



```
classDiagram
    class StateOfMessageType {
        currentState: StateOfMessageType
        description: String
        name: UTF8String {id}
        <<signal>> deprecate()
        <<signal>> readyForFullUse()
        <<signal>> readyForTrialUse()
    }
    class StateOfMessageType2 {
        <<signal>> deprecate()
        <<signal>> readyForFullUse()
        <<signal>> readyForTrialUse()
    }
    StateOfMessageType --> StateOfMessageType2
```

The diagram illustrates the structure of the `StateOfMessageType` class and its relationship with another class. The `StateOfMessageType` class is defined with the following attributes and operations:

- Attributes:
 - `currentState: StateOfMessageType`
 - `description: String`
 - `name: UTF8String {id}`
- Operations (Signals):
 - `<<signal>> deprecate()`
 - `<<signal>> readyForFullUse()`
 - `<<signal>> readyForTrialUse()`

A directed association exists from `StateOfMessageType` to a second class, which also defines the same three signal operations: `<<signal>> deprecate()`, `<<signal>> readyForFullUse()`, and `<<signal>> readyForTrialUse()`.