;; Auto-generated. Do not edit!

(when (boundp 'schunk\_ezn64::set\_position)

(if (not (find-package "SCHUNK\_EZN64"))

(make-package "SCHUNK\_EZN64"))

(shadow 'set\_position (find-package "SCHUNK\_EZN64")))

(unless (find-package "SCHUNK\_EZN64::SET\_POSITION")

(make-package "SCHUNK\_EZN64::SET\_POSITION"))

(unless (find-package "SCHUNK\_EZN64::SET\_POSITIONREQUEST")

(make-package "SCHUNK\_EZN64::SET\_POSITIONREQUEST"))

(unless (find-package "SCHUNK\_EZN64::SET\_POSITIONRESPONSE")

(make-package "SCHUNK\_EZN64::SET\_POSITIONRESPONSE"))

(in-package "ROS")

(defclass schunk\_ezn64::set\_positionRequest

:super ros::object

:slots (\_goal\_position \_goal\_speed ))

(defmethod schunk\_ezn64::set\_positionRequest

(:init

(&key

((:goal\_position \_\_goal\_position) 0.0)

((:goal\_speed \_\_goal\_speed) 0.0)

)

(send-super :init)

(setq \_goal\_position (float \_\_goal\_position))

(setq \_goal\_speed (float \_\_goal\_speed))

self)

(:goal\_position

(&optional \_\_goal\_position)

(if \_\_goal\_position (setq \_goal\_position \_\_goal\_position)) \_goal\_position)

(:goal\_speed

(&optional \_\_goal\_speed)

(if \_\_goal\_speed (setq \_goal\_speed \_\_goal\_speed)) \_goal\_speed)

(:serialization-length

()

(+

;; float32 \_goal\_position

4

;; float32 \_goal\_speed

4

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; float32 \_goal\_position

(sys::poke \_goal\_position (send s :buffer) (send s :count) :float) (incf (stream-count s) 4)

;; float32 \_goal\_speed

(sys::poke \_goal\_speed (send s :buffer) (send s :count) :float) (incf (stream-count s) 4)

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; float32 \_goal\_position

(setq \_goal\_position (sys::peek buf ptr- :float)) (incf ptr- 4)

;; float32 \_goal\_speed

(setq \_goal\_speed (sys::peek buf ptr- :float)) (incf ptr- 4)

;;

self)

)

(defclass schunk\_ezn64::set\_positionResponse

:super ros::object

:slots (\_goal\_accepted ))

(defmethod schunk\_ezn64::set\_positionResponse

(:init

(&key

((:goal\_accepted \_\_goal\_accepted) nil)

)

(send-super :init)

(setq \_goal\_accepted \_\_goal\_accepted)

self)

(:goal\_accepted

(&optional (\_\_goal\_accepted :null))

(if (not (eq \_\_goal\_accepted :null)) (setq \_goal\_accepted \_\_goal\_accepted)) \_goal\_accepted)

(:serialization-length

()

(+

;; bool \_goal\_accepted

1

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; bool \_goal\_accepted

(if \_goal\_accepted (write-byte -1 s) (write-byte 0 s))

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; bool \_goal\_accepted

(setq \_goal\_accepted (not (= 0 (sys::peek buf ptr- :char)))) (incf ptr- 1)

;;

self)

)

(defclass schunk\_ezn64::set\_position

:super ros::object

:slots ())

(setf (get schunk\_ezn64::set\_position :md5sum-) "9d30216bd0f9755a614e054b1e5b684d")

(setf (get schunk\_ezn64::set\_position :datatype-) "schunk\_ezn64/set\_position")

(setf (get schunk\_ezn64::set\_position :request) schunk\_ezn64::set\_positionRequest)

(setf (get schunk\_ezn64::set\_position :response) schunk\_ezn64::set\_positionResponse)

(defmethod schunk\_ezn64::set\_positionRequest

(:response () (instance schunk\_ezn64::set\_positionResponse :init)))

(setf (get schunk\_ezn64::set\_positionRequest :md5sum-) "9d30216bd0f9755a614e054b1e5b684d")

(setf (get schunk\_ezn64::set\_positionRequest :datatype-) "schunk\_ezn64/set\_positionRequest")

(setf (get schunk\_ezn64::set\_positionRequest :definition-)

"float32 goal\_position

float32 goal\_speed

---

bool goal\_accepted

")

(setf (get schunk\_ezn64::set\_positionResponse :md5sum-) "9d30216bd0f9755a614e054b1e5b684d")

(setf (get schunk\_ezn64::set\_positionResponse :datatype-) "schunk\_ezn64/set\_positionResponse")

(setf (get schunk\_ezn64::set\_positionResponse :definition-)

"float32 goal\_position

float32 goal\_speed

---

bool goal\_accepted

")

(provide :schunk\_ezn64/set\_position "9d30216bd0f9755a614e054b1e5b684d")