Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
3	When a jp instruction in embedded assembly code is the next to be executed, the DC debugger doesn't highlight the jp instruction in the source window.	Look at the assembly window when single stepping through assembly code.	6.04-6.57P2
12	Behavior similar to bug #3 with call instructions.	Look at the assembly window when single stepping through assembly code.	6.04-6.57P2

26	The following program generates an error about not finding an internal compiler label.
	cofunc cof2(){}
	cofunc cof1(){
	costate {
	wfd cof2();
	}}
	main(){}
		None	6.04-6.57P2

39	Compiling a simple program like that below causes DC to enter an inconsistent state so that DC must be restarted. #nodebug main() {}	User must exit and restart DC	6.04-6.57P2

40	In the following program the address of greet assigned to x is bad (0x0000 and 0x0002).
	/*** BeginHeader greet */
	extern unsigned long greet;
	/*** EndHeader */
	xdata greet {"Hello"};
	void main(){
	   unsigned long x; x = greet;
	}
		Don’t use extern declaration.	6.04-6.57P2

45	strtol does not work properly with hex value or tail pointers (when the number is too large the tail/end pointer gets miscalculated).	None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

53	integer pointer arithmetic generates extra instruction when used with sizeof
	int *iPt;
	int *iPt2;
	main()
	{
	// this code generate a superfluous ld.
	iPt2=(int*) (iPt+sizeof(*iPt));
	//...this code does not
	iPt2=(int*) (iPt+sizeof(iPt));
	}
		None	6.04-6.57P2

74	Runwatch does not work correctly with auto variables. In the following program, if a watch is set on i, and ctrl-u is used to update the watch window, the value of i will be incorrect.
	void main(){
	   auto unsigned i;
	   while(1) runwatch();
	}
		If i is changed to a static variable, its value is updated correctly.	6.04-6.57

82	Editor limitation on line length can cause weird errors. A warning "line too long" should be put up, or too long lines should be disallowed altogether	keep lines under 90 characters in length, especially in library headers	6.04-6.57P2
85	lcall should be able to take a constant argument or arguments i.e. lcall 0xabcd,0xef or lcall 0xf9bcd correct strategy TBD	DB it db 0xcf,0xcd,0xab,0xef	6.04-6.57P2
87	DC crashes when compiling a file in a directory where the user does not have write permission. An error message window opens to report the file could not be opened, and then DC crashes after the OK button is clicked.	Work in a directory where you have permission.	6.04-6.57P2
107	if you do the following you will crash dc: 1. open dc 2. open any program, compile and run (everything should be ok) 3. open dc again. 4. attempt to compile and run....gpf	Use the “Close Serial Port” command on the Run menu in the first copy before compiling with the second.	6.04-6.57P2
112	The http libraries were using a 16 bit counter which is not sufficient for larger images.	None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

113	This program does not produce error about reassigning 'p',
	which is initialized at its declaration.
	main()
	{
	static char a;
	static int *p = &a;
	p = &b; // should not compile
	}
		None	6.04-6.57P2

114	When char c is assigned in foo(), the compiler produces an error about type mismatch
	Under similar conditions, initializing char c to be int i separate from declaration in main() produces no error.
	This should be allowed, but with demotion warning.
	void foo(void);
	main(){
	int i = 97;
	char c;
	c = i;
	}
	void foo(void){
	int j = 98;
	char c = j;
	}
		None	6.04-6.57P2

115	In the following program, assignment of p seems to be initializing value POINTED TO BY p, not address STORED IN p.
	char c = 'a';
	main() {
	static int *p = c;
	printf("c is char%c, int%d\n", c, c);
	printf("p is int%d\n", p);
	printf("*p is char %c, int%d\n", *p, *p);
	}
	int x = 10;
	int y = x; // y is set to address of x, not 10
	main() {
	printf("%x, %x\n", x, y);
	}
		None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

117	The following should generate a compiler error:
	char c[0x10008UL];
	char c[0x10008L];
	char c[0x10008];
		No work around. Do not make arrays larger than signed 16 bit integer.	6.04-6.57P2

119	A previously run program may have initialized the com port differently than DC needs it initialized. For instance, if a terminal program is run, and that program sets up the com port to use xon/xoff, Dynamic C will not be able to reset the attached Rabbit if it receives xoff as a checksum returned from the controller.	Use a terminal program to set up the com port the way that Dynamic C needs it set up, making certain that xon/xoff flow control is disabled. When Dynamic C runs again, xon/xoff flow control will be disabled since DC relies on the information about the port as returned by Windows.	6.04-6.57P2
120	function prolog/epilogs should be generated inline if compiling optimized for speed, and should only be used when optimizing for size if the function includes debug information, or if the function is nodebug and uses auto variables.	None	6.04-6.57P2

123	//This program causes a gpf/endless loop.
	// SizeOfSource is not decremented because //SizeOfDest is zero
	cofunc cof_indexed[8](char* my_name, int starting, int countby, int times)
	{while(times--) {
	printf("%s:%d\n",
	my_name, starting+=countby);
	}}
	main()
	{for(;;) {
	wfd {
	cof_indexed[0]("cof_indexed[0]", 0, 1, 8);
	cof_indexed[1]("cof_indexed[1]", 1, 2, 5);
	cof_indexed[2]("cof_indexed[2]", 2, 4, 5);
	cof_indexed[3]("cof_indexed[3]",3,6,5);
	cof_indexed[4]("cof_indexed[4]", 4, 8, 5);
	cof_indexed[5]("cof_indexed[5]", 5, 10, 5);
	cof_indexed[6]("cof_indexed[6]", 6, 12, 5);
	cof_indexed[7]("cof_indexed[7]", 7, 14, 5);
	}}}
		None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
124	The limit on the nesting level for #ifdef, #if, and #ifndef prevents the preprocessor from testing for multiple defines above 3-4 levels deep (depending on where the code is located).	None	6.04-6.57P2
126	pint() incorrectly outputs asterisks when the signed integer to be output is -32768 (i.e.: 0x8000) and the field width is specified as "%6d". plint() similarly incorrectly outputs asterisks when the signed long integer to be output is -2147483648 (i.e.: 0x8000000) and the field width is specified as "%11ld". plint() also outputs "--2147483648" (has two minus signs).	None	6.04-6.57P2
130	When landscape mode is selected, Dynamic C does not renumber the pages. Therefore, not all pages get printed. Only the same number that would have printed in Portrait mode.	use portrait mode	6.04-6.57P2
131	Using a macro name that is too long causes dynamic C to crash or hang. Dynamic C should report that the name is too long and/or accept longer names. 30 chars seem to crash it #ifdef INCLUDE_MODBUS_SERIAL_PROTOCOL #endif	keep macro names shorter	6.04-6.55
135	HTTP header field names and SSI commands were parsed in a case-sensitive manner. This prevented recognition of header field names and SSI commands that were not in the exact same case. (Case sensitive string functions now handle these cases appropriately).	None	6.04-6.54

138	This program compiles successfully with an unclosed comment block:
	main()
	{
	}
	/*
		None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
142	Version 6.52 of the Jackrabbit RS-485 libraries had been changed to work correctly with the newer Jackrabbit boards . The hardware change was to get around the problem of D/A 1 interfering with the RS-485 channel, but software support was also needed. Unfortunately, at least one version of Dynamic C was released AFTER the RS-485 library changes were made but before the Jackrabbit I/O library changes were made.	None	6.52-6.54

144	When the compiler reports the use of an undefined global label, the error line given is always '1'.
	// Sample code:
	void main(){
	undefined_label();
	}
		None	6.04-6.57P2

146	When Dynamic C starts up, the _BOARD_TYPE_ macro used in DEFAULT.H is incorrectly assigned the JackRabbit code value, no matter what board type is actually present. All recompilations of the BIOS after that one have the correct _BOARD_TYPE_ setting.	None	6.50 - 6.54

147	The following code compiles and runs. However, it should give an error because all case expressions must be different.
	int i;
	main(){
	i=0;
	while (1){
	switch (i) {
	case 1:
	i++;
	break;
	case 1:
	i--;
	break;
	default:
	i=1;
	break;}}}
		Make sure that each switch statement has unique case expressions.	6.04 -6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

154	Short circuit evaluation is not implemented correctly. In the following expression, j becomes 2.
	main (){
	int i,j,k;
	j=3;
	i = 3;
	i = 0 && (j = 2) && (k != 2); // Subexpression j = 2 is evaluated, but should not be.
	}
		None	6.32-6.57P2

156	The filename \\zintranet\sweng\tcpip\joel\iproute\dual2.c causes Dynamic C to either have an error creating the object file or crash when compiling.	copy the file to a regular disk file name	6.04 -6.57P2
157	RS232, 7Bit data transfers need to have MSbit cleared on incoming.	None	6.04-6.57P2
163	A long path name can cause the following error: Unable to open object file	None	6.04-6.57P2

164	The watch incorrectly promotes an int to unsigned int. For example:
	int i;
	i = -1;
	The watch window will display i as 65535.
		None	6.04-6.57P2

165	The packet receive buffers used by DCRTCP.LIB and the packet driver libraries are stored as a list of available buffers. Currently, retrieving a buffer from this list does not mark it as used. This is not a problem at the moment, since buffers are currently only used one at a time. This fix is necessary for planned extensions to TCP/IP code. Alterations must be made to the function _pktentry() in each packet driver library.	None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

172	Keyword are not reserved in Dynamic C. This program should not compile, but does.
	float(*typedef)();
	float float(float int){
	return int;
	}
	main(){
	int do;
	float char = 30.0;
	1,do = 1;
	printf("float is%f\n", float(char));
	typedef=float;
	do {
	printf("float is %f\n", typedef(char));
	} while(do--);
	}
		None	6.04-6.57P2

174	RS232 routines: serXread, cof_serXgets, and cof_serXread do not correctly handle timing out before any data has been received. In the case of serXread(), a time-out occurs immediately if there are no characters to read. In the case of the cofunctions, a timeout will never occur if no characters have been received.	None	6.04-6.57P2

176	This program compiles and should not.
	main() {
	int x 3; // should be error
	}
		None	6.04-6.57P2

180	The function description block in the source code and the desciption in the manual do not state that the function will stop reading and return if it receives a linefeed or carriage return character. This omits an important feature and allows for confusion.	None	6.04-6.57P2
182	The function read_rtc() only checks a single byte to see if the RTC was upating while it was being read. The ripple-counter nature of the RTC makes it possible for an unvalid bit to be possible (though rare) in any bit of the RTC registers.	None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

184	pow function generates run time error if 1st argument is negative
	float z;
	main (){
	float x, y;
	x = -3;
	y = 3;
	z = pow (x, y);
	}
	Note that if x is negative and y is not an integral value, a domain error should result, but if y is integral it should work.
		give the absolute value for the parameter and adjust the result.	6.04-6.57P2

185	In the cofunctions cof_serXputch(), cof_serXwrite() and cof_serXputs(), the abandon clauses make calls to rdunlock() instead of wrunlock(). This could cause a lock to not be freed and prevent further writing to the serial port. Fortunately, the abandon clauses are only used under unusual circumstances.	None	6.04-6.57P2

187	Compiling the following program with Dynamic C causes a general protection fault. The workaround is to remove the syntax error.
	union (// syntax error should be a curly brace instead
	  int a;
	  int b;
	};
	main() {}
		None	6.19-6.57P2

192	pd_resetinterface() (called from sock_init()) could lock due to a faulty delay loop.	Remove the following 3 lines from PKTDRV.LIB: pdri_loop0: ; bit is unreliable? tickwait should be enough. ioe bit 7,(hl) jr z,pdri_loop0	6.50, 6.53, 6.57T, 6.57T2

Bugs fixed as of Dynamic C 7.04P3

<table>
<tr><th>Reference Number</th><th>Description</th><th>Work-Around</th><th>Version(s) Affected</th></tr>
<tr><td>193</td><td>The sprintf function adds a space where not needed.
main(){
char ch[20];
float f;
f=(float)123;
sprintf(ch, "%4.0f", f); // extected 4 characters but come with 3 characters
printf("|%s|\n", ch);
f=0.123;
sprintf(ch, "%-10.6f", f); // extected 10 characters but come with 11 characters
// problem if 0.00 < f < 1.00
printf("|%s|\n", ch);
sprintf(ch, "%10.6f", f); // extected 10 characters but come with 11 characters
printf("|%s|\n", ch);
}</td><td>None</td><td>6.04-6.57P2</td></tr>
<tr><td>195</td><td>This should be an extremely rare bug when debugging over a serial connection, but if not fixed could cause much less rare problems with TCP/IP debugging.
Steps to cause crash:
1. Start dynamic C.
2. After BIOS is compiled, go to Inspect | Dump at Address
3. After dump window is open, disconnect programming cable
4. Scroll the dump window
5. Crash</td><td>None</td><td>6.04-6.57P2</td></tr>
<tr><td>196</td><td>After the BIOS is compiled, if the programming cable is detached and two attempts are made to disassemble, Dynamic C will crash. This should be an extremely rare bug when debugging across a serial connection.
To cause crash:
1. Start dc and compile bios.
2. Disconnect programming cable
3. Go to inspect | disassemble at address
4. choose an address and press OK.
5. Press OK on the error dialog box.
6. press OK again on the disassemble from address dialog
7. Crash.</td><td>None</td><td>6.04-6.57P2</td></tr>
</table>

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

197	When a UDP socket receives an ICMP Port Unreachable message, it causes the packets to be sent with an invalid IP type. The socket functions should return -1 when the socket has become invalid so the user can detect this condition close and reopen the socket.	Customers experiencing this problem can check the socket by:
		if(_chk_socket(s)) {
		        sock_close(s);
		        // reopen it.
		}
			6.04-6.57P2

198	Similar to defect #80. Except this is for chars and unsigned chars
	main(){
	  char n;
	  n = 0;
	  while(n--)
	    printf("%d\n",n); // should not execute!
	}
		None	6.04-6.57P2

201	Float comparisons <, >, <=, and >= in which the first operand is a float and the second is a dereferenced float do not always produce the correct result. For < and >, if the two operands are equal, then the comparison produces true when it should be false. For <= and >=, if the two operands are equal, then the comparison produces false when it should be true. The following code, which prints the string "Failure", demonstrates the problem:
	void main(void){
	   float f;
	   float *fptr;
	   f = 1.0;
	   fptr = &f;
	   if (f >= *fptr) {
	      printf("Success\n");
	   } else {
	      printf("Failure\n");
	   }}
		Swapping the two operands and changing to the corresponding comparison operator fixes the problem. That is, make the dereferenced float be the first operand and the actual float be the second operand.	6.04- 6.57P2

202	The xstring keyword does not give the address of the table. It appears that table is appended to the end of the strings.	use xdata	6.04-6.57P2
204	The assembler accepts and generates code for the nonexistent opcode "ld a, (hl+x)" in the format consistent with the actual opcodes "ld a, (ix+x)", etc.	None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
205	Placing a #ximport at end of the source file causes compiler error. If a semi-colon is placed on the line following the error goes away.	If a semi-colon is placed on the line following the error goes away.	6.04-6.57P2

208	The compiler incorrectly reports an error when trying to long jump to a forward reference.
	main() {
	;
	#asm
	ljp foo
	#endasm
	}
	xmem foo() {
	}
		None	6.04-6.57P2

209	Bug occurs with sample program core_flowcontrol.c under the following conditions 1. compile and run to root 2. compile and run to xmem (add #memmap xmem to top of file) This causes a "target not responding"-type error message. If you then recompile it, compiling the bios this time, the program runs fine. Similar bug happens under 6.57 where the program simply jumps to dkLoop instead of losing target communication. The problem also happens when you switch from xmem to root.	None	6.50-6.57P2

211	unsigned long mt();
	main(){
	unsigned long a;
	char chr;
	chr =20;
	// This expression
	// does not evaluate correctly
	a=mt()-(chr%15);
	}
	unsigned long mt(){return 10000;}
		None	6.04-6.57P2

213	The loop for transmitting and receiving packets is incorrect when it is transmitting packets with an event 256 multiple or receiving packets are are 14 more than an exact multiple. There is a small chance of data corruption when receiving packets.	None	6.04-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
215	A lost carry bit in the flash driver's _unlockQuad and _lockQuad functions could prevent flash writes in very special situations.	None	6.04-6.57P2
219	Sometimes when compilers errors occur in libraries subsequent compilations open a new error message window with closing the previous one.	None	6.04-6.57P2
222	If an e-mail is received by the POP3.LIB, and the e-mail has no body, the message might not be receive by the library properly.	Have something (even one empty line) in the body of the message.	6.53-6.57P2
223	When a task is deleted in uC/OS-II its stack is not returned to the "pool".	None	6.52-6.57P2

224	In special circumstances, forward references in xmem can cause invalid "Out of root code space" errors. This happens when the forward reference is located at logical address 0xe000.
	For example, this bug could happen in the following program if the compiler generates the lcall instruction to foo at address f8:efff so that the forward reference lies at f9:e000 (i.e., f8:f000).
	xmem bar() { foo(); }
	xmem foo() {}
		The work around is to add or subtract a byte from the program so that it will compile.	6.04-6.57P2

225	The compiler defined macro symbol for #ximport is defined as an int. It should be defined as a long.
	Example
	#ximport "bios\coldload.bin" coldloader
	main() {
	long z = 0
	long x;
	/* addition below will wrap to negative inte ger if coldloader > 0x6000 */
	x = coldloader + 4 + 0x4100 + z;
	}
		Cast it to a long.
		#ximport "bios\coldload.bin" coldloader
		main() {
		long z = 0
		long x;
		/* addition below will wrap to negative integer if coldloader > 0x6000 */
		x = (long)coldloader + 4 + 0x4100 + z;
		}
			6.51-6.57

227	When macros for assigning ports and bit numbers to the flow control lines are not defined, a set of default macros are defined in RS232.LIB instead if flow control functions are used. flow control code that depends on these macros is only compiled if these macros are defined. This makes it dependent on the order in which certain user functions are compiled. Therefore, some programs won't work.	The macros for flow control should be expicitly defined in the user's program	6.52-6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
228	There was a problem a extremely high (attack levels) traffic for the packet driver. This change quietly discards a number of packets.	None	6.53, 6.54, 6.57P/P2, 6.57T/T2
233	A structure is allocated on the stack during a DNS lookup that is > 512 bytes long, so it could overflow a 512-byte stack in uC/OS-II.	None	6.51-6.57P2
234	When running in RAM, xalloc can succeed when it has run out of space.	None	6.50-6.57P2
235	The PUSH flag should be set on TCP packets that contain data. As it stands, using the FTP browser in Internet Explorer in Windows does not interact nicely with our FTP server. Duplicate packets are sent from the FTP server to the Windows box, since the Windows machine does not ACK the first data packet it gets, but only the retransmitted data packet (with the PUSH flag set).	This inefficiency could be removed by setting the PUSH flag on all TCP packets that contain data.	6.51 - 6.57P2
240	The following does not compile in 6.57: /*** BeginHeader FL_ANA_PLANE */ #ximport "c:\jlc\rabbit\Customer_Pgms\Bester.lib" FL_ANA_PLANE /*** EndHeader */	If a semi-colon is placed at the end of the ximport line then it does compile	6.04-6.57P2
241	If an If-Modified-Since header is included an an HTTP request, the HTTP server is incorrectly closing the connection. We should really ignore that header (at least until we have a method for providing real support for it). This bug is especially visible in Netscape when going through the proxy server--every other load of the page results in a "This page contains no data" error (since the HTTP connection has been prematurely ended by the server).	None	6.51 - 6.57P2

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

242	The first condition of the second if statement contains a bad jump fixup. The bug is dependent on the first
	if statement.
	struct {
	char t1;
	char ta;
	char e[12];
	char g[12];
	char r1[12];
	char r2[12];
	}t[5];
	void main()
	{
	int j,l;
	// initialize structure/arrays with various values
	memset(t, 0, sizeof(t));
	t[0].g[0]='P';
	t[0].g[1]='G';
	t[0].g[2]=0;
	t[0].e[0]=0;
	t[0].e[1]=1;
	t[0].e[2]=0xFF;
	j=0;
	if ((t[0].g[j]=='P') && (t[0].e[j] != 0xFF)) // removing this line removes the bug
	{
	if (t[0].g[j] == 'G' && t[0].e[j] != 0xFF) // first half of && generates a bad jump
	{
	j++;
	}
	}
	}
	}
		None	6.17-7.02P

248	Mail sent through the SMTP client was not readable by some mail readers. SMTP.LIB should insert a blank line between the end of the mail headers and the beginning of the mail body. This problem can be seen when using Netscape Communicator as a mail client (it works correctly with Eudora). In Netscape, the subject displays correctly, but the body is lost.	No workaround.	6.51-7.02P3

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
254	spkrOut() was not included in ctrl-H function lookup.	None	7.02-7.04P
259	The spkrOut() function only disables the timer B interrupt if the frequency is below the valid range, NOT if volume is set to zero! This should cause timer B to be disabled.	To disable the timer B interrupt, call spkrOut with a frequency of 0.	6.57-7.04P

261	WriteFlash() isn't working correctly after the changes for the 7.02 flash driver, at least for certain flash types. The data you want to write is correctly transferred, but the rest of the flash sector is filled with garbage.	change the first call in WriteFlash in LIB\XMEM.LIB
		from this:
		root2xmem(FLASH_BUF_PHYS, (void *)flashAddr, _FlashInfo.sectorSize);
		to this:
		xmem2xmem(FLASH_BUF_PHYS, flashPtr&sectorMask, _FlashInfo.sectorSize);
			7.02-7.02P2

265	sock_bytesready() sometimes failed with ASCII TCP sockets. sock_bytesready() will return a random value when there is no '\r' or '\n' in the socket buffer because it is not properly checking the return value of xmemchr(). This can be exposed in sock_wait_input() on an ASCII socket, since it will return sooner than it should if the socket does not yet have a complete line (but has received some data in that line).	No Workaround	7.02-7.02P3
267	The checksums for block data are calculated as a 1-s complement sum of 16-bit words. This causes the checksum to fail if an odd number of bytes are written.	None	7.02P-7.02P3

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
278	The following program generates an Out of Xmem error: #ximport "c:\jlc\areacode.txt" InputFile// 16 KB #ximport "c:\jlc\areacode.txt" InputFile1// 16 KB #ximport "c:\jlc\areacode.txt" InputFile2// 16 KB xdata XdataValues {10}; main () {;} File size must be greater than 35KB	No error is noted if the xdata statement is moved to precede the #ximports. Also, no error is noted if there are only 2 #ximports.	6.04-7.03P
280	GPF occurs when duplicate function names are used in three or more libraries 1. Create three libraries: library.lib, library2.lib and library3.lib. 2. Add a function "foo" to all three of these libraries. 3. Add these libraries to LIB.DIR. 4. Compile Dynamic C 5. Once loaded, open the "Help=>Function Lookup/Insert" option. 6. GPF	Do not duplicate library function names more than twice.	6.04-7.03P
281	pack_dom() can sometimes overflow the DNS packet buffer when creating the packed form of the address to look up. If it doesn't overflow the buffer, it does still result in an invalid DNS lookup, to which the server replies "No such name". The packet is formed correctly on retry.	None	All TCPIP versions from 6.53 - 7.04P
283	A bug introduced in the cloning library MAY cause cloning to crash after 15 seconds or so, depending on whether the function it tries to call actually does exist in flash or not...	None	7.02-7.03P
288	There is a missing function in modem.lib(ModemSetDTR()) and an error in the description for PPPEscape() in ppplink.lib	None	7.02P-7.03P

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
289	NOTE: this only affects communications between the target and the RabbitLink console -- remote programming and debugging will not be affected! A RabbitLink and a target board will lose console communication if both boards lose power. If the target board starts up while the RabbitLink has the SMODE pins low, it will go into run mode which by default disables stdio (which is used to communicate with the RabbitLink console). The same effect also occurs if F4 is pressed after the program is running. Polling is disabled when F4 is pressed, which also disables STDIO.	The workaround for the 7.03 release (RabbitLink code version 1.03) is to use serial communications between the target and the RabbitLink console. The fix will be for the RabbitLink, after a power cycle, to: a) reset the target board; b) put it into debug mode to get the stdio correctly processing; c) put it into run mode. The F4 problem will be solved by not disabling polling when F4 is pressed.	7.021-7.04P

290	In TargetlessComp GUI, floating point val-

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected

298	Some packets were being broadcasted when they should have been sent directly. There is a bug in the ARP.LIB _arp_resolve function which causes a packet to be marked broadcast when it has not verified that the packet was on the same physical network. This causes some unnecessary traffic	work around:
		change:
		/* check for local broadcase address */
		if(0xffffffff == (sin_mask | ina)) {
		to:
		/* check for local broadcast address */
		if((((ina ^ my_ip_addr) & sin_mask)==0) && (0xffffffff == (sin_mask | ina))) {
			All TCPIP versions from 6.53 - 7.04P

302	Function description blocks in modem.lib have an incorrect library name	None	7.02P-7.04P
307	delays in CofModemInit and CofModemHangup do not work properly. This causes them to work intermittently.	None	7.02P-7.04P
309	The following two lines occurred multiple times in the FLASH.LIB library: ioi ld (MB3CR), a ioi ld (MB3CRShadow), a which would cause inadvertent writes to whatever Rabbit 2000 register coincided with the current address of the MB3CRShadow variable.	None	7.03P-7.04P
315	During the Realtek initialization sequence on a TCP/IP Dev Kit, there is a race condition between the Realtek coming up after its reset and the external I/O strobes being set.	None	6.57-7.04P
316	The ModemOpen() function will not currently work with some external modems when the Rabbit first starts up. The TX line must be explicitly set high (idle) and the DTR line must be expicitly set low (active), before starting to talk to the modem.	Before calling ModemOpen: set TX pin high with BitWrPortI(PCDR, &PCDRShadow, 1, 2); set DTR active with ModemSetDTR(1) or a call to BitWrPortI on the DTR pin	7.02P - 7.04P
318	sock_recv_from() now returns the IP address and port number in host byte order instead of network byte order. Note that this does affect code that depends on the old behavior.	Apply ntohl() to the IP address and ntohs() to the IP port number.	6.51-7.04P

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
327	Cloning now works for CPU clock frequencies > 19MHz.	for all boards with the CRYSTAL frequency less than 19.584MHz is to disable the clock doubler.	6.57-7.04P
328	512kb flash device support was fixed	None	6.57-7.04P
336	When a TCP/IP program does an open without sending data it continuously resends the ACK.	None	
343	Auto variables corrupt when certain transcendental functions are called from useix functions.	don't call transcendentals from useix functions or make bdSeries useix.	7.02-7.04P
344	The spa_init() function called by serAopen() incorrectly uses the shadow register for PADR instead of PCDR to set initial values for the outputs. This could affect other outputs on parallel port C that are not set up as serial ports.	None	6.53 - 7.04P
350	Several flash devices have problems when reporting back the flash type + manufacturer ID. In particular, the Atmel AT29C040-15 is known not to work.	The workaround is to hard-code the flash ID in the _lookupFlash function in the FLASHWR.LIB library by commenting out the call to _GetFlashID and adding the line ld hl, 0x1FA4 // hard-code the Atmel AT29C040A right after the call (substitute in whatever device+manufacturer ID is appropriate for the flash installed).	6.57-7.04P
352	xmem2root and root2xmem are not reentrant, this causes the http server to not work in MuC/OS.	None	
354	RES,res,SET,set should be in root	same as fix, add root to the util.lib function definitions and prototypes.	6.04 - 7.04P
362	tcp_reserveport() only works the first time it is called. trying to reserve a second port fails.	None	All TCPIP versions from 6.53 - 7.04P

Bugs fixed as of Dynamic C 7.04P3

Reference Number	Description	Work-Around	Version(s) Affected
363	The powerspectrum function and ff_hann functions give erroneous output when the input has 256 or 512 data points.	None	6.04-7.04P
365	When you change the TCP port the rabbitlink listens on, and that value is saved in flash, it is not restored properly. (the restore from the flash happens after the rabbitlink has already listened on the default value)	None	All TCPIP versions from 6.57 - 7.04P
375	A number of shadow registers were not being filled by the BIOS. In particular, PBDR and PCDR should have initial values written to them as well as the shadow being filled.	None	6.04-7.04P
383	Targetless compilation GUI board list, accessed via "Compile | Compile to .bin file | Define target configuration" menu selection, did not contain all Rabbit boards.	None	7.02-7.04P
387	The RCM2100 uses PD4..PD7 to control the Realtek chip. This should leave PD0..PD3 available to the user. This is not the case. SSI2 was modified to attempt using PD0. This caused the program to not function - Ethernet communication was lost to the RCM2100.	None	7.02-7.04P
396	Functions in RS232 that use MS_TIMER to determine if a timeout has occured, will not work correctly when the MS_TIMER value rolls over.	None	6.52 - 7.04P