;; Auto-generated. Do not edit!

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

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

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

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

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

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

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

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

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

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

(in-package "ROS")

(defclass robotnik\_msgs\_new::SetByteRequest

:super ros::object

:slots (\_value ))

(defmethod robotnik\_msgs\_new::SetByteRequest

(:init

(&key

((:value \_\_value) 0)

)

(send-super :init)

(setq \_value (round \_\_value))

self)

(:value

(&optional \_\_value)

(if \_\_value (setq \_value \_\_value)) \_value)

(:serialization-length

()

(+

;; byte \_value

1

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; byte \_value

(write-byte \_value s)

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; byte \_value

(setq \_value (sys::peek buf ptr- :char)) (incf ptr- 1)

(if (> \_value 127) (setq \_value (- \_value 256)))

;;

self)

)

(defclass robotnik\_msgs\_new::SetByteResponse

:super ros::object

:slots (\_success \_message ))

(defmethod robotnik\_msgs\_new::SetByteResponse

(:init

(&key

((:success \_\_success) nil)

((:message \_\_message) "")

)

(send-super :init)

(setq \_success \_\_success)

(setq \_message (string \_\_message))

self)

(:success

(&optional \_\_success)

(if \_\_success (setq \_success \_\_success)) \_success)

(:message

(&optional \_\_message)

(if \_\_message (setq \_message \_\_message)) \_message)

(:serialization-length

()

(+

;; bool \_success

1

;; string \_message

4 (length \_message)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; bool \_success

(if \_success (write-byte -1 s) (write-byte 0 s))

;; string \_message

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

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; bool \_success

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

;; string \_message

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

;;

self)

)

(defclass robotnik\_msgs\_new::SetByte

:super ros::object

:slots ())

(setf (get robotnik\_msgs\_new::SetByte :md5sum-) "cb2fe72c5204ad899462b90ec1ebb6ff")

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

(setf (get robotnik\_msgs\_new::SetByte :request) robotnik\_msgs\_new::SetByteRequest)

(setf (get robotnik\_msgs\_new::SetByte :response) robotnik\_msgs\_new::SetByteResponse)

(defmethod robotnik\_msgs\_new::SetByteRequest

(:response () (instance robotnik\_msgs\_new::SetByteResponse :init)))

(setf (get robotnik\_msgs\_new::SetByteRequest :md5sum-) "cb2fe72c5204ad899462b90ec1ebb6ff")

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

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

"byte value

---

bool success

string message

")

(setf (get robotnik\_msgs\_new::SetByteResponse :md5sum-) "cb2fe72c5204ad899462b90ec1ebb6ff")

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

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

"byte value

---

bool success

string message

")

(provide :robotnik\_msgs\_new/SetByte "cb2fe72c5204ad899462b90ec1ebb6ff")