内で処理しなければいけません。

『kcdk.control: arg』

function専用の汎用変数です。引数として呼び出す側と呼ば出される側の両方の関数に使われます。フォルダの指定はありませんが、処理の主題がスコアボードである場合は<data/kcdk/functions/score/>に配置する必要があります。

return変数同様、前回変数に格納された値は、必ず関数内で初期化または上書きされるようにしてください。またarg変数を用いる関数は呼び出しから値の読み取りまでを同じtick内で処理しなければいけません。

スコアボードの詳細は< KCDK 2.0/data/kcdk/functions/common/initial_settings >の「scoreboard.mfunction」をご覧ください。

PTP (Pass To Plugin)

※『～』はコマンドタワーの名前およびスコアボードオブジェクトを指しています。

キット内拡張機能の処理を基本機能から実行する方法です。〈data/kcdk/functions/ptp/〉のファイルに処理を追加します。ptp に直接処理を書くのではなく、拡張機能の処理を指定するようにしましょう。

『always.mcfunction』

常に処理を実行します。使用する場合は『kcdk.config』などで処理を無効化できるようにしておきましょう。

『reset.mcfunction』

『reset』が実行されたときに処理を実行します。

『setup.mcfunction』

セットアップの実行時に処理を実行します。

『start_end_game.mcfunction』

『start/end game』が実行されたときに処理を渡します。

その他

- **ボスバーにプレイヤーの体力を表示する:**
下記のコマンド (1) を常時実行し、プレイヤーの体力をボスバーに代入します。
(2) のコマンドを実行してプレイヤーにボスバーを表示させます。

```
1. execute if score phase kcdk.control matches 2 store result  
   bossbar kcdk:general value run scoreboard players get <player>  
   health
```

解説:ボスバーに <player> の体力を代入します。

```
2. bossbar set kcdk:general players @a
```

解説:ボスバーを表示します。

- **ランダム整数出力:**
〈data/kcdk/functions/score/return/dice〉はランダムな整数を「return」に出力します。
1～10 の数、1/2 ～ 1/10 の範囲を出力することができます。

```
1. function kcdk:score/return/dice/10
```

解説:1～10 のいずれかの数字を出力します。

勝利判定の詳細:

victory: 1~4 の他に、以下の勝利判定があります。

victory: 5	青チームの敗北
victory: 6	緑チームの敗北
victory: 7	赤チームの敗北
victory: 8	黄色チームの敗北
victory: 9..12	ゲーム終了 (title表示)
victory: 13..	ゲーム終了 (subtitle表示)

tips: ビーコン戦³の際に『start/end game』にビーコン設置コマンドを置いておくことでビーコンの設置し忘れを防止することができます。

トラブルシューティング

- チームの操作ができない:
チームが有効になっていない、もしくはチームが存在していない可能性があります。
チームが存在しない場合は、以下のコマンドを実行してチームを作成してください。

```
1. function kcdk:team/add/batch
```

チームが有効になっていない場合は、スコアボードの説明の章を確認して有効に設定してください。

- functionが正常に実行されない:
データパックが競合している可能性があります。KCDKを複数導入している場合は、有効なKCDKを1つにしてください。
- リスポーン地点が設定できない:
リスポーン地点が塞がれている可能性があります。ブロック等で塞がれていないか確認してください。水中にリスポーン地点を設定することはできませんのでご注意ください。

³ ビーコン戦: 特定のブロック(ビーコン)の破壊が勝利条件のゲーム形式のこと。

コマンドチェックリスト

コマンドのミスをなるべく減らすために、設定した項目をもう一度確認しましょう。

設定	コマンドの場所	備考
制限時間の設定	set timer	
勝利条件の設定	victory conditions	条件の記述ミスと勝利判定の書き忘れに注意。
プレイヤーの配置	set deploy	配置地点に水があるとリスポーン出来なくなるので注意。
アイテムの設定	set items	ビーコン戦でのピッケルの配布し忘れに注意。
エフェクトの設定	set effects	
死亡時の設定	death	
役職の設定	set role	
終了時の subtitle 表示	announce start/end	