

Variables, procédures et évènements de la balise «Ranger Interface board» disponibles avec Aseba

Variables, procédures et évènements sont triés par fonctionnalités.

Contenu

Leds.....	3
leds.RGBcolor()	3
leds.RGBcolor_single().....	4
leds.off()	4
leds.on().....	4
Range & Bearing	5
RAB.off()	5
RAB.on().....	5
Lolettes	6
lolettes.stat.....	6
lolettes.stat1.....	6
lolettes.stat2.....	7
lolettes.stat3.....	7
onevent lolette	7
Accéléromètre	8
acc.x.....	8
acc.y.....	8
acc.z	8
acc.tilt	8
onevent acc	8

Battery.....	9
battery.mV.....	9
battery.mA.....	9
battery.level	9
battery.temperature	9
battery.chargeStat.....	9
battery.underVoltageError.....	9
Battery.UpdateStatus()	10
onevent battery.....	10
Timer	10
onevent timer.01s	10
onevent timer.05s	10
Data (microSD card)	11
data.error	11
Data.Save.char()	11
Data.Save.ASCII()	11
Data.Save.nb()	12
Data.Save.acc()	12
Data.Save.acc_lol()	12
Divers.....	13
Sleep.WaitLolette().....	13
_fwversion	13

Leds

leds.RGBcolor()

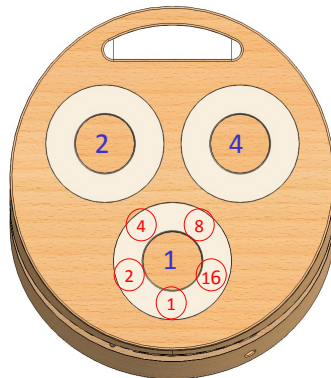
Procédure: `leds.RGBcolor(group, leds, RGBcolor[3])`

Fonctionnalité: - Affiche une couleur sur les leds sélectionnées.

Paramètres: - group - Numéro de groupe(1, 2, 4) ou combinaison des 3 groupes circulaires de leds (1 - 7).

- leds - Numéro de led (1, 2, 4, 8, 16) ou combinaison des 5 leds RGB à l'intérieur d'un groupe (1 - 31).

- RGBcolor - Trois champs contenant respectivement la quantité de couleur rouge, vert, bleu; valeurs de 0 à 255.



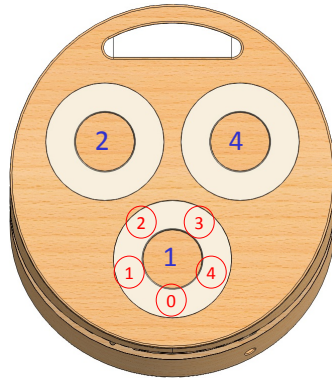
leds.RGBcolor_single()

Procédure: leds.RGBcolor_single (group, leds, RGBcolor[3])

Fonctionnalité: - Affiche une couleur sur la led sélectionnée, dans les groupes choisis.

Paramètres: - group - Numéro de groupe(1, 2, 4) ou combinaison des 3 groupes circulaires de leds (1 - 7).

- led - Numéro de led (0-4) à l'intérieur d'un groupe.
- RGBcolor - Trois champs contenant respectivement la quantité de couleur rouge, vert, bleu; valeurs de 0 à 255.



leds.off()

Procédure: leds.off()

Fonctionnalité: - Eteint les leds; la configuration de couleur reste inchangée.

Paramètres: aucun.

leds.on()

Procédure: leds.on()

Fonctionnalité: - Allume les leds avec les couleurs précédemment configurées.

Paramètres: aucun.

Range & Bearing

RAB.off()

Procédure: RAB.off()

Fonctionnalité: - Désactive les fonctionnalités du Range & Bearing et l'éteint.

Paramètres: aucun.

RAB.on()

Procédure: RAB.on()

Fonctionnalité: - Alimente le Range & Bearing et active ses fonctionnalités.

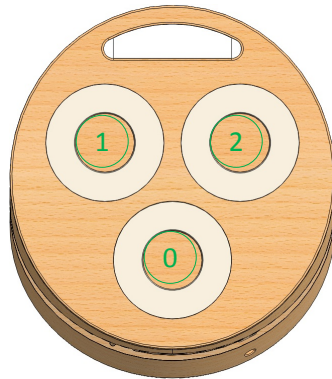
Paramètres: aucun.

Loettes

loettes.stat

Variable: loettes.stat

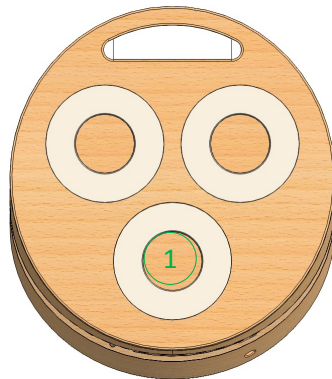
Description: - Champ de trois bits, indiquant la présence d'une lolette à l'endroit concerné (1: présent).



loettes.stat1

Variable: loettes.stat

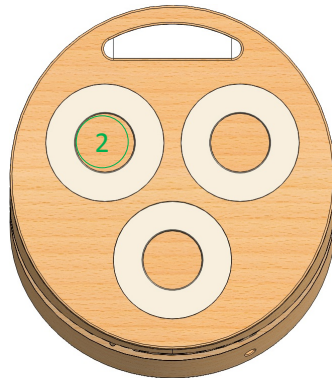
Description: - Indique la présence de la lolette 1 (1: présent).



lolettes.stat2

Variable: lolettes.stat

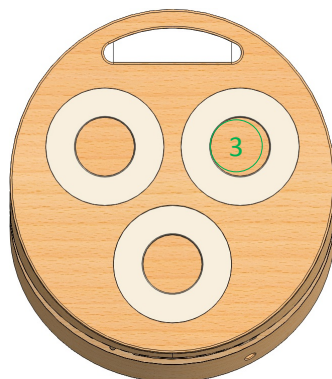
Description: - Indique la présence de la lolette 2 (1: présent).



lolettes.stat3

Variable: lolettes.stat

Description: - Indique la présence de la lolette 3 (1: présent).



onevent lolette

Evènement: lolette

Description: - Evènement généré lorsqu'un lolette est insérée ou enlevée.

Accéléromètre

acc.x

Variable: acc.x

Description: - Accélération selon l'axe x.

- De -1.5g à +1.45g, par pas de environ 0.047g (1.5g/32).
- Mise à jour en même temps que l'évènement.

acc.y

Variable: acc.y

Description: - Accélération selon l'axe y.

- De -1.5g à +1.45g, par pas de environ 0.047g (1.5g/32).
- Mise à jour en même temps que l'évènement.

acc.z

Variable: acc.z

Description: - Accélération selon l'axe z.

- De -1.5g à +1.45g, par pas de environ 0.047g (1.5g/32).
- Offset de gravitation initiale de -20 LSB.
- Mise à jour en même temps que l'évènement.

acc.tilt

Variable: acc.tilt

Description: - Champ de bits contenant les statuts de mouvement et d'orientation.

- bit[1:0]: 01 -> positionnement sur le devant (**front**),
10 -> positionnement sur le dos (**back**),
autre -> positionnement inconnu.
- bit[4:2]: 001 -> positionnement paysage sur la gauche (**left**),
010 -> positionnement paysage sur la droite (**right**),
101 -> positionnement portrait inversé (**down**),
110 -> positionnement portrait normal (**up**),
autre -> positionnement inconnu.
- bit[5]: tape détectée (1: tape).
- bit[7]: secousse détectée (1: secousse).

onevent acc

Evènement: acc

Description: - Evènement généré lorsqu'une nouvelle valeur de l'accéléromètre est présente.

Environ chaque 9 [ms]; 0.1[s] au maximum.

Battery

battery.mV

Variable: battery.mV

Description: - Dernière tension obtenue à partir de la batterie.

- Résolution d'environ 5 [mV].
- Le système se coupe lorsque la tension descend en dessous de 3.2 [V];
il est de plus impossible à rallumer la balise si cette tension est inférieure à 3.2 [V].
- Mise à jour en même temps que l'évènement.

battery.mA

Variable: battery.mA

Description: - Dernière tension obtenue à partir de la batterie.

- Résolution de 1 [mA].
- Positif -> en charge, négatif en décharge.
- Mise à jour en même temps que l'évènement.

battery.level

Variable: battery.level

Description: - Capacité en [%] (0 - 100).

- Mise à jour en même temps que l'évènement.

battery.temperature

Variable: battery.temperature

Description: - Température de l'accu [°C/100].

- Résolution de environ 0.12°C
- Mise à jour en même temps que l'évènement.

battery.chargeStat

Variable: battery.chargeStat

Description: -Etat de la charge.

- 0: charge désactivée, 1: pas d'alimentation de charge, 2: en charge, 3: pas en charge,
4: complètement chargé.

battery.underVoltageError

Variable: battery.underVoltageError

Description: - Si différent de 0, indique que l'accu s'est retrouvé déchargé.

- Ecrire 0 dedans pour réinitialiser l'indicateur d'erreur.
- Peut-être lu pour savoir si l'alimentation a dû être coupée pour cause d'énergie insuffisante. Cela indique que le RAB ont et les leds ont dû est momentanément éteints, le fichier dans la carte SD fermé et les évènements ignorés ou désynchronisés.

Battery.UpdateStatus()

Procédure: Battery.UpdateStatus()

Fonctionnalité: - Demande une mise à jour des valeurs de la batterie.

- Pas très utile à appeler, car automatiquement appelée chaque 0.5 [s].

Paramètres: aucun.

onevent battery

Evènement: battery

Description: - Evènement généré lorsqu'une nouvelle information sur la batterie est présente.

Survient après un appel à Battery.UpdateStatus(), chaque 0.5[s], ou lors du changement d'état de charge.

Timer

onevent timer.01s

Evènement: timer.01s

Description: - Evènement chaque 0.1[s].

onevent timer.05s

Evènement: timer.05s

Description: - Evènement chaque 0.5[s].

Data (microSD card)

data.error

Variable: data.error

Description: - Indique si une erreur s'est produite lors du dernier appel à une fonction Data (Data.Save.acc); notamment lorsque la carte microSD est absente.
- Contient 0, si aucune erreur; 1, si c'est le cas.

Data.Save.char()

Procédure: Data.Save.char(char[])

Fonctionnalité: - Sauve les données présentes en paramètre, sur la carte SD à la suite dans le fichier ouvert.

Crée et ouvre un nouveau fichier (log?????.txt) lorsqu'aucun n'est ouvert.

Le fichier est fermé lors de la mise en veille au travers de Sleep.WaitLolette().

Paramètres: - char - tableau contenant un certain nombre de caractères ASCII (jusqu'à 255).

Seul le byte le moins significatif de ces caractères codés sur 16bits est pris en compte.

Valeur de retour: La variable data.error est mise à jour.

Data.Save.ASCII()

Procédure: Data.Save.ASCII(char[])

Fonctionnalité: - Identique à Data.Save.char(), à l'exception du codage des caractères qui est différent. Cette fonction nécessite moins de ressources (RAM et instructions) mais est moins pratique à utiliser.

Paramètres: - char - tableau contenant un certain nombre de paires de caractères ASCII (jusqu'à 255 caractères).

Chaque élément du tableau étant de 16bits, il contient deux caractères; le stockage est en little endian. ([0x1100, 0x3322], écrit dans l'ordre: 0x00, 0x11, 0x22, 0x33).

Si l'on veut écrire un nombre de caractère impaire, mettre 0x00 dans le dernier caractère du tableau; c'est-à-dire 0x00?? dans le dernier élément. Les autres 0 du tableau sont écrits comme caractère NULL.

Valeur de retour: La variable data.error est mise à jour.

Data.Save.nb()

Procédure: Data.Save.nb(nb)

Fonctionnalité: - Sauve le nombre donné en paramètre, sur la carte SD à la suite dans le fichier ouvert.

Crée et ouvre un nouveau fichier (log?????.txt) lorsqu'aucun n'est ouvert.

Le fichier est fermé lors de la mise en veille au travers de Sleep.WaitLolette().

Paramètres: - nb - Nombre entier signé codé sur 16 bits.

Valeur de retour: La variable data.error est mise à jour.

Data.Save.acc()

Procédure: Data.Save.acc()

Fonctionnalité: - Sauve les données courantes de l'accéléromètre sur la carte SD, à la suite dans le fichier ouvert. (acc.x acc.y acc.z acc.tilt)

Crée et ouvre un nouveau fichier (log?????.txt) lorsqu'aucun n'est ouvert.

Le fichier est fermé lors de la mise en veille au travers de Sleep.WaitLolette().

Paramètres: aucun.

Valeur de retour: La variable data.error est mise à jour.

Data.Save.acc_lol()

Procédure: Data.Save.acc_lol()

Fonctionnalité: - Sauve les données courantes de l'accéléromètre, ainsi que l'emplacement des lolettes présentes, sur la carte SD à la suite dans le fichier ouvert. (acc.x acc.y acc.z acc.tilt lolettes)

Crée et ouvre un nouveau fichier (log?????.txt) lorsqu'aucun n'est ouvert.

Le fichier est fermé lors de la mise en veille au travers de Sleep.WaitLolette().

Paramètres: aucun.

Valeur de retour: La variable data.error est mise à jour.

Divers

Sleep.WaitLolette()

Procédure: Sleep.WaitLolette()

Fonctionnalité: - Met le système en mode veille, jusqu'à ce qu'une lolette soit insérée.

Aucun autre évènement n'est possible.

Eteint également les leds

Paramètres: aucun.

_fwversion

Variable: _fwversion[2]

Description: - Indique la version du firmware (_fwversion[0]) et la sous-version (_fwversion[1]).

- Pas de protection en écriture!