

CALL  
CALL

#### Format

CALL subprogram-name[(parameter-list)]

#### Cross Reference

SUB

#### Description

The CALL instruction transfers program control to the specified subprogram.

You can use CALL as either a program statement or a command.

The CALL instruction transfers program control to the subprogram specified by the subprogram-name.

The optional parameter-list consists of one or more parameters separated by commas. Use of a parameter-list is determined by the subprogram you are calling. Some subprograms require a parameter-list, some do not use a parameter-list, and with some a parameter is optional.

You can use CALL as a program statement to call either a built-in MYARC Advanced BASIC subprogram or to call a subprogram that you write. After the subprogram is executed, program control returns to the statement immediately following the CALL statement.

You can use CALL as a command only to call a built-in MYARC Advanced BASIC subprogram, not to call a subprogram that you write.

Each of the following built-in subprograms is discussed separately in this manual:

BCOLOR	ECOLOR	LOAD	PEEK	SPRITE2
BTIME	ERR	LOCATE	PEEKV	SPRITESET
CHAR	FILL	LPR	POINT	STCR
CHARPAT	FILES	MAGNIFY	POKEV	TCOLOR
CHARSET	GCHAR	MARGINS	POSITION	TIME
CIRCLE	GPOINT	MEMSET	PSET	VCHAR
CLEAR	GRAPHICS	MKEY	RECTANGLE	VERSION
COINC	HCHAR	MLOC	RESETPLT	
COLOR	HIDEMOUSE	MOTION	SAY	
DATE	INIT	MOUSEDRAG	SCREEN	
DCOLOR	INP	MREL	SEEMOUSE	
DELSprite	JOYST	MYART	SCHAR	
DISTANCE	KEY	OUTP	SOUND	
DRAW	LDCR	PALETTE	SPGET	
DRAWTO	LINK	PATTERN	SPRITE	

#### Examples

```
CALL GRAPHICS(4)
CALL LINK("filename" [,parameters])
CALL RESETPLT
```