

Planet Express

Ahogy a programozó látta...

Egy kis visszatekintés: 2008 őszén határoztuk el, hogy elindulunk a Magyarok a Marson versenyen. (Turczel József - Turci, Kerényi Pál - Kepa, és én (Bernard Balázs)). A teljes feladat megvalósítása két részből áll: kell egy vízen és szárazföldön egyaránt jól közlekedő robot és egy PC oldali program, ami az élő kamerakép földolgozásával szabályozza azt. Mivel mindhárman mikrovezérlő programozásában és hardvertervezésben vagyunk / voltunk inkább jártasak, adódott a kérdés, hogy akkor ki írja a PC oldali szabályozószoftvert. A robothoz hozzá tudunk fogni, tudtuk merre induljunk, de a képfeldolgozást szinte teljes sötétséggel fedte a homály. Turci és én voltunk erre esélyesebbek, végül az első ~~köszönés~~ csapatmegbeszélésen elvállaltam, de Turci is ajánlkozott, hogy ha kap tőlem X és Y koordinátákat, akkor Ő a vezérlést megírja jávában. Végül rám maradt a teljes program kitalálása és megírása. Kepa és Turci nekiugrottak a robotnak én meg álltam a szakadék szélén és próbáltam repülni. A gond az volt, hogy utoljára kb. 10 éve tanultam a CA-VO-t (csávó) - mert kötelező volt - mint 4GL-t, amit már akkor se nagyon szerettem. Viszont a webkamera programozása régóta piszkálta a fantáziámat, de időlt időhiány miatt sosem tudtam nekifogni, ezért megmaradt egy eltemetett szándék szintjén. A csapat nevét hamar sikerült tisztázni, egyhangúan a Planet Express név mellett döntöttünk utalva az alapműnek számító rajzfilmsorozatra. A roverünk neve meg APC lett, mint az Aliens film csapatszállító járműve. A Microsoft Excel helpjéből, makrózási és VBA környezetéből próbáltam tanulni. Egész jól ment, egyszerűnek látszott a dolog. Eleinte a tömörítetlen, 24 bites .bmp formátumot próbáltam feldolgozni assembly-s megközelítésben, azaz: Binary Access segítségével megnyitottam a képfájlt és az 55-ik bájtra álltam rá. Ezután már térdig gázolhattam az RGB komponensekben. A feladat egy darab 255, 0, 0 - azaz Red - pixel megtalálása volt, egy olyan .bmp fájlban, ami 3 pixel széles és 1 pixel magas volt. (Ilyeneket gyártottam Photoshopban) Ezután jött a kétdimenziós világ: 4x4-es fájlban kerestem az adott pixelt és kiírtam a sor és oszlopkoordinátáit a felületre. Közben leszedtem a netről a Visual Basic, Microsoft által tanulási céllal ingyenessé tett környezetét. Ez persze egészen más volt, mint az Excel VBA-s "inkubátora". A következő lépés egy 800x600-as képben megtalálni az előzőleg beleszínezett Red (255, 0, 0) pixelt, aztán pixeleket, végül az utolsónak megtalált két pixel távolságát X és Y, valamint az összekötő egyenes hosszában megadva. Sajnos ez a módszer nagyon lassúnak bizonyult. Ekkor áttértem a Visual Basic által biztosított lehetőségekre, a képfájlban való "manuális" lépdelés helyett. Ez a megoldás már gyorsabbnak mutatkozott. A fájl fejlécéből már csak a dimenzióadatokat olvastam ki a későbbi ciklushatárolásokhoz.

A www.magyarokamarson.hu -ról letölthetővé vált a webkamera állóképsorozata. Ezt felhasználva Paint-ben firkáltam roverszerű (nem rovarszerű, ROVER-szerű) pacát egy-egy kék és zöld pixellel az elején és a végén, mint irányfények. Ennek megtalálása és összekötése volt a feladat. Miután ez már állóképre ment simán, adódott a gondolat: mi legyen a szabályozó rendszer alapötlete? Föltettem a kérdést: hogyan döntöm el 3 pixelről, hogy merre menjen a rover? Ha azt tudom, hogy milyen színű az első és a hátsó irányfény és tudom az elérendő pontot, akkor mi legyen a képlet, a mögöttes matematika? A megoldás azonnal beugrott: az első és hátsó irányfények megtalálása már nem gond az ismert színinformációk alapján. Az útvonal kijelölésénél a fölvetett (végigkattintgatott) koordináták is ismertek, tehát a szabályozás külső ciklusa egy For Next (az elsőnek fölvetett koordinátától az utolsóig), azon belül pedig Do ... Loop Until Elertukasarokpontot=True ciklusban terelgetjük a rovert a soronkövetkező (a For ciklus szerint éppen aktuális) sarokpontig. A rover egyenesének (az első és hátsó irányfényeken átmenő vonal) alapján felállítok egy normált koordináta-rendszert, az egyszerűség kedvéért a hátsó irányfényt az origóba helyezem. Ez alapján már simán számolható a rover egyenesének egyenlete. Innen a pont és egyenes távolságszámítására alkalmas képlettel megvan az eltérés mértéke. Az eltérés előjele megadja az irányt is (a rover szögállásától függően). Lényegében ez a megoldás vitte végig a versenyen a roverünket az Európa II -ig. Innen már "csak" a soros port kezelését kellett megtanulni, Szetáp opciót beépíteni, tolerancia lehetőségét megvalósítani, ügyesen "szétvágni" a kódot, egy Földbázisra és egy Marsbázisra, megoldani a két szoftver közötti biztonságos (visszanyugtázott) kommunikációt, beépíteni a szabályozásba a hibatűrési képességet, azaz találja meg a rovert, ha elvesztette de úgy, hogy az ne járjon az egész kép feldolgozásával, kézivezérlési lehetőséget, ha közbe kell avatkozni, stb., stb. Mindeközben Kepa és Turci agyaltak a roboton, szervót beleztek, hogy az ne csak szögtartományban legyen képes forogni, hanem mint egy DC motor... stb. Én is szívesen ott lettem volna velük, de míg Ők a robotot

fejlesztették, nekem a Visual Basic -et kellett annyira megismernem, hogy az abban megírt program képes legyen ellátni a feladatát. A kocsmában persze gondolatban beleraktunk hőelhajláson alapuló dőlésérzékelőt, gyorsulásérzékelőt, szivattyúkat a gyorsabb mozgáshoz, stb. Ezekből sajnos nem lett semmi, egyrészt mert sokmindent nem lehetett kapni, amiről szerettünk volna, másrészt mert rendkívül kevés időnk volt a sulis és a munka mellett... Az első tesztelésünk meglepően jól sikerült. Ekkor láttam esélyt arra, hogy a rover képes lesz a szabályozó rendszer szerint arra haladni, amerre kijelöltük számára a bejárandó útvonalat. Persze alig-alig haladt a rover, de a teszt rengeteg tapasztalati hozadéka később meghozta gyümölcsét. Ekkor még a webkamerát a saját programom kezelte (WDM driveren keresztül) és persze még nem volt külön földi és marsi bázis sem. A tús pukkasztás viszont teljesen kudarcnak bizonyult: a rover a 12 cm átmérőre fűjt, parkettára leragacsolt lufit, félig magaalá gyűrve legázolta, a tű bemélyedt a lufiba másfél centit, de az nem pukkant ki... Komoly problémát okozott a kép betöltése webkameráról: én a WheresJames ingyenes szoftvert használtam a fejlesztésnek ebben a szakaszában, azaz a képet fájlba mentette 1 képkocka / sec sebességgel, és én meg onnan olvastam be. Igen ám, de gyakran "jöttek" fél és negyedképek, hiszen a WheresJames még írta a képfájlt, amikor én már olvastam azt, így előfordult, hogy a betöltött képnek csak a felső harmada volt kép, a többi csak zaj... A WebcamXP-vel nem voltam kibékülve, teljesen leterhelte az idén 11 éves, P2 533MHz-es, celeronos kövületet. A fejlesztő gépem meg egy villámcsapott berendezés, így egyetlen usb port sem működött rajta. Egy középkategória aljának számító webkamerát vettem (Logitech 3500), azzal próbálkoztunk. A kezdeti időkben a breadboardon készítettem irányfényeket: színes led fényét visszavettem, és ráborítottam egy színezett papírt. A kamera az íróasztalról nézett le, a padlón meg a lábammal tologattam a breadboardot és figyeltem, mit is csinál a program. Később is ez maradt az irányfénytész, mert a robot Kepánál volt. (A második (első medencés) próba után megkaptam egy hétre, őt akkor terhelte meg a sulis.) A tesztek során hamar kiderült, hogy az irányfények megtalálása kritikus az egész szabályozási rendszerre nézve. Leddel "optikánlóve" a kamerát, az kivérzéses jelenséget szenved: a led által módosított pixelhalmaz színinformációval nem fog rendelkezni csak, mint fehér paca lesz látható a képen. Időközben átköltöztem feleségem laptopjára, azt hurcibáltuk teszteléshez. (Ez azért is volt üdvöztető lépés, mert a VB ekkor már rendszeresen - és mentés nélkül - kilépett, így az el nem mentett változtatások "elszálltak". Ezért szinte minden új sor után célszerű volt menteni egyet, sőt váratlan kilépés után meg kellett várnom, míg a HDD-intenzív programbezáródás végetér, ismét indítani a VB-s projekteket (Marsbázis-Földbázis), megkeresni, hogy épp hol editáltam a kódot, és ismét feltölteni a koponyacsontom által határolt memóriaterületre a kód éppen módosítás alatt lévő összefüggéseit, és azok



hatásait a programrendszer többi részére. Ráadásul a monitorom is csak fél óra üzemidőt követően kezdett el élesedni, a teljes élességet csak a végtelenbe tartó idő alatt érné el, ezt viszont nem tudtam kivárni...) Az első medencés tesztnél Kepánál állítottunk fel egy gyermekmedencét a lakásban (a fényviszonyok miatt), amiről a teszt során derült ki, hogy két helyen is lukas. Itt már elkülönült a Földbázis és a Marsbázis, valamint a WebcamXP-vel kibékülve a feleségem laptopja, mint képszervert jelent meg a lokálneten. Marsbázisnak egy selejtezett szervert vittem, ami kb. 60 kilós tömegével és kb. 70 decibeles zajszennyezésével, de két processzorával és több, mint 1 gigabájtnyi memoárjával már simán ellátta a marsi bázis funkcióját. (A versenyre szedelődsködvé Kepa

javasolta, hogy vigyük magunkkal - az eredeti terv ez volt, de Turci közben szerzett egy 1 U magas (rackmount) vasat, ami kb. 50 kilóval volt könnyebb, mint az én "gépállatom" - biztos ami biztos alapon, de végül otthonhagytuk.) Kepa laptopja avanszált Földbázissá.

Az utolsó teszten a medencét már nálunk a kertben, egy féltető alatt állítottuk föl, a kamera kb. 3,3 méteren volt. Az a teszt nagyon jól sikerült, bár az irányfények problémája még nem oldódott meg. Ekkorra már sikerült venni egy gyermekeknek szánt (teszkós), hálós gyűjtőben, 100 darabos kivitelben kapható műanyag labdát, homogén színekben. Ezeket félbevágva (csiszolópapírral érdesítve)

megoldottuk az irányfények borítását. Turci ott helyben megtervezte a kör alakú nyákot smd ledekkel. Ekkor már csak egy hét volt hátra a versenyig.

Közben persze záporoztak az ötletek a lufi kipukkasztására: Kepa gyártott egy ollóskart, (az ekkor még nem működött) aminek a szervója - mint később kiderült - annyira megrántotta a tápot, hogy elhasalt tőle a rádió. Az utolsó hétre szinte több megvalósítandó feladat maradt, mint a megelőző egy hónapra! Ekkor merült föl a pengés megoldás, ami Kepánál a parkettán jól ment, a versenyen a tesztek során viszont siralmas eredményt hozott: a rover nekitolta a lufit a pálya szélének és folyamatos előremenetben sem tudta kidurrintani azt. Ez után kapott tüket az első lökhárító, a zsilippengék közé.

Persze a verseny előtti napokban át kellett írnom a programot, hogy föl lehessen venni az útvonal mellett pukkasztási zónákat is: az így megadott területen belül az ollóskart kell működtetni. A leutazás előtti napon derült ki, hogy az ollókar szervója "elkaszálja" a rádiót, így ezt aztán rapid módon kellett kigyomlálnom a szoftverből. Az utolsó vizes tesztre - verseny előtt egy héttel - írtam egy moziprogramot, ami lényegében annyi, hogy a multimaster kommunikációjú



Marsbázis - Földbázis páros közül ekkor a Mrasbázis folyamatosan közvetíti az általa megtalált irányfények X és Y koordinátáit, amit a Földbázis felrajzol a kijelölt útvonallal (az útvonal sarokpontjaiban elhelyez egy kis kört, mint dizájnelemet), és a pukkasztási zónákat. (Ezeket sárgával (lufi színe.)) A Marsbázis által "visszalőtt" első és hátsófények X és Y koordinátái alapján pedig az irányfények színével megegyező színnel szintén egy-egy kis kört, amelyeknek közepe az a pont, ahol a képfeldolgozás során a Marson futó szabályozószoftver megtalálni vélte a fényeket. Ez nagyon látványos volt, de sajnos lassította a program futását a folyamatosan közvetített információtömeg, a versenyen nem is használtuk....

A rover szaggatottan haladt, ezt szerettük volna rászabályozással "folyamatosítani": Biztonsági okokból a rover szoftvere a kapott parancsot csak 500 milisecundumig hajtotta végre. A szabályozó szoftverből, a Szetápban állíthatóvá tettem a soros port adásidőt: azaz a szabályozás eredményeként kialakult parancsot a Marsbázis csak az ebben a változóban tárolt ideig hagyta a roverre, az idő lejártá után küldött a rovernek egy Állj parancsot. Ennek elhagyása a szabályozó programból és a rover programjából folyamatosá tette volna a haladást. Illetve elegendő lett volna a roverben mondjuk 600-ra állítani, a képfeldolgozás és szabályozás ennyi idő alatt lefut a komolyabb hardveren, így a rover oldali biztonsági - 600 milisec -es - védelem sosem jutott volna érvényre csak, ha a szoftver elveszíti az irányfényeket hosszabb időre. Mindez persze dugába dőlt egy másik gond miatt: amíg a rover a kapott parancsot végrehajtja - azaz működnek a szervói - addig a rádió nem tud venni semmit a Marsbázistól. A szabályozó szoftverből kigyomlált kódrészleteket a verseny napján szögeltem vissza. Szerettük volna "összelőni" ezt az időzítést hiszen ha a PC oldali szabályozás pl. 340 milisec. -enként küld új parancsot a rovernek, akkor a rover oldalon 300 milisec. -et beállítva két parancsvégrehajtás között csak 40 milisecundumot állt volna a szervó (feltéve, hogy a parancsot ebben a 40 milisecundumban meg tudja "venni" a Marsbázistól).... Ezzel szerettük volna áthidalni a HopeRF eszköz silányságából fakadó tetemes idővesztést. Ez sajnos nem jött össze, mert én a verseny előtti éjszaka kidőltem, Turci és Kepa ezalatt az irányfényeket „hangolta”, és mire megpróbáltuk valamelyest csökkenteni – hajnaltájt –

ezt az idővesztést, az akkuk teljesen lemerültek. (A normális persze az lett volna, ha tudunk kommunikálni a roverrel miközben az halad. Tehát nem kell megállítanunk, ha parancsot adunk neki....)

A szabályozó szoftverbe az utolsó teszt előtti héten bekerült egy Kézivezérlés funkció is. Erre kattintva - a kötelező késleltetés letelte után - a Marsbázis szoftvere "elfelejti" a bejárandó útvonalat, a rovernek küld egy Állj parancsot és lehetővé válik a rover manuális irányítása. A versenyszabályok értelmében minden irányító gomb lenyomása után kötelezően várni kell 15 másodpercet. Viszont egy üzenetsomagban lehet több utasítás is, így beépítettem a szoftverbe az egyes parancsok felvételének lehetőségét, így a Rec gombra kattintva több irányító gombot is lenyomhattunk igényeinknek megfelelő sorrendben, amiket egyetlen 15 másodperces késleltetéssel, egyetlen üzenetsomagban juttat el a rendszer a Marsra, ahol a korábban beszétápolt, Sorosportadásidő változóban tárolt késleltetéssel "lejátsza" a rovernek, amiket az végrehajt. (A versenyen erre sem volt szükség.) Már a második tesztén nyilvánvalóvá vált, hogy ha a szabályozó szoftver megtalálja az irányfényeket, akkor simán kézbentartja a rover mozgását. A Marsbázis szoftverben az ellenőrzés miatt összekötöttem a megtalált irányfényeket. A képfeldolgozás gyorsítására pedig kitaláltam egy lokális szkennelést: azaz a Marsbázis szabályozószoftvere elsőként meghatározza a teljes Munkaterületen a rover pozícióját. Munkaterület az a terület, amelybe befér a 8x8 méteres pálya, amit a Földbázisról lehet fölvenni. Már ez kisebb, mint a teljes 640x480 -as képterület.



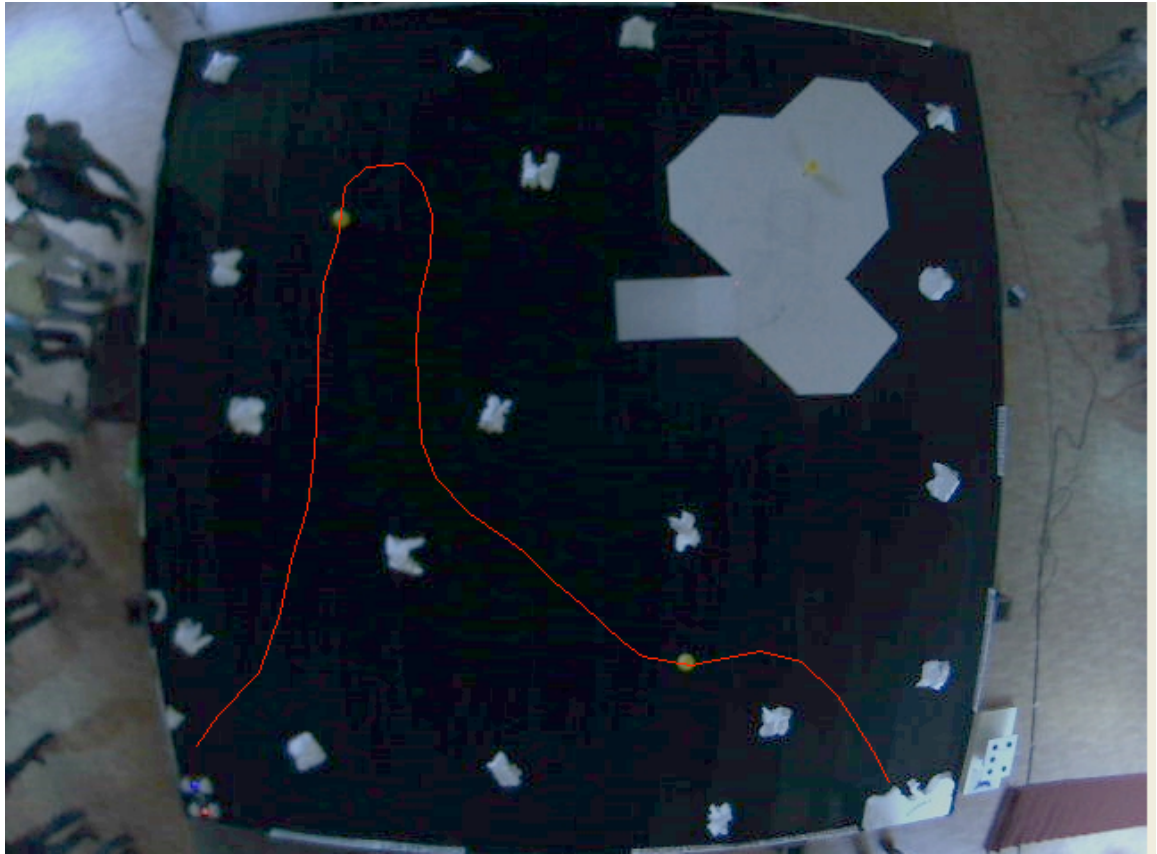
A cél nem csak az volt, hogy a pálya mellett, azon kívül elhelyezkedő robotok, csapatok, közönség színei / ruházata ne zavarjon, hanem a kisebb képterület arányosan rövidebb idő alatt dolgozható föl. Ha az irányfények jól láthatóak a szoftver számára is folyamatosan, akkor soha többet nem kell az egész Munkaterületet földolgozni. Ekkor ugyanis a szoftver meghatároz egy befoglaló síkidomot, amelybe - némi ráátartással - beleférnek az irányfények és a soronkövetkező koordináta sarokpont. Ezt a síkidomot meg is rajzolja a Marsbázis. (Ez inkább a korábbi fejlesztési szakaszból maradt meg.) Ennek a síkidomnak a területét dolgozza föl az algoritmus: az adott pixelt szétbontja R G B komponensekre, és a Földbázis Szetáp -jában megadott toleranciával szűrve vagy megvannak az irányfények, vagy halad tovább a feldolgozó rutin. Később ez is finomodott (a verseny előtti napokban), azaz a rutin végigszkennelte az összes pixelt a befoglaló síkidomban akkor is, ha megtalálta az első és hátsó irányfények pixeleit. Ugyanis beépítettem egy belső szűrést, ami vizsgálta az eredetinek - a Földbázison - fölvevett irányfény R G B összetevők mindegyikénél a középértéktől való eltérést, és ha talált olyat a rutin, amely R G B komponenshármassal közelebb esett ehhez a középértékhez, akkor azt tekintette a megtalált -

első vagy hátsó - irányfénynek. (Tehát az eredetileg fölvevett pixel színéhez a színben legközelebb esőt választotta ki.) A pixel pozíciójából adódott a sor és oszlopkoordináta, azaz az X és Y érték. Mivel a játéklabda, ami az irányfények borítását adta, alig volt nagyobb átmérőjű, mint 6 cm, ezért a műholdképen egy pár pixelből álló foltocska jelent meg. A verseny során a Marsbázis szabályozó szoftverének Lokálszken - befoglaló téglalapot feldolgozó - algoritmus, pár alkalommal elvesztette az irányfényeket. Ekkor viszont egy Látószög nevű változó értékével megnövelt befoglaló síkidomot dolgozott föl ismét a rutin. Ha nincs meg az irányfények közül legalább az egyik, akkor a Látószög növelésével a műholdkép adott részterületét szkennelte ismét végig. Tehát a rutin a műholdkép azon területén "kapirgált" az irányfények után, ahol az valószínűleg felbukkan. Ez úgy alakult ki, hogy tartani kellett a rover ki / elsodródásától. Főleg akkor, amikor az ollós lufipukkasztó ötlete kezdett körvonalazódni. Előfordulhatott ugyanis, hogy az ismét szétnyíló ollószerkezet pont ellöki a rovert a lufitól, vagy a rover egyik széle pont beleütközik valamibe és arról kifordulva kikerül a befoglaló síkidom területéről. Ezért volt látható a téglalap megnövelt kirajzolása a Marsbázison, amikor tolerancián kívülre került a rover legalább egyik irányfénye.

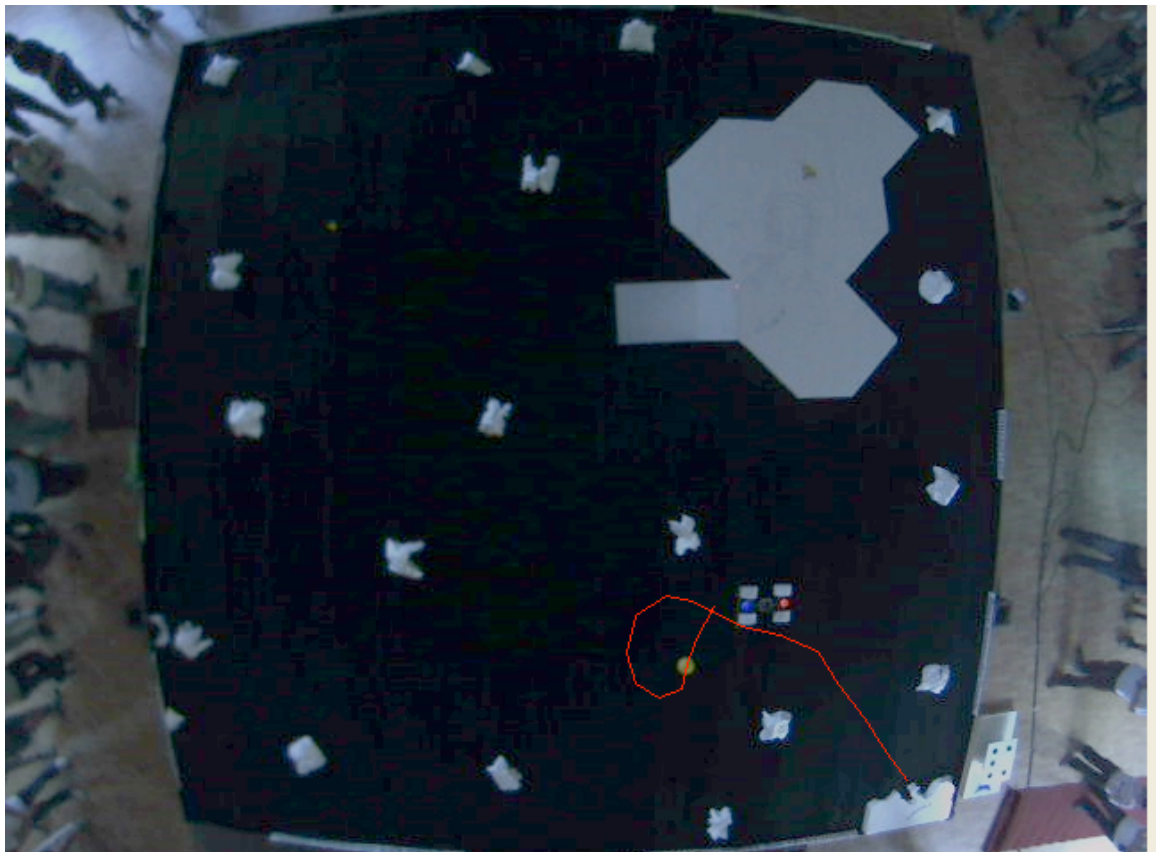
Látványos lett volna kiadni a Marsbázis képét a projektorra, de a szerveren nem volt, csak egy monitorcsatlakozó, az meg kellett nekünk. Feleségem laptopja - amit Földbázisnak használtunk - lelassult volna, ha a második monitorkimenetre ráaggattuk volna a projektort. A tesztek során csatlakoztattunk hozzá külső megjelenítőt, de a laptop érezhetően komótosabbá vált tőle, így ezért protestáltam, amikor a szervezők kérték a projektorkábel bedugását. De kézikamerával vették a műsort, remélem mi is kapunk belőle legalább egy .avi -t. A Marsbázis kapott egy folyamatjelző funkciót is: néhány Shape objektummal jeleztem, hogy melyik fázisban dolgozik éppen a program. Az egyes fázisok (Képbetöltés, Képfeldolgozás, Szabályozás) végrehajtása során a megfelelő Shape objektumot zöldre színeztem, amíg az adott fázis végrehajtodik, így könnyedén látható volt az egyes fázisok végrehajtási sebessége és időaránya. Ez csak a fejlesztés kezdeti szakaszában rendelkezett jelentőséggel, de a "végleges" verzióban is benne maradt. A kövület gépemén a soros portot nem ismerte föl az operációs rendszer (mert az a gép annyira régi), ezért a Marsbázis szoftvere kapott egy Rádió engedélyezése parancsgombot, amivel az "alapban" (defaultként) tiltott rádiót lehetett engedélyezni. Az Exit és About gombok mellett már csak a GO parancsgomb az, amivel a Marsbázis szoftvere rendelkezett. Ezzel indulhatott a szabályozási folyamat. A "végleges" verzióból ki szerettem volna hagyni a Rádió engedélyezése és a GO gombokat, de erre sem maradt idő. Viszont ezekre csak egyszer kellett kattintani, innentől kezdve minden konfigurálás, parancs és kérés (pl. mozi üzemmódhoz) a Földbázisról érkezett és a Marsbázis automatikusan reagált rá.

Másképpen: a Földbázisról szétápoltam mindent, jelöltem ki az útvonalat, munkaterületet, vetem föl az irányfényeket egy (azaz két) kattintással, mindez "átpergett" lokáltneten a Marsbázisra, amelynek szabályozó algoritmus szimpla parancsokkal irányította a rovert, amely a rádión keresztül kapott utasításokat végrehajtotta. A rover programja döntéseket nem hozott, mindenben a Marsbázis szabályozási rendszere döntött. (Pl.: ha elvesztette az irányfényeket, akkor a Látószög növelésével egyre nagyobb területet dolgozott volna föl a rover legutóbbi helyzetének ismeretében, végül a teljes Munkaterületre keresett volna 3 alkalommal. Ezután - a kötelező 15 sec. - es késleltetéssel - a Földbázison megjelenik az irányfények új felvételének szükségességét jelző felszólítás.)

A Földbázison a műholdkép lekérése is 15 sec. -es késleltetésű volt (Szetáploáshoz), hiszen a szabályok értelmében a Föld-Mars távolságot így kötelező szimbolizálni, és a műhold a Mars körül kering! Az első terv az volt, hogy a pálya széléről indulva (a rover hátsó kerekeivel érintve a szélét), kikerüljünk az első akadályt, "föl"megyünk a lufiért, majd "le" a másikért és végül érintjük az Európa II -öt. Ezután megyünk a jégmezőre a többiekért. Mindez így is indult, de az első lufi pukkasztása után, amikor már jött "lefelé" a rover, Kepa javasolta, hogy menjünk a jégre.



(Az irányfényeket stabilan látta a szoftver - egy-két alkalom volt, amikor a Látószög változóval kellett a szabályozásnak növelnie a keresést, nem azért, mert elsodródott hanem, mert elvesztette a fényeket.) Ekkor a Kézivezérléssel közbeszóltam, a 15 másodperc késleltetés alatt a rover eljutott a rámpa "magasságáig". Új útvonal felvételével felvittem a jégre, fordulás balra "föl" a lufin túlra, fordulás vissza, jobbra és "lefelé", végül a rámpa mellőzésével lecsobbantunk a jégmező "alján" a lenti lufi felé. Kanyar a lufira és irány az Európa II.





A lufit tartó konzolba némileg "belegabalyodott", de az alacsony merülésnek köszönhetően, mindig le tudott vergődni a vasakról. Amikor Kepa a sorsoláson az első helyet csípte meg, (pedig súlykoltam az ukázt: "csak az első helyet ne, bármit, csak azt NE!", de a szerencse már csak ilyen...) akkor erős aggályaim voltak: a szoftver nem volt készen, a Visual Basic le sem fordította, mert a legutóbbi módosítás "nyomai" nem stimmeltek szintaktikailag, ezeket villámgyorsan el kellett varrni, a szoftver felületén a leutazás napján (pénteken) rátett címkedoboz (Label elem – hibakeresési célzattal (valami változó értékét

monitoroztam)) még ott éktelenkedett, és egy alapos átgondolás és teszt kellett volna, hogy némi esélyt lássak a sikerre. A verseny napján kigyomláltam a programból a pénteki, sebtében elkövetett módosítások maradványát és teszteltünk volna, (mentünk kb. 110 centit) de az akkuk pont ekkor merültek le teljesen. Kepa az akkutöltőkkel varázsol, így egy átdrótolt megoldással, markáns töltőárammal próbáltuk feltölteni az akkukat. Szóval 9:56 -kor derült ki, hogy 10:00 -kor indul a versenyünk.... a szoftver még átmeneti állapotban, az akukk félig üresek, stb... Menet közben kiderült, hogy az Űrközpontnak épített paraván mögötti infrasrúktúra nem működik: az UTP kábel nem linkelt föl, (szerencsére volt másik), a műholdképet sugárzó számítógép nem áll szóba semmilyen egérrel, majd lefagy, végül folyamatosan sípol. Attila négykézláb szerelte, mi meg az ő feje fölött kábelezünk. Mire a gépek leálltak, mire átrógeroltuk őket az Űrközpontba, mire a sípolós géppel lett valami (Hatalmas köszönet ezért is Sipos Attilának, hiszen pikk- pakk megoldotta valahogy a gondunkat), mire fölálltak a gépeink, mire látták a műholdképet, és egymást, mire indult a Marsbázis szabályozó szoftvere, mire indult a Földbázis... ekkorra már kb. 30 perces késésben voltunk és már elindult a küldetési idő visszaszámlálása is....

Ekkor még össze kellett lőni a Földbázis szoftverét a Marsbáziséval, beállítani a soros port paramétereit, indítani a feldolgozó folyamatot a Marsbázison, hogyha a Földről megérkeznek az adatok, akkor legyen ami szabályoz... stb., stb. (Volt olyan csapat, amelynek a robotja 8 másodperccel a visszaszámlálás indulása után már megmozdult, a miénk még percekig ott vesztegelt, mire az alapbeállításokat megtettük, fölvettem az útvonalat és kattinthatam a GO-ra, hogy a kötelező 15 sec. után a Marsbázis hozzálasson a rover terelgetéséhez. Ráadásul a portszámot nem írjuk be (alapértelmezett értéken maradt), így le kellett lőni a szabályozást és újraszétapolni a portszámot, újabb 15 sec... A Marsbázis lefordított .exe -ből futott, a Földbázist viszont a VB-s környezetből, hiszen nem volt idő befejezni a programot, féltem, hogy valami ottmaradt dologba lép és akkor .exe esetén totál lerobban, környezetből futtatva még van esély továbblékdödni a végrehajtást. Még kellett volna egy hét és egy 20 perces teszt a pályán, hogy "késznek" nyilvánítsam - adott funkcionalitás mellett - a programrendszer. A Földbázis és a Marsbázis közötti visszanyugtázott átvitelt még a leutazás napján finomítottam, így volt esély, hogy adott kontextusban eldobja a szinkront a két program (ha esetleg elrontottam a logikát valahol...). De aztán nagyon stabilnak bizonyult, pedig nagyon egyszerű, "kettős parancsküldéses, visszanyugtázott, multimaster" megoldást találtam ki. Ahol lehetett az egyszerű, túlbonyolításoktól mentes megoldásokra törekedtem a szabályozó szoftverrendszer megírása során. Így gyorsabb a kódvégrehajtás, és átláthatóbb a kód. Jövőre a visszaszámlálás indulásakor mi is csak a GO-ra kattintunk, (ha nem is 8 másodperc, de azért a kötelező késleltetés lejárta után azonnal indulna a roverünk) feltöltjük az akkukat és összehangoljuk a rovert a szabályozó szoftverrel. És persze a mostani, kezdeti tapasztalataimat fölhasználva - feltéve, ha jövőre is én írom a képfeldolgozó-szabályozó programot -, alaposabb és több teszteléssel egy "készebb" szoftverrel vágunk majd neki a Mars meghódításának.

Nagyon jó mulatság volt, fantasztikus volt a hangulat, remek megoldásokat láthattunk, nagyszerű embereket ismerhettem meg, és végre személyesen is találkozhattam Géczi Gáborral (aki nem mellesleg törött szervóval vitte végig a robotját és így is a negyedik helyezést szerezte meg!)? Kepa írt már a robotról, de egy-két gondolattal én is szeretnék hozzájárulni a leírásához: a versenyen többen keresték a masszív vasat a robotban, de a puritán nyáklap alatt csak a kifacsart akkuk pihentek. Páran alig akarták elhinni, hogy ilyen kevés alkatrésszel megvalósítható egy robot. Pedig meglepően sokminden befér egy

mikrovezérlőbe! A többi csak a táp, a rádió, és a szervók illesztése volt. (Az viszont biztos, hogy legközelebb messze elkerüljük a HopeRF termékeket. A silány rádió miatt szinte annyit állt a rover, amennyit haladt. Vagyis a rover haladási ideje – ezáltal a küldetési idő – a kétszeresére nőtt...) Kepa és Turci egy minden ízében jól átgondolt robotot hoztak össze! A zseniális kerékprofilnak köszönhető oldalazási képességért – Kepa ötlete – a csapat megosztott különdíjat is kapott. (Illetve a díjból csak az oklevelet.) A PC oldali programban is hasonlóra törekedtem: egyszerű - gyors lefutású - szabályozási kör épüljön föl.

A versenyre való készülés hatalmas toleranciát igényelt feleségem részéről, hiszen amikor a munkából fáradtan hazaestem, (némi táplálék és nem kevés koffeinmennyiség bevitelét követően) szinte csak a Visual Basic megismerésével és a program fejlesztésével foglalkoztam. Hatalmas köszönet illeti Öt ezért, sőt a végén még a laptopját is elragadtam magammal, hogy az legyen a Földbázisunk. Köszönet a szervezőknek, a zsűrinek, hatalmas köszönet jár Sipos Attilának a rengeteg munkáért, ami lehetővé tette, hogy ennyien csobbanhassunk a Marson! Reméljük lesz jövőre is verseny!