

# tiempos estándares mediante microcomputadoras: la aplicación de las microcomputadoras a la medición del trabajo

MARCOS MOYA N.\*

## RESUMEN

*Este artículo presenta un programa de computadora que determina tiempos estándares mediante el sistema de tiempos predeterminados MTM-2.*

*Trabaja en forma interactiva y deben dársele como datos los movimientos por ejecutar y las distancias o pesos por transportar. El programa arroja como resultado el tiempo estándar obtenido en TMU, segundos y minutos.*

## INTRODUCCION

Ultimamente se ha visto cómo la generación de microcomputadoras está teniendo gran auge en el mercado, debido a que las casas fabricantes están ofreciendo gran cantidad de equipo de bajo costo. Esto está permitiendo que los paquetes de programas (Software) que antes tenían que ser adquiridos en el exterior a precios muy altos, puedan ahora ser diseñados en el país con la consiguiente disminución en los costos.

Desde la época de Taylor, se ha considerado la conveniencia de definir tiempos estándares a las divisiones básicas de una actividad u operación como una forma de elevar la eficiencia de la administración de las organizaciones industriales.

Desde 1945 ha venido aumentando el interés en el uso de movimientos básicos sintéticos, como un método rápido y moderno para estimar tiempos estándares sin el uso de cronómetros u otros instrumentos de medición de tiempo.

Con el sistema de tiempos predeterminados se logra una reducción de costos importante cuando

los tiempos de ciclo de las operaciones de producción son muy cortos. Esto por cuanto, si usamos el muestreo para determinar los estándares, por lo corto del tiempo de ciclo de la operación, el porcentaje de error de los datos es muy alto, lo que causaría un aumento desproporcionado en el tamaño de la muestra. Esto lógicamente, produce un impacto en los costos, elevándolos por el mayor volumen de información que es preciso procesar.

Otra de las ventajas del sistema de tiempos predeterminados es que permite fijar tiempos estándares aún cuando las operaciones sean nuevas (no se tienen estimaciones previas).

En este artículo se pretende poner a disposición de los usuarios de las técnicas de tiempos predeterminados, un programa de computadora que calcula tiempos estándares mediante el método del MTM-2 (Measurement Time Method-2).

Este programa ha sido desarrollado para usarse en una microcomputadora que disponga de unidad de diskette, pues está escrito en lenguaje Fortran y por tanto requiere de este compilador para su ejecución.

## MTM-2

Como un esfuerzo para extender la aplicación del MTM a áreas de trabajo donde MTM-1 resultaría muy costoso, la Dirección Internacional de la Asociación MTM inició un proyecto de investigación para desarrollar datos que fueran menos detallados, pero apropiados para la mayor parte de las secuencias de movimientos. De esta investigación resultó el MTM-2, que consiste de:

- Movimientos MTM básicos sencillos
- Combinaciones de movimientos MTM básicos

El MTM-2 tiene aplicaciones en aquellos trabajos donde:

Profesor del Departamento de Producción Industrial del Instituto Tecnológico de Costa Rica.

- El tiempo de ciclo de trabajo es de más de un minuto de duración
- El ciclo no es altamente repetitivo
- La parte del ciclo que es manual no implica un gran número de movimientos manuales complejos o simultáneos.

MTM-2 considera 11 categorías que son:

CATEGORIA	SIMBOLO
Get (obtener)	G
Put (poner)	P
Get weight (tomar peso)	GW
Put weight (poner peso)	PW
Regrasp (volver a asir)	R
Apply pressure (aplicar presión)	A
Eye action (acción del ojo)	E
Foot action (acción del pie)	F
Step (paso)	S
Bend & arise (doblar y subir)	B
Crank (acción de manivela)	C

En MTM-2 las distancias se estiman por clases y afectan los tiempos de las categorías Get y Put.

En estas dos clases de movimientos, tres variables afectan al tiempo requerido. Estas son:

- el caso considerado
- la distancia recorrida
- el peso manejado

El programa que aquí se presenta está basado en el MTM-2. Las tablas de MTM-2 proveen los tiempos para cada una de las categorías arriba descritas.

Hay cinco segmentos del programa que proveen los tiempos para las 11 categorías ya descritas.

- El segmento -GET- provee los tiempos para la categoría G.
- El segmento -PUT- provee los tiempos para la categoría P.
- El segmento -GETW- provee los tiempos para la categoría GW.
- El segmento -PUTW- provee los tiempos para la categoría PW.
- El segmento -MOV- provee los tiempos para las categorías A, R, E, C, S, F y B.

Cada uno de estos movimientos, así como las

variables requeridas, son pedidas por el programa al usuario.

El programa que se incluye como anexo está limitado a 50 elementos para una operación. Sin embargo este número puede ser aumentado si la capacidad del computador lo permite. Esto se logra modificando el estatuto DIMENSION. Está escrito en lenguaje FORTRAN IV y fue probado en una microcomputadora APPLE II.

Lo que el programa puede hacer se muestra en el siguiente ejemplo:

Ejemplo de aplicación práctica:

Considérese una persona que se encuentra ensamblando un componente y que realiza los siguientes movimientos:

Mano izquierda	Mano derecha
G15A	G25B
P15C	P30A
A	A
GW5	GW5
PW5	PW5

El usuario inserta su diskette y teclea < X > MTM-2. En la pantalla aparece este mensaje:

ITCR  
DEPARTAMENTO DE PRODUCCION  
INDUSTRIAL

MTM-2

Deseas ver los cuadros (SI/NO)

SI

CUADRO DE ELEMENTOS

CODIFICACION	GA	GB	GC	PA	PB	PC
5 .....	3	7	14	3	10	21
15 .....	6	10	19	6	15	26
30 .....	9	14	23	11	19	30
45 .....	13	18	27	15	24	36
80 .....	17	23	32	20	30	41

GW-1/KG PW-1/5 KG

A	R	E	C	S	F	B
14	6	7	15	18	9	61

CUADRO DE HOLGURAS

GEC	GEC5		
PEB	GEC5		
PEC	PEC5	PEC5	PEC5
GEC	PEB	PEC	

DATOS

Quando no desee introducir más datos teclee la palabra NO en el momento en que en la pantalla se le pida el código y la categoría.

Dar el elemento y el código (1 M.IZQ., 2 M.DER.)  
GE1

Dar la distancia y la categoría  
15A

Dar el elemento y el código (1 M.IZQ., 2, M.Der.)  
GE2

Dar la distancia y la categoría  
25B

.  
. .  
. .

Dar el elemento y el código (1 M.IZQ., 2 M.DER.)  
PW2

Dar el peso  
05

Dar el elemento y el código (1. M.IZQ., 2 M.Der)  
NO  
Teclee < RETURN > y aparecerá en la pantalla:

CUADRO DE ELEMENTOS

1	GE	15	A	1
2	GE	25	B	2
3	PE	15	C	1
.				
.				
.				
10	PW	5		2

Desea hacer algún cambio (SI/NO)

SI

Dar el número de elementos que desea cambiar

03

Teclee el nuevo elemento y el nuevo código

PE1

Dar la distancia y la categoría

45C

Al presionar la tecla < RETURN > aparecerá en la pantalla

TABLA DE ELEMENTOS

1	GE	15	A	1
2	GE	25	B	2
3	PE	45	C	1
.				
.				
.				
10	PW	5		2

Desea hacer algún cambio (SI/NO)

NO

Al teclear < RETURN > aparecerá en la pantalla:

MTM—2

TIEMPOS PREDETERMINADOS

<u>MANO IZQUIERDA</u>		<u>MANO DERECHA</u>		
Elemento	Tiempo	Tiempo	Tiempo	Elemento
GE15A	6	14	14	GE25B
PE45C	36	57	11	PE30A
A	14	14	14	A
GW5	5	5	5	GW5
PW5	1	1	1	PW5

El tiempo total de la operación en TMU es: 91

El tiempo total de la operación en minutos es: 0,0546

El tiempo total de la operación en segundos es: 4

FIN

LITERATURA CONSULTADA

Niebel, B.W. **Ingeniería Industrial**, Representaciones y Servicios de Ingeniería, S.A., México, 2ed., 1980.

```

DIMENSION DTA(50),CATT(50),PSSC(50),CODC(50)
DIMENSION TIP(50)
INTEGER DT,COD,TIME,TIME1,TIME2,TMAY(50),D(50),PSO,PSO1
INTEGER DTA,CODD,PSSC
CHARACTER TIPO*2,GE*2,PE*2,GW*2,PW*2,CAT*1,N*2
CHARACTER TIP*2,CATT*1,S*2,DEC*2,A*1,B*1,C*1
DATA GE/'GE'/,PE/'PE'/,GW/'GW'/,PW/'PW'/
DATA D/50*0/,N/'NO'/,S/'SI'/,A/'A'/,E/'E'/,C/'C'/
IADD = 1
WRITE(*,4000)
WRITE(*,3700)
WRITE(*,4300)
READ(*,100) DEC
IF(DEC.EQ.N) GO TO 21
WRITE(*,3500)
WRITE(*,3800)
WRITE(*,3900)
WRITE(*,4600)
21 WRITE(*,4400)
WRITE(*,4500)
DO 11 I = 1,50
WRITE(*,2600)
WRITE(*,4200)
READ(*,100) TIP(I),CODD(I)
TIPO = TIP(I)
IF(TIPO.EQ.N) GO TO 12
IF(TIPO.EQ.GE.OR.TIPO.EQ.PE) WRITE(*,2700)
WRITE(*,4200)
IF(TIPO.EQ.GE.OR.TIPO.EQ.PE) READ(*,200) DTA(I),CATT(I)
IF(TIPO.EQ.GW.OR.TIPO.EQ.PW) WRITE(*,2800)
WRITE(*,4300)
IF(TIPO.EQ.GW.OR.TIPO.EQ.PW) READ(*,300) PSSC(I)
IADD = IADD + 1
11 CONTINUE
12 NELEM = IADD - 1
17 WRITE(*,3500)
WRITE(*,3600)
DO 24 I = 1,NELEM
IF(TIP(I).NE.GE.AND.TIP(I).NE.PE) GO TO 18
WRITE(*,2900) I,TIP(I),DTA(I),CATT(I),CODD(I)
GO TO 24
18 IF(TIP(I).NE.GW.AND.TIP(I).NE.PW) GO TO 19
WRITE(*,3000) I,TIP(I),PSSC(I),CODD(I)
GO TO 24
19 WRITE(*,3100) I,TIP(I),CODD(I)
24 CONTINUE
WRITE(*,3600)
WRITE(*,3200)
WRITE(*,4300)
READ(*,100) DEC
IF(DEC.NE.S) GO TO 22
WRITE(*,3300)
WRITE(*,4300)
READ(*,3400) LL
WRITE(*,4100)
WRITE(*,4200)
READ(*,100) TIP(LL),CODD(LL)
IF(TIP(LL).NE.GE.AND.TIP(LL).NE.PE) GO TO 20
WRITE(*,2700)
WRITE(*,4200)
READ(*,200) DTA(LL),CATT(LL)
GO TO 17
20 IF(TIP(LL).NE.GW.AND.TIP(LL).NE.PW) GO TO 17
WRITE(*,2800)
WRITE(*,4300)
READ(*,300) PSSC(LL)
GO TO 17
22 K = 1
L = 1
ISUM = 0
DO 50 I = 1,NELEM
TIPO = TIP(I)

```

```

COD = CODD(I)
IF(TIPO.NE.GE.AND.TIPO.NE.PE) GC TO 13
DT = DTA(I)
CAT = CATT(I)
GO TO 14
13 IF(TIPO.NE.GW.AND.TIPO.NE.PW) GC TO 16
PSO = PSSO(I)
14 IF(TIPO.EQ.GE.AND.COD.EQ.1) GO TO 1
IF(TIPO.EQ.GE.AND.COD.EQ.2) GO TO 2
IF(TIPO.EQ.PE.AND.COD.EQ.1) GO TO 3
IF(TIPO.EQ.PE.AND.COD.EQ.2) GO TO 4
IF(TIPO.EQ.GW.AND.COD.EQ.1) GO TO 5
IF(TIPO.EQ.GW.AND.COD.EQ.2) GO TO 6
IF(TIPO.EQ.PW.AND.COD.EQ.1) GO TO 7
IF(TIPO.EQ.PW.AND.COD.EQ.2) GO TO 8
16 IF(COD.EQ.1) GO TO 9
IF(COD.EQ.2) GO TO 10
1 CALL GET(DT,CAT,TIME1)
GO TO 15
2 CALL GET(DT,CAT,TIME2)
GO TO 35
3 CALL PUT(DT,CAT,TIME1)
GO TO 15
4 CALL PUT(DT,CAT,TIME2)
GO TO 35
5 CALL GETW(PSO,TIME1)
GO TO 15
6 CALL GETW(PSO,TIME2)
GO TO 35
7 CALL PUTW(PSO,TIME1)
GO TO 15
8 CALL PUTW(PSO,TIME2)
GO TO 35
9 CALL MOV(TIPO,TIME1)
GO TO 15
10 CALL MOV(TIPO,TIME2)
GO TO 35
15 D(K) = TIME1
TMAY(L) = D(K)
NUMEL = MOD(NELEM,2)
IF(I.EQ.NELEM.AND.NUMEL.NE.C) ISUM = ISUM + TMAY(L)
GO TO 40
35 D(K) = TIME2
TMAY(L) = MAX(D(K),D(K-1))
IF(TIP(I).EQ.GE.AND.CATT(I).EQ.C) GC TO 36
IF(TIP(I).EQ.PE.AND.CATT(I).EQ.B) GC TO 36
IF(TIP(I).EQ.PE.AND.CATT(I).EQ.C) GC TO 37
GO TO 35
36 IF(TIP(I-1).EQ.GE.AND.CATT(I-1).EQ.C) TMAY(L)=TMAY(L)+14
GO TO 38
37 IF(TIP(I-1).EQ.PE.AND.CATT(I-1).EQ.E) TMAY(L)=TMAY(L)+21
IF(TIP(I-1).EQ.GE.AND.CATT(I-1).EQ.C) TMAY(L)=TMAY(L)+21
GO TO 39
38 IF(TIP(I-1).EQ.PE.AND.CATT(I-1).EQ.E) TMAY(L)=TMAY(L)+14
39 IF(TIP(I-1).EQ.PE.AND.CATT(I-1).EQ.C) TMAY(L)=TMAY(L)+21
ISUM = ISUM + TMAY(L)
L = L + 1
K = K + 1
40 CONTINUE
SUM = ISUM * 0.0006
ISUM1 = SUM * 60. + 1.
WRITE(*,900)
IF(NUMEL.NE.0) NELEM = NELEM + 1
WRITE(*,2500)
WRITE(*,2400)
WRITE(*,1900)
WRITE(*,2400)
WRITE(*,2000)
DO 99 I = 1,NFLEM,2
J = I/2 + 1
IF(TIP(I).EQ.GE.OR.TIP(I).EQ.PE) GC TO 60
IF(TIP(I).EQ.GW.OR.TIP(I).EQ.PW) GC TO 75

```

```

        GC TO 90
60      IF (TIP(I+1).NE.GE.AND.TIP(I+1).NE.PE) GC TO 65
        WRITE(*,1000) TIP(I),DTA(I),CATT(I),C(I),TMAY(J),D(I+1),TIP(
*I+1),DTA(I+1),CATT(I+1)
        GC TO 99
65      IF (TIP(I+1).NF.GW.AND.TIP(I+1).NE.PW) GC TO 70
        WRITE(*,1100) TIP(I),DTA(I),CATT(I),C(I),TMAY(J),D(I+1),TIP(
*I+1),PSSO(I+1)
        GC TO 99
70      WRITE(*,1200) TIP(I),DTA(I),CATT(I),C(I),TMAY(J),D(I+1),TIP(
*I+1)
        GC TO 99
75      IF (TIP(I+1).NF.GE.AND.TIP(I+1).NE.PE) GC TO 80
        WRITE(*,1300) TIP(I),PSSO(I),D(I),TMAY(J),D(I+1),TIP
*(I+1),DTA(I+1),CATT(I+1)
        GC TO 99
80      IF (TIP(I+1).NF.GW.AND.TIP(I+1).NE.PW) GC TO 85
        WRITE(*,1400) TIP(I),PSSO(I),D(I),TMAY(J),D(I+1),TIP
*(I+1),PSSO(I+1)
        GC TO 99
85      WRITE(*,1500) TIP(I),PSSO(I),D(I),TMAY(J),D(I+1),TIP
*(I+1)
        GC TO 99
90      IF (TIP(I+1).NE.PE.AND.TIP(I+1).NE.GE) GC TO 95
        WRITE(*,1600) TIP(I),D(I),TMAY(J),D(I+1),TIF(I+1),DTA(I+1),C
*ATT(I+1)
        GC TO 99
95      IF (TIP(I+1).NE.GW.AND.TIP(I+1).NE.PW) GC TO 96
        WRITE(*,1700) TIP(I),D(I),TMAY(J),D(I+1),TIF(I+1),PSSO(I+1)
        GC TO 99
96      WRITE(*,1800) TIP(I),D(I),TMAY(J),D(I+1),TIF(I+1)
99      CCNTINUE
        WRITE(*,2100) ISUM
        WRITE(*,2200) SUM
        WRITE(*,2300) ISUM1
100     FORMAT(A2,I1)
200     FORMAT(I2,A1)
300     FORMAT(I2,I1)
400     FORMAT(5X,I5)
800     FORMAT(A1)
900     FORMAT(4(/))
1000    FORMAT(1X,A2,I2,A1,5X,I2,5X,I5,10X,I2,5X,A2,I2,A1)
1100    FORMAT(1X,A2,I2,A1,5X,I2,5X,I5,10X,I2,5X,A2,I2,I1)
1200    FORMAT(1X,A2,I2,A1,5X,I2,5X,I5,10X,I2,5X,A2)
1300    FORMAT(1X,A2,I2,6X,I2,5X,I5,10X,I2,5X,A2,I2,A1)
1400    FORMAT(1X,A2,I2,6X,I2,5X,I5,10X,I2,5X,A2,I2,I1)
1500    FORMAT(1X,A2,I2,6X,I2,5X,I5,10X,I2,5X,A2)
1600    FORMAT(1X,A2,3X,I2,5X,I5,10X,I2,5X,A2,I2,A1)
1700    FORMAT(1X,A2,3X,I2,5X,I5,10X,I2,5X,A2,I2,I1)
1800    FORMAT(1X,A2,3X,I2,5X,I5,10X,I2,5X,A2)
1900    FORMAT(1X,'MANO IZQUIERDA',15X,'MANO DERECHA')
2000    FORMAT(2(/),1X,44('-',))
*/,'ELEM.',4X,
*,'TIEMPO',4X,'TIEMPO',4X,
*,'TIEMPO',4X,
*,'ELEM.',/,1X,44('-',),2(/))
2100    FORMAT(1X,44('-',),/,,' EL TIEMPO TOTAL DE LA OPERAC
*ICN EN TMU ES ',I5)
2200    FORMAT(1X,'EL TIEMPO TOTAL DE LA OPERACION EN MINUT
*OS ES ',F10.5)
2300    FORMAT(1X,'EL TIEMPO TOTAL DE LA OPERACION EN SEGUN
*OS ES ',I5)
2400    FORMAT(1X,14('-',),15X,12('-',))
2500    FORMAT(3(/),15X,'M T M 2',/,8X,'TIEMPOS PREDETERM
*INADOS',3(/))
2600    FORMAT(' DAR EL ELEMENTO Y EL CODIGO')
2700    FORMAT(' DAR LA DISTANCIA Y LA CATEGORIA')
2800    FORMAT(' DAR EL PESO ')
2900    FORMAT(5X,I2,5X,A2,5X,I2,5X,A1,5X,I1)
3000    FORMAT(5X,I2,5X,A2,5X,I2,11X,I1)
3100    FORMAT(5X,I2,5X,A2,16X,I1)
3200    FORMAT(' DESEA HACER ALGUN CAMBIO (SI/NO)')

```

```

3300  FORMAT(' DAR EL NUM. DEL ELEM.GUE DESEAS CAMEIAR.')
3400  FORMAT(I2)
3500  FORMAT(25(/),8X,18(' '),/,8X,'TABLA DE ELEMENTOS',/,8X,18(' '),
*3(/))
3600  FORMAT(5X,28(' '))
3700  FORMAT(1X,'DESEAS VER LAS TABLAS (SI/NO)')
3800  FORMAT(
*
* CODIFICACION      GA      GB      GC      PA      FB      PC
*   5.....          3       7      14      3       10     21
*  15.....          6      10     19      6       15     26
*  30.....          9      14     23     11      19     30
*  45.....         13     18     27     15      24     36
*  80.....         17     23     32     20      30     41
*-----
* GW-1/KG      PW-1/5 KG ',7(/))
3900  FORMAT(
*
*   A   R   E   C   S   F   B
*  14   6   7   15  18   9   61
*-----
*                                     ',5(/))
4000  FORMAT(7(/),
*
*           I T E S M ',3(/),
*   DEPARTAMENTO DE INGENIERIA',///,
*   INDUSTRIAL Y DE SISTEMAS',///,
*   SISTEMA DE TIEMPOS PREDETERMINADOS',///,
*           M T M 2 ',5(/))
4100  FORMAT(1X,'TECLEE EL NUEVO ELEMENTO Y EL NUEVO CODIGO.')
4200  FORMAT('---')
4300  FORMAT('---')
4400  FORMAT(5(/),12X,9(' '),/,12X,'C A T O S',/,12X,9(' '),5(/))
4500  FORMAT(
*
* CUANDO NO DESFE INTRODUCIR',/,
* MAS DATOS TECLEE UN - NO -',/,
* CUANDO LE PIDA EL ELEMENTO',/,
* Y EL CODIGO',5(/))
4600  FORMAT(8X,17(' '),/,8X,'TABLA DE HCLGRAS',/,8X,17(' '),3(/),
*
*   GEC  GEC5 ',/,
*   PEB  GEC5 ',/,
*   PEC  PEC5  FEC5  PEC5 ',/,
*-----
*           GEC  FEB  PEC ',/,
*-----
*)
STOP
END
SUBROUTINE GET(DT,CAT,TIME)
INTEGER DT,TIME
CHARACTER CAT*1,A*1,E*1,C*1
DATA A/'A'/,B/'B'/,C/'C'/
IF(DT.EQ.5) GO TO 10
IF(DT.EQ.15) GO TO 20
IF(DT.EQ.30) GO TO 30
IF(DT.EQ.45) GO TO 40
IF(DT.EQ.80) GO TO 50
10 IF(CAT.EQ.A) TIME = 3
IF(CAT.EQ.B) TIME = 7
IF(CAT.EQ.C) TIME = 14
GO TO 60
20 IF(CAT.EQ.A) TIME = 6
IF(CAT.EQ.B) TIME = 10
IF(CAT.EQ.C) TIME = 15
GO TO 60
30 IF(CAT.EQ.A) TIME = 9
IF(CAT.EQ.B) TIME = 14
IF(CAT.EQ.C) TIME = 23
GO TO 60
40 IF(CAT.EQ.A) TIME = 13
IF(CAT.EQ.B) TIME = 18
IF(CAT.EQ.C) TIME = 27
GO TO 60

```

```

50         IF(CAT.EQ.A) TIME = 17
           IF(CAT.EQ.B) TIME = 23
           IF(CAT.EQ.C) TIME = 32
60         RETURN
END
SUBROUTINE PUT(DT,CAT,TIME)
INTEGER DT,TIME
CHARACTER A*1,E*1,C*1,CAT*1
DATA A/'A'/,B/'B'/,C/'C'/
        IF(DT.EQ.5) GO TO 10
        IF(DT.EQ.15) GO TO 20
        IF(DT.EQ.30) GO TO 30
        IF(DT.EQ.45) GO TO 40
        IF(DT.EQ.80) GO TO 50
10        IF(CAT.EQ.A) TIME = 3
        IF(CAT.EQ.B) TIME = 10
        IF(CAT.EQ.C) TIME = 21
        GO TO 60
20        IF(CAT.EQ.A) TIME = 6
        IF(CAT.EQ.B) TIME = 15
        IF(CAT.EQ.C) TIME = 26
        GO TO 60
30        IF(CAT.EQ.A) TIME = 11
        IF(CAT.EQ.E) TIME = 19
        IF(CAT.EQ.C) TIME = 30
        GO TO 60
40        IF(CAT.EQ.A) TIME = 15
        IF(CAT.EQ.B) TIME = 24
        IF(CAT.EQ.C) TIME = 36
        GO TO 60
50        IF(CAT.EQ.A) TIME = 20
        IF(CAT.EQ.B) TIME = 30
        IF(CAT.EQ.C) TIME = 41
60        RETURN
END
SUBROUTINE GETW(PSO,TIME)
INTEGER TIME,PSC
        TIME = 1 * PSC
        RETURN
END
SUBROUTINE PUTW(PSO,TIME)
INTEGER TIME,PSC,PSC1
        PSC1 = PSO/5.
        TIME = 1 * PSC1
        RETURN
END
SUBROUTINE MOV(TIPO,TIME)
INTEGER TIME
CHARACTER A*2,R*2,E*2,C*2,S*2,F*2,E*2,TIFC*2
DATA A/'A'/,R/'R'/,E/'E'/,C/'C'/
DATA S/'S'/,F/'F'/,B/'B'/
        IF(TIPO.EQ.A) TIME = 14
        IF(TIPO.EQ.R) TIME = 6
        IF(TIPO.EQ.E) TIME = 7
        IF(TIPO.EQ.C) TIME = 15
        IF(TIPO.EQ.S) TIME = 18
        IF(TIPO.EQ.F) TIME = 9
        IF(TIPO.EQ.B) TIME = 61
        RETURN
END

```