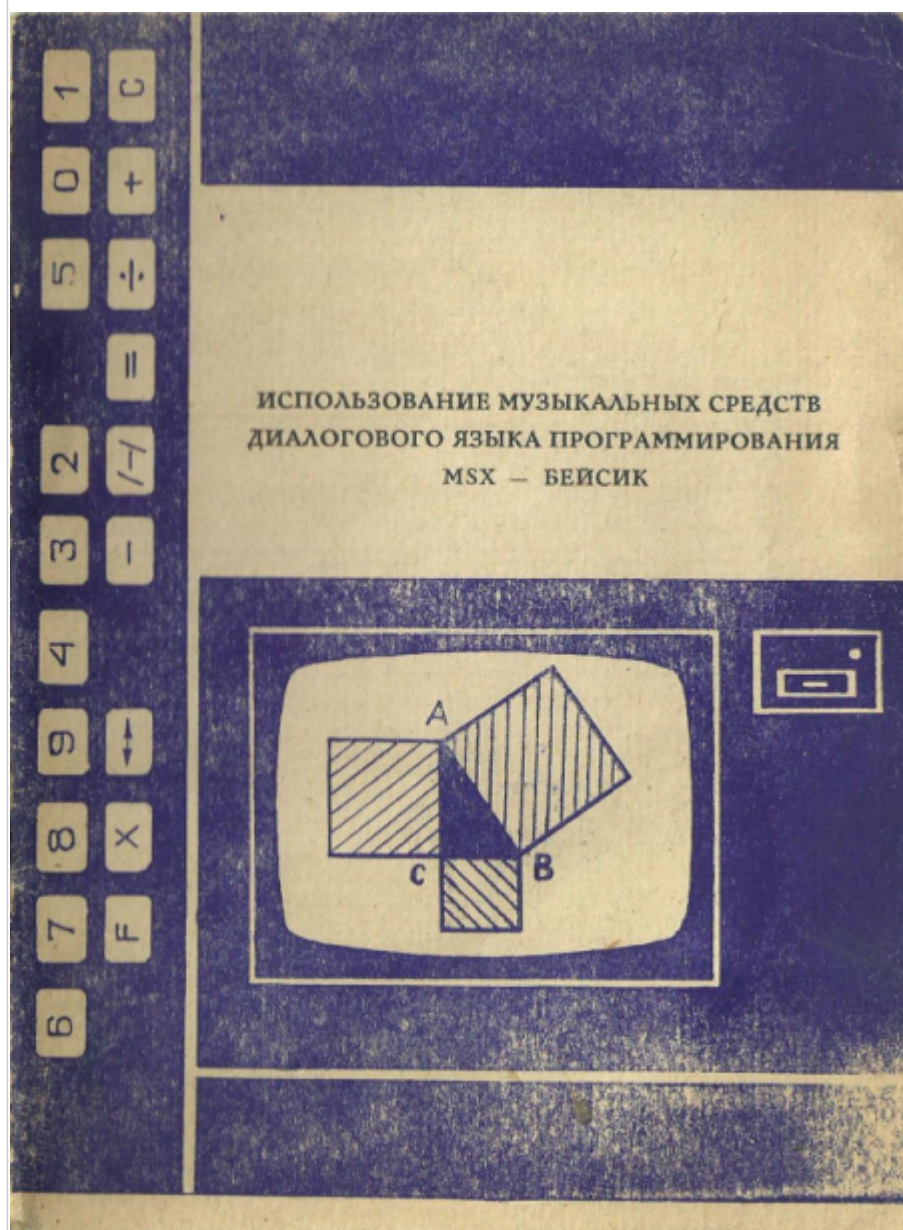


□ Использование музыкальных средств диалогового языка программирования MSX BASIC



Министерство просвещения СССР. Омский ордена «Знак Почета» Государственный педагогический институт имени А.М. Горького

Методические рекомендации для студентов всех специальностей педагогических институтов

Омск 1987

Утверждено Управлением учебных заведений Министерства просвещения СССР.

© Министерство просвещения СССР, 1987 г., – 36 стр.

Подписано к печати 5.05.87 Формат 60х90/16

Заказ 178 Тираж 660 Объем 1,75 п.л.

Бесплатно Оперативная печать. Бумага газетная

Отпечатано в Межвузовской типографии ОмПИ



Методические рекомендации адресованы студентам вузов, учащимся средних учебных заведений, а также всем желающим ознакомиться с музыкальными средствами ведущего диалогового языка программирования MSX BASIC. В них обсуждаются синтаксис и семантика основных команд и функций макро-языка MML, приведены примеры иллюстративного характера и тексты программ конкретных музыкальных произведений.

Рекомендации рассчитаны на использование персонального MSX-компьютера.

Составители: А.Р. Есаян, Л.П. Лапицкая, Т.Н. Карякина (Тульский государственный педагогический институт имени Л.Н. Толстого), В.И. Ефимов (Минпрос СССР)

Научный редактор: А.Р. Есаян, к.ф.-м.н., доцент ТГПИ

Рецензенты: Д.П. Мокров, д.ф.-м.н., профессор Тульского Ордена Трудового Красного Знамени политехнического института А.Г. Луценко, к.ф.-м.н. Тульского государственного педагогического института им. Л.Н. Толстого

1. Звуковой сигнал

Для синтеза звука в **MSX BASIC** имеются довольно широкие возможности, предоставляемые командами: **BEEP**, **PLAY**, **SOUND**, функцией **PLAY()** и кодом **CHR\$(7)**.

Наиболее простым способом вывода звука является команда:

```
BEEP
```

по которой исполнение текущей музыки прекращается, выдаётся короткий сигнал и устанавливаются в начальное состояние все регистры **PLAY** и **SOUND**. В трёх случаях команда **BEEP** реализуется автоматически: при выдаче сообщения об ошибке, нажатии на клавишу **STOP** и при выполнении команды **STOP**.

Команда

```
PRINT CHR$(7)
```

реализуется аналогично команде **BEEP**.

Команду **SOUND** мы описывать не будем, в лишь кажем, что она обеспечивает доступ к 14 имеющимся регистрам звукового генератора и используется для создания разнообразных звуковых эффектов типа: «гудок паровоза», «падающая бомба», «взрыв» и т.п. (см. [1]).

Другие «звуковые» возможности **MSX BASIC** подробно описаны в следующем разделе.

2. Музыкальный макро-язык

Всякая мелодия представляет собой организованную последовательность звуков, отличающихся друг от друга высотой, длительностью, громкостью и некоторыми другими параметрами. Общепринятая стандартная нотная запись музыки позволяет отображать слуховые явления в условные зрительные формы. Певец или музыкант, интерпретируя ноты реализует обратный процесс превращения их в звуки различной высоты и длительности, связанные между собой музыкальным смыслом.

Язык MML(Musical Macro-Language) также создан для наглядного изображения звуков и их характеристик и в этом смысле он аналогичен стандартной нотной записи. Однако любое произведение на MML легко интерпретируется не только человеком, но и компьютером. Для этого, разумеется, оно должно быть зафиксировано в памяти ЭВМ. Простые и эффективные способы синтеза звука с помощью средств MML обеспечивают

возможность одновременного исполнения до 3 голосов. Причём музыка может звучать вместе с выполнением интерпретатором последующих команд программы. Это важное свойство позволяет использовать MML не только для простого исполнения конкретных музыкальных произведений, но и для организации фонового звукового сопровождения при работе с педагогическими программными средствами.

Реализуется вывод музыки командой

```
PLAY α [β][γ]
```

где:

- PLAY (играть) — служебное слово;
- α, β, γ — строковые выражения, значения которых представляют последовательности команд на MML.

Количество строковых выражений в команде PLAY определяет соответственно музыку для одного, двух или трёх голосов. Некоторые команды MML имеют локальное действие. Например, «проигрывание» конкретной ноты. Другие фиксируют режим исполнения музыки и действуют до появления соответствующей команды переопределения этого режима. Команды MML могут следовать друг за другом непрерывно или через разделители. В качестве последних разрешается использовать одиночные знаки «_» (пробел) или «;» или их последовательности. Наиболее наглядной и не слишком громоздкой получается запись в том случае, когда разделителем является пробел и вставляется он не между любыми командами, а лишь между группами из разных тактов. Разделитель «;» не может располагаться за символом «.».

Пример 1 В значениях, присваиваемых символьным переменным A\$, B\$, C\$ команды MML расположены соответственно подряд, через пробел и через символ «;».

```
10 A$="CE-G05C2.04B-A-GF2.03B-04DFB-2.A-GFE-2."
20 B$="05 E2 D C D2 C D E2 G E D2. R4"
30 C$="05;E2;D;C;D2;C;D;E2;G;E;D2. R4"
```

Теперь можно непосредственно приступить к описанию синтаксиса и семантики команд MML. При этом мы будем опираться на стандартную нотную запись музыкальных произведений и некоторые «азбучные» элементы музыкальной грамоты. Методику изложения учебного материала будем строить так, чтобы дать пользователю инструмент механического перевода обычной записи музыки на MML.

2.1 Обозначения звуков

Основные звуки в MML имеют обычное название и записываются буквами (нотами):

C	D	E	F	G	A	B
до	ре	ми	фа	соль	ля	си

Для модификации основных нот используют так называемые знаки альтерации: диоз и бемоль. Их запись в MML и стандартной форме и значение указаны в [таблице 1](#).

Таблица 1
Знаки модификации основных нот

Название знака альтерации	Запись в MML	Запись в стандартной форме	Назначение
диоз	# или +		повышение на 1/2 тона
бемоль	-		понижение на 1/2 тона

На [рис. 1](#) приведены обозначения и названия нот и их соответствие клавишам одной октавы рояля (см. [табл. 2](#)).

Рекомендуем обратить внимание на локальный характер действия в MML рассмотренных выше знаков диоза и бемоля. При обычном нотном письме эти знаки альтерации, вообще говоря, имеет глобальный характер. Они бывают

двух типов: ключевые и случайные. Первые из них выставляются на основных или дополнительных линиях нотного стана или между ними сразу за скрипичным, басовым или иным ключом и относятся к нотам всего музыкального произведения (или его части). Вторые знаки выставляются непосредственно перед нотой и действуют до конца данного такта. При всех случаях локальная отмена бемоля и диеза реализуется записью перед требуемой нотой


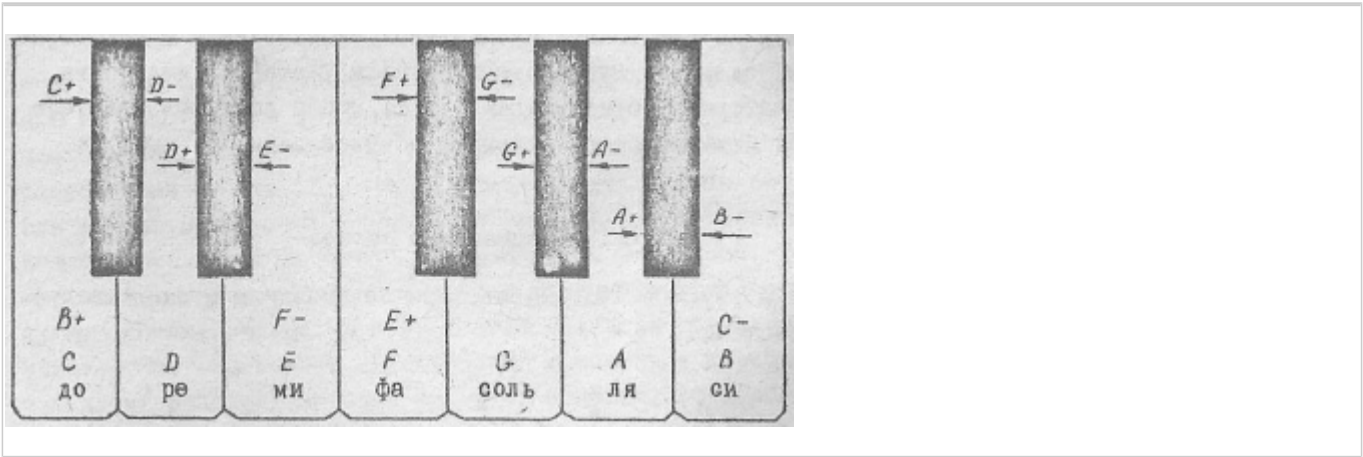
знака бекар «».



Рисунок 1. Обозначения и названия нот и их соответствие клавишам рояля одной октавы



2.2 Высота звука. Октава

Одно из значений термина «октава» характеризует интервал высот звучания группы последовательных нот. В этом смысле октавами называют отрезки звукоряда от до до си включительно, содержащие до 12 полутонов (см. [рис. 1](#)).

C, C+, D, D+, E, F, F+, G, G+, A, A+, B (1)

В MML допускается 8 октав, каждой из которых дано специальное название и обозначение (см. [табл. 2](#)).

По умолчанию в MML реализуется выбор октавы O4 (O—буква), соответствующей первой октаве рояля. Для задания ноты вне текущей октавы необходимо предварительно выполнить команду установки требуемой октавы. Записывается такая команда в одной из следующих форм:

O_μ (2)
O=t;

где

- $\mu \in \{1,2,3,4,5,6,7,8\}$,
- t — числовая переменная со значением μ .

Таблица 2
Название и обозначение октав

№	Стандартное название октав	Название октав в MML	Обозначение октав в MML
1	субконтроктава		
2	контроктава	первая	01
3	большая	вторая	02
4	малая	третья	03
5	первая	четвёртая	04
6	вторая	пятая	05
7	третья	шестая	06
8	четвёртая	седьмая	07
9	пятая	восьмая	08

Выполнение (2) приводит к установке октавы O_μ для всех последующих нот до очередного её переназначения или сброса на 04 командой ВЕЕР.

Необходимо помнить, что C- и B, а также B+ и C соответствуют одной и той же текущей октаве и не выходят за её пределы.

Из сказанного выше вытекает, что звуки фиксируются в MML заданием октавы и ноты (1). Другой способ записи звуков состоит и присваивании каждому из них индивидуального имени. В MML 96 допустимым звукам от самого низкого до самого высокого даны имена:

N1, N2, N3. ..., N96 ,

называемые также нотами (N-нотами). Допустимо и такое задание нот:

$N=t$,

где t — числовая переменная, значение которой — целое число из диапазона 1...96.

Отметим, что имеет место следующее соответствие:

...	...
N36	— "до" 1 октавы (04C)
N37	— { "до диэз" } 1 октавы { [04C+] } { "ре бемоль" } { [04D-] }
N38	— "ре" 1 октавы (04D)
...	...

На рисунке приведено обозначение нот в MML и их расположение на нотном стане в некоторых октавах при скрипичном и басовом ключах.

2.3 Длительность звука

Продолжительность звучания или, по-другому, длительность звука является одним из основных его свойств. Нас будут интересовать длительности звуков по отношению друг к другу.

В нотном письме графическое изображение звука содержит указание на его продолжительность. За единицу длительности принята продолжительность звучания целой ноты «О». По отношению к ней рассматривают такие ноты:

 половинная	 четверть	 восьмая
 шестнадцатая	 тридцать вторая	...

Их длительности в порядке следования равны соответственно 1/2, 1/4, 1/8, 1/16, 1/32 и т.д.

В языке MML длительность всех нот по умолчанию равна 1/4. Её переопределение возможно с любого места по команде:

L_{μ} (3)

где

$\mu \in \{1, 2, 3, \dots, 64\}$ (4)

Выполнение (3) фиксирует длительность всех последующих нот равной $1/\mu$ и сохраняет её до появления новой команды L и сбрасывании регистров по BEEP на L4.

третья октава (в MML O6)

вторая октава (в MML O5)

первая октава (в MML O4)

	C	D	E	F	G	A	B
	до	ре	ми	фа	соль	ля	си

первая октава (в MML O4)

малая октава (в MML O3)

большая октава (в MML O2)



Рис. 2. Обозначение нот в MML и их расположение на нотном стане в некоторых октавах и скрипичном и басовом ключах

С другой стороны, длительность можно назначить и локально, указав её «значение» сразу за нотой в форме

$$\alpha_{\mu} \quad (5)$$

где α — нота, μ — удовлетворяет условию (4)

Соотношение (5) определяет длительность только ноты α , устанавливая её равной $1/\mu$.

Отметим больший приоритет локального задания длительности в самой ноте по сравнению с её глобальной фиксацией по команде (3).

В MML, как и при обычной записи, к нотам можно добавлять любое количество точек, каждая из которых удлинит ноту на половину.

Пример 2. Используя имеющиеся, пока ещё весьма ограниченные средства, мы уже можем «писать» на MML простейшую музыку. Для сопоставления здесь обычная запись украинской народной песни «Солнышко низенько» и снизу её текст по тактам, перекодированный на MML.

Проиграть эту мелодию можно по программе:

[02.bas](#)

[02.bas](#)

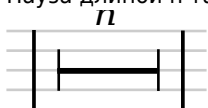
```
10 A$="05E04BB LBBA05C04BAG F+GA4A4
    AF+BAGF+ GAB4B4 BA05C04BAG
    F+GA4A4 AF+BAGF+ L405C04B.D+B
    F+E"
20 PLAY A$
30 END
```

2.4 Паузы

Перерывы звучания в музыкальном произведении задаются паузами, которые подобно нотам имеют определённую длительность обозначаемую при обычном нотном письме особыми знаками:

		
целая	половинная	четверть
		
восьмая	шестнадцатая	тридцать вторая

Пауза длиной n тактов (n — целое) коротко записывается так:



На языке MML пауза задаётся специальной, так называемой незвучащей нотой R. По умолчанию длительность R равна 4 (1/4). Локальные модификации длительности у R проводятся так же, как и у любой другой ноты. Задание длительности командой L на R не распространяется.

Пример 3. Этот фрагмент показывает, как проводится кодирование мелодии на 2 голоса и как выглядит соответствующая программа на MML для её проигрывания. Обратите внимание на большое количество пауз в произведении.

«Журавель» А. Аренский



Программа на MML:

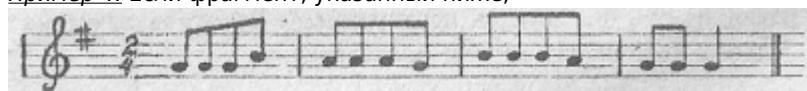
[03.bas](#)

 [03.bas](#)

```
10 A$="L8FR8GR8AR8B-R8 A05C04G4R2 FR8GR8AR8B-R8
    A05C04G4R2 EGEFGA4 R1 A05C04A05C04B-05CDF R1"
20 B$="L8R1 R2FDC4 R1 R2FDC4 R1 EGEGFEDC R1 AR8GRBF2"
30 PLAY A$,B$
```

Отметим возможность использования в MML малых по длительности пауз для разделения друг от друга нот одной высоты. При вставке подобных пауз в текст для сохранения размера такта требуется уменьшение длительности соответствующих нот. Иногда это приводит к «разбуханию» программы.

Пример 4. Если фрагмент, указанный ниже,



закодировать обычным способом,

```
L8GGGB AAAG BBBG GGG4
```

то при проигрывании одинаковые ноты каждого такта будут сливаться в единый звук. Подправленная запись с промежутком между нотами, равным 1/32, выглядит так:

2.5 Темп

Скорость исполнения музыки называют темпом. Темп произведения зависит от его содержания, характера и настроения. Правильный темп — важное условие выразительного звучания музыки. При обычной нотной записи наиболее часто темп обозначают итальянскими терминами, хотя многие композиторы указывают его на родном языке. Ниже приведён список некоторых итальянских слов, используемых для задания темпа и их значения на русском языке.

1	Largo	очень широко
2	Lento	весьма медленно
3	Adagio	медленно
4	Andante	не спеша
5	Moderato	умеренно
6	Andantino	неторопливо
7	Allegretto	оживлённо
8	Allegro	скоро
9	Vivo	живо
10	Presto	очень быстро
11	Accelerando	ускоряя
12	Stringendo	
13	Ritardando	замедляя
14	Ritardando	
15	Molto	очень
16	Assai	
17	Non troppo	не слишком
18	Poco	немного
19	Tempo rubato	свободный темп
20	Ad libitum	по желанию
21	A piacere	

Отметим, что словесное задание темпа носит весьма приблизительный характер. В MML темп может быть равен от 32 до 255. По умолчанию он равен 120. Его переопределение возможно с любого места по одной из следующих команд:

T_{μ} (6)
T=t;

где $\mu \in \{32,33,...,255\}$,
t — числовая переменная со значением μ .

Выполнение (6) приводит к установке темпа μ до очередного его переназначения по (6) или сброса на 120 командой ВЕЕР.

Пример 5.





2.6 Громкость

Громкость является одной из характеристик звука и в обычной нотной записи задаётся динамическими оттенками, определяющими силу звучания. Практически всё разнообразие динамических оттенков можно отнести к 6 основным градациям:

ff	fortissimo	очень громко
f	forte	громко
mf	mezzo forte	умеренно громко
mp	mezzo piano	умеренно тихо
p	piano	тихо
pp	pianissimo	очень тихо

Переход от одной градации к другой отмечается терминами и знаками:

crescendo		постепенно усиливая
diminuendo		постепенно стихая
decrescendo		

Как и словесное определение темпа задание силы звучания весьма условно. В MML громкость может быть равна от 0 до 15. По умолчанию она равна 8. Её переопределение возможно с любого места по одно из следующих команд:

$$V_{\mu} \quad (7)$$

$$V=t;$$

где $\mu \in \{0,1,\dots,15\}$,
 t — числовая переменная со значением μ .

Выполнение (7) приводит к установке громкости μ до очередного его переназначения по (6) или сброса на 8 командой VEER.

Пример 6.

V6T170LB05C4. и т.д.

2.7 Музыкальные подпрограммы

Для того, чтобы компактно записать на MML музыкальное произведение имеется специальный механизм обращения из значения α (β , γ) строкового выражения **PLAY** к подпрограммам на MML, являющимися значениями строковых переменных. Команда обращения в «подпрограмме» $\mu\$$ может находиться в любом месте α и записывается в виде

$$X\mu\$; \quad (8)$$

При этом $\mu\$$ может также содержать команду типа (8) и т.д.

Использование «подпрограмм» диктуется, как правило, наличием повторяющихся фрагментов в произведении или желанием сделать его запись обозримой и легко читаемой.

Пример 7. Вернёмся к программе [примера 2](#) и выделим в ней подпрограмму P\$. Тогда текст мелодии можно переписать в виде:

07.bas

 07.bas

```
10 P$="04L8BA05C04BAG F+GA4A4 AF+BAGF+"
20 A$="05E04BBXP$;GAB4B4XP$;"
30 PLAY A$
40 B$="L405C04B.D+B F+ER4"
50 PLAY B$
```

Обратите внимание на наличие в программе двух операторов PLAY. Дело в том, что использование подпрограмм всегда приводит к существенному уменьшению общей возможной длины музыкальных фрагментов, подготавливаемых для проигрывания одной командой PLAY. И в данном случае мы не смогли бы обойтись командой:

```
PLAY A$+B$
```

Конец фрагмента по ней звучать не будет.

2.8 Некоторые дополнительные команды

Для создания различного рода эффектов можно использовать дополнительные средства языка MML.

По любой из команд: M_μ

$M=t$;

где

- $\mu \in \{1, 2, 3, \dots, 32767\}$,
- t — числовая переменная со значением μ ,

производится установка несущей частоты звукового генератора на значение μ . По умолчанию $\mu=255$.

По любой из команд: S_μ

$S=t$;

где

- $\mu \in \{0, 1, \dots, 15\}$,
- t — числовая переменная со значением μ ,

устанавливается определённая форма волны (μ — волна). По умолчанию $\mu=13$.

Установки как M , так и S , действуют до их очередного переназначения или сброса по команде ВЕЕР. Кроме того, S и M сбрасываются при выполнении команды V. Поэтому используя S и M , мы фактически теряем контроль над громкостью и с этим необходимо считаться.

Для придания звуку оттенка «стакато» (отрывистости) значения строковых переменных PLAY можно назначить командами:

```
S3M3000
```

Пример 8.



08.bas

 08.bas

```
10 A$="S3M300005CC04BB AAS13M255G2 S3M300005FFEE DDRXA1$;"
20 A1$="S13M255G2S3M3000EC S13M25504G2S3M3000EC S13M255D2R2 C2R2"
30 PLAY A$
```

Другой пример формирования оттенка «стаккато» — это уменьшение длительности звучания соответствующих нот на малую величину (1/64) и добавление за счёт этого пауз.

2.9 Музыкальные очереди

При выполнении команды `PLAY α [β][γ]` значения строковых выражений преобразуются во внутренний формат или, как будем говорить, для каждого голоса создается своя «музыкальная очередь». Делается это довольно быстро. Но размер соответствующей области памяти для хранения очередей мал. Поэтому обычно `PLAY` реализуется за несколько приёмов. Сразу после того, как очереди сформированы, генератор начинает «проигрывать» их. Затем в очереди загоняется следующая часть музыкального фрагмента и т.д. до тех пор, пока не будут выполнены все команды `MMI`, указанные в α (β, γ). Заметим, что работа по программе возобновляется только при начале проигрывания последних очередей для данной команды `PLAY`.

Длины очередей зависят от количества операндов в `PLAY`.

Пример 9.

По приведённой программе «под музыку» проводится индикация квадратов натуральных чисел от 1 до 500. Причем. Вывод организуется не сразу после запуска программы, а только вместе с исполнением последней очереди музыкального фрагмента (от команд второй строки в значении `A$`).

[09.bas](#)

 [09.bas](#)

```
10 A$="T90e805e.f8g8.f16e8.f16d2.o4eo5d.
    e8f8.e16d8.e16c2.o4ao5g.a8b-8a8
    g8a8f2.da.b8o6c8o5b8a8b8g#2."
20 B$="T80r4o2ao3ao4co3bo2a8o3f8o4d8c8o3
    b2o2g+8o3e8o4deo3g+o2ao4eeeo2ao4
    ec+eo3d8a8o4e8d8o3a2f8a8oedbc8o3
    do4eo3b"
30 C$="T80r8r1r1r1r4o3aaar4o4c#o3gar2d2r2.ae2."
40 PLAY A$,B$,C$
50 FOR K=1 TO 500:PRINT K^2;:NEXT
```

С помощью булевой функции

`PLAY (n)`

где $n \in \{0, 1, 2, 3\}$, можно определить, есть ли хотя бы в одной очереди музыка ($n = 0$) или, есть ли она в конкретной очереди n ($n = 1, 2, 3$). При наличии музыки функция `PLAY` принимает значение `TRUE` (истина), а при отсутствии её `FALSE` (ложь).

Пример 10. Здесь функция `PLAY` используется для задержки выполнения программы, пока играет музыка.


[10.bas](#)

 [10.bas](#)

```
10 ' ПОДМОСКОВНЫЕ ВЕЧЕРА          В.Соловьев-Седой
20 ' -----
30 A$="T80L804CE-GE- F4E-D G4F4 C2 E-GB-16.R32B- 05C404B-A- G2XA1$;"
40 A1$="D4A4B405DC04G4 GD4C G.F16A-4 A-4B-A-G4FE- G4F4"
50 PLAY A$:PLAY "05C2"
60 PLAY A1$+"04C2CR32"
70 IF PLAY(1) THEN 70
80 ' продолжение программы
90 ' ... ..
```

Пример 11. Здесь функция `PLAY` используется для многократного проигрывания музыкального фрагмента (выход по `CTRL+STOP`).

[11.bas](#)

 [11.bas](#)

```
10 ' МЕНУЭТ                      Х.Хаслер
```

```

20 '-----
30 A$="V6T140L4B-05D04B- F2. 05CE-C 04F2.XA1$;"
40 B$="V6T140L403B-2. B-04D03B- F2. F04E-CxB1$;"
50 A1$="05L8D04B-05E-04B-05F04V7B- 05V8F+GFE-DC 04B05CDE-04G05C
    04B-2A4xA2$;"
60 B1$="03B-V7GV8D E-2. E-CE- F02F03FxB2$;"
70 A2$="V9B-405D04B-AB- F2. D5C4E-C04B05C 04F2."
80 B2$="B-2. B-04D803B-8A8B-8 F2. F04E-8C803A8F8"
90 A3$="05D04B-V1005E-04B-V1105F04B- 05F+GFE-V10DC L4V9DVBE-V7C
    04V6B-R2"
100 B3$="B-V10GV11D E-2V10E- V9FV802FV703F 02B-02B-R"
110 C3$="V6T140R2. R2. L404B-05C04A R2."
120 PLAY A$,B$
130 PLAY A3$,B3$,C3$
140 IF PLAY(0) THEN 140 ELSE 120

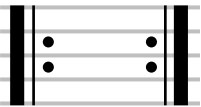
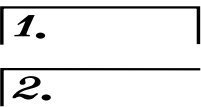




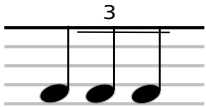
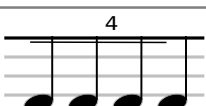
```

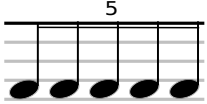

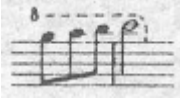


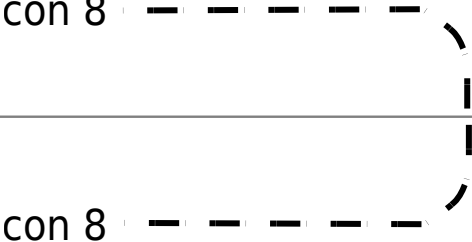

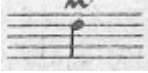

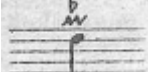

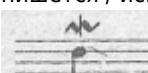





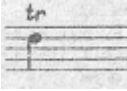
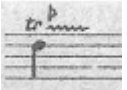
2.10 Примеры программ

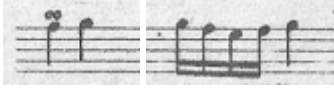
Пусть задана нотная запись музыкального произведения. Учитывая знаки сокращения, применяемые в нотном письме, часть из которых приведены в [таблице 3](#), нетрудно перекодировать это произведение на языке MML. Причём сделать это может даже человек, имеющий весьма поверхностное представление о музыкальной грамоте, но, разумеется, освоивший язык MML и весьма аккуратно реализующий соответствующее кодирование.

Таблица 3

Некоторые знаки сокращения, применяемые в нотном письме (аббревиатура)

№	Знак	Название	Пояснения и (или) примеры
1		Реприза(знак повторения)	Повторяющийся нотный текст, заключённый между этими знаками
2		Вольта	Обозначение, применяемое вместе со знаком репризы и указывающее на различие в окончании какого либо фрагмента при его повторении. Первый раз исполнение проводится до репризы а при повторе часть «1» опускается и играется окончание «2»
3		Знак повторения такта	Повторяется текст предыдущего такта
4		Знак повторения двух тактов	Повторяется текст двух предыдущих тактов
5		Знак повторения последней связки	
6		Дуполь	Ритмическая фигура, состоящая из 2 нот и равная по времени звучания 3 обычным нотам той-же тотальности
7		Триполь	Ритмическая фигура, состоящая из 3 нот и равная по времени звучания 2 обычным нотам той-же тотальности
8		Квартоль	Ритмическая фигура, состоящая из 4 нот и равная по времени звучания 3 обычным нотам той-же тотальности


№	Знак	Название	Пояснения и (или) примеры
9		Квинтоль	Ритмическая фигура, состоящая из 5 нот и равная по времени звучания 4 обычным нотам той-же тотальности
10		Перенос звуков на октаву вверх	
10		Перенос звуков на октаву вниз	
11		Знаки удвоения	Проигрываются ноты и одновременно соответствующие ноты октавой выше (ниже)
12		Мордент простой	пишется / исполняется 
12			пишется / исполняется 
13		Мордент перечёркнутый	пишется / исполняется 
13			пишется / исполняется 
14		Форштаг	Маленькая нота перед «нормальной» нотой. Если форштаг не перечёркнут, то он называется долгим и отнимает обычно половину длительности ноты, перед которой он стоит. Если форштаг перечёркнут, то он называется коротким и заимствует незначительную часть ноты перед которой стоит.
14			
15		Трель	пишется / исполняется 
15			пишется / исполняется 

№	Знак	Название	Пояснения и (или) примеры
16	∞	Группетто	<p>Мелодическое украшение из 4-5 нот. Возможны различные варианты символа ∞ в зависимости от его расположения, наличие перечёркивающей линии и меток альтерации. Ограничимся одним примером. пишется / исполняется</p> 

Ниже приведены тексты двух программ, по которым проигрываются романсы А.Петрова [4].

Пример 12.

[12.bas](#)

 [12.bas](#)


```

10 ' ПРИМЕР 12
20 ' -----
30 '      ПОД ЛАСКОЙ ПЛЮШЕВОГО ПЛЕДА
40 '      А.Петров      на стихи М.Цветаевой
50 ' -----
60 A$=" ":C$=" ":B$="V7T80L802EB03GEBGEG"
70 A1$="T80L8 R805G16.R32G16.R32GF#ED04AxA2$;"
80 B1$="02A03E04C03GDF#04C03F#xB2$;"
90 A2$="05C404B4R4.B 05C04BA#B05F#4.D#xA3$;"
100 B2$="02G03DGDBGDG 02B03D#AD#BAF+AxB3$;"
110 A3$="04B1 R805G16.R32G16.R32GF#ED04A"
120 B3$="02EB03GEBGEG 02A03E04C03GDF#04C03F#"
130 A4$="05C404B4R8B05DC 04B2R8AGAxA5$;"
140 B4$="02G03DGDBGDG F#A04D#03A02B03A04D#03F#xB5$;"
150 A5$="B2R8B16.R32B05C D16.R32DE04B05D4C4"
160 B5$="FA04D03FEG#04D03G# 02B03FG#04D03CE04C03G"
170 A6$="V8R8R16.R32E16.R32E16.R32E16.R32EGC# V9F#1xA7$;"
180 B6$="V802B-03EG04C02F#03C#EA# V902B03DF#B02A#03DF#BxB7$;"
190 A7$="F#16.R32F#GA06C.05B32R32BB32F#16. A4G4"
200 B7$="02A03DF#04C02D03DF#04C 02G03DBD"
210 A8$="04R8B05C#D#G.F#32R32F#F#32D#16.F#4E4R8EF#G"
220 B8$="02B03F#AF#04D03AF#02B 03CE04C03G04EC03EG"
230 A9$="V8C2R804F#GA V7B1"
240 B9$="V802A03EF#G04C403B4 V702EB03GEBGEG"
250 X$="V805C2R804F#GA V7B1"
260 Y$="V802A03EF#G04C403B4 V702EB03GEBG04EG BGB05EB2"
270 PLAY A$,B$,C$
280 PLAY A1$,B1$,C1$:PLAY A4$,B4$,C4$
290 PLAY A6$,B6$,C6$:PLAY A8$,B8$,C8$
300 PLAY A9$,B9$,C9$
310 PLAY A1$,B1$,C1$:PLAY A4$,B4$,C4$
320 PLAY A6$,B6$,C6$:PLAY A8$,B8$,C8$
330 PLAY A9$,B9$,C9$
340 PLAY A1$,B1$,C1$:PLAY A4$,B4$,C4$
350 PLAY A6$,B6$,C6$:PLAY A8$,B8$,C8$
360 PLAY X$,Y$,Z$

```

Пример 13.

[13.bas](#)

 [13.bas](#)

```

10 ' ПРИМЕР 13
20 ' -----
30 '      А НА ПОСЛЕДОК Я СКАЖУ
40 '      А.Петров      на стихи Б.Ахмадулиной
50 ' -----

```

```

60 A$="T70V8L805RBGAB-V9AG4R8 D4.04B- V805D4V7R4.04B-16.R32
    B-16.R32B-16.R32xA1$;"
70 B$="T70V803GFV9E-.R8 R4D V802G03GV702G03GxB1$;"
80 A1$="B-A4.05C4.04B- G4GR2G 05D4.04G15.R32G16.R32GF+GxA2$;"
90 B1$="CGDF+ 02G03G02D03G 02G1603D16F16B1604D02G03FxB2$;"
100 A2$="05D4C4R4.G16.R32 G.F32R32F4FCEF- D4D16.R32D"
110 B2$="C16C64E-16E-64G16.04C02G16G6403E-16E-64G16.04C
    02F03F02A03F 02B-03F"
120 A3$="E-4E-16.R32E-D04AB-05C E-4D4R4.04V6B-16.R32
    B-.A32.R32A4A16.R32AB-05Cx4$;"
130 B3$="02A03GL8F+02F+GA B-03CDGB-AB-04D
    V6L403F+04C03F04Cx4$;"
140 A4$="V7D4D16.R32D V8C.04B32R32B4B16.R32B05CD
    V9A-2G4R8B-16.R32xA5$;"
150 B4$="V702B-03F VB02A-03F02G03F V9C16E-16A-16
    04C16E-03C02B-xB5$;"
160 A5$="B-.A32.R32A4A16.R32A06C05B- A4G4R8G16.
    R32G16.R32G16.R32 V8G404A4V7R8AG+A"
170 B5$="A1603E-16G1604C1603G02D03F+ 02E-16E-64
    G16G64B-16.03E-16E-64G16G64B-16.04E-8R4. C2C8R4."
180 A6$="05E-2D4.R8 V8R8GAB-V9AG4R8 D4.04B-xA7$;"
190 B6$="R802D8A803C8D8A804C8D8 03GFE-.R8 R4DxB7$;"
200 A7$="V805D4R4.V704B-16.R32B-16.R32B-16.R32
    B-4.A05C4.04B- G4GR4GF+GxA8$;"
210 B7$="02G03G02G03G CGDF+ 02G03G02D03GxB8$;"
220 A8$="05D2R804GF+G 05D4D4R4.G16.R32.G.F32.R32F4FCFE-"
230 B8$="02G1603D1GF16B1604D02G03F C16C64E-16E-64G16.04C
    02G16G6403E-16E-64G16.04C 02F03F02A03F"
240 A9$="D4R8D E-4E-16.R32E-D04AB-05C E-4D4R8
    04B-16.R32B-16.R32B-16.R32xC1$;"
250 B9$="02B-03F 02A03GLBF+02F+GA B-03CDGB-DB-04DxD1$;"
260 C1$="B-A4.R4B-05C V8D4D16.R32D C04BR8V9B16.R32B16.R32B05CD"
270 D1$="L403F+04C03F04C 02B-03F 02A-03F02G03F"
280 C2$="A-2G4R8V10B-16.R32 B-.A32.R32A4A16.R32A06C04B-
    A4G4R8G16.R32G16.R32G16.R32xC3$;"
290 D2$="L16CE-A-04CL4E-03C02B- L16A03E-G04CL403G02D03F#
    02E-16E-64G16G64B-16.03E-16E-64G16G64B-16.02E8R4.xD3$;"
300 C3$="V9G.04A32.R32A4AV8R4A V705E-2D4.R8 V8R8GAB-AG4R8xC4$;"
310 D3$="02A.03E-8G804C8G8R8 R802D8A803C8D8A804C8D8 03GFE-.R8xD4$;"
320 C4$="D4.04B- 05D2.R4 B-1"
330 D4$="R4D R802G803D8A8 03L16B-04DAB-05D04B-05DAL4 R1"
340 PLAY A$,B$
350 PLAY A3$,B3$
360 PLAY A6$,B6$
370 PLAY A9$,B9$
380 PLAY C2$,D2$
390 IF PLAY(0) THEN 390 ELSE 340

```

Отметим, что одним из способов упрощения перевода нотной записи на MML является создание какого-либо редактора, представляющего собой программу на [MSX BASIC](#). Она должна позволять в интерактивном режиме осуществлять ввод и редактирование музыкального произведения в обычной нотной записи с одноимённым генерированием и индикацией соответствующих команд на MML.

Литература

1. Справочное руководство по языку программирования Бейсик для КУВТ на базе персональных компьютеров "Ямаха (перевод с английского), 1986.
2. Ю.Булучевский, В. Фомин. Краткий музыкальный словарь для учащихся. —Л.: Музыка, 1984.
3. С.Максимов, Музыкальная грамота. —М.: Музыка, 1979.

4. А.Петров, Романсы из кинофильма «Жестокий романс». —М.: Советский композитор, 1984.

Ссылки

[Использование музыкальных средств диалогового языка программирования MSX — БЕЙСИК | Hardwarium, Оригинал](#)

Нотные символы

http://sysadminmosaic.ru/msx/using_musical_means_in_basic/using_musical_means_in_basic

2022-09-09 22:15

