Codelist - Arma

This page provides useful code snippets for Arma development and/or mission design.

It has been curated since Arma 1 and may contain outdated information. Always verify before using in large Multiplayer missions.

Returns & Definitions

```
comment "Return definition of marker 1 (mark1)";
copyToClipboard (str (getMarkerType "mark1")); hint format ["%1",getMarkerType "mark1"];
```

```
comment " Return Position of MARKER1 to your clipboard";
MP = getMarkerPos "MARKER1"; hint format["%1",(getMarkerPos "MARKER1")]; copyToClipboard (str
(MP));
```

```
comment "Return current weapon classname you are using";
copyToClipboard (str (currentWeapon (vehicle player))); hint format ["%1",currentWeapon
(vehicle player)];
```

```
comment "Return current Magazine classnames your weapon using";
copyToClipboard (str (currentMagazine (vehicle player))); hint format ["%1",currentMagazine
(vehicle player)];
```

```
comment "Return hit points of cursored vehicle";
HP = ((getAllHitPointsDamage cursorObject) select 0); systemChat format["%1",
((getAllHitPointsDamage cursorObject) select 0)] ; copyToClipboard (str (HP));
```

```
comment "Return classname of current vehicle";
copyToClipboard (str (vehicleVarName this)); hint format ["%1",vehicleVarName this];
```

```
comment "Return animation list back to player (useful for getting door/gate animations";
copyToClipboard (str (animationNames cursorTarget)); hint format ["%1",animationNames
cursorTarget];
```

```
comment "Return current turret of players seat";
copyToClipboard (str (assignedVehicleRole player)); hint format ["%1", assignedVehicleRole
player];
```

```
comment "Return current Weapon of turret";
copyToClipboard (str (vehicle player weaponsTurret [3])); hint format ["%1", (vehicle player)
weaponsTurret [3]];
```

```
comment "Return current Magazine of turret";
copyToClipboard (str (vehicle player magazinesTurret [3])); hint format ["%1", (vehicle
player) magazinesTurret [3]];
```

```
comment "Get specific aspects of a vehicle";
copyToClipboard str (animationNames cursorObject);     //--- Animation names
copyToClipboard str (getObjectMaterials cursorObject); //--- Materials
copyToClipboard str (getObjectTextures cursorObject);  //--- Textures
```

```
comment "Return position of the players location and copy to clipboard";
houseposATL=getPosATL player; hint format["%1",(getPosATL player)]; copyToClipboard (str
(houseposATL));
houseposASL=getPosASL player; hint format["%1",(getPosASL player)]; copyToClipboard (str
(houseposASL));
houseposWOP=getPosWorld player; hint format["%1",(getPosWorld player)]; copyToClipboard (str
(houseposWOP));
```

```
comment "Returns the direction (0 to 360) of the players location and copy to clipboard";
dirpos=getDir player; hint format["%1",(getDir player)]; copyToClipboard (str (dirpos));
```

```
comment "Return height of the players location and copy to clipboard";
heightpos=[(getPosATL player) select 2]; hint format["%1",(getPosATL player)select 2];
copyToClipboard (str (heightpos));
```

```
comment "Return array of Actions the player just executed and copy to clipboard ";
actions=inGameUISetEventHandler ["Action", "copyToClipboard str (_this select 3); false"];
```

```
comment "Return classname of object the player is pointing to";
objectClass = copyToClipboard (str (typeOf cursorTarget)); hint format ["Classname of
cursorTarget: %1",typeOf cursorTarget];
```

```
comment "Return classname of bullet that has been fired by the player";
bulletClass = player addEventHandler ["FiredMan",{systemChat (typeOf (_this select 6))}];
```

General Codes

```
comment "Get Arsenal as AddAction";
player addAction ["Arsenal", {["Open",true] spawn BIS_fnc_arsenal}];
```

```
comment "Get Garage (Arsenal) as AddAction";
player addAction ["Garage", {["Open",true] spawn BIS_fnc_garage}];
```

```
comment "Get Zeus instantly without Modules and with all entities as editable";
curGrp =  createGroup EAST; theCurator  = curGrp createUnit ["ModuleCurator_F", player, [], 0,
"NONE"];
theCurator setCuratorEditingAreaType true; theCurator addCuratorEditableObjects [entities
"all",true];
```

```
comment "Sets NATO hostile with AAF (just there for easy c&p)";
west setFriend [resistance, 0];
resistance setFriend [west, 0];
```

```
comment "Periscope down at startup";
vehicleName animate ["periscope", 3]; vehicleName animate ["Antenna", 3]; vehicleName animate
["HideScope", 3]; vehicleName animate["display_on_R", 1];
```

```
comment "Open Ghosthawk doors";
vehicleName animateDoor ['door_R', 1];
vehicleName animateDoor ['door_L', 1];
```

```
comment "Animation on a AI unit also in isMultiplayer";
0 = this spawn {waitUntil {time > 0};
[_this, "Acts_Kore_TalkingOverRadio_loop"] remoteExecCall ["switchMove"];
_animEH = _this addEventHandler ["AnimDone", {(_this select 0) switchMove (_this select
1)}];
_this setVariable ["BIS_animEH", _animEH];
_killedEH = _this addEventHandler ["Killed", {(_this select 0) removeEventHandler ["AnimDone",
(_this select 0) getVariable "BIS_animEH"]}];
_this setVariable ["BIS_killedEH", _killedEH]};
```

```
comment "Change formation for whole group";
{_x setFormation "LINE"} forEach units group groupName
```

```
comment "Destroy specific object via ID";
(getPos this nearestObject 19305) setDamage 1;
```

```
comment "Spawn AI correctly in houses & towers";
unitName setPos (nearestBuilding unitName buildingPos 2);
```

```
comment "Place boats correctly on sea on height level 0";
this setPos [position this select 0, position this select 1, 0];
```

```
comment "Set height of player to 2500 above ground";
player setPos [position player select 0, position player select 1, 2500];
```

```
comment "Move a group in a vehicle";
{_x moveInCargo vehicleName} forEach units group groupName;
```

```
comment "Stop AI group from moving without disabling the moveFunction (re-enable with stop
false;) (obsolete, see code beolow this)";
{_x stop true} forEach units group groupName;
```

```
comment "Stop AI at mission start to prevent formation changes done in editor";
doStop (units groupName);
```

```
comment "Stop AI from moving around while disabling the PATH FSM";
{_x disableAI "PATH"} forEach units group groupName;
```

```
comment "Let AI group target a specific unit or object";
{_x doTarget vehicleName} forEach units group groupName;
```

```
comment "Registers the unitName in the helicopter and reserves a cargo / driver / gunner seat
for him";
unitName assignAsCargo vehicleName; /**/ unitName assignAsDriver vehicleName; /**/ unitName
assignAsGunner vehicleName;
```

```
comment "Get out of the helicopter under normal conditions (when it lands). He leaves at the
first opportunity";
unassignVehicle unitName;
```

```
comment "Execute local command for mp enviroment";
[_this, "acts_carfixingWheel"] remoteExecCall ["switchMove"]; // SwitchMove example
[_this, "ru_incoming"] remoteExecCall ["say3D"]; // say3D example
```

```
comment "Prevent helicopter from falling down like a stone";
vehicleName setVelocity [0,0,25];
```

```
comment "Activate a trigger or script only for Land Units";
count (thisList unitsBelowHeight 2) > 0;
```

```
comment "Driver in trigger needs to be from the east side";
 {side (driver _x) == east} count thisList > 0
```

```
comment "Run script code when unitName is in vehicle/helicopter";
if (unitName in vehicleName) then { your_code_here };
```

```
comment "10 - in meters";
vehicleName flyInHeight 10;
```

```
comment "Check if the helicopter is two meters above the ground or less";
position vehicleName select 2 < 2;
```

```
comment "Check if trigger has atleast 1 car inside";
{typeOf _x == "car"} count thisList > 0;
```

```
comment "Move unit in vehicle turret 4";
unitName moveInTurret [vehicleName, [4]];
```

```
comment "Remove TIEquipment of disasembled HMG's / GMG's";
vehicleName addEventHandler ["WeaponAssembled", {params ["", "_obj"]; _obj disableTIEquipment
true;}];
```

```
comment "Unlimited Ammo for all Weapons";
vehicleName addEventHandler ["Fired",{[_this select 0,getNumber (configFile/"CfgAmmo"/(_this
select 4)/"explosive")]
spawn {if (_this select 1==1) then {sleep 0.75};_this select 0 setVehicleAmmo 1}}];
```

```
comment "AddAction for opening and closing the MedicPod";
vehicleName setVariable ['bis_disabled_Door_4', 0]; vehicleName setVariable
```

```
['bis_disabled_Door_6', 0];
```

```
comment "Damage the TailRotor (ArmA 2)";
vehicleName setHit ["mala vrtule", 0.95];
```

```
comment "Damage a TailRotor (ArmA 3)";
vehicleName setHitPointDamage ["HitVRotor",1];
```

```
comment "Will make a Taru (CSAT Heli) Black.";
[vehicleName, ["Black", 1], []] call bis_fnc_initVehicle;
```

```
comment "Will make a AAF YF-55 Hellcat Green";
[vehicleName, ["green",1], []] call bis_fnc_initVehicle;
```

```
comment "Reactive Armor for the MBT-52 KUMA";
[vehicleName, ["HideTurret", 1, "HideHull", 1], []] call bis_fnc_initVehicle;
```

```
comment "Will apply Dazzle camo to the armed NATO prowler";
[vehicleName, ["Dazzle", 1], []] call bis_fnc_initVehicle;
```

```
comment "Change color of a vehicle with RGBA";
vehicleName setObjectTexture [0,"#(argb,8,8,3)color(1,0,0,1)"];
```

```
comment "No Autoattack UAV";
vehicleName disableAI "AUTOTARGET";
vehicleName disableAI "TARGET";
vehicleName setCombatMode "BLUE";
vehicleName setBehaviour "CARELESS";
vehicleName flyInHeight 500;
```

```
comment "UAV Darter drone with Machinegun instead of Laserdesignator";
vehicleName removeWeaponTurret ["Laserdesignator_mounted",[0]];
vehicleName addWeapon "LMG_Zafir_F"; vehicleName addMagazines ["150Rnd_762x54_Box_Tracer", 6];
```

```
comment "Delete vehicle (wreck, crate) the player is looking at in MP";
if (name player == "aebian") then {deleteVehicle cursorTarget};
```

```
comment "Add Item to player (MP)";
if (name player == "aebian") then {player addWeapon "launch_I_Titan_F"};
```

```
comment "Set base texture of a vehicle";
vehicleName setObjectTextureGlobal [0, "itsAebian\images\CH146_0.paa"];
```

```
comment "Open doors of a Medic Tent";
objectName animate ["door1_hide",1] and objectName animate ["door2_hide",1]
```

3DEN Specific Codes

```
comment "Show Objects IDs in the Editor again";
do3DENAction "ToggleMapIDs";
```

```
comment "Get ASL postion of selected object.";
getPosASL (get3DENSelected "object" select 0)
```

```
comment "Returns the attachTo coordinates of objects you 'placed' in 3DEN that are needed. So you can actually attach items via 3DEN now. Object first, then things you want to attach.";
attachPos = (get3DENSelected "object" select 0) worldToModel (getPosATL (get3DENSelected "object" select 1)); copyToClipboard (str (attachPos));
```

```
comment "Returns direction of selected 3DEN object from 0 to 360";
copyToClipboard str(getDir (get3DENSelected "object" select 0));
```

```
comment "Return array of numbers for setVectorDirAndUp when object faces north";
vectorPos = [vectorDir (get3DENSelected "object" select 0), vectorUp (get3DENSelected "object" select 0)]; copyToClipboard (str (vectorPos));
```

```
comment "Return array of numbers for setVectorDirAndUp relative to the base objects position.";
vectorPosRel = [(get3DENSelected "object" select 0) vectorWorldToModel vectorDir (get3DENSelected "object" select 1), (get3DENSelected "object" select 0) vectorWorldToModel vectorUp (get3DENSelected "object" select 1)]; copyToClipboard (str (vectorPosRel));
```

```
comment "Rotate selected objects 180 degrees";
{_x set3DENAttribute ["rotation", [(_x get3DENAttribute "rotation" select 0 select 0), (_x get3DENAttribute "rotation" select 0 select 1), ((_x get3DENAttribute "rotation" select 0 select 2) + 180)]]} forEach get3DENSelected "object";
```

```
comment "Rotate selected objects 90 degrees";
_sel = get3DENSelected "object"; _center = [0,0,0]; {_center = _center vectorAdd (getPosATL _x)} forEach _sel; _center = _center vectorMultiply (1/(count _sel)); {_relPos = (getPosATL _x) vectorDiff _center; _newPos = [(_relPos select 1), -(_relPos select 0), _relPos select 2]; _x set3DENAttribute ["position", _center vectorAdd _newPos]; _x set3DENAttribute ["rotation", [(_x get3DENAttribute "rotation" select 0 select 0), (_x get3DENAttribute "rotation" select 0 select 1), ((_x get3DENAttribute "rotation" select 0 select 2) + 90)]]} forEach _sel;
```

```
comment "Copy selected object and move it on the X-Axis by its width 2 times (rotation aware). modelFix variable can be adjusted if object doesn't form a straight line due to model alignment issues. Set _dir to 1 for positive X/y, -1 for negative X/y";
private _obj = get3DENSelected "object" select 0;
private _dir = 1;
private _modelFix = _dir * 0;
private _size = (_obj call BIS_fnc_boundingBoxDimensions) select 0;
private _type = typeOf _obj;
private _rotation = _obj get3DENAttribute "rotation" select 0;
private _currentPos = _obj get3DENAttribute "position" select 0;

for "_i" from 1 to 2 do
{
    _currentPos = _currentPos vectorAdd (_obj vectorModelToWorld [(_dir * _size - _modelFix), 0, 0]);
    _currentPos = [
        parseNumber ((_currentPos select 0) toFixed 3),
        parseNumber ((_currentPos select 1) toFixed 3),
        parseNumber ((_currentPos select 2) toFixed 3)
    ];
    _newObj = create3DENEntity ["Object", _type, [0, 0, 0]];
    _newObj set3DENAttribute ["position", _currentPos];
    _newObj set3DENAttribute ["rotation", _rotation];
};
```

ZEUS Specific Codes

```
comment "get ASL pos of selected object in ZEUS";
getPosASL ((curatorSelected select 0) select 0);
```

RHS (rhsmods.org) Specific Codes

```
comment "Will make a RHS Humvee Olive Green";
[vehicleName, ["Olive", 1], []] call bis_fnc_initVehicle;
```

```
comment"Fold mainrotor @AH-1Z // 0 = unfold | 1 = fold";
[vehicleName,["mainRotor_folded",1]] call BIS_fnc_initVehicle;
```

```
comment "Will create a Light inside a C-130J";
Light = createMine ["placed_chemlight_yellow", position player, [], 0];
Light2 = createMine ["placed_chemlight_yellow", position player, [], 0];
Light setVectorUp [0,0,-1]; Light attachTo [vehicleName, [0, 6, 2.5] ];
Light2 setVectorUp [0,0,-1]; Light2 attachTo [vehicleName, [0, 4, 2.5] ];
```

```
comment "AI will prepare the SCUD and will fire the missile to target";
[vehicleName,1] spawn rhs_fnc_ss21_AI_prepare;
[vehicleName, getMarkerPos "target"]spawn rhs_fnc_ss21_AI_launch;
```

```
comment "Detonate Tochka-U Nuke without launch vehicle. 100 is the radius you can change";
[(getPos objectName),100] call rhs_fnc_ss21_nuke;
```

```
comment "Animate rhino // Down = 1 | Up = 0";
vehicleName animateDoor ['rhino',1,true];
```

```
comment "Animate back-door on FMTV Command Post Variant // Open = 1 | Closed = 0";
vehicleName animate ["DoorB", 1];
```

```
comment "Animate back-ladder on FMTV Command Post Variant // Hide = 0 | Show = 1";
vehicleName animate ["hide_steps", 1];
```

```
comment "Turn off the RHS Decals (could save some performance)";
//Run Function RHSDecalsOff = true;
//Disable all vehicle decals
vehicleName setVariable ["RHS_Decal_Enabled", false];
// Disable vehicle symbol decals
vehicleName setVariable ["RHS_Decal_Symbol_Enabled", false];
// Disable vehicle number decals
```

```
vehicleName setVariable ["RHS_Decal_Number_Enabled", false];
```

```
comment "Removes the M260 rocket pods and ammo on a AH-6M-L so that only the Miniguns are visible and installed";
{vehicleName removeMagazineTurret [_x, [-1]]} forEach ["rhs_pod_empty","rhs_pod_empty","rhs_pod_FFAR_7","rhs_pod_FFAR_7","RHS_14Rnd_M151_Magazine_MELB"];
{vehicleName removeWeaponTurret [_x, [-1]]} forEach ["RHS_M260_MELB"];
```

ACE3 Specific Codes

```
comment "Define unit as ACE Medic // 0 = no medic; 1 = medic; 2 = doctor;";
unitName setVariable ["ace_medical_medicClass", 2, true];
```

```
comment "Define unit as ACE Engineer // 0 = no engineer; 1 = repair specialist; 2 = engineer;";
unitName setVariable ["ACE_IsEngineer", 1, true];
```

```
comment "Define vehicle as medical facility";
vehicleName setVariable ["ace_medical_isMedicalFacility",1 ,true];
```

```
comment "Define vehicle as repair facility";
vehicleName setVariable ['ace_isRepairFacility',1, true];
```

```
comment "Full heal of a player trough ACE3";
[player] call ace_medical_treatment_fnc_fullHealLocal;
```

```
comment "Get ACE Full Arsenal";
[objectName, true, true] call ace_arsenal_fnc_initBox;
```

```
comment "Change G Effects of player";
unitName setVariable ["ACE_GForceCoef", 0];
```

```
comment "Block a vehicles engine (like a EMP would do) ";
[vehicleName, "blockEngine", "EMP", true] call ace_common_fnc_statusEffect_set;
```

```
comment "Do specific action when soldier use specific medicine";
["ace_medical_treatmentAdvanced_medicationLocal", {
  params ["_patient", "_medication"];
  if (_medication == "Atropine") then {  };
}] call CBA_fnc_addEventhandler;
```

Poppy Specific Codes

```
comment "Set loadout via Poppy";
call{[this, "Loadout"] call Poppy_fnc_applyLoadout;}
```

```
comment "Preview Loadouts in 3DEN"
  call{[(get3DENSelected "object" select 0), "Loadout"] call Poppy_fnc_applyLoadout;}
```

```
comment "Check if Loadout was given";
[player, playerLoadoutGiven] call Poppy_fnc_preInit;
```

FUPS Specific Codes

```
comment "Initialize KI group for FUPS patrol";
[groupName,"markerName"] call FUPS_fnc_main;
```

Revision #17
Created 2025-08-12 16:52:52 UTC by Alexander (nethavn)
Updated 2025-09-22 20:38:39 UTC by Alexander (nethavn)