AOZ Studio Beta - Bug #782

Bsave transpiles improperly. (%end is improperly embedded)

06/13/2021 11:58 PM - Brian Flanagan

Status:	Rejected	**Start date:**	06/13/2021
Priority:	Normal	**Due date:**	
Assignee:	Baptiste Bideaux	**% Done:**	0%
Category:		**Estimated time:**	0:00 hour
Target version:	none		
Affected version:	1.0.0 (B10) u16		

Description

For example:

```
// Enter your magic spell here!
Reserve As Data 15,50000
Print "Reserved 15"
Print Length(15) ' Length works by itself.
' ...but not in an expression on bSave:
Bsave "Bank15.bin",Start(15),Start(15)+Length(15)-1
Print "Done!"
End
```

The resulting improper traspilation of the Bsave is:

```
        // Bsave "Bank15.bin",Start(15),Start(15)+Length(15)-1
        aoz.sourcePos="0:5:0";
        return{type:12,waitThis:aoz.moduleFilesystem,callFunction:"bSave",waitFunction:"load_wait"
,args:["Bank15.bin",aoz.banks.getStart(15),aoz.banks.getStart(15)+%end]};
```

The %end should not be there.

History

#1 - 06/14/2021 12:05 AM - Brian Flanagan

- Subject changed from Length(n) transpiles improperly when inside an expression. (%end is embedded instead) to Bsave transpiles improperly. (%end is improperly embedded)

Well... not exactly, it is the BSave command itself which is not transpiling properly.
Even if I set Length(15) to the variable L, it fails the same way. (I've changed the topic accordingly.)

Example:

```
L=Length(15)
Bsave "Bank15.bin",Start(15),Start(15)+L-1
```

Result:

```
        // Bsave "Bank15.bin",Start(15),Start(15)+Length(15)-1
        aoz.sourcePos="0:5:0";
        return{type:12,waitThis:aoz.moduleFilesystem,callFunction:"bSave",waitFunction:"load_wait",args:["Bank
15.bin",aoz.banks.getStart(15),aoz.banks.getStart(15)+%end]};
```

(The 3rd item in args is set improperly on Base.)

#2 - 06/14/2021 11:03 AM - Brian Flanagan

- Assignee set to Baptiste Bideaux

#3 - 06/26/2021 09:37 PM - David Baldwin

- Status changed from New to Resolved

Fixed by Francois, wrong variable in instruction definition.

#4 - 07/15/2021 10:30 AM - Brian Flanagan

- Status changed from Resolved to Feedback

Transpiles now, however, Bsave still fails.
(Tested in 1.0.0 (B8) u15 on 7/15/2021)

#5 - 09/18/2021 02:21 AM - Brian Flanagan

- Affected version changed from 1.0.0 (B8) to 1.0.0 (B10) u16

Re-tested in 1.0.0 (B10) u16 (9/17)
Still fails.

#6 - 09/21/2021 08:57 AM - Francois Lionet

- Status changed from Feedback to Rejected

Not a bug! Instructions that in the original AMOS had a "To" to indicate the end now have a second syntax, WITHOUT the "To".
BSave PATH$, START To END // END is an ADDRESS. You should add the start of the bank if you use this syntax.
BSave PATH$, START, LENGTH // LENGTH is the number of byt5es to save.

You example should be either:
BSave PATH$, Start(BANK) To Start(BANK) + Length(BANK)
or
BSave PATH$, Start(BANK), Length(BANK)

Nothing much I can do to auto detect the mistake... but the doc should be ammended.

#7 - 09/22/2021 12:35 AM - Brian Flanagan

In performing the test, I was following the syntax shown in the F5 help.

According to the F5 help, the syntax is: **BSave path$,start,end**, however, the example shown there uses the original syntax (as it should) of BSave path$,Start To End, soooo... apparently the F5 help just needs to be corrected.

I should've just used the AMOS Pro Manual (as I usually do), but then I wouldn't have found that error. ;-)