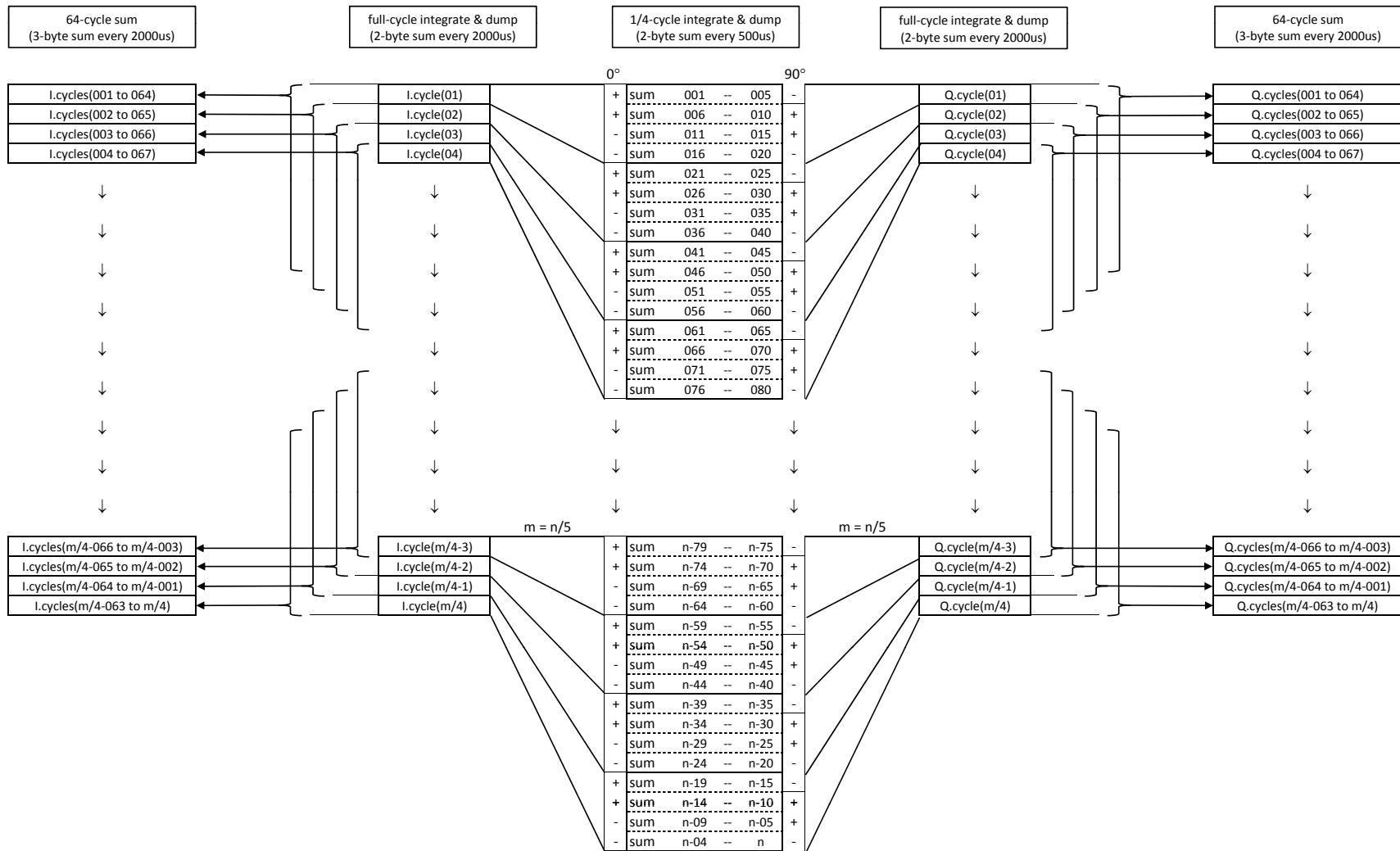


		2000us																				2000us																				2000us																											
0° 90°		001	002	003	004	005	006	007	008	009	010	011	012	013	014	015	016	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031	032	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047	048	049	050	051	052	053	054	055	056	057	058	059	060	061	062	063	064				
+	-	sum	001	--	005																																																																
+	+	sum	006	--	010																																																																
-	+	sum	011	--	015																																																																
-	-	sum	016	--	020																																																																
+	-	sum	021	--	025																																																																
+	+	sum	026	--	030																																																																
-	+	sum	031	--	035																																																																
-	-	sum	036	--	040																																																																
+	-	sum	041	--	045																																																																
+	+	sum	046	--	050																																																																
-	+	sum	051	--	055																																																																
-	-	sum	056	--	060																																																																
+	-	sum	061	--	065																																																																
+	+	sum	066	--	070																																																																
-	+	sum	071	--	075																																																																
-	-	sum	076	--	080																																																																

↓ ↓

↓ ↓

+	-	sum	921	--	925																																																																
+	+	sum	926	--	930																																																																
-	+	sum	931	--	935																																																																
-	-	sum	936	--	940																																																																
+	-	sum	941	--	945																																																																
+	+	sum	946	--	950																																																																
-	+	sum	951	--	955																																																																
-	-	sum	956	--	960																																																																
+	-	sum	961	--	965																																																																
+	+	sum	966	--	970																																																																
-	+	sum	971	--	975																																																																
-	-	sum	976	--	980																																																																
+	-	sum	981	--	985																																																																
+	+	sum	986	--	990																																																																
-	+	sum	991	--	995																																																																
-	-	sum	996	--	1000																																																																



memory 64 x 2B	revolution of circular buffer							
	00	01	02	03	04	05	06
address	data							
00	cycle-001	cycle-065	cycle-129	cycle-193	cycle-257	cycle-321	cycle-385
01	cycle-002	cycle-066	cycle-130	cycle-194	cycle-258	cycle-322	cycle-386
02	cycle-003	cycle-067	cycle-131	cycle-195	cycle-259	cycle-323	cycle-387
03	cycle-004	cycle-068	cycle-132	cycle-196	cycle-260	cycle-324	cycle-388

↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

61	cycle-062	cycle-126	cycle-190	cycle-254	cycle-318	cycle-382	cycle-446
62	cycle-063	cycle-127	cycle-191	cycle-255	cycle-319	cycle-383	cycle-447
63	cycle-064	cycle-128	cycle-192	cycle-256	cycle-320	cycle-384	cycle-448

initialize "running sum"

zero "cicular buffer" (64 2B values)

zero "running sum" register

set "input pointer" to "00"

set "ouput pointer" to "63"

update "running sum"

get "next cycle value" from "full-cycle integrate & dump"

store "next cycle value" at "input pointer" address

add "current cycle value" to "running sum"

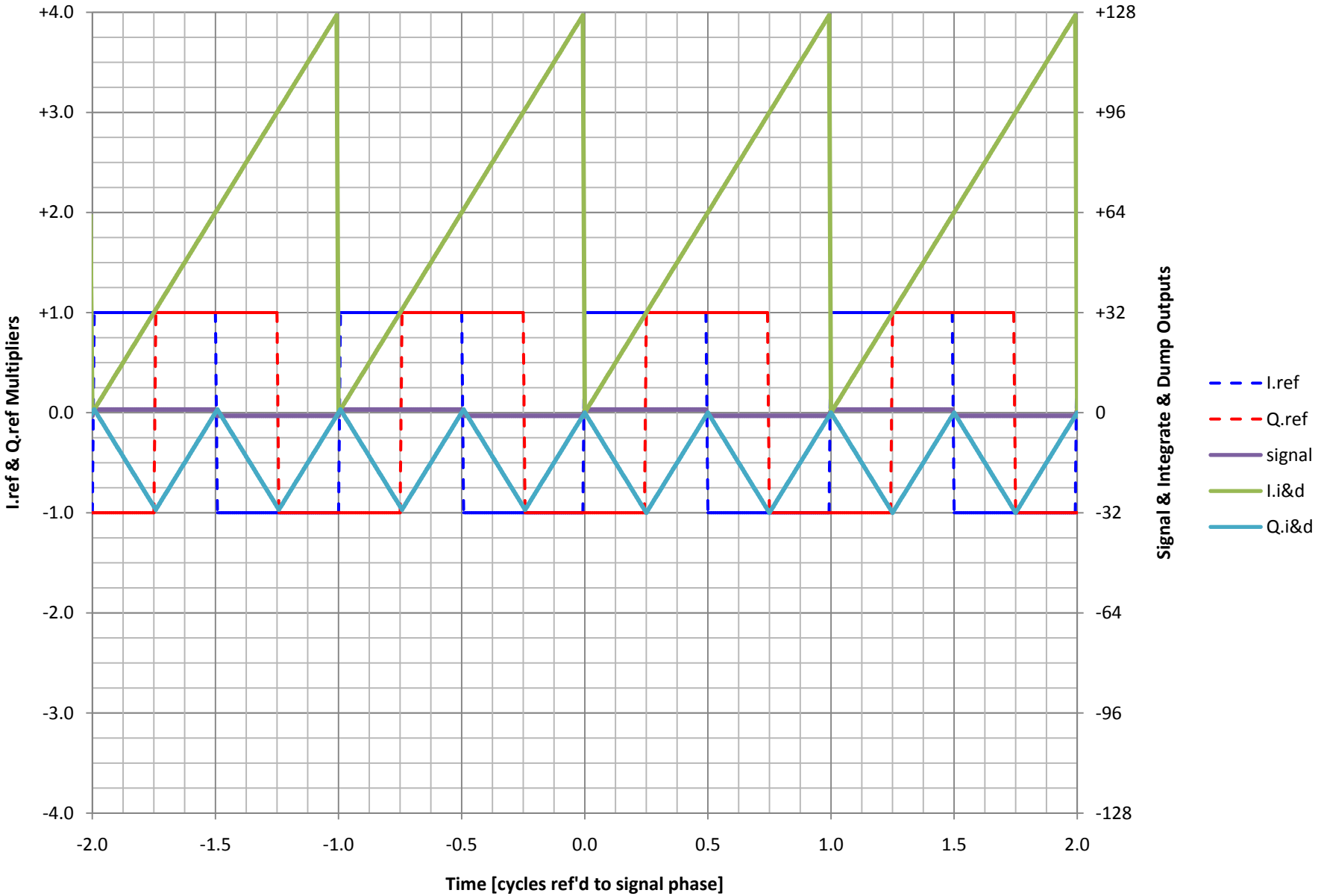
subtract "cycle value" at "output pointer" address from "running sum"

pass "running sum" to subsequent processing functions

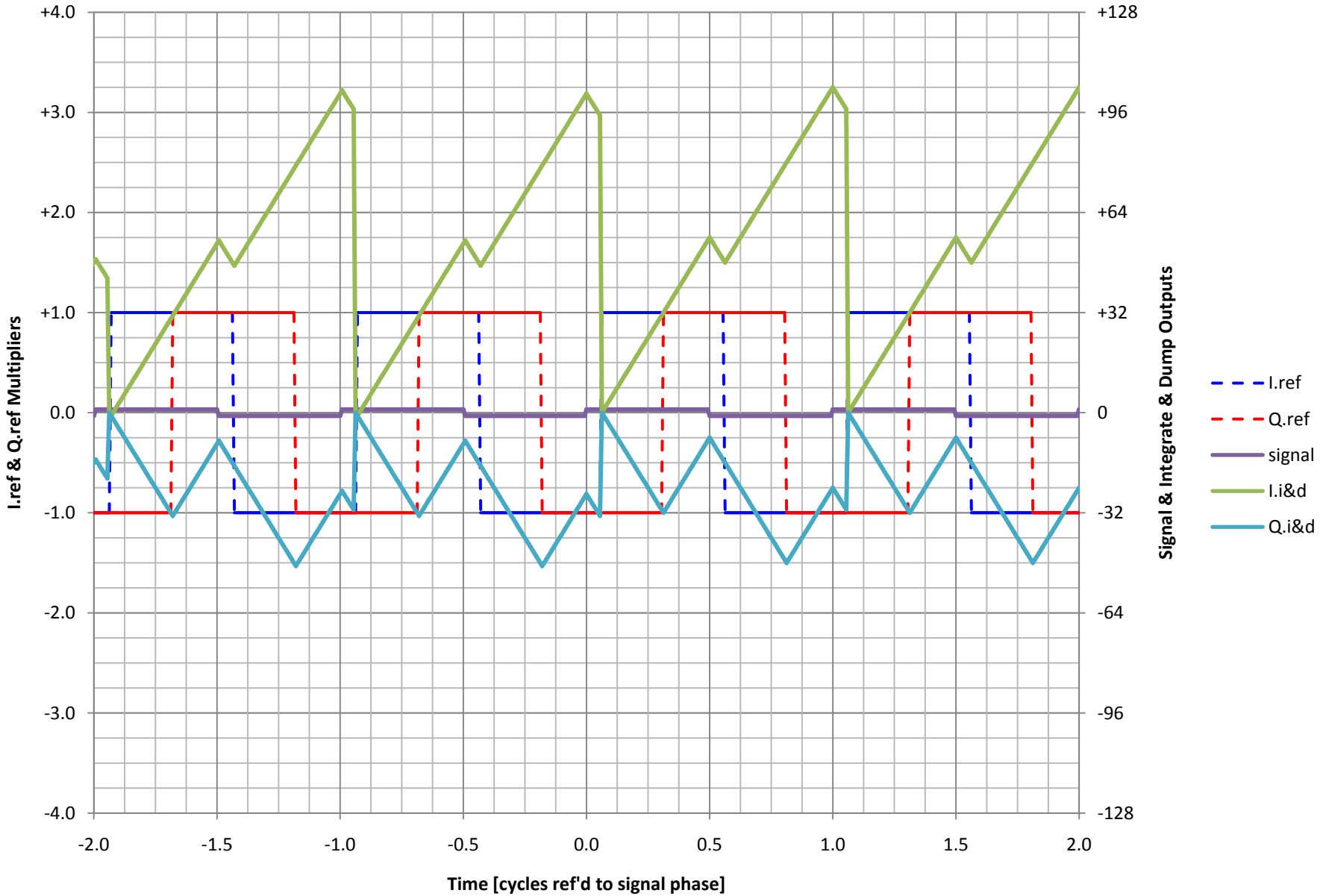
increment "input pointer" and "output pointer" by +1

next update

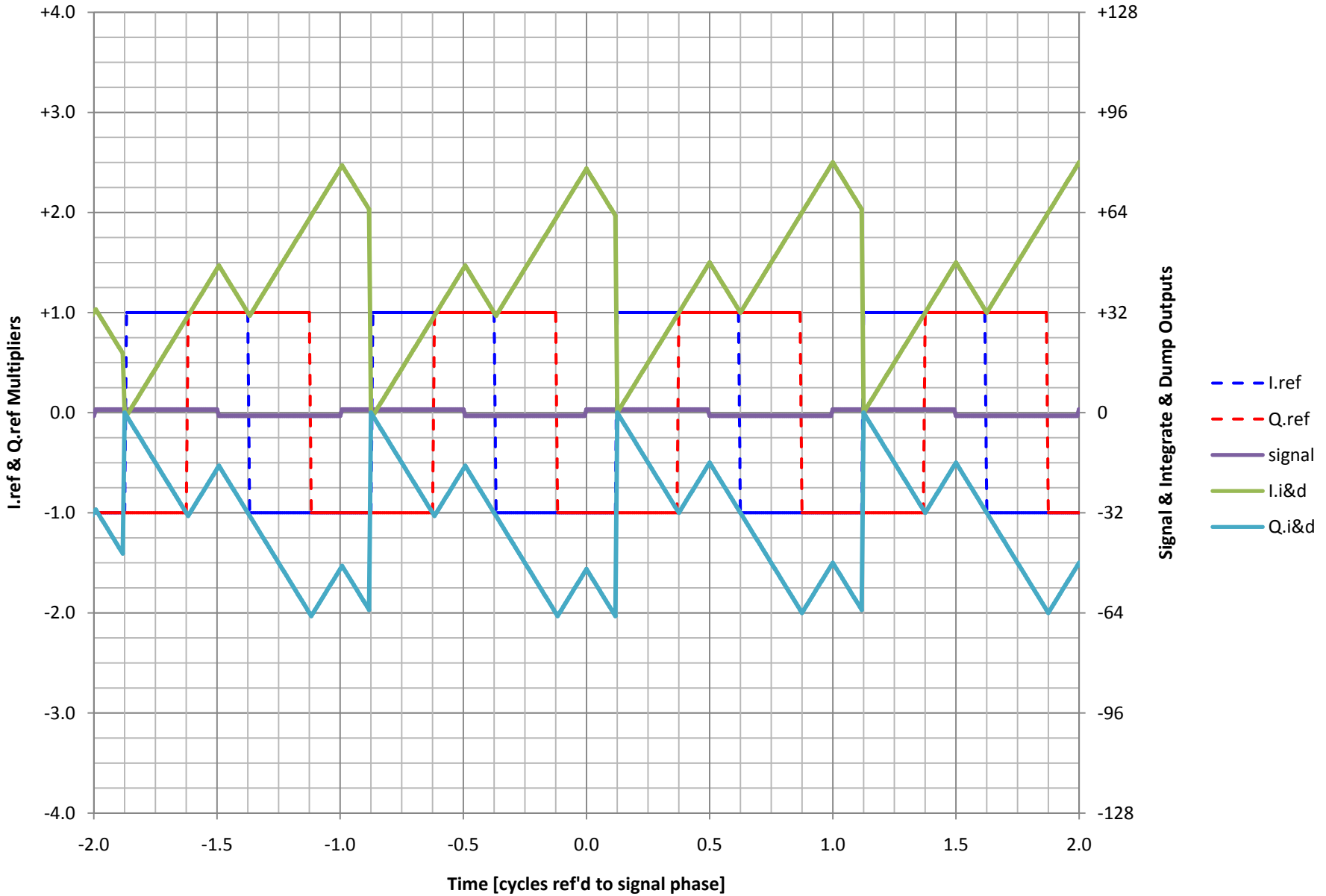
Integrate & Dump Waveforms (I.ref offset = 0/128 cycles)



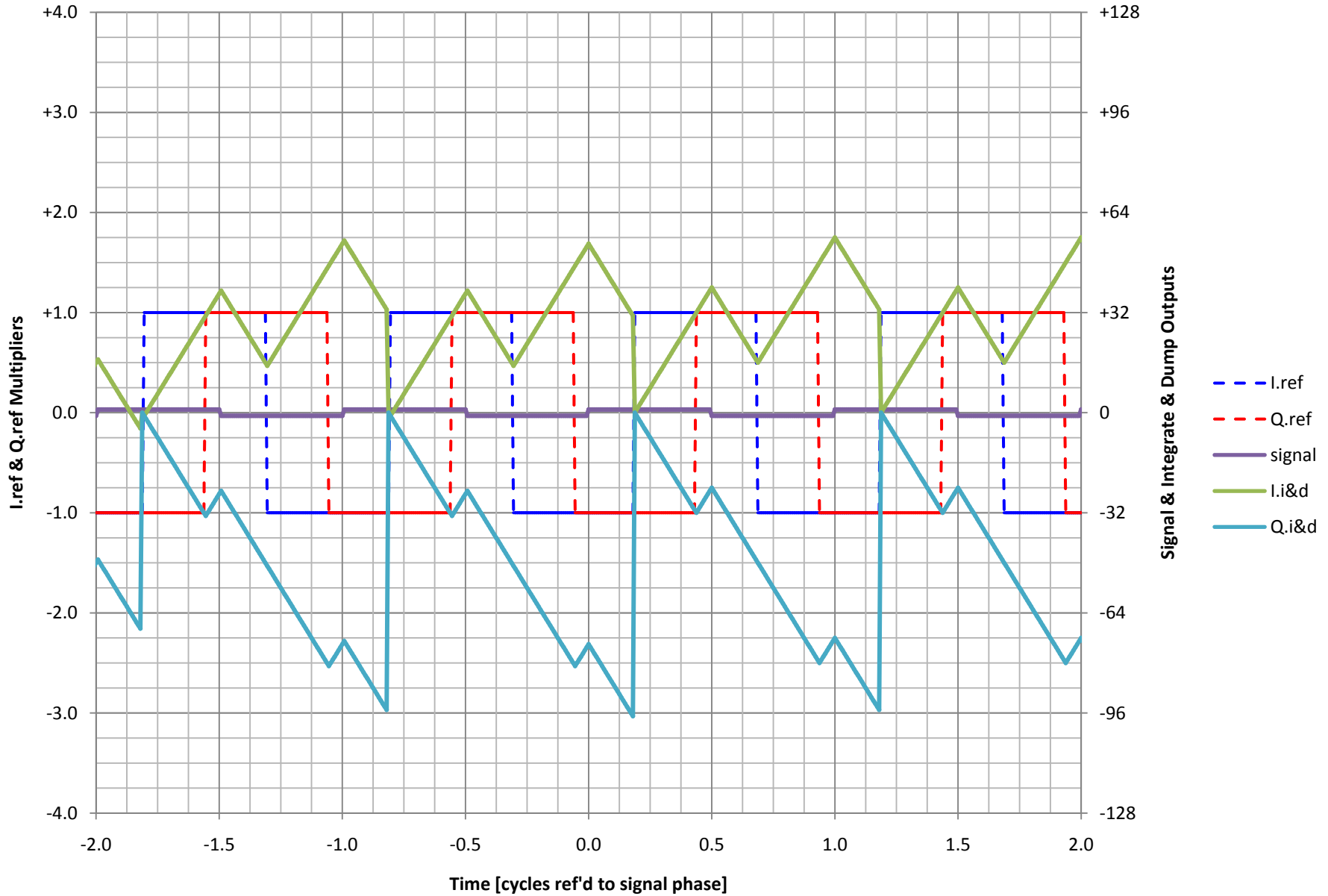
Integrate & Dump Waveforms (I.ref offset = 8/128 cycles)



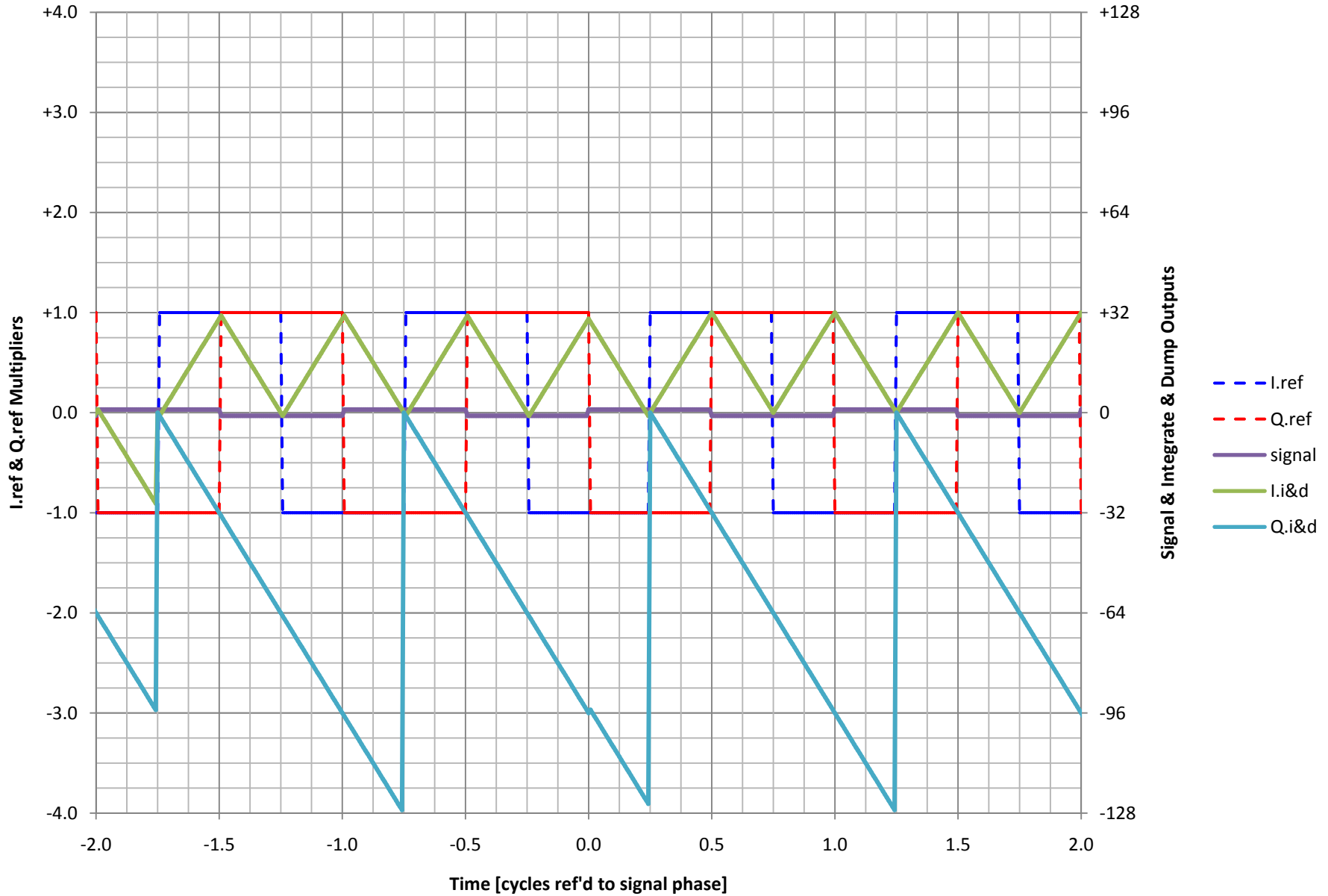
Integrate & Dump Waveforms (I.ref offset = 16/128 cycles)



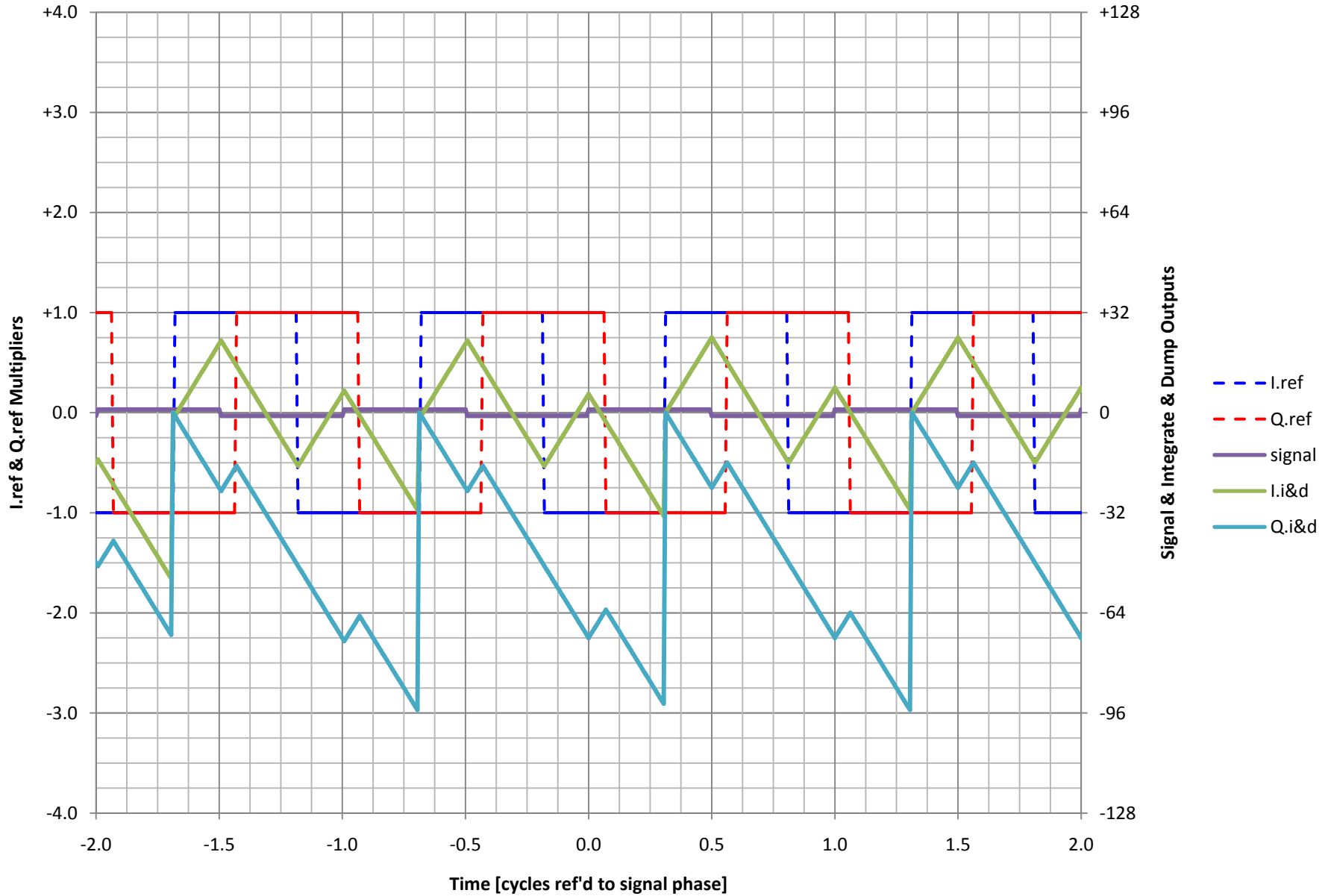
Integrate & Dump Waveforms (I.ref offset = 24/128 cycles)



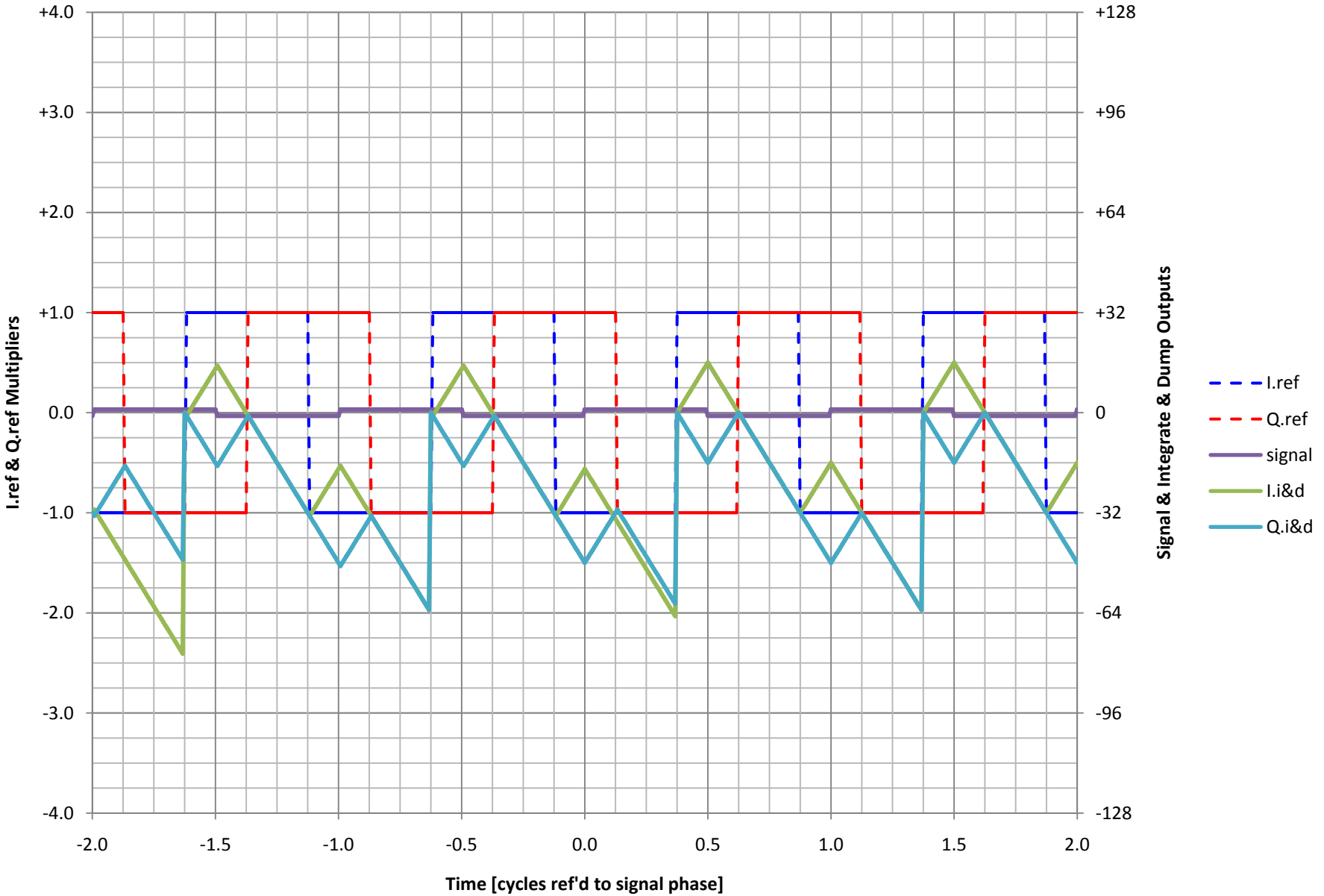
Integrate & Dump Waveforms (I.ref offset = 32/128 cycles)



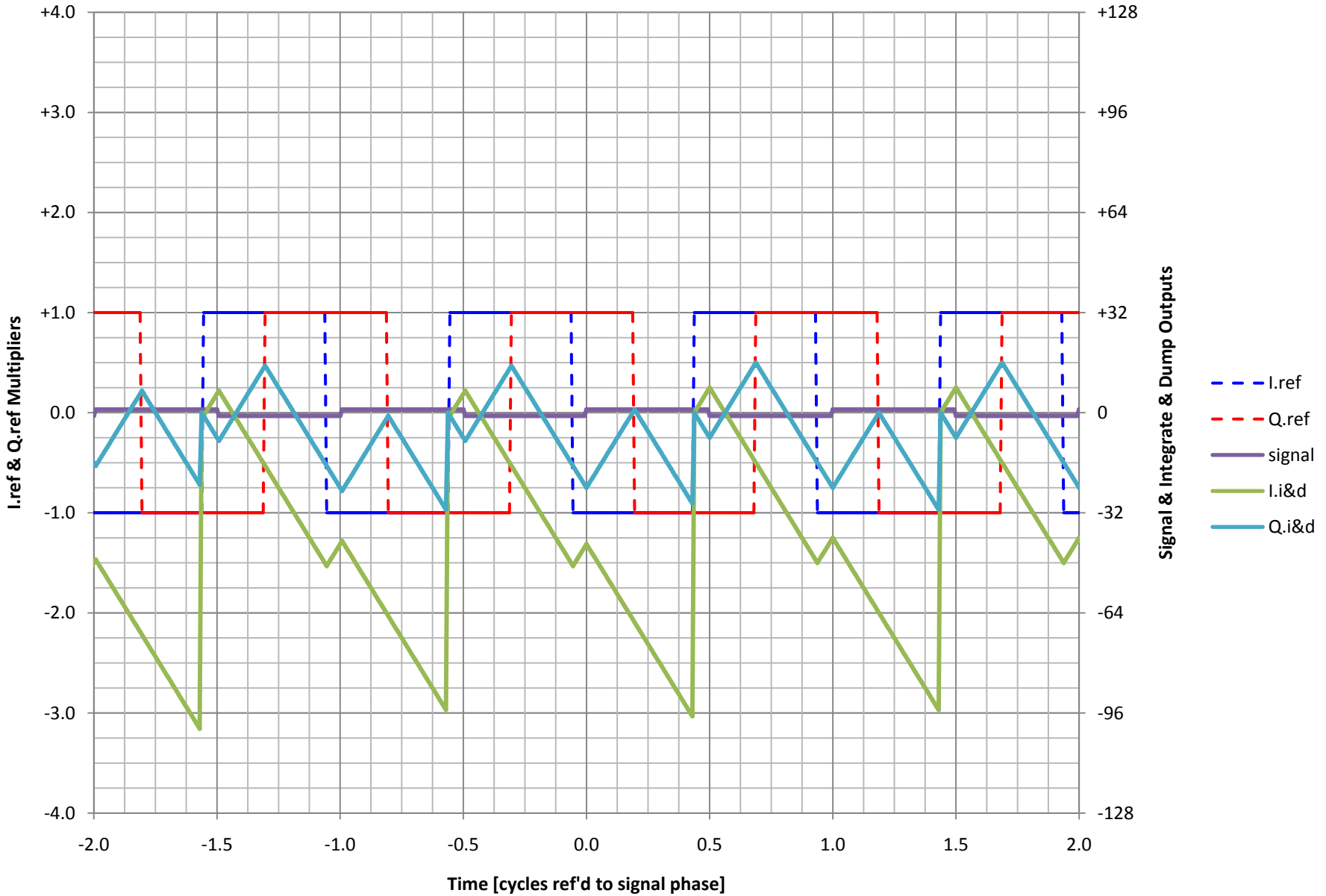
Integrate & Dump Waveforms (I.ref offset = 40/128 cycles)



Integrate & Dump Waveforms (I.ref offset = 48/128 cycles)



Integrate & Dump Waveforms (I.ref offset = 56/128 cycles)



Integrate & Dump Waveforms (I.ref offset = 64/128 cycles)

