動作モデルの機能追加方法について

2013/06/28

高橋友也

●MoveCode

1. FieldMoveModelTypes.h

　Enum MvCode にタイプを追加

1. MoveCodeBaseを継承したクラスを作成

1. MoveProcessManager::m\_pTblにクラスを登録

●EvType

1. FieldMoveModelTypes.h

　Enum EventTypeにタイプを追加

1. EvTypeBaseを継承したクラスを作成

1. MoveProcessManager:: m\_pEvTypeTblにクラスを登録

●DrawCode

1. FieldMoveModelTypes.h

　Enum DrawCodeにタイプを追加

1. DrawCodeBaseを継承したクラスを作成

1. DrawProcessManager:: m\_pTblにクラスを登録

●Acmd

1. FieldPawnTypes.h

　Enum Acmdにタイプを追加

1. ActionCmdBaseを継承したクラスを作成

1. ActionCmdManager:: m\_pTblにクラスを登録

●動作モデルエフェクト

　①FieldMoveModelEffect.h

MoveModelEffectManager::Typeにタイ峰を追加

　②MoveModelEffectBaseを継承したクラスを作成

　③MoveModelEffectManager:: m\_pArrayにクラスを登録

●アナログ移動の種類

　アナログ移動は、Acmd、MoveModelBase:: SetInputWay、MoveModelBase:: GetOrder系関数を組み合わせて実装します。

　▽情報伝達の流れは以下のようになります。

1. 移動量、方向を制御するデータを設定
   1. MoveModelBase::SetInputWay
2. アクションコマンドを発行
   1. MoveModel::SetAcmd
3. アクションコマンドのプロセスで、移動スピード、加速度、方向を設定
   1. GetInputWayで取得した入力情報から、スピード、加速、方向をMoveModelBase::GetOrder系関数で取得
   2. 取得したスピード、加速、方向で、アナログ移動させる。
   3. 例）FieldActionCmdAnalog.hを参照

　▽追加方法

　　①Acmdを追加

　②MoveModelBaseにアナログ移動の専用GetOrder処理を作成

　　③①で作成したクラスに実装

　▽GetOrder系関数について、注意事項

　　この関数により、入力方向、パワーからそのフレームの移動方向、移動量を算出することができます。

　　この関数は、コリジョン判定でも使用します。

　　コリジョン判定時とAcmdでの移動時での、入力方向、パワーに差分があると、

　　壁へのめり込みなどが発生するため、注意が必要です。

以上。