;; Auto-generated. Do not edit!

(when (boundp 'robotnik\_msgs\_new::ack\_alarm)

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

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

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

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

(make-package "ROBOTNIK\_MSGS\_NEW::ACK\_ALARM"))

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

(make-package "ROBOTNIK\_MSGS\_NEW::ACK\_ALARMREQUEST"))

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

(make-package "ROBOTNIK\_MSGS\_NEW::ACK\_ALARMRESPONSE"))

(in-package "ROS")

(defclass robotnik\_msgs\_new::ack\_alarmRequest

:super ros::object

:slots (\_type \_group ))

(defmethod robotnik\_msgs\_new::ack\_alarmRequest

(:init

(&key

((:type \_\_type) "")

((:group \_\_group) "")

)

(send-super :init)

(setq \_type (string \_\_type))

(setq \_group (string \_\_group))

self)

(:type

(&optional \_\_type)

(if \_\_type (setq \_type \_\_type)) \_type)

(:group

(&optional \_\_group)

(if \_\_group (setq \_group \_\_group)) \_group)

(:serialization-length

()

(+

;; string \_type

4 (length \_type)

;; string \_group

4 (length \_group)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; string \_type

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

;; string \_group

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

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; string \_type

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

;; string \_group

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

;;

self)

)

(defclass robotnik\_msgs\_new::ack\_alarmResponse

:super ros::object

:slots (\_ret ))

(defmethod robotnik\_msgs\_new::ack\_alarmResponse

(:init

(&key

((:ret \_\_ret) nil)

)

(send-super :init)

(setq \_ret \_\_ret)

self)

(:ret

(&optional \_\_ret)

(if \_\_ret (setq \_ret \_\_ret)) \_ret)

(:serialization-length

()

(+

;; bool \_ret

1

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; bool \_ret

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

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; bool \_ret

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

;;

self)

)

(defclass robotnik\_msgs\_new::ack\_alarm

:super ros::object

:slots ())

(setf (get robotnik\_msgs\_new::ack\_alarm :md5sum-) "5691e91cfff5344662d430d7da60497e")

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

(setf (get robotnik\_msgs\_new::ack\_alarm :request) robotnik\_msgs\_new::ack\_alarmRequest)

(setf (get robotnik\_msgs\_new::ack\_alarm :response) robotnik\_msgs\_new::ack\_alarmResponse)

(defmethod robotnik\_msgs\_new::ack\_alarmRequest

(:response () (instance robotnik\_msgs\_new::ack\_alarmResponse :init)))

(setf (get robotnik\_msgs\_new::ack\_alarmRequest :md5sum-) "5691e91cfff5344662d430d7da60497e")

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

(setf (get robotnik\_msgs\_new::ack\_alarmRequest :definition-)

"string type

string group

---

bool ret

")

(setf (get robotnik\_msgs\_new::ack\_alarmResponse :md5sum-) "5691e91cfff5344662d430d7da60497e")

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

(setf (get robotnik\_msgs\_new::ack\_alarmResponse :definition-)

"string type

string group

---

bool ret

")

(provide :robotnik\_msgs\_new/ack\_alarm "5691e91cfff5344662d430d7da60497e")