;; Auto-generated. Do not edit!

(when (boundp 'robotnik\_msgs\_new::SubState)

(if (not (find-package "ROBOTNIK\_MSGS\_NEW"))

(make-package "ROBOTNIK\_MSGS\_NEW"))

(shadow 'SubState (find-package "ROBOTNIK\_MSGS\_NEW")))

(unless (find-package "ROBOTNIK\_MSGS\_NEW::SUBSTATE")

(make-package "ROBOTNIK\_MSGS\_NEW::SUBSTATE"))

(in-package "ROS")

;;//! \htmlinclude SubState.msg.html

(defclass robotnik\_msgs\_new::SubState

:super ros::object

:slots (\_substate \_substate\_description \_msg ))

(defmethod robotnik\_msgs\_new::SubState

(:init

(&key

((:substate \_\_substate) 0)

((:substate\_description \_\_substate\_description) "")

((:msg \_\_msg) "")

)

(send-super :init)

(setq \_substate (round \_\_substate))

(setq \_substate\_description (string \_\_substate\_description))

(setq \_msg (string \_\_msg))

self)

(:substate

(&optional \_\_substate)

(if \_\_substate (setq \_substate \_\_substate)) \_substate)

(:substate\_description

(&optional \_\_substate\_description)

(if \_\_substate\_description (setq \_substate\_description \_\_substate\_description)) \_substate\_description)

(:msg

(&optional \_\_msg)

(if \_\_msg (setq \_msg \_\_msg)) \_msg)

(:serialization-length

()

(+

;; int32 \_substate

4

;; string \_substate\_description

4 (length \_substate\_description)

;; string \_msg

4 (length \_msg)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

(make-string-output-stream (send self :serialization-length)))))

;; int32 \_substate

(write-long \_substate s)

;; string \_substate\_description

(write-long (length \_substate\_description) s) (princ \_substate\_description s)

;; string \_msg

(write-long (length \_msg) s) (princ \_msg s)

;;

(if (null strm) (get-output-stream-string s))))

(:deserialize

(buf &optional (ptr- 0))

;; int32 \_substate

(setq \_substate (sys::peek buf ptr- :integer)) (incf ptr- 4)

;; string \_substate\_description

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_substate\_description (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;; string \_msg

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_msg (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;;

self)

)

(setf (get robotnik\_msgs\_new::SubState :md5sum-) "d67ef59716a56f2892e9ebccd174a9e5")

(setf (get robotnik\_msgs\_new::SubState :datatype-) "robotnik\_msgs\_new/SubState")

(setf (get robotnik\_msgs\_new::SubState :definition-)

"

# substate of the component

# each component has its own state machines on each state

int32 substate

# Description of the substate

string substate\_description

# Text message - additional status information

string msg

")

(provide :robotnik\_msgs\_new/SubState "d67ef59716a56f2892e9ebccd174a9e5")