

MCU-alapú üvegtörés-érzékelő tervezése – 2.

Kripasagar Venkat – alkalmazástechnikai mérnök, Texas Instruments

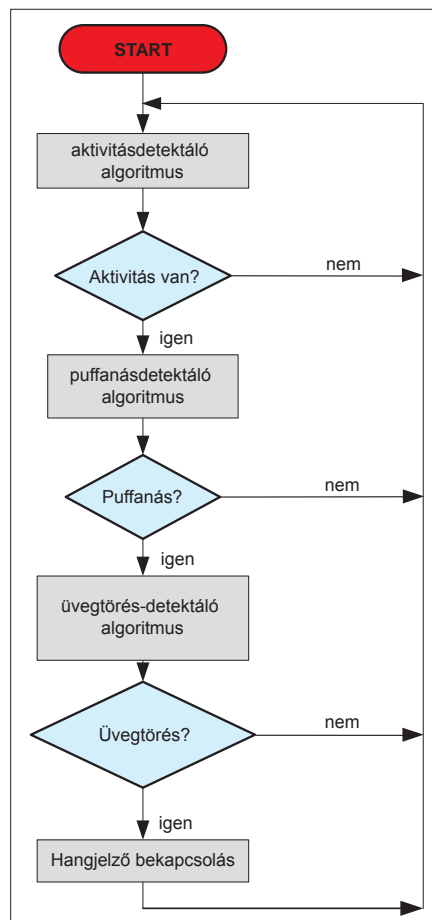
2010. 9. számunkban olyan esettanulmány közlésébe kezdtünk, amely egy biztonságtechnikai üvegtörésjelző tervezésén át mutatja be, hogyan lehet összetett, valós idejű feladatot olcsón, kis áramfelvétellel, a felhasznált mikrokontroller alapos kihasználásával megoldani. A második, befejező rész a szoftvertervezés és a konkrét hardvermegvalósítás szempontjait ismerteti, részletesen elemezve a teleptáplálásnál létfontosságú áramfelvétel kérdését.

Szoftvertervezési specifikációk

A mikrofonból származó analóg jelet a 20 kHz vágási frekvenciájú AAF szűri. Ennek a jelnek a digitalizálásához 40 kHz-nél sűrűbb mintavételre van szükség, amiből következik, hogy olyan AD-átalakítót kell választani, amely képes erre. A kívánt feldolgozást az egymást követő minták közötti időben be kell tudni fejezni ahhoz, hogy valós idejű feldolgozásról beszélhessünk. Például, ha a CPU órafrekvenciája 12 MHz, 40 kHz-es mintasebességnél két minta között nagyjából 300 gépi ciklusnyi számítási kapacitás marad a soron levő minta feldolgozására – ez rendkívül „szűk keret” egy összetettebb jelfeldolgozáshoz. Ha viszont gyorsabb CPU-t választunk, és ezzel további gépi ciklusokat bocsátunk a feldolgozó algoritmus rendelkezésére, ennek ára a megnövekedett teljesítményfelvétel és a tápellátást végző elem rövidebb élettartama. Ezért fontos feladat, hogy a feladat bonyolultsága és az elem élettartama között egyensúlyt találjunk.

Ebben a fejezetben az üvegtörés detektálására alkalmas algoritmus tervezési szempontjait tárgyaljuk. Amint az előző rész 1. és 2. ábráin láthattuk, az üvegtörés egy alacsony frekvenciájú „puffanással” kezdődik, majd sok magas frekvenciájú összetevőből álló, sok nullátmenettel és csúccsal jellemezhető „csörömpöléssel” folytatódik. Fontos megjegyezni, hogy a puffanás

és az ütközés zaja számos más hangjelenségnek is része: egy faajtó becsapódása, egy szekrény bezárása, a földre eső tárgyak, taps, kopogtatás az ajtón stb. is hasonló hangokat tartalmaznak. Viszont ezeknek a hangjelenségeknek egyikére sem jellemzők azok az utána közvetlenül következő, nagyfrekvenciás összetevők, amelyet egy jellegzetes üvegtörési zajtól várunk. De ez fordítva is érvényes: egy kávéőrölő, zenei felvétel, motorverseny a tévében, borospohár leesése és törése stb. tartalmazza az üvegtörésre jellemző nagyfrekvenciás komponenseket, de ezeket nem előzi meg az alacsony frekvenciájú, puffanásszerű zaj. A továbbiakban leírt üvegtörés-felismerő algoritmus a frekvencia- és az időtartomány-beli viselkedés elemzésével állapítja meg, hogy ez a két – jól elkülöníthető – jellemző jelen van-e a hangjelben, és a sorrendje megfelel-e a fentiekben leírt ismérveknek.



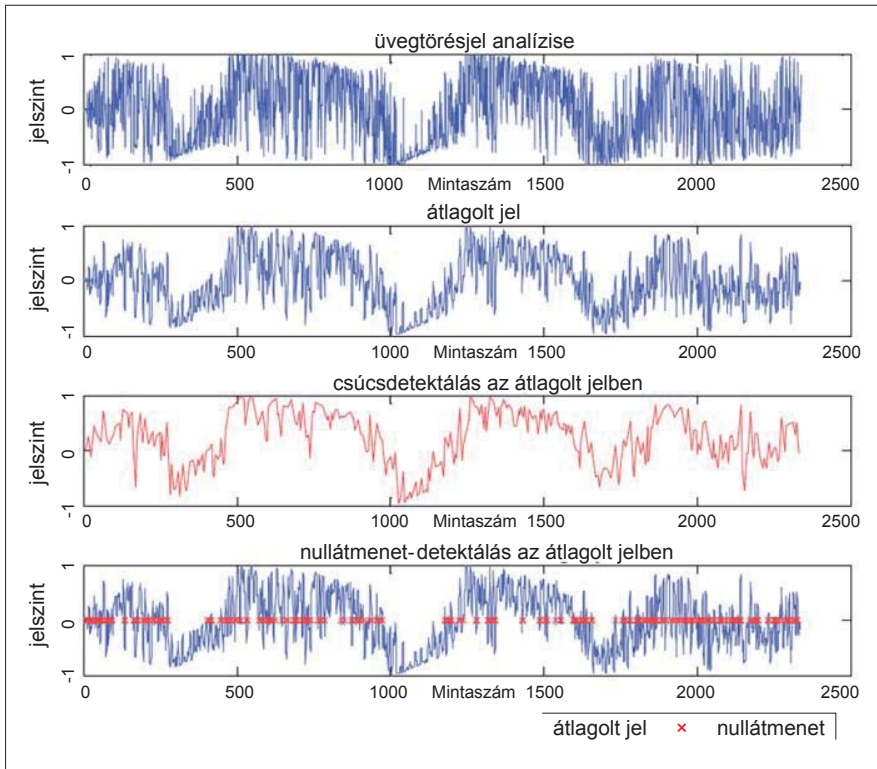
4. ábra Az üvegtörés-detektáló algoritmus folyamatábrája

Az algoritmus

A 4. ábra az algoritmus magas szintű folyamatábráját mutatja. Ennek három fő része az aktivitás detektálása, a „puffanásdetektálás” és a „csörömpölés” érzékelése az előfordulásuknak ebben a sorrendjében. Nagyjából minden 2,5 ms után a mikrofon és az OA1 műveleti erősítő „feléled”, hogy ellenőrizze, van-e folyamatban valamiféle akusztikus aktivitás. Ha semmilyen értékelhető hangjelenséget nem tapasztalunk, az erősítő és a mikrofon kikapcsolódik, az MCU pedig alacsony fogyasztású állapotba kapcsol. Ha viszont van valamiféle hangjelenség folyamatban, a szoftver futása a „puffanásdetektálással” folytatódik. Ekkor az ADC bekapcsolódik, elkezdődik a mintavételezés, és a mintasorozaton végrehajtódik a puffanásszerű hangot felismerő jelfeldolgozás. Csak ha a puffanásdetektáló programrész felismerte az üvegtörést megelőző ütés alacsonyfrekvenciás összetevőit, csupán akkor folytatódik a program az üvegtörés nagyfrekvenciás, sok nullátmenettel és csúccsal jellemezhető szakaszát felismerő programrészrel. Ha ez a szakasz is sikeres felismeréssel zárul, a panelre épített led vagy hangjelző bekapcsolódik, az ÜTJ-algoritmus pedig visszatér a kezdeti, aktivitásfelismerő szakaszához.

Az **aktivitásérzékelés** abból áll, hogy az ADC bemeneti jelét egyszerűen összehasonlítjuk egy előre meghatározott fix kü-

szöbértékkel a nullaszint két oldalán. Ezzel a valódi jel és a zaj egymástól elválasztható. Amint már említettük, a „puffanás” egy alacsony frekvenciájú összetevő, amely nagyjából 300 Hz körül jelentkezik. Mivel a „puffanás” a törést kiváltó ütközéskeletkezik, csak a bemeneti jel első néhány mintáját kell szűrni. Ezt egy 350 Hz vágási frekvenciájú, digitális aluláteresztő szűrővel lehet megvalósítani. A szűrt mintákat összegyűjtjük, átlagoljuk és összehasonlítjuk egy előre meghatározott energiaküszöbértékkel. Ha az energia ezt a küszöbértéket meghaladja, a „puffanásdetektáló” és a töréskor keletkező nagyfrekvenciás komponensek jelenlétét érzékelő programrészt kell aktiválni. Annak érdekében, hogy a digitális aluláteresztő szűrő kisebb legyen, de hatékony maradjon, a mintavételi frekvencia ebben a korai szakaszban sokkal alacsonyabb – mindössze 4 kHz.



5. ábra A Jelanalízis1 programrész jelreprezentációja

Viszont a korábban meghatározott 20 kHz-es vágási frekvenciájú AAF helyett 2 kHz-es AAF-vágási frekvencia adódik az algoritmusnak ebben a szakaszában.

Az **üvegtörés-detektáló** algoritmus összetettebb a „puffanás-érzékelésnél”, és két jelanalizáló szakaszra van osztva: Jelanalízis1 (JA1) és Jelanalízis2 (JA2). A JA1 a feldolgozás első szakasza, és minden mintán végrehajtódik, miután a puffanás detektálása megtörtént. A JA1 során 20 kHz-es AAF-et választunk, és az ADC mintafrekvenciáját 40 kHz-re növeljük. A JA1 műveletei közt jel-átlagolást, nullátmenet-érzékelést és csúcsdetektálást találunk. Ez nagyjából 60 ms-ig, mintegy 2400 mintán keresztül tart. Ha a JA1 véget ért, elindul a JA2 szakasz, amely befejezi a teljes jelanalízis folyamatát. Az 5. ábrán a JA1 jelreprezentációját, a 6. ábra pedig a jelanalizáló programrész folyamatábráját mutatja.

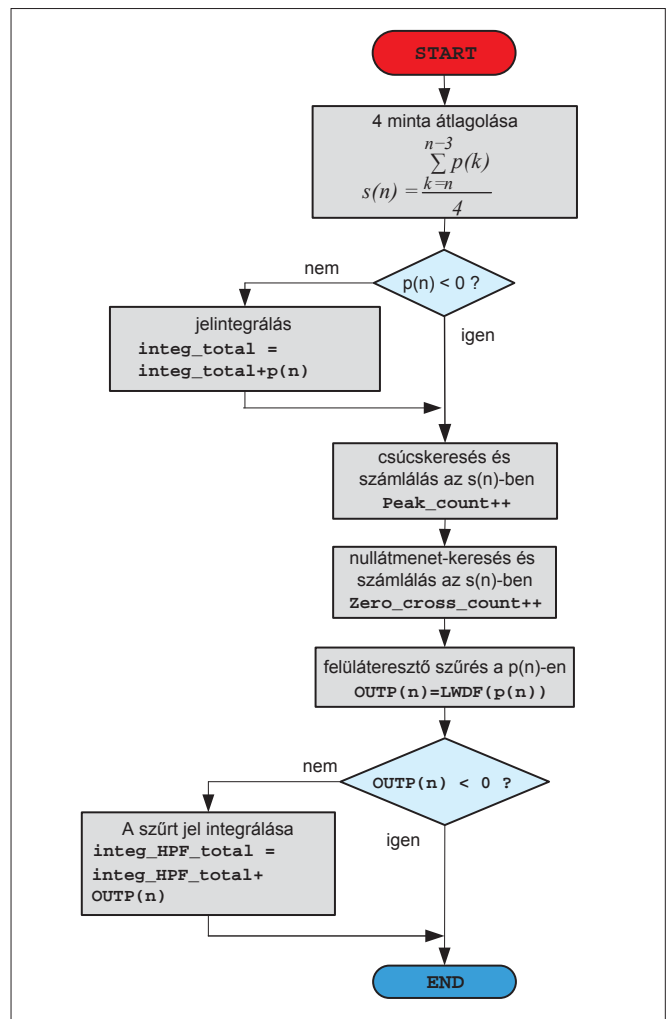
A $p(n)$ jelű bejövő mintasorozaton először egy egyszerű, mozgó átlagú, átlagoló szűrés hajtódik végre a zaj elnyomása érdekében, melynek eredménye az elsősorban a jelre jellemző $s(n)$ szűrt mintasorozat. A $p(n)$ integrálása során csak a pozitív előjelű mintákat összegezzük a jelenergia meghatározása érdekében. Ez az érték az `integ_total` változóban keletkezik, amelyet később majd a JA2 fázis használ fel. Az $s(n)$ mintasorozaton megtörténik a csúcsok és nullátmenetek megszámlálása. A jel nagyfrekvenciás komponenseiről le kell választani az alacsonyfrekvenciás összetevőket, amelyet egy $F_s/4$ vágási frekvenciájú, felüláteresztő szűrő (High Pass Filter – a továbbiakban HPF) végez el, és a $p(n)$ jelsorozat minden elemén végrehajtódik. Ezzel egyidejűleg a szűrt kimenetnek csak a pozitív mintái összegződnek egy `integ_HPF_total` nevű változóban, amelyet a JA2 fázisban használunk majd fel. A valós idejű feldolgozás feltételének teljesítése érdekében az egész SA1-et minden mintán végre kell hajtani, és a feldolgozást be kell fejezni, mielőtt a következő $p(n+1)$ minta beérkezik – figyelembe véve, hogy az ehhez rendelkezésre álló processzor-ciklusok száma a processzor órafrekvenciájának és a 40 kHz-es mintafrekvenciának a hányadosa. A szűrés rendszerint nagyon időigényes művelet. A hatékonyság érdekében a Lattice Wave

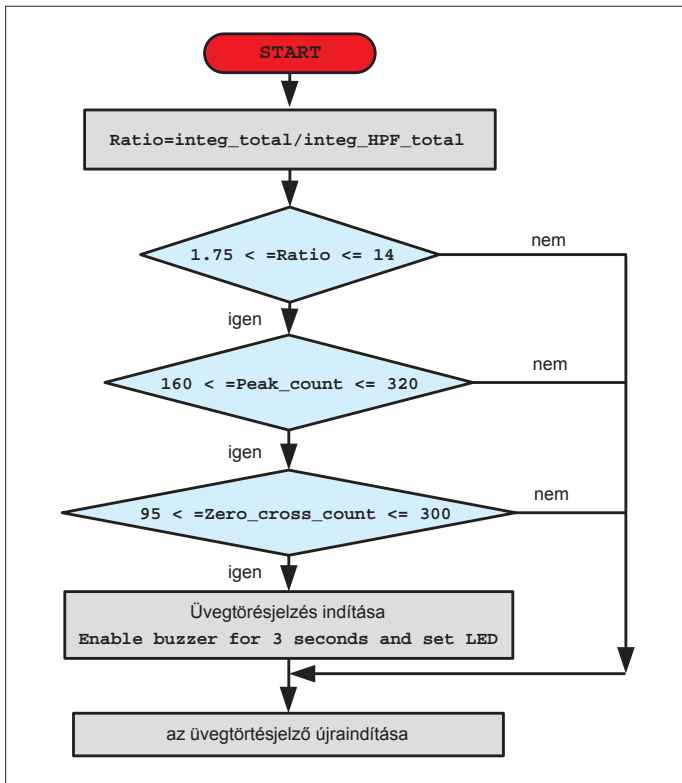
Digital Filters (LWDF) és a Horner-algoritmus került felhasználásra a puffanásdetektálás aluláteresztő szűrőjének, valamint az SA1 felüláteresztő szűrőjének megvalósításához.

Ha a 60 ms-os mintasorozatot már feldolgozta a JA1, az algoritmus a JA2-vel, a feldolgozás második fázisával folytatódik, amely nem igényel valós idejű működést. A műveletekről a 7. ábra ad áttekintést. Végül, ha a jelsorozat minden szempontból az üvegtörésre jellemző tulajdonságot hordozza, ezt a JA2 eljárás végén jelzi az eszköz a külvilágnak.

A szoftver kiszámítja a jelsorozat teljes energiáját és a felüláteresztő szűrés utáni jelenergiát. A kettő arányát a program egy küszöbértékkel veti össze. A vizsgálatok azt mutatják, hogy a jelenség üvegtörésnek csak akkor minősíthető, ha ez az arány 1,75 és 14 közé esik. Hasonlóképpen az üvegtörésre jellemző, ha a „csörömpölés” fázisában a csúcsértékek száma 160 és 320 közé, a nullátmenetek száma pedig 95 és 300 közé esik. Valódi üvegtörésnek akkor minősíti a program a jelenséget, ha ez a három kritérium egyszerre teljesül. Ha bármelyik ezek közül nem valósul meg, az üvegtörésjelző újrainicializálódik és visszaáll a legelső, aktivitásdetektáló üzemmódba.

6. ábra A Jelanalízis1 programrész folyamatábrája





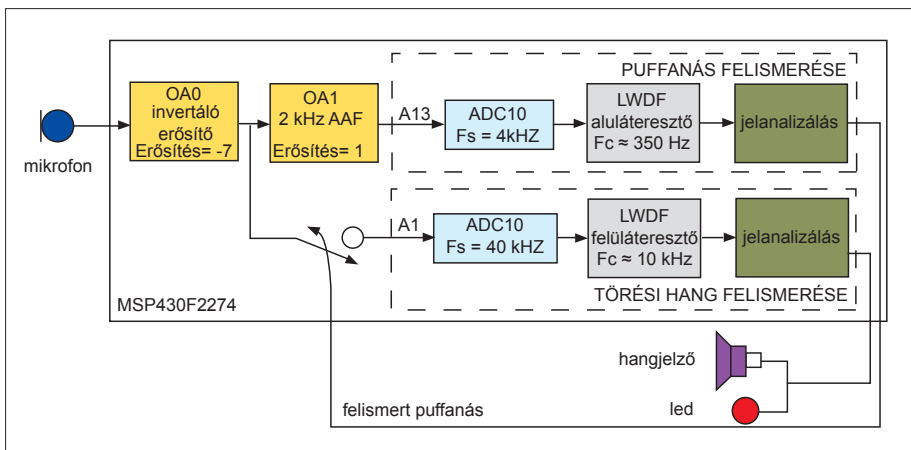
7. ábra A Jelanalízis2 programrész folyamatábrája

A megadott közbértékeken némileg módosíthat a helyiség akusztikája, az ÜTJ elhelyezése és a környezeti háttérzaj stb.

A mikrokontroller megvalósítása

A Texas Instruments (TI) MSP430™ MCU-platformja az eszköz széles választékát tartalmazza. Az MSP430F2274 16 bites MCU az MSP430 alacsony teljesítményigényű platform 2xx-családjának tagja. Az MCU 16 MHz-es órajelre működőképes. Van egy nagyon kis fogyasztású, nagyon alacsony frekvencián működő oszcillátora (Very Low Frequency Oscillator – VLO), amely szobahőmérsékleten 12 kHz-es órajelet állít elő. Az eszközben van továbbá két 16 bites időzítő és egy 10 bites AD-átalakító (ADC10), amely akár 200 kHz-es mintafrekvenciával is képes működni. Az ADC10 úgy is konfigurálható, hogy számára az analóg jelkondicionálási feladatokat a csipre integrált OA0 és OA1 műveleti erősítő lássa el. Az MCU áramfelvétele a 3. készenléti üzemmódban (Low Power Mode 3 – LPM3) csupán 0,7 µA, de a fogyasztása aktív állapotban is mindössze

8. ábra Az üvegtöréssjelző rendszerszintű tömbvázlata az MSP430-as MCU-platform felhasználásával



250 µA, amely ideális választás azoknak a fejlesztőknek, akik teleptáplálású eszközökhöz terveznek vezérlőegységet.

A 8. ábra mutatja az MSP430F2274 MCU-val és beépített perifériáival megvalósított ÜTJ rendszerszintű tömbvázlatát. Mivel a mikrofon 20 Hz...20 kHz között sáváteresztő karakterisztikájú, és az MSP430F2274-ben csak két műveleti erősítő van, megengedjük magunknak, hogy a 20 kHz-es AAF-et kihagyjuk a megvalósításból. Bár ez a mintavételi tétel nyilvánvaló megsértése, azt tapasztaltuk, hogy a 20 kHz-es antialiasing szűrő elhagyása nem jár semmilyen hátrányos következménnyel. Ha viszont még egy műveleti erősítőt rászánunk, a 20 kHz-es AAF-et is megvalósíthatjuk vele.

Az MSP430F2274-ben két, szoftveresen konfigurálható műveleti erősítőt (OA0 és OA1) találunk. Az OA0-t 7-szeres erősítésű, invertáló erősítőként mikrofonerősítőnek használjuk, míg az OA1-gyel egységnyi erősítésű, másodrendű Butterworth-konfigurációjú, aluláteresztő szűrőt valósítunk meg a Sallen-Key-architektúra szerint. A szűrő 3 dB-es vágási frekvenciája 2 kHz-en van. Az OA0 és OA1 erősítők kimenete belülről csatlakozik az A1 és A13 csatornákra.

Áramfelvétel

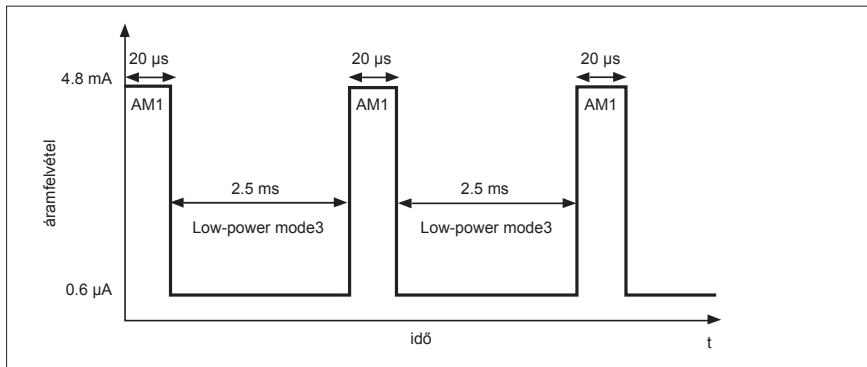
Az egész ÜTJ felépítésénél széles körben kihasználtuk – a használat igényeinek megfelelően – a MCU alacsony áramfelvételű üzemmódjait és a perifériák szelektív ki/bekapcsolását. Az MSP430-cal megvalósított rendszer áramfelvételi profilját a háromféle üzemmódban a 9., 10. és 11. ábra mutatja.

A 9. ábrán az aktivitásdetektálási fázis áramfelvételi profilja látható. Az eszköz minden 2,5 ms-ban „feléled”, 20 µs-ig ellenőrzi, hogy van-e a zajszintet meghaladó hangjel. Eközben a processzor órajele 12 MHz-re van beállítva. Ha a külső mikrofonon aktivitásra utaló jel nincs, az eszköz visszatér a készenléti üzemmódba (LPM3). A készenléti állapotból való periodikus felélesztést a csipre integrált időzítő végzi, amelynek órajelét ilyenkor a – szobahőmérsékleten – 12 kHz-es órajelet előállító VLO adja.

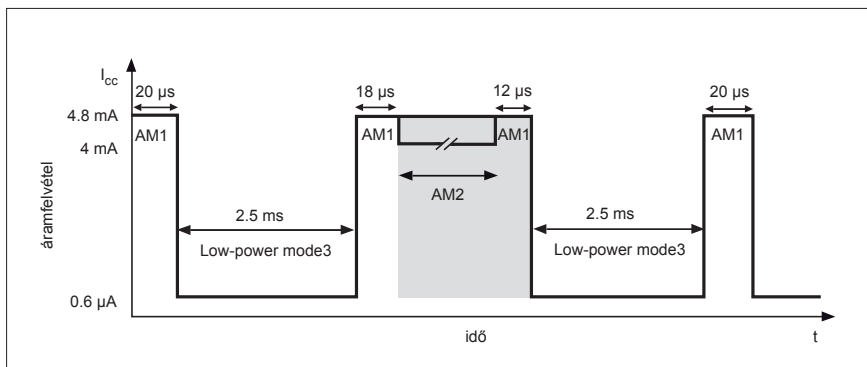
A 10. ábra mutatja azt az áramfelvételi profilt, amikor a mikrofon felől aktivitást észlel a rendszer, és az algoritmus a puffanásdetektálással folytatódik. Az MCU ekkor 18 µs-ra, az AM1 aktív üzemmódba kerül, órajele 8 MHz. Ezalatt hajtódnak végre a puffanásdetektálás végrehajtásának előkészítéséhez szükséges konfigurációs beállítások. Az MCU ezután az AM2 aktív üzemmódba kerül. Ebben az üzemmódban az ADC 4 kHz-es mintasebességre áll be. Ettől kezdve 32 ms-on keresztül minden egyes mintán jelanalízis hajtódik végre – összesen 128 minta vizsgálata és feldolgozása történik meg. Ha puffanás-ként értékelhető jelet nem talál az algoritmus, az eszköz 12 µs-ra visszalép az AM1-üzemmódba. Ezalatt megtörténik az eszköz újrainicializálása, és végrehajtnak az aktivitásdetektáláshoz való visszatéréshez szükséges konfigurációs lépések.

A 11. ábrán az az áramfelvételi profil látható, amikor az eszköz már felismert egy puffanászerű hangjelenséget, és az eszköz átlépett az üvegtörés-detektálást végző üzemmódba. Ennek során a CPU órajelét 12 MHz-re állítjuk be, és az eszköz az AM3 aktív üzemmódjába lép. Ekkor az ADC mintasebessége 40 kHz. Egy 60 ms-os perióduson át minden mintán végrehajtnak a jelanalízáló algoritmus (ez a 40 kHz-es mintasebességet figyelembe véve 2400 mintát jelent). Ha az elemzés eredményeképpen az

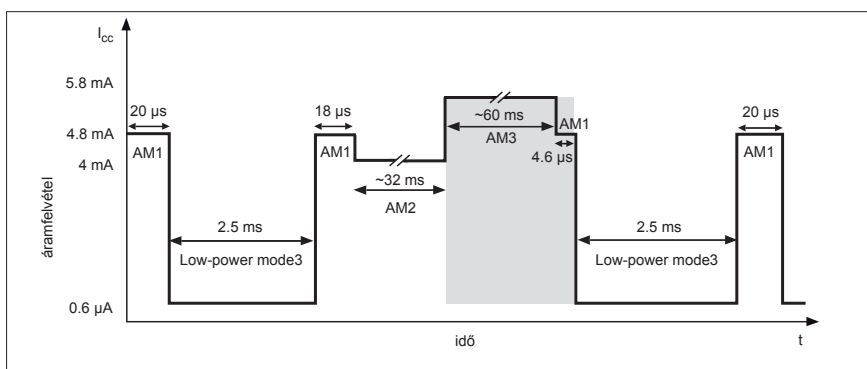
9. ábra Az eszköz áramfelvételi profilja az aktivitásdetektálás fázisában



10. ábra A puffanásdetektálás áramfelvételi profilja



11. ábra Az üvegtörés-detektálás áramfelvételi profilja



adódik, hogy a jel nem felel meg az üvegtörés jellemző tulajdonságainak, az eszköz 4,6 ms-ra visszatér az AM1-üzemmódba, mialatt az eszköz újrainicializálása és a 2,5 ms-onkénti periodikus feléledéssel működő, aktivitásdetektálási üzemmódbhoz szükséges konfigurációs folyamat végrehajtódik. Ha viszont a jelsorozat minden szempontból megfelel az üvegtörés jellemzőinek, az eszköz bekapcsolja a beépített ledet és 3 másodpercre a hangjelzést, majd visszatér az aktivitásdetektálás üzemmódjába.

Az 1. táblázat felsorolja az összes perifériát és órajelet, amelyek a működés különböző fázisaiban bekapcsolt állapotba kerülnek. Ennek az a jelentősége, hogy az aktív perifériák áramfelvétele hozzájárul az egyes működési fázisokra jellemző, összesített áramfelvételhez.

Az MSP430-alapú ÜTJ-kártya áramellátását két AAA-méretű elem szolgáltatja, amely 800 mAh töltést tárol. Habár az ilyen alkalmazásban nem könnyű megjósolni az elemek élettartamát, feltételezve, hogy üvegtörés nem történt, a teljes áramfelvétel nagyjából 80 µA, amiből 416 napos elemélettartam számítható ki. Ez tovább növelhető, ha az aktivitásdetektálás periódusidejét megnöveljük, viszont ez a veszéllyel jár, hogy megnövekszik az egyáltalán nem detektált akusztikus események valószínűsége.

Összefoglalás

A cikk bemutatta egy egyszerű, de mégis robusztus felépítésű, üvegtörésjelző eszköz megoldását, az üvegtörés hangjelenségeinek jellemzőit és azok felismerésének és elkülönítésének módját. Kitértünk a hardver- és szoftverkövetelményekre és azok realizációjára egy MCU segítségével, és betekintést adtunk a valós idejű megvalósítás energiaigényének kalkulációjába is.

www.ti.com/hu
<http://www.ti.com/ww/hu/cikkek-szakirodalom.html>

1. táblázat Az MCU periféria-rendszerének jellegzetes áramfelvételi és időzítési adatai

Feltétel/üzemmód	Perifériák bekapcsolva	Órajelek	Áramfelvétel	Bekapcsolási idő
LPM3	Időzítő bekapcsolva, (előreszámlálás üzemmód)	MCLK=DC0=OFF SMLCK=DC0=OFF ACLK=VLO ~ 12 kHz TALCK=ALCK=VLO	0,6 µA	2,5 ms
Aktivitásdetektálás AM1 (aktív üzemmód 1)	Mikrofon bekapcsolva, OA0 bekapcsolva	MCLK=DC0=12 MHz SMLCK=DC0=12 MHz ACLK=VLO ~ 12 kHz	4,8 mA	20 µs
Puffanásfelismerés AM2 (aktív üzemmód 2)	Időzítő bekapcsolva, (előreszámlálás üzemmód)	MCLK=DC0=8 MHz SMLCK=DC0=8 MHz ACLK=VLO ~ 12 kHz TBCLK=SMCLK=8 MHz	4 mA	32 ms
Üvegtöréshang felismerése AM3 (aktív üzemmód 3)	Mikrofon bekapcsolva, OA0 bekapcsolva. ADC10 bekapcsolva	MCLK=DC0=12 MHz SMLCK=DC0=12 MHz ACLK=VLO ~ 12 kHz	5,8 mA	60 ms