

```
/*
```

```
001-Generated C code for functional specification 'GET_FREE_MACHINE'
```

```
VERSION: 3.2.3.9 C-RAT
```

```
: 1.6.0.0
```

```
AUTHOR: Hamilton Technologies Inc. Copyright 1991-2015.
```

```
OPERATION: GET_FREE_MACHINE
```

```
GENERATED: Tue Aug 25 16:55:23 2015
```

```
SCCSID: @(#) %M% %I% of %G%.
```

```
*/
```

```
#include "MACHINES.h"
```

```
#include "BOOLEAN.h"
```

```
#include "CHAR.h"
```

```
#include "MACHINE.h"
```

```
#include "JOB.h"
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
#include <errno.h>
```

```
#include "BOOLEAN.h"
```

```
#include "NAT.h"
```

```
fGET_FREE_MACHINE(V0JOB,V0MACHINES,  
                  V0M,V0MS)
```

```
IDECLARE_JOB(V0JOB)
```

```
IDECLARE_MACHINES(V0MACHINES)
```

```
ODECLARE_MACHINE(V0M)
```

```
ODECLARE_MACHINES(V0MS)
```

```
{
```

```
/* __LOCAL_VARIABLE_DECLARATIONS__ */
```

```
DECLARE_CHAR(C1)
```

```
DECLARE_MACHINES(V0MSM)
```

```
DECLARE_CHAR(C3)
```

```

DECLARE_MACHINES(V0MSN)
DECLARE_CHAR(C5)
DECLARE_BOOLEAN(V0WILLDO)
DECLARE_MACHINE(V0M0)
DECLARE_MACHINES(V0MS1)
DECLARE_CHAR(C9)
DECLARE_BOOLEAN(D0D1)
DECLARE_BOOLEAN(D0D4)
DECLARE_BOOLEAN(D0D8)
        /* __ITERATION_VARIABLE_DECLARATIONS__ */
int rec4FIND;
DECLARE_JOB(R40JOB)
DECLARE_MACHINES(R40MS1)
        /* __CONSTANT_DECLARATIONS_AND_ASSIGNMENTS__ */
DOT_K_CHAR('R',C1)
DOT_K_CHAR('L',C3)
DOT_K_CHAR('R',C5)
DOT_K_CHAR('<',C9)
        /* __FUNCTION_SOURCE_CODE_BEGINNING__ */
ISEMPTY_MACHINES(V0MACHINES,D0D1)
    if(D0D1<1)
{if(D0D1 == REJECT_BOOLEAN) {REJECT_TEST_BOOLEAN()}
  NEXT_MACHINES(C9,V0MACHINES,V0MS1)
R40JOB=V0JOB;
R40MS1=V0MS1;
rec4FIND=1;
while(rec4FIND--){
    ATNULL_MACHINES(V0MS1,D0D4)
    if(D0D4<1)
{if(D0D4 == REJECT_BOOLEAN) {REJECT_TEST_BOOLEAN()}
  MOVETO_MACHINES(V0MS1,V0M0)
  FWILL_MACHINE_DO_JOB(V0JOB,V0M0,&V0WILLDO);
  CLONE_BOOLEAN(V0WILLDO,D0D8)
    if(D0D8<1)
{if(D0D8 == REJECT_BOOLEAN) {REJECT_TEST_BOOLEAN()}
  NEXT_MACHINES(C5,V0MS1,V0MSN)

```

```
    rec4FIND=1;
    V0JOB=V0JOB;
    VOMS1=VOMSN;
}/*FALSE*/
else{/*WILLDOJOB*/
    NEXT_MACHINES(C3,VOMS1,VOMSM)
    EXTRACT_MACHINES(C1,VOMSM,*VOM,*VOMS)
}/*TRUE*/
}/*FALSE*/
else{/*ATNULL*/
    CLONE_MACHINES(VOMS1,*VOMS)
    KREJECT_MACHINE(VOMS1,*VOM)
}/*TRUE*/
}
V0JOB=R40JOB;
VOMS1=R40MS1;
}/*FALSE*/
else{/*NO_MACHINES_AVAILABLE*/
    CLONE_MACHINES(VOMACHINES,*VOMS)
    KREJECT_MACHINE(VOMACHINES,*VOM)
}/*TRUE*/

return;
}
/* ----- end of source -----*/
```