

BUGGER

FW = Folge Wand
W = Warte
FaW = Finde andere Wand

zb_u_ns = Zielfeld belegt u. nicht Start?
zf = Zielfeld frei?
zb = Zielfeld belegt?
zf_u_ns = Zielfeld frei u. nicht Start?
zs = Zielfeld Start?
zw = Zielfeld Wand?
zf_u_nw = Zielfeld frei u. nicht Wand?

	zb_u_ns	zf	zb	zf_u_ns	zs	zw	zf_u_nw
→ FW	W	–	–	FW	FaW	–	–
W	–	FW	W	–	–	–	–
FaW	–	–	–	–	–	FW	FaW

PUSHER¹

HB = Horizontale Bewegung
VB = Vertikale Bewegung
Z = Zerstört

zf = Zielfeld frei (Oder Roboter, aber verschiebbar)?
zw_o_pbb = Zielfeld Wand (o. block. Rob.) o. Patchbot-Breite erreicht?
zw_o_pbh = Zielfeld Wand (o. block. Rob.) o. Patchbot-Höhe erreicht?
zg = Zielfeld gefährlich?

	zf	zw_o_pbb	zw_o_pbh	zg
→ HB	HB	VB	–	Z
VB	VB	–	HB	Z
Z	–	–	–	–

DIGGER¹

HB = Horizontale Bewegung
VB = Vertikale Bewegung
Z = Zerstört

zf = Zielfeld frei (Oder Roboter, aber verschiebbar)?
zw_o_pbb = Zielfeld Wand (o. block. Rob.) o. Patchbot-Breite erreicht?
zw_o_pbh = Zielfeld Wand (o. block. Rob.) o. Patchbot-Höhe erreicht?
zg = Zielfeld gefährlich?

	zf	zw_o_pbb	zw_o_pbh	zg
--	----	----------	----------	----

¹ Hier gibt es weitere Events, die auf den gleichen Zustand zeigen, z.B. Bewegung oder Verschieben (lt. Aufgabenstellung). Es werden hier zur Vereinfachung und zur Kompakthaltung der States und Events etwaige Nebeneffekte (z.B. Verschieben oder Graben) mit der Bewegung an sich zusammengefasst. Das v.a., da die Aufgabenstellung meiner Meinung nach sehr inkonsistent formuliert ist.

→ HB	HB	VB	–	Z
VB	VB	–	HB	Z
Z	–	–	–	–

SWIMMER²

HB = Horizontale Bewegung
 VB = Vertikale Bewegung
 Z = Zerstört

zf_o_w = Zielfeld frei (Oder Roboter, aber verschiebbar) o. Wasser?
 zw_o_pbb = Zielfeld Wand (o. block. Rob.) o. Patchbot-Breite erreicht?
 zw_o_pbh = Zielfeld Wand (o. block. Rob.) o. Patchbot-Höhe erreicht?
 zg = Zielfeld gefährlich?

	zf_o_w	zw_o_pbb	zw_o_pbh	zg
→ HB	HB	VB	–	Z
VB	VB	–	HB	Z
Z	–	–	–	–

FOLLOWER³

V = Verfolge
 W = Warte

pb_is = Patchbot in Sicht und erreichbar?
 pb_as = Patchbot außer Sicht oder nicht erreichbar?

	pb_is	pb_as
→ W	V	W
V	V	W

HUNTER³

V = Verfolge
 W = Warte
 J = Jage

pb_is = Patchbot in Sicht und erreichbar?
 pb_as = Patchbot außer Sicht oder nicht erreichbar?

² s. Fußnote 1

³ Dass sich Patchbot „in Sicht“ befindet, heißt hier, dass jedes Tile t mit x und y zwischen Patchbot pb mit x und y und Gegner g mit x und y: $x_t: x_{pb} \leq x_t \leq x_g \wedge y_t: y_t = \frac{y_g - y_{pb}}{x_g - x_{pb}} x_t + y_g - \frac{y_g - y_{pb}}{x_g - x_{pb}} x_g \vee x_t: x_t = \frac{(y_t - y_g - \frac{y_g - y_{pb}}{x_g - x_{pb}} x_g)}{\frac{y_g - y_{pb}}{x_g - x_{pb}}} \wedge y_t: y_{pb} \leq y_t \leq y_g$ nicht vom

Typ Wand ist. Typ Wand sei hier terrain::CONCRETE_WALL, terrain::ROCK_WALL, terrain::AUTOMATIC_DOOR, terrain::MANUAL_DOOR oder terrain::SERET_PASSAGE. Wohl gemerkt sind die Türen geschlossen. Wenn das Ergebnis der o.g. Funktionen direkt auf ein Tile fällt, d.h. $y_t = \lfloor y_t \rfloor \vee x_t = \lfloor x_t \rfloor$, je nachdem ob $\Delta x > \Delta y$, werden auch die diagonale nächsten Tiles mit geprüft (also x-1, y und x, y-1).

	pb_is	pb_as
→ w	V	-
v	V	J
j	V	J

SNIFFER

V
W

= Verfolge
= Warte

pe
pne

= Patchbot erreichbar?
= Patchbot nicht erreichbar?

	pe	pne
→ w	V	W
v	V	W