## changeGiven: boolean = false changeGiver: ChangeGiver {readOnly} coinReceiver: CoinReceiver {readOnly} custPanel: CustomerPanel dispenseCtrl: DispenseController {readOnly drinkDispensed: boolean = false mainCtrl: MainController {readOnly} price: int = 0 selection: int = -1 state: TransactionControllerState cancelTransaction(): void closeDown(): void completeTransaction(): void «interface» displayCustomerPanel(): void TransactionControllerState endMaintenance(): void getChangeGiver(): ChangeGiver cancelTransaction(): void getCoinReceiver(): CoinReceiver completeTransaction(): void getCustomerPanel(): CustomerPanel endMaintenance(): void getDispenseController(): DispenseControlle processMoneyReceived(int): void getMainController(): MainController startMaintenance(): void startTransaction(int): void getPrice(): int getSelection(): int terminateFault(): void isChangeGiven(): boolean isDrinkDispensed(): boolean nullifyCustomerPanel(): void processMoneyReceived(int): void refreshCustomerPanel(): void refreshMachineryDisplay(): void setChangeGiven(boolean): void setDrinkDispensed(boolean): void setPrice(int): void setSelection(int): void setState(TransactionControllerState): void startMaintenance(): void startTransaction(int): void terminateFault(): void TransactionController(MainController) FaultState MaintenanceState **TransactionState** Id leState coinReceiver: CoinReceiver {readOnly} txCtrl: TransactionController {readOnly} changeGiver: ChangeGiver {readOnly} changeGiver: ChangeGiver {readOnly} custPanel: CustomerPanel {readOnly} coinReceiver: CoinReceiver {readOnly} coinReceiver: CoinReceiver {readOnly} + cancelTransaction(): void dispenseCtrl: DispenseController {readOnly custPanel: CustomerPanel {readOnly} dispenseCtrl: DispenseController {readOnly completeTransaction(): void txCtrl: TransactionController {readOnly} dispenseCtrl: DispenseController {readOnly drinksBrand: DrinksBrand endMaintenance(): void price: int = 0 storeItem: StoreItem cancelTransaction(): void MaintenanceState(TransactionController selection: int = -1 txCtrl: TransactionController {readOnly} completeTransaction(): void processMoneyReceived(int): void txCtrl: TransactionController {readOnly} endMaintenance(): void startMaintenance(): void + cancelTransaction(): void FaultState(TransactionController) startTransaction(int): void cancelTransaction(): void completeTransaction(): void processMoneyReceived(int): void terminateFault(): void completeTransaction(): void endMaintenance(): void startMaintenance(): void IdleState(TransactionController) endMaintenance(): void startTransaction(int): void processMoneyReceived(int): void processMoneyReceived(int): void terminateFault(): void startMaintenance(): void startMaintenance(): void startTransaction(int): void startTransaction(int): void terminateFault(): void terminateFault(): void TransactionState(TransactionController)

**TransactionController**